



Open Source Used In App-BOM-ination 1.0

Cisco Systems, Inc.

www.cisco.com

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

Text Part Number: 78EE117C99-1196572276

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

In your requests please include the following reference number 78EE117C99-1196572276

Contents

- 1.1 java-api 2.6.0**
 - 1.1.1 Available under license
- 1.2 freemarker 2.3.13**
 - 1.2.1 Available under license
- 1.3 activemq 5.8.0**
 - 1.3.1 Available under license
- 1.4 commons-logging 1.1.1**
 - 1.4.1 Available under license
- 1.5 protobuf-java 2.6.0**
- 1.6 hawtbuf 1.9**
- 1.7 mysql-jdbc 8.0.11**
 - 1.7.1 Available under license
- 1.8 spring-test 2.5.6**
 - 1.8.1 Available under license
- 1.9 hibernate-validator 5.2.0.Final**
 - 1.9.1 Available under license
- 1.10 struts 2.1.6**
 - 1.10.1 Available under license
- 1.11 commons-io 1.3.2**
 - 1.11.1 Available under license
- 1.12 activemq-broker 5.8.0**
 - 1.12.1 Available under license
- 1.13 jboss-logging 3.2.1.Final**
 - 1.13.1 Available under license
- 1.14 geronimojmspec 1.1.1**
 - 1.14.1 Available under license

1.15 bean-validation-api 1.1.0.Final

1.16 junit 3.8.1

1.16.1 Available under license

1.17 ognl 2.6.11

1.17.1 Available under license

1.18 activemq-client 5.8.0

1.18.1 Available under license

1.19 xwork 2.1.2

1.19.1 Available under license

1.20 commons-fileupload 1.2.1

1.20.1 Available under license

1.21 slf4j 1.7.21

1.21.1 Available under license

1.22 commons-lang3 3.4

1.22.1 Available under license

1.23 geronimoj2eemanagement 1.0.1

1.23.1 Available under license

1.24 classmate 1.1.0

1.24.1 Available under license

1.25 activemq-openwire-legacy 5.8.0

1.25.1 Available under license

1.1 java-api 2.6.0

1.1.1 Available under license :

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

```
// Copyright 2008 Google Inc. All rights reserved.  
// Redistribution and use in source and binary forms, with or without  
// modification, are permitted provided that the following conditions are  
// * Redistributions of source code must retain the above copyright  
// notice, this list of conditions and the following disclaimer.  
// * Redistributions in binary form must reproduce the above  
// copyright notice, this list of conditions and the following disclaimer  
// in the documentation and/or other materials provided with the  
// * Neither the name of Google Inc. nor the names of its  
// this software without specific prior written permission.
```

Found in path(s):

```
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-  
jar/com/google/protobuf/ExtensionRegistry.java  
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-  
jar/com/google/protobuf/InvalidProtocolBufferException.java
```

* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/ProtocolStringList.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/MessageOrBuilder.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/AbstractMessageLite.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/ServiceException.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/BlockingService.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/BlockingRpcChannel.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/GeneratedMessageLite.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/AbstractParser.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/Parser.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/TextFormat.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/UnmodifiableLazyStringList.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/Service.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/Descriptors.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/LazyField.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/LazyStringArrayList.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/Message.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/ExtensionRegistryLite.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/CodedOutputStream.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/MessageLite.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/RpcCallback.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/BoundedByteString.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/ByteString.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/UnknownFieldSet.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/RepeatedFieldBuilder.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-

jar/com/google/protobuf/AbstractMessage.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/LiteralByteString.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/LazyFieldLite.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/WireFormat.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/ProtocolMessageEnum.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/RopeByteString.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/DynamicMessage.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/FieldSet.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/Internal.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/RpcController.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-jar/com/google/protobuf/Utf8.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/CodedInputStream.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/SmallSortedMap.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/MessageReflection.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/Extension.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/UninitializedMessageException.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/GeneratedMessage.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/RpcChannel.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/MessageLiteOrBuilder.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/RpcUtil.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/SingleFieldBuilder.java
* /opt/cola/permits/1000883908_1607475857.85/0/protobuf-java-2-6-0-sources-
jar/com/google/protobuf/LazyStringList.java

1.2 freemarker 2.3.13

1.2.1 Available under license :

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

/*

* Copyright (c) 2003-2007 The Visigoth Software Society. All rights reserved.

*

* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

*

* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

*

* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

*

* 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgement:

* "This product includes software developed by the
* Visigoth Software Society (<http://www.visigoths.org/>)."

* Alternately, this acknowledgement may appear in the software itself, if and wherever such third-party acknowledgements normally appear.

*

* 4. Neither the name "FreeMarker", "Visigoth", nor any of the names of the project contributors may be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact visigoths@visigoths.org.

*

* 5. Products derived from this software may not be called "FreeMarker" or "Visigoth" nor may "FreeMarker" or "Visigoth" appear in their names without prior written permission of the Visigoth Software Society.

*

* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE VISIGOTH SOFTWARE SOCIETY OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*

=====

*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Visigoth Software Society. For more
* information on the Visigoth Software Society, please see
* <http://www.visigoths.org/>
*/

Found in path(s):

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/beans/OverloadedVarArgMethod.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/beans/SimpleMemberModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/beans/MethodUtilities.java

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

/*
* Copyright (c) 2006 The Visigoth Software Society. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution, if
* any, must include the following acknowledgement:
* "This product includes software developed by the
* Visigoth Software Society (<http://www.visigoths.org/>)."
* Alternately, this acknowledgement may appear in the software itself,
* if and wherever such third-party acknowledgements normally appear.
*
* 4. Neither the name "FreeMarker", "Visigoth", nor any of the names of the
* project contributors may be used to endorse or promote products derived
* from this software without prior written permission. For written
* permission, please contact visigoths@visigoths.org.
*
* 5. Products derived from this software may not be called "FreeMarker" or "Visigoth"
* nor may "FreeMarker" or "Visigoth" appear in their names
* without prior written permission of the Visigoth Software Society.
*
*/

* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE VISIGOTH SOFTWARE SOCIETY OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

* =====

*

* This software consists of voluntary contributions made by many
* individuals on behalf of the Visigoth Software Society. For more
* information on the Visigoth Software Society, please see
* <http://www.visigoths.org/>
*/

Found in path(s):

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/ExistsExpression.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/DefaultToExpression.java

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

Copyright 2003-2007 Sun Microsystems, Inc. All rights reserved.

The contents of this file are subject to the terms of either the
GNU General Public License Version 2 only ("GPL") or the Common
Development and Distribution License("CDDL") (collectively, the
"License"). You may not use this file except in compliance with
particular file as subject to the "Classpath" exception as
provided by Sun in the GPL Version 2 section of the License file
CONFIDENTIAL. NONE means that the application does not
changed in transit. CONFIDENTIAL means that the application
CONFIDENTIAL flag will indicate that the use of SSL is
<xsd:enumeration value="CONFIDENTIAL"/>

Found in path(s):

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/jsp/web-app_2_5.xsd

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

/*

* Copyright (c) 2005 The Visigoth Software Society. All rights
* reserved.
*

* Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 *
 * 3. The end-user documentation included with the redistribution, if
 * any, must include the following acknowledgement:
 * "This product includes software developed by the
 * Visigoth Software Society (<http://www.visigoths.org/>)."
 * Alternately, this acknowledgement may appear in the software itself,
 * if and wherever such third-party acknowledgements normally appear.
 *
 * 4. Neither the name "FreeMarker", "Visigoth", nor any of the names of the
 * project contributors may be used to endorse or promote products derived
 * from this software without prior written permission. For written
 * permission, please contact visigoths@visigoths.org.
 *
 * 5. Products derived from this software may not be called "FreeMarker" or "Visigoth"
 * nor may "FreeMarker" or "Visigoth" appear in their names
 * without prior written permission of the Visigoth Software Society.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE VISIGOTH SOFTWARE SOCIETY OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
 * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
 * OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 * =====
 *
 * This software consists of voluntary contributions made by many
 * individuals on behalf of the Visigoth Software Society. For more
 * information on the Visigoth Software Society, please see
 * <http://www.visigoths.org/>
 */

Found in path(s):

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/UnaryPlusMinusExpression.java

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

/*

* Copyright (c) 2003-2006 The Visigoth Software Society. All rights reserved.

*

* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

*

* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

*

* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

*

* 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgement:

* "This product includes software developed by the

* Visigoth Software Society (<http://www.visigoths.org/>)."

* Alternately, this acknowledgement may appear in the software itself, if and wherever such third-party acknowledgements normally appear.

*

* 4. Neither the name "FreeMarker", "Visigoth", nor any of the names of the project contributors may be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact visigoths@visigoths.org.

*

* 5. Products derived from this software may not be called "FreeMarker" or "Visigoth" nor may "FreeMarker" or "Visigoth" appear in their names without prior written permission of the Visigoth Software Society.

*

* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE VISIGOTH SOFTWARE SOCIETY OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF

* SUCH DAMAGE.

* =====

*

* This software consists of voluntary contributions made by many
* individuals on behalf of the Visigoth Software Society. For more
* information on the Visigoth Software Society, please see
* <http://www.visigoths.org/>
*/

Found in path(s):

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/Configuration.java

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

distributed under licenses restricting their use, copying,
CONFIDENTIAL. NONE means that the application does not
changed in transit. CONFIDENTIAL means that the application
CONFIDENTIAL flag will indicate that the use of SSL is
<xsd:enumeration value="CONFIDENTIAL"/>

Found in path(s):

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/jsp/web-app_2_4.xsd

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

Copyright (c) 2000 Sun Microsystems, Inc.,
901 San Antonio Road,
Palo Alto, California 94303, U.S.A.
All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to
technology embodied in the product that is described in this document.
In particular, and without limitation, these intellectual property
rights may include one or more of the U.S. patents listed at
<http://www.sun.com/patents> and one or more additional patents or
pending patent applications in the U.S. and in other countries.

This document and the product to which it pertains are distributed
under licenses restricting their use, copying, distribution, and
decompilation. This document may be reproduced and distributed but may
not be changed without prior written authorization of Sun and its
licensors, if any.

Third

guarantee element specifies that the communication
between client and server should be NONE, INTEGRAL, or
CONFIDENTIAL. NONE means that the application does not require any
transport guarantees. A value of INTEGRAL means that the application

requires that the data sent between the client and server be sent in such a way that it can't be changed in transit. CONFIDENTIAL means that the application requires that the data be transmitted in a fashion that prevents other entities from observing the contents of the transmission. In most cases, the presence of the INTEGRAL or CONFIDENTIAL flag will indicate that the use of SSL is required.

Used in: user

Found in path(s):

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/jsp/web-app_2_3.dtd

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

/*

* Copyright (c) 2003 The Visigoth Software Society. All rights reserved.

*

* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

*

* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

*

* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

*

* 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgement:

* "This product includes software developed by the

* Visigoth Software Society (<http://www.visigoths.org/>)."

* Alternately, this acknowledgement may appear in the software itself, if and wherever such third-party acknowledgements normally appear.

*

* 4. Neither the name "FreeMarker", "Visigoth", nor any of the names of the project contributors may be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact visigoths@visigoths.org.

*

* 5. Products derived from this software may not be called "FreeMarker" or "Visigoth" nor may "FreeMarker" or "Visigoth" appear in their names without prior written permission of the Visigoth Software Society.

*

* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE VISIGOTH SOFTWARE SOCIETY OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

* =====

*

* This software consists of voluntary contributions made by many
* individuals on behalf of the Visigoth Software Society. For more
* information on the Visigoth Software Society, please see
* <http://www.visigoths.org/>
*/

Found in path(s):

- * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/log/NullLoggerFactory.java
- * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/beans/ArrayModel.java
- * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/util/IdentityHashMap.java
- * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/xml/JdomNavigator.java
- * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/ParenteticalExpression.java
- * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/beans/OverloadedMethodModel.java
- * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/AndExpression.java
- * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/RegexBuiltins.java
- * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/xml/Navigator.java
- * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/log/JDK14LoggerFactory.java
- * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/jsp/JspContextModel.java
- * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/SimpleDate.java
- * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/ant/JythonAntTask.java
- * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/SimpleNumber.java
- * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-

jar/freemarker/ext/beans/DateModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/utility/Constants.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/jython/JythonNumberModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/utility/ClassUtil.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/ListLiteral.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/Expression.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/NumericalOutput.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/log/Log4JLoggerFactory.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/NonNumericalException.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/AttemptBlock.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/DollarVariable.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/Case.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/beans/StaticModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/beans/IteratorModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/GeneralPurposeNothing.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/beans/SimpleMethodModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/debug/impl/RmiDebuggerService.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/BooleanLiteral.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/EscapeBlock.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/TransformBlock.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/TemplateCollectionModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/TemplateMethodModelEx.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/jython/JythonModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/RecurseNode.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/ReturnInstruction.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/beans/BeansWrapper.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-

jar/freemarker/core/CompressedBlock.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/core/ArithmeticExpression.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/beans/CollectionModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/template/utility/Execute.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/jsp/EventForwarding.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/debug/Breakpoint.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/core/IteratorBlock.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/util/ModelCache.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/core/NumberLiteral.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/template/utility/SecurityUtilities.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/beans/NumberModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/core/FreeMarkerTree.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/template/TemplateHashModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/jython/JythonWrapper.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/debug/DebugModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/template/TemplateExceptionHandler.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/template/TemplateException.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/template/SimpleObjectWrapper.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/VisitNode.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/beans/EnumModels.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/debug/DebuggerClient.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/log/AvalonLoggerFactory.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/core/ArithmeticEngine.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/template/WrappingTemplateModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/template/TemplateModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-

jar/freemarker/ext/xml/NodeOperator.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/TransformControl.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/ConditionalBlock.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/ant/UnlinkedJythonOperations.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/beans/MethodMap.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/beans/OverloadedFixArgMethod.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/utility/DOMNodeModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/jsp/TaglibFactory.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/beans/StaticModels.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/debug/Debugger.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/RecoveryBlock.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/BreakInstruction.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/cache/URLTemplateLoader.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/SwitchBlock.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/NewBI.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/TemplateNumberModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/TemplateDateModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/BooleanExpression.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/MethodCall.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/NotExpression.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/util/ModelFactory.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/cache/URLTemplateSource.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/TemplateMethodModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/servlet/HttpRequestHashModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/debug/impl/DebuggerService.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/Range.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/log/LoggerFactory.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/EvaluationUtil.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/TemplateTransformModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/servlet/ServletContextHashModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/Dot.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/TemplateHashModelEx.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/jython/JythonHashModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/ObjectWrapper.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/beans/SimpleMapModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/SimpleList.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/Comment.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/cache/FileTemplateLoader.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/servlet/HttpRequestParametersHashModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/TemplateSequenceModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/DynamicKeyName.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/AssignmentInstruction.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/cache/ClassTemplateLoader.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/Macro.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/jython/JythonSequenceModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/StringLiteral.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/ant/UnlinkedJythonOperationsImpl.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/jsp/FreemarkerTag.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/Identifier.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/StopException.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/cache/MultiTemplateLoader.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/SimpleHash.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/TemplateScalarModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-

jar/freemarker/template/SimpleSequence.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/beans/MapModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/template/utility/DeepUnwrap.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/TextBlock.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/core/NoEscapeBlock.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/servlet/FreemarkerServlet.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/template/SimpleScalar.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/core/StopInstruction.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/beans/ResourceBundleModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/template/TemplateModelException.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/Assignment.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/core/NonBooleanException.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/IfBlock.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/ant/FreemarkerXmlTask.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/beans/ClassString.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/core/FallbackInstruction.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/core/ComparisonExpression.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/template/DefaultObjectWrapper.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/xml/NodeListModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/core/OrExpression.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/beans/ClassBasedModelFactory.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/cache/TemplateLoader.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/template/utility/OptimizerUtil.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/template/utility/ObjectConstructor.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/core/MixedContent.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/ext/jsp/TagTransformModel.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/Configurable.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/servlet/HttpSessionHashModel.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/jsp/FreeMarkerPageContext.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/beans/BeanModel.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/InvalidReferenceException.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/beans/OverloadedMethod.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/xml/Namespaces.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/TemplateModelIterator.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/cache/WebappTemplateLoader.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/debug/impl/DebuggerServer.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/FlushInstruction.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/xml/DomNavigator.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/util/WrapperTemplateModel.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/beans/StringModel.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/servlet/AllHttpScopesHashModel.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/CustomAttribute.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/utility/StandardCompress.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/SimpleCollection.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/TemplateElement.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/beans/EnumerationModel.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/TemplateNodeModel.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/TemplateBooleanModel.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/TemplateObject.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/BodyInstruction.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/UnifiedCall.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/beans/BooleanModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/DebugBreak.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/jdom/NodeListModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/xml/JaxenNamespaces.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/NumericalRange.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/NonStringException.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/utility/ToCanonical.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/utility/Collections12.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/xml/Dom4jNavigator.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/LocalContext.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/Interpret.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright (c) 2003 The Visigoth Software Society. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution, if
* any, must include the following acknowledgement:
* "This product includes software developed by the
* Visigoth Software Society (<http://www.visigoths.org/>)."
* Alternately, this acknowledgement may appear in the software itself,
* if and wherever such third-party acknowledgements normally appear.
*
* 4. Neither the name "FreeMarker", "Visigoth", nor any of the names of the
* project contributors may be used to endorse or promote products derived

* from this software without prior written permission. For written
 * permission, please contact visigoths@visigoths.org.
 *
 * 5. Products derived from this software may not be called "FreeMarker" or "Visigoth"
 * nor may "FreeMarker" or "Visigoth" appear in their names
 * without prior written permission of the Visigoth Software Society.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE VISIGOTH SOFTWARE SOCIETY OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
 * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
 * OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 * =====
 *
 * This software consists of voluntary contributions made by many
 * individuals on behalf of the Visigoth Software Society. For more
 * information on the Visigoth Software Society, please see
 * <http://www.visigoths.org/>
 */

Found in path(s):

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/Include.java
 * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
 jar/freemarker/core/StringArraySequence.java
 * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
 jar/freemarker/ext/dom/Transform.java
 * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
 jar/freemarker/ext/dom/DocumentTypeModel.java
 * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
 jar/freemarker/ext/dom/CharacterDataNodeModel.java
 * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
 jar/freemarker/ext/dom/XalanXPathSupport.java
 * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
 jar/freemarker/ext/dom/DocumentModel.java
 * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
 jar/freemarker/core/AddConcatExpression.java
 * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
 jar/freemarker/ext/dom/AttributeNodeModel.java
 * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/BuiltIn.java
 * /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
 jar/freemarker/template/utility/NormalizeNewlines.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/dom/SunInternalXalanXPathSupport.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/NumericalBuiltins.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/utility/XmlEscape.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/cache/CacheStorage.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/TemplateModelListSequence.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/dom/NodeModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/NodeBuiltins.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/dom/XPathSupport.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/utility/JythonRuntime.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/cache/MruCacheStorage.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/HashLiteral.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/dom/PINodeModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/cache/TemplateCache.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/log/Logger.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/ParseException.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/cache/SoftCacheStorage.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/Environment.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/utility/StringUtil.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/CollectionAndSequence.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/BuiltinVariable.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/TrimInstruction.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/StringBuiltins.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/PropertySetting.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/LibraryLoad.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/BlockAssignment.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/utility/HtmlEscape.java

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/dom/ElementModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/cache/StrongCacheStorage.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/SequenceBuiltins.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/dom/NodeListModel.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/dom/NodeOutputter.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/utility/CaptureOutput.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/dom/JaxenXPathSupport.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/CommandLine.java
* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/template/Template.java

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

distributed under licenses restricting their use, copying,

Found in path(s):

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/jsp/web-jsptaglibrary_2_0.xsd

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

Copyright 1999 Sun Microsystems, Inc. 901 San Antonio Road,
Palo Alto, CA 94303, U.S.A. All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or documentation may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any.

Third party software, including font technology, is copyrighted and licensed from Sun suppliers.

Sun, Sun Microsystems, the Sun Logo, Solaris, Java, JavaServer Pages, Java Naming and Directory Interface, JDBC, JDK, JavaMail and Enterprise JavaBeans, are trademarks or registered trademarks of Sun Microsystems, Inc in the U.S. and other countries.

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

PostScript is a registered trademark of Adobe Systems, Inc.

Federal Acquisitions: Commercial Software

guarantee element specifies that the communication between client and server should be NONE, INTEGRAL, or CONFIDENTIAL. NONE means that the application does not require any transport guarantees. A value of INTEGRAL means that the application requires that the data sent between the client and server be sent in such a way that it can't be changed in transit. CONFIDENTIAL means that the application requires that the data be transmitted in a fashion that prevents other entities from observing the contents of the transmission. In most cases, the presence of the INTEGRAL or CONFIDENTIAL flag will indicate that the use of SSL is required.

Found in path(s):

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/jsp/web-app_2_2.dtd

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

/*

* Copyright (c) 2003 The Visigoth Software Society. All rights reserved.

*

* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

*

* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

*

* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

*

* 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgement:

* "This product includes software developed by the

* Visigoth Software Society (<http://www.visigoths.org/>)."

* Alternately, this acknowledgement may appear in the software itself, if and wherever such third-party acknowledgements normally appear.

*

* 4. Neither the name "FreeMarker", "Visigoth", nor any of the names of the project contributors may be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact visigoths@visigoths.org.


```

*
* 5. Products derived from this software may not be called "FreeMarker" or "Visigoth"
*   nor may "FreeMarker" or "Visigoth" appear in their names
*   without prior written permission of the Visigoth Software Society.
*
* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE VISIGOTH SOFTWARE SOCIETY OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* =====
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Visigoth Software Society. For more
* information on the Visigoth Software Society, please see
* http://www.visigoths.org/
*/

```

options

```

{
  STATIC=false;
  UNICODE_INPUT=true;
  // DEBUG_TOKEN_MANAGER=true;
  // DEBUG_PARSER=true;
}

```

PARSER_BEGIN(FMParser)

package freemarker.core;

```

import freemarker.template.*;
import freemarker.template.utility.StringUtil;
import freemarker.template.utility.DeepUnwrap;
import java.io.*;
import java.util.*;

```

/**

* This class is generated by JavaCC from a grammar file.

*/

```

public class FMParser {

```

```

// Necessary for adding macros and setting location info.
Template template;
private String templateName;

// variables that keep track of whether we are in a loop or a switch.
private int loopNesting, switchNesting;
private boolean inMacro, inFunction, stripWhitespace, stripText;
private LinkedList escapes = new LinkedList();
private int contentNesting; // for stripText

/**
 * Create an FM expression parser using a string.
 */
static public FMParser createExpressionParser(String s) {
    SimpleCharStream scs = new SimpleCharStream(new StringReader(s), 1, 1, s.length());
    FMParserTokenManager token_source = new FMParserTokenManager(scs);
    token_source.SwitchTo(FMParserConstants.FM_EXPRESSION);
    return new FMParser(token_source);
}

/**
 * Constructs a new parser object.
 * @param template The template associated with this parser.
 * @param reader The character stream to use as input
 * @param strictEscapeSyntax Whether FreeMarker directives must start with a #
 */
public FMParser(Template template, Reader reader, boolean strictEscapeSyntax, boolean stripWhitespace) {
    this(reader);
    this.template = template;
    token_source.strictEscapeSyntax = strictEscapeSyntax;
    this.templateName = template != null ? template.getName() : "";
    token_source.templateName = templateName;
    this.stripWhitespace = stripWhitespace;
}

public FMParser(Template template, Reader reader, boolean strictEscapeSyntax, boolean stripWhitespace, int
tagSyntax) {
    this(template, reader, strictEscapeSyntax, stripWhitespace);
    switch (tagSyntax) {
        case Configuration.AUTO_DETECT_TAG_SYNTAX :
            token_source.autodetectTagSyntax = true;
            break;
        case Configuration.ANGLE_BRACKET_TAG_SYNTAX :
            token_source.altDirectiveSyntax = false;
            break;
        case Configuration.SQUARE_BRACKET_TAG_SYNTAX :
            token_source.altDirectiveSyntax = true;
            break;
    }
}

```

```

        default : throw new IllegalArgumentException("Illegal argument for tagSyntax");
    }
}

public FMParser(String template) {
    this(null, new StringReader(template), true, true);
}

private String getErrorStart(Token t) {
    return "Error in template: " + template.getName()
        + "\nnon line " + t.beginLine + ", column " + t.beginColumn;
}

/**
 * Throw an exception if the expression passed in is a String
 * Literal
 */
private void notStringLiteral(Expression exp, String expected) throws ParseException {
    if (exp instanceof StringLiteral) {
        String msg = "Error " + exp.getStartLocation()
            + "\nFound string literal: " + exp
            + "\nExpecting: " + expected;
        throw new ParseException(msg, exp);
    }
}

/**
 * Throw an exception if the expression passed in is a Number
 * Literal
 */
private void notNumberLiteral(Expression exp, String expected) throws ParseException {
    if (exp instanceof NumberLiteral) {
        String msg = "Error " + exp.getStartLocation()
            + "\nFound number literal: " + exp.getCanonicalForm()
            + "\nExpecting " + expected;
        throw new ParseException(msg, exp);
    }
}

/**
 * Throw an exception if the expression passed in is a boolean
 * Literal
 */
private void notBooleanLiteral(Expression exp, String expected) throws ParseException {
    if (exp instanceof BooleanLiteral) {
        String msg = "Error " + exp.getStartLocation()
            + "\nFound: " + exp.getCanonicalForm()
            + "\nExpecting " + expected;
    }
}

```

```

        throw new ParseException(msg, exp);
    }
}

/**
 * Throw an exception if the expression passed in is a Hash
 * Literal
 */
private void notHashLiteral(Expression exp, String expected) throws ParseException {
    if (exp instanceof HashLiteral) {
        String msg = "Error " + exp.getStartLocation()
            + "\nFound hash literal: " + exp.getCanonicalForm()
            + "\nExpecting " + expected;
        throw new ParseException(msg, exp);
    }
}

/**
 * Throw an exception if the expression passed in is a List
 * Literal
 */
private void notListLiteral(Expression exp, String expected)
    throws ParseException
{
    if (exp instanceof ListLiteral) {
        String msg = "Error " + exp.getStartLocation()
            + "\nFound list literal: " + exp.getCanonicalForm()
            + "\nExpecting " + expected;
        throw new ParseException(msg, exp);
    }
}

/**
 * Throw an exception if the expression passed in is a literal
 * other than of the numerical type
 */
private void numberLiteralOnly(Expression exp) throws ParseException {
    notStringLiteral(exp, "number");
    notListLiteral(exp, "number");
    notHashLiteral(exp, "number");
    notBooleanLiteral(exp, "number");
}

/**
 * Throw an exception if the expression passed in is
 * not a string.
 */
private void stringLiteralOnly(Expression exp) throws ParseException {

```

```

    notNumberLiteral(exp, "number");
    notListLiteral(exp, "number");
    notHashLiteral(exp, "number");
    notBooleanLiteral(exp, "number");
}

/**
 * Throw an exception if the expression passed in is a literal
 * other than of the boolean type
 */
private void booleanLiteralOnly(Expression exp) throws ParseException {
    notStringLiteral(exp, "boolean (true/false)");
    notListLiteral(exp, "boolean (true/false)");
    notHashLiteral(exp, "boolean (true/false)");
    notNumberLiteral(exp, "boolean (true/false)");
}

private Expression escapedExpression(Expression exp) {
    if(!escapes.isEmpty()) {
        return ((EscapeBlock)escapes.getFirst()).doEscape(exp);
    }
    return exp;
}

private boolean getBoolean(Expression exp) throws ParseException {
    TemplateModel tm = null;
    try {
        tm = exp.getAsTemplateModel(null);
    } catch (Exception e) {
        throw new ParseException(e.getMessage()
            + "\nCould not evaluate expression: "
            + exp.getCanonicalForm()
            + exp.getStartLocation(), exp);
    }
    if (tm instanceof TemplateBooleanModel) {
        try {
            return ((TemplateBooleanModel) tm).getAsBoolean();
        } catch (TemplateModelException tme) {
        }
    }
    if (tm instanceof TemplateScalarModel) {
        try {
            return StringUtil.getYesNo(((TemplateScalarModel) tm).getAsString());
        } catch (Exception e) {
            throw new ParseException(e.getMessage()
                + "\nExpecting yes/no, found: " + exp.getCanonicalForm()
                + exp.getStartLocation(), exp);
        }
    }
}

```

```

    }
    throw new ParseException("Expecting boolean (yes/no) parameter" + exp.getStartLocation(), exp);
}
}

```

PARSER_END(FMParser)

/**

- * The lexer portion defines 5 lexical states:
- * DEFAULT, FM_EXPRESSION, IN_PAREN, NO_PARSE, and EXPRESSION_COMMENT.
- * The DEFAULT state is when you are parsing
- * text but are not inside a FreeMarker expression.
- * FM_EXPRESSION is the state you are in
- * when the parser wants a FreeMarker expression.
- * IN_PAREN is almost identical really. The difference
- * is that you are in this state when you are within
- * FreeMarker expression and also within (...).
- * This is a necessary subtlety because the
- * ">" and ">=" symbols can only be used
- * within parentheses because otherwise, it would
- * be ambiguous with the end of a directive.
- * So, for example, you enter the FM_EXPRESSION state
- * right after a \${ and leave it after the matching }.
- * Or, you enter the FM_EXPRESSION state right after
- * an "<if" and then, when you hit the matching ">"
- * that ends the if directive,
- * you go back to DEFAULT lexical state.
- * If, within the FM_EXPRESSION state, you enter a
- * parenthetical expression, you enter the IN_PAREN
- * state.
- * Note that whitespace is ignored in the
- * FM_EXPRESSION and IN_PAREN states
- * but is passed through to the parser as PCDATA in the DEFAULT state.
- * NO_PARSE and EXPRESSION_COMMENT are extremely simple
- * lexical states. NO_PARSE is when you are in a comment
- * block and EXPRESSION_COMMENT is when you are in a comment
- * that is within an FTL expression.

*/

TOKEN_MGR_DECLS :

{

/**

The noparseTag is set when we enter
a block of text that the parser more or less ignores.
These are <noparse> and <comment>. This variable
tells us what the closing tag should be, and when
we hit that, we resume parsing. Note that with this

scheme, <comment> and <noparse> tags cannot nest recursively, but it is not clear how important that is.

```
*/
String noparseTag;

/**
Keeps track of how deeply nested
we have the hash literals.
This is necessary since we need to be
able to distinguish the } used to close
a hash literal and the one used to
close a ${
*/
private int hashLiteralNesting;
private int parenthesisNesting;
private int bracketNesting;
private boolean inFTLHeader;
boolean strictEscapeSyntax,
    onlyTextOutput,
    altDirectiveSyntax,
autodetectTagSyntax,
    directiveSyntaxEstablished,
inInvocation;
String templateName;

// This method checks if we are in a strict mode where all
// FreeMarker directives must start with <#

private void strictSyntaxCheck(Token tok, int newLexState) {
    if (onlyTextOutput) {
        tok.kind = PRINTABLE_CHARS;
        return;
    }
    char firstChar = tok.image.charAt(0);
    if (autodetectTagSyntax && !directiveSyntaxEstablished) {
        altDirectiveSyntax = (firstChar == '[');
    }
    if ((firstChar == '[' && !altDirectiveSyntax) || (firstChar == '<' && altDirectiveSyntax)) {
        tok.kind = PRINTABLE_CHARS;
        return;
    }
    if (!strictEscapeSyntax) {
        SwitchTo(newLexState);
        return;
    }
    if (!altDirectiveSyntax) {
        if (!tok.image.startsWith("<#") && !tok.image.startsWith("</#")) {
```

```

        tok.kind = PRINTABLE_CHARS;
        return;
    }
}
directiveSyntaxEstablished = true;
SwitchTo(newLexState);
}

private void unifiedCall(Token tok) {
    char firstChar = tok.image.charAt(0);
    if (autodetectTagSyntax && !directiveSyntaxEstablished) {
        altDirectiveSyntax = (firstChar == '[');
    }
    if (altDirectiveSyntax && firstChar == '<') {
        tok.kind = PRINTABLE_CHARS;
        return;
    }
    if (!altDirectiveSyntax && firstChar == '[') {
        tok.kind = PRINTABLE_CHARS;
        return;
    }
    directiveSyntaxEstablished = true;
    SwitchTo(NO_SPACE_EXPRESSION);
}

private void unifiedCallEnd(Token tok) {
    char firstChar = tok.image.charAt(0);
    if (altDirectiveSyntax && firstChar == '<') {
        tok.kind = PRINTABLE_CHARS;
        return;
    }
    if (!altDirectiveSyntax && firstChar == '[') {
        tok.kind = PRINTABLE_CHARS;
        return;
    }
}

private void closeBracket(Token tok) {
    if (bracketNesting > 0) {
        --bracketNesting;
    } else {
        tok.kind = DIRECTIVE_END;
        if (inFTLHeader) {
            eatNewline();
            inFTLHeader = false;
        }
        SwitchTo(DEFAULT);
    }
}

```



```
}
```

```
private void eatNewline() {  
    int charsRead = 0;  
    try {  
        while (true) {  
            char c = input_stream.readChar();  
            ++charsRead;  
            if (!Character.isWhitespace(c)) {  
                input_stream.backup(charsRead);  
                return;  
            } else if (c=='r') {  
                char next = input_stream.readChar();  
                ++charsRead;  
                if (next != '\n') {  
                    input_stream.backup(1);  
                }  
                return;  
            } else if (c=='\n') {  
                return;  
            }  
        }  
    } catch (IOException ioe) {  
        input_stream.backup(charsRead);  
    }  
}
```

```
private void ftlHeader(Token matchedToken) {  
    if (!directiveSyntaxEstablished) {  
        altDirectiveSyntax = matchedToken.image.charAt(0) == '[';  
        directiveSyntaxEstablished = true;  
        autodetectTagSyntax = false;  
    }  
    String img = matchedToken.image;  
    char firstChar = img.charAt(0);  
    char lastChar = img.charAt(img.length() - 1);  
    if ((firstChar == '[' && !altDirectiveSyntax) || (firstChar == '<' && altDirectiveSyntax)) {  
        matchedToken.kind = PRINTABLE_CHARS;  
    }  
    if (matchedToken.kind != PRINTABLE_CHARS) {  
        if (lastChar != '>' && lastChar != ']') {  
            SwitchTo(FM_EXPRESSION);  
            inFTLHeader = true;  
        } else {  
            eatNewline();  
        }  
    }  
}
```

```
}  
}
```

TOKEN:

```
{  
<#BLANK : [" ", "\t", "\n", "\r"]>  
|  
<#START_TAG : "<" | "<#" | "#">  
|  
<#END_TAG : "<" | "<#" | "#">  
|  
<#CLOSE_TAG1 : (<BLANK>)* (">" | "]">  
|  
<#CLOSE_TAG2 : (<BLANK>)* ("/")? (">" | "]">  
|  
<ATTEMPT : <START_TAG> "attempt" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken, DEFAULT);}  
|  
<RECOVER : <START_TAG> "recover" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken, DEFAULT);}  
|  
<IF : <START_TAG> "if" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}  
|  
<ELSE_IF : <START_TAG> "elseif" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}  
|  
<LIST : <START_TAG> "list" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}  
|  
<FOREACH : <START_TAG> "foreach" <BLANK>> {strictSyntaxCheck(matchedToken,  
FM_EXPRESSION);}  
|  
<SWITCH : <START_TAG> "switch" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}  
|  
<CASE : <START_TAG> "case" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}  
|  
<ASSIGN : <START_TAG> "assign" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}  
|  
<GLOBALASSIGN : <START_TAG> "global" <BLANK>> {strictSyntaxCheck(matchedToken,  
FM_EXPRESSION);}  
|  
<LOCALASSIGN : <START_TAG> "local" <BLANK>> {strictSyntaxCheck(matchedToken,  
FM_EXPRESSION);}  
|  
<INCLUDE : <START_TAG> "include" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}  
|  
<IMPORT : <START_TAG> "import" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}  
|  
<FUNCTION : <START_TAG> "function" <BLANK>> {strictSyntaxCheck(matchedToken,  
FM_EXPRESSION);}  
|  
<MACRO : <START_TAG> "macro" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}
```

```

|
<TRANSFORM : <START_TAG> "transform" <BLANK>> {strictSyntaxCheck(matchedToken,
FM_EXPRESSION);}
|
<VISIT : <START_TAG> "visit" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}
|
<STOP : <START_TAG> "stop" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}
|
<RETURN : <START_TAG> "return" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}
|
<CALL : <START_TAG> "call" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}
|
<SETTING : <START_TAG> "setting" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}
|
<COMPRESS : <START_TAG> "compress" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken,
DEFAULT);}
|
<COMMENT : <START_TAG> "comment" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken,
NO_PARSE); noparseTag="comment";}
|
<TERSE_COMMENT : ("<" | "[" | "#--" > {noparseTag = "--"; strictSyntaxCheck(matchedToken, NO_PARSE);
}
|
<NOPARSE : <START_TAG> "noparse" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken, NO_PARSE);
noparseTag="noparse";}
|
<END_IF : <END_TAG> "if" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<END_LIST : <END_TAG> "list" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<END_RECOVER : <END_TAG> "recover" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken,
DEFAULT);}
|
<END_ATTEMPT : <END_TAG> "attempt" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken,
DEFAULT);}
|
<END_FOREACH : <END_TAG> "foreach" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken,
DEFAULT);}
|
<END_LOCAL : <END_TAG> "local" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<END_GLOBAL : <END_TAG> "global" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<END_ASSIGN : <END_TAG> "assign" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<END_FUNCTION : <END_TAG> "function" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken,
DEFAULT);}
|

```

```

<END_MACRO : <END_TAG> "macro" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<END_COMPRESS : <END_TAG> "compress" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken,
DEFAULT);}
|
<END_TRANSFORM : <END_TAG> "transform" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken,
DEFAULT);}
|
<END_SWITCH : <END_TAG> "switch" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<ELSE : <START_TAG> "else" <CLOSE_TAG2>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<BREAK : <START_TAG> "break" <CLOSE_TAG2>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<SIMPLE_RETURN : <START_TAG> "return" <CLOSE_TAG2>> {strictSyntaxCheck(matchedToken,
DEFAULT);}
|
<HALT : <START_TAG> "stop" <CLOSE_TAG2>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<FLUSH : <START_TAG> "flush" <CLOSE_TAG2>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<TRIM : <START_TAG> "t" <CLOSE_TAG2>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<LTRIM : <START_TAG> "lt" <CLOSE_TAG2>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<RTRIM : <START_TAG> "rt" <CLOSE_TAG2>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<NOTRIM : <START_TAG> "nt" <CLOSE_TAG2>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<DEFAULT : <START_TAG> "default" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|
<SIMPLE_NESTED : <START_TAG> "nested" <CLOSE_TAG2>> {strictSyntaxCheck(matchedToken,
DEFAULT);}
|
<NESTED : <START_TAG> "nested" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}
|
<SIMPLE_RECURSE : <START_TAG> "recurse" <CLOSE_TAG2>> {strictSyntaxCheck(matchedToken,
DEFAULT);}
|
<RECURSE : <START_TAG> "recurse" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}
|
<FALLBACK : <START_TAG> "fallback" <CLOSE_TAG2>> {strictSyntaxCheck(matchedToken,
DEFAULT);}
|
<ESCAPE : <START_TAG> "escape" <BLANK>> {strictSyntaxCheck(matchedToken, FM_EXPRESSION);}
|
<END_ESCAPE : <END_TAG> "escape" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken, DEFAULT);}
|

```

```

<NOESCAPE : <START_TAG> "noescape" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken,
DEFAULT);}
|
<END_NOESCAPE : <END_TAG> "noescape" <CLOSE_TAG1>> {strictSyntaxCheck(matchedToken,
DEFAULT);}
|
<UNIFIED_CALL : "<@" | "["@> {unifiedCall(matchedToken);}
|
<UNIFIED_CALL_END : ("<" | "[" | "/"@ ((<ID>) (".<ID>")*)? <CLOSE_TAG1>>
{unifiedCallEnd(matchedToken);}
|
<FTL_HEADER : ("<#ftl" | "#ftl") <BLANK>> {ftlHeader(matchedToken);}
|
<TRIVIAL_FTL_HEADER : ("<#ftl" | "#ftl") ("")? (">" | "]"> {ftlHeader(matchedToken);}
|
<UNKNOWN_DIRECTIVE : ("[#" | "[/#" | "<#" | "</#" | ("a"- "z", "A"- "Z", "_")+>
{
    if (!directiveSyntaxEstablished) {
        matchedToken.kind = PRINTABLE_CHARS;
    } else {
        char firstChar = matchedToken.image.charAt(0);
        if (firstChar == '<' && !altDirectiveSyntax) {
            matchedToken.kind = PRINTABLE_CHARS;
        } else if (firstChar == '[' && !altDirectiveSyntax) {
            matchedToken.kind = PRINTABLE_CHARS;
        }
        else if (strictEscapeSyntax) {
            String s = matchedToken.image;
            int index = s.indexOf('#');
            s = s.substring(index);
            String msg = "Unknown directive: "
            + s
            + " on line: " + matchedToken.beginLine
            + ", column: " + matchedToken.beginColumn + 1
            + ", in template: " + templateName;
            throw new TokenMgrError(msg, TokenMgrError.LEXICAL_ERROR);
        }
    }
}
}
}

<DEFAULT, NODIRECTIVE> TOKEN :
{
    <WHITESPACE : ([ "\n", "\r", "\t", " " ])+>
    |
    <PRINTABLE_CHARS : (~["$", "<", "#", "[", "{", "\n", "\r", "\t", " " ])+>
    |
    <FALSE_ALERT : "$" | "#" | "<" | "[" | "{"> // to handle a lone dollar sign or "<" or "#" or "<@" with whitespace

```

after"

```
|  
<OUTPUT_ESCAPE : "${" : FM_EXPRESSION  
|  
<NUMERICAL_ESCAPE : "#{" : FM_EXPRESSION  
}
```

<FM_EXPRESSION, IN_PAREN, NAMED_PARAMETER_EXPRESSION> SKIP :

```
{  
<(" " | "\t" | "\n" | "\r" )+ >  
|  
<["<", "[" ["#", "!"] "--" : EXPRESSION_COMMENT  
}
```

<EXPRESSION_COMMENT> SKIP:

```
{  
<(~["-", ">", "]")+ >  
|  
<">">  
|  
<"]">  
|  
<"-">  
|  
<"--" | "--"> {if (parenthesisNesting >0) SwitchTo(IN_PAREN); else if (inInvocation)  
SwitchTo(NAMED_PARAMETER_EXPRESSION); else SwitchTo(FM_EXPRESSION);}  
}
```

<FM_EXPRESSION, IN_PAREN, NO_SPACE_EXPRESSION, NAMED_PARAMETER_EXPRESSION>

TOKEN :

```
{  
<#ESCAPED_CHAR : "\""  
  (  
    ["n", "t", "r", "f", "b", "g", "l", "a", "\\", "", "\'", "$", "{"]  
    |  
    ("x" ["0"- "9", "A"- "F", "a"- "f"])  
  )  
>  
|  
<STRING_LITERAL :  
  ("\""  
  ((~["'", "\"] | <ESCAPED_CHAR>)*  
  "\"")  
  |  
  ("\""  
  ((~["'", "\"] | <ESCAPED_CHAR>)*  
  "\"")  
  )
```

```

>
|
|
|<RAW_STRING : "r" ((\"\" (~[\"\"])* \"\") | (\"\" (~[\"\"])* \"\"))>
|
|<FALSE : "false">
|
|<TRUE : "true">
|
|<INTEGER : ([\"0\"-\"9\"])+>
|
|<DECIMAL : <INTEGER> "." <INTEGER>>
|
|<DOT : ".">
|
|<DOT_DOT : "..">
|
|<BUILT_IN : "?">
|
|<EXISTS : "??">
|
|<EQUALS : "=">
|
|<DOUBLE_EQUALS : "==">
|
|<NOT_EQUALS : "!=">
|
|<LESS_THAN : "<" | "\\lt" | "<" | "&lt;">
|
|<LESS_THAN_EQUALS : "<=" | "\\lte" | "<=" | "&lt;=">
|
|<ESCAPED_GT : ">" | "\\gt" | "&gt;">
|
|<ESCAPED_GTE : ">=" | "\\gte" | "&gt;=">
|
|<PLUS : "+">
|
|<MINUS : "-">
|
|<TIMES : "*">
|
|<DOUBLE_STAR : "**">
|
|<ELLIPSIS : "...">
|
|<DIVIDE : "/">
|
|<PERCENT : "%">
|

```

```

<AND : "&" | "&&" >
|
<OR : "|" | "||">
|
<EXCLAM : "!">
|
<COMMA : ",">
|
<SEMICOLON : ";">
|
<COLON : ":">
|
<OPEN_BRACKET : "["> {++bracketNesting;}
|
<CLOSE_BRACKET : "]"> {closeBracket(matchedToken);}
|
<OPEN_PAREN : "(">
{
    ++parenthesisNesting;
    if (parenthesisNesting == 1)
        SwitchTo(IN_PAREN);
}
|
<CLOSE_PAREN : ")">
{
    --parenthesisNesting;
    if (parenthesisNesting == 0) {
    if (inInvocation) SwitchTo(NAMED_PARAMETER_EXPRESSION);
        else SwitchTo(FM_EXPRESSION);
    }
}
|
<OPEN_BRACE : "{"> {++hashLiteralNesting;}
|
<CLOSE_BRACE : "}">
{
    if (hashLiteralNesting == 0)
        SwitchTo(DEFAULT);
    else
        --hashLiteralNesting;
}
|
<IN : "in">
|
<AS : "as">
|
<USING : "using">
|

```


< ID: <LETTER> (<LETTER>|<DIGIT>)* >

|

< #LETTER:

[

"\u0024",
"\u0040"-" \u005a",
"\u005f",
"\u0061"-" \u007a",
"\u00c0"-" \u00d6",
"\u00d8"-" \u00f6",
"\u00f8"-" \u00ff",
"\u0100"-" \u01ff",
"\u0304"-" \u0318",
"\u0330"-" \u0337",
"\u0340"-" \u03d2",
"\u04e0"-" \u09ff",
"\uf900"-" \ufaff"

]

>

|

< #DIGIT:

[

"\u0030"-" \u0039",
"\u0060"-" \u0069",
"\u00f0"-" \u00f9",
"\u0966"-" \u096f",
"\u09e6"-" \u09ef",
"\u0a66"-" \u0a6f",
"\u0ae6"-" \u0aef",
"\u0b66"-" \u0b6f",
"\u0be7"-" \u0bef",
"\u0c66"-" \u0c6f",
"\u0ce6"-" \u0cef",
"\u0d66"-" \u0d6f",
"\u0e50"-" \u0e59",
"\u0ed0"-" \u0ed9",
"\u1040"-" \u1049"

]

>

}

<FM_EXPRESSION, NO_SPACE_EXPRESSION, NAMED_PARAMETER_EXPRESSION> TOKEN :

{

<DIRECTIVE_END : ">">

{

if (inFTLHeader) eatNewline();
inFTLHeader = false;
if (altDirectiveSyntax) {

```

        matchedToken.kind = NATURAL_GT;
    } else {
        SwitchTo(DEFAULT);
    }
}
|
<EMPTY_DIRECTIVE_END : "/>" | "/"> {if (inFTLHeader) eatNewline(); inFTLHeader = false;
SwitchTo(DEFAULT);}
}

```

<IN_PAREN> TOKEN :

```

{
<NATURAL_GT : ">">
|
<NATURAL_GTE : ">=">
}

```

<NO_SPACE_EXPRESSION> TOKEN :

```

{
<TERMINATING_WHITESPACE : ("\\n", "\\r", "\\t", " ")> : FM_EXPRESSION
}

```

<NAMED_PARAMETER_EXPRESSION> TOKEN :

```

{
<TERMINATING_EXCLAM : "!" ("\\n", "\\r", "\\t", " ")> : FM_EXPRESSION
}

```

<NO_PARSE> TOKEN :

```

{
<TERSE_COMMENT_END : "-->" | "--]">
{
    if (noparseTag.equals("-->")) {
        boolean squareBracket = matchedToken.image.endsWith("]");
        if ((altDirectiveSyntax && squareBracket) || (!altDirectiveSyntax && !squareBracket))
            SwitchTo(DEFAULT);
    }
}
|

```

<MAYBE_END :

```

("<" | "[")
"/"
("#")?
(["a"- "z", "A"- "Z"])+
(" " | "\\t" | "\\n" | "\\r")*
(">" | "]" )

```

>

```

{

```

```

StringTokenizer st = new StringTokenizer(image.toString(),
    "\t\n\r<>[]/##",
    false);
if (st.nextToken().equals(noparseTag)) {
    SwitchTo(DEFAULT);
}
}
|
<KEEP_GOING : (~["<", "[", "-"])+>
|
<LONE_LESS_THAN_OR_DASH : ["<", "[", "-"]>
}

```

```

// Now the actual parsing code, starting
// with the productions for FreeMarker's
// expression syntax.

```

```

/**
 * This is the same as OrExpression, since
 * the OR is the operator with the lowest
 * precedence.
 */

```

```

Expression Expression() :

```

```

{
    Expression exp;
}
{
    exp=OrExpression()
    {
        return exp;
    }
}

```

```

/**
 * Lowest level expression, a literal, a variable,
 * or a possibly more complex expression bounded
 * by parentheses.
 */

```

```

Expression PrimaryExpression() :

```

```

{
    Expression exp;
}
{
    (
        exp=NumberLiteral()
        |
        exp=HashLiteral()
    )
}

```

```

|
exp=StringLiteral(true)
|
exp=BooleanLiteral()
|
exp=ListLiteral()
|
exp=Identifier()
|
exp=Parenthesis()
|
exp=BuiltinVariable()
)
(
LOOKAHEAD(<DOT>
|<OPEN_BRACKET>
|<OPEN_PAREN>
|<BUILT_IN>
|<EXCLAM>
|<TERMINATING_EXCLAM>
|<EXISTS>)
exp=AddSubExpression(exp)
)*
{
return exp;
}
}

```

Expression Parenthesis() :

```

{
Expression exp, result;
Token start, end;
}
{
start=<OPEN_PAREN>
exp=Expression()
end=<CLOSE_PAREN>
{
result = new ParentheticalExpression(exp);
result.setLocation(template, start, end);
return result;
}
}

```

/**

- * A primary expression preceded by zero or
- * more unary operators. (The only unary operator we
- * currently have is the NOT.)

```

*/
Expression UnaryExpression() :
{
    Expression exp, result;
    boolean haveNot = false;
    Token t = null, start=null;
}
{
    (
        result=UnaryPlusMinusExpression()
        |
        result=NotExpression()
        |
        result=PrimaryExpression()
    )
    {
        return result;
    }
/*
    (
        t=<EXCLAM>
        {
            haveNot = !haveNot;
            if (start == null)
                start = t;
        }
    )*
    exp=PrimaryExpression()
    {
        result = exp;
        if (haveNot) {
            booleanLiteralOnly(exp);
            result = new NotExpression(exp);
            result.setLocation(template, start, exp);
        }
        return result;
    }
*/
}

```

```

Expression NotExpression() :
{
    Token t;
    Expression exp, result=null;
    ArrayList nots = new ArrayList();
}
{
    (

```

```

    t=<EXCLAM> {nots.add(t);}
)+
exp=PrimaryExpression()
{
    for (int i=0; i<nots.size(); i++) {
        result = new NotExpression(exp);
        Token tok = (Token) nots.get(nots.size() -i -1);
        result.setLocation(template, tok, exp);
        exp = result;
    }
    return result;
}
}

```

Expression UnaryPlusMinusExpression() :

```

{
    Expression exp, result;
    boolean isMinus = false;
    Token t;
}
{
    (
        t=<PLUS>
        |
        t=<MINUS> {isMinus = true;}
    )
    exp=PrimaryExpression()
    {
        result = new UnaryPlusMinusExpression(exp, isMinus);
        result.setLocation(template, t, exp);
        return result;
    }
}

```

Expression AdditiveExpression() :

```

{
    Expression lhs, rhs, result;
    boolean plus;
}
{
    lhs=MultiplicativeExpression() {result = lhs;}
    (
        LOOKAHEAD(<PLUS>|<MINUS>)
        (
            <PLUS> {plus = true;}
            |
            <MINUS> {plus = false;}
        )
    )
}

```

```

    )
    )
    rhs=MultiplicativeExpression()
    {
        if (plus) {
            // plus is treated separately, since it is also
            // used for concatenation.
            result = new AddConcatExpression(lhs, rhs);
        }
        else {
            numberLiteralOnly(lhs);
            numberLiteralOnly(rhs);
            result = new ArithmeticExpression(lhs,
                rhs,
                ArithmeticExpression.SUBTRACTION);
        }
        result.setLocation(template, lhs, rhs);
        lhs = result;
    }
    )*
    {
        return result;
    }
}

/**
 * A unary expression followed by zero or more
 * unary expressions with operators in between.
 */
Expression MultiplicativeExpression() :
{
    Expression lhs, rhs, result;
    int operation = ArithmeticExpression.MULTIPLICATION;
}
{
    lhs=UnaryExpression() {result = lhs;}
    (
        LOOKAHEAD(<TIMES>|<DIVIDE>|<PERCENT>)
        (
            <TIMES> {operation = ArithmeticExpression.MULTIPLICATION;}
            |
            <DIVIDE> {operation = ArithmeticExpression.DIVISION;}
            |
            <PERCENT>{operation = ArithmeticExpression.MODULUS;}
        )
    )
    rhs=UnaryExpression()

```

```

    {
        numberLiteralOnly(lhs);
        numberLiteralOnly(rhs);
        result = new ArithmeticExpression(lhs, rhs, operation);
        result.setLocation(template, lhs, rhs);
        lhs = result;
    }
)*
{
    return result;
}
}

```

Expression EqualityExpression() :

```

{
    Expression lhs, rhs, result;
    Token t;
}
{
    lhs=RelationalExpression() {result = lhs;}
[
    LOOKAHEAD(<NOT_EQUALS>|<EQUALS>|<DOUBLE_EQUALS>)
    (
        t=<NOT_EQUALS>
        |
        t=<EQUALS>
        |
        t=<DOUBLE_EQUALS>
    )
    rhs=RelationalExpression()
    {
        notHashLiteral(lhs, "scalar");
        notHashLiteral(rhs, "scalar");
        notListLiteral(lhs, "scalar");
        notListLiteral(rhs, "scalar");
        result = new ComparisonExpression(lhs, rhs, t.image);
        result.setLocation(template, lhs, rhs);
    }
]
{
    return result;
}
}

```

Expression RelationalExpression() :

```

{
    Expression lhs, rhs, result;

```



```

Token t;
}
{
  lhs=RangeExpression() {result = lhs;}
  [
LOOKAHEAD(<NATURAL_GTE>|<ESCAPED_GTE>|<NATURAL_GT>|<ESCAPED_GT>|<LESS_THAN_E
QUALS>|<LESS_THAN_EQUALS>|<LESS_THAN>)
  (
    t=<NATURAL_GTE>
    |
    t=<ESCAPED_GTE>
    |
    t=<NATURAL_GT>
    |
    t=<ESCAPED_GT>
    |
    t=<LESS_THAN_EQUALS>
    |
    t=<LESS_THAN>
  )
  rhs=RangeExpression()
  {
    notHashLiteral(lhs, "scalar");
    notHashLiteral(rhs, "scalar");
    notListLiteral(lhs, "scalar");
    notListLiteral(rhs, "scalar");
    notStringLiteral(lhs, "number");
    notStringLiteral(rhs, "number");
    result = new ComparisonExpression(lhs, rhs, t.image);
    result.setLocation(template, lhs, rhs);
  }
  ]
  {
    return result;
  }
}

```

```

Expression RangeExpression() :
{
  Expression lhs, rhs=null, result;
}
{
  lhs=AdditiveExpression() {result = lhs;}
  [
  LOOKAHEAD(<DOT_DOT>)
  <DOT_DOT>
  [
    LOOKAHEAD(Expression())

```

```

    rhs=AdditiveExpression()
  ]
  {
    numberLiteralOnly(lhs);
    if (rhs != null) {
      numberLiteralOnly(rhs);
    }
    Range range = new Range(lhs, rhs);
    if (rhs != null) {
      range.setLocation(template, lhs, rhs);
    } else {
      range.setLocation(template, lhs, lhs);
    }
    result = range;
  }
]
{
  return result;
}
}

```

Expression AndExpression() :

```

{
  Expression lhs, rhs, result;
}
{
  lhs=EqualityExpression() {result = lhs;}
(
  LOOKAHEAD(<AND>)
  <AND>
  rhs=EqualityExpression()
  {
    booleanLiteralOnly(lhs);
    booleanLiteralOnly(rhs);
    result = new AndExpression(lhs, rhs);
    result.setLocation(template, lhs, rhs);
    lhs = result;
  }
)*
{
  return result;
}
}

```

Expression OrExpression() :

```

{
    Expression lhs, rhs, result;
}
{
    lhs=AndExpression() {result = lhs;}
    (
        LOOKAHEAD(<OR>)
        <OR>
        rhs=AndExpression()
        {
            booleanLiteralOnly(lhs);
            booleanLiteralOnly(rhs);
            result = new OrExpression(lhs, rhs);
            result.setLocation(template, lhs, rhs);
            lhs = result;
        }
    )*
    {
        return result;
    }
}

```

ListLiteral ListLiteral() :

```

{
    ArrayList values = new ArrayList();
    Token begin, end;
}
{
    begin=<OPEN_BRACKET>
    values=PositionalArgs()
    end=<CLOSE_BRACKET>
    {
        ListLiteral result = new ListLiteral(values);
        result.setLocation(template, begin, end);
        return result;
    }
}

```

Expression NumberLiteral() :

```

{
    Token op = null, t;
}
{
    (
        t=<INTEGER>
        |
        t=<DECIMAL>
    )
}

```

```

{
    String s = t.image;
    Expression result = new NumberLiteral(template.getArithmeticEngine().toNumber(s));
    Token startToken = (op != null) ? op : t;
    result.setLocation(template, startToken, t);
    return result;
}
}

```

Identifier Identifier() :

```

{
    Token t;
}
{
    t=<ID>
    {
        Identifier id = new Identifier(t.image);
        id.setLocation(template, t, t);
        return id;
    }
}

```

Expression IdentifierOrStringLiteral() :

```

{
    Expression exp;
}
{
    (
        exp=Identifier()
        |
        exp=StringLiteral(false)
    )
    {
        return exp;
    }
}

```

BuiltinVariable BuiltinVariable() :

```

{
    Token dot, name;
}
{
    dot=<DOT>
    name=<ID>
    {
        BuiltinVariable result = null;
        try {
            result = new BuiltinVariable(name.image);

```

```

    } catch (ParseException pe) {
        pe.lineNumber = dot.beginLine;
        pe.columnNumber = dot.beginColumn;
        throw pe;
    }
    result.setLocation(template, dot, name);
    return result;
}
}

/**
 * Production that builds up an expression
 * using the dot or dynamic key name
 * or the args list if this is a method invocation.
 */
Expression AddSubExpression(Expression exp) :
{
    Expression result = null;
}
{
    (
        result=DotVariable(exp)
        |
        result=DynamicKey(exp)
        |
        result=MethodArgs(exp)
        |
        result=BuiltIn(exp)
        |
        result=DefaultTo(exp)
        |
        result=Exists(exp)
    )
    {
        return result;
    }
}

```

```

Expression DefaultTo(Expression exp) :
{
    Expression rhs = null;
    Token t;
}
{
    (
        t=<TERMINATING_EXCLAM>
        |
    )

```

```

    t=<EXCLAM>
    [
        LOOKAHEAD(Expression())
        rhs=Expression()
    ]
)
)
{
    DefaultToExpression result = new DefaultToExpression(exp, rhs);
    if (rhs ==null) {
        result.setLocation(template, exp, t);
    }
    else {
        result.setLocation(template, exp, rhs);
    }
    return result;
}
}

```

```

Expression Exists(Expression exp) :
{
    Token t;
}
{
    t=<EXISTS>
    {
        ExistsExpression result = new ExistsExpression(exp);
        result.setLocation(template, exp, t);
        return result;
    }
}

```

```

Expression BuiltIn(Expression exp) :
{
    Token t=null;
}
{
    <BUILT_IN>
    t=<ID>
    {
        BuiltIn result = null;
        try {
            result = BuiltIn.newBuiltIn(exp, t.image, t, templateName);
        } catch (ParseException pe) {
            pe.lineNumber = t.beginLine;
            pe.columnNumber = t.beginColumn;
            throw pe;
        }
    }
}

```

```

        result.setLocation(template, exp, t);
        return result;
    }
}

/**
 * production for when a key is specified by <DOT> + keyname
 */
Expression DotVariable(Expression exp) :
{
    Token t;
}
{
    <DOT>
    (
        t=<ID> | t=<TIMES> | t=<DOUBLE_STAR>
        |
        (
            t=<LESS_THAN>
            | t=<LESS_THAN_EQUALS>
            | t=<ESCAPED_GT>
            | t=<ESCAPED_GTE>
            | t=<FALSE>
            | t=<TRUE>
            | t=<IN>
            | t=<AS>
            | t=<USING>
        )
    )
    {
        if (!Character.isLetter(t.image.charAt(0))) {
            String msg = getErrorStart(t)
                + "\n" + t.image + " is not a valid identifier.";
            throw new ParseException(msg, t.beginLine, t.beginColumn);
        }
    }
}
{
    notListLiteral(exp, "hash");
    notStringLiteral(exp, "hash");
    notBooleanLiteral(exp, "hash");
    Dot dot = new Dot(exp, t.image);
    dot.setLocation(template, exp, t);
    return dot;
}
}

/**

```

```

* production for when the key is specified
* in brackets.
*/
Expression DynamicKey(Expression exp) :
{
    Expression arg;
    Token t;
}
{
    <OPEN_BRACKET>
    arg=Expression()
    t=<CLOSE_BRACKET>
    {
        notBooleanLiteral(exp, "list or hash");
        notNumberLiteral(exp, "list or hash");
        DynamicKeyName dkn = new DynamicKeyName(exp, arg);
        dkn.setLocation(template, exp, t);
        return dkn;
    }
}

```

```

/**
* production for an arglist part of a method invocation.
*/

```

```

MethodCall MethodArgs(Expression exp) :
{
    ArrayList args = new ArrayList();
    Token end;
}
{
    <OPEN_PAREN>
    args=PositionalArgs()
    end=<CLOSE_PAREN>
    {
        args.trimToSize();
        MethodCall result = new MethodCall(exp, args);
        result.setLocation(template, exp, end);
        return result;
    }
}

```

```

StringLiteral StringLiteral(boolean interpolate) :
{
    Token t;
    boolean raw = false;
}
{
    (

```



```

t=<STRING_LITERAL>
|
t=<RAW_STRING> {raw = true;}
)
{
    String s = t.image;
    // Get rid of the quotes.
    s = s.substring(1, s.length() -1);
    if (raw) {
        s=s.substring(1);
    }
    else try {
        s = StringUtil.FTLStringLiteralDec(s);
    } catch (ParseException pe) {
        pe.lineNumber = t.beginLine;
        pe.columnNumber = t.beginColumn;
        throw pe;
    }
    StringLiteral result = new StringLiteral(s);
    result.setLocation(template, t, t);
    if (interpolate && !raw) {
        if (t.image.indexOf("${") >=0 || t.image.indexOf("#{") >=0)
            result.checkInterpolation();
    }
    return result;
}
}

```

Expression BooleanLiteral() :

```

{
    Token t;
    Expression result;
}
{
    (
        t=<FALSE> {result = new BooleanLiteral(false);}
        |
        t=<TRUE> {result = new BooleanLiteral(true);}
    )
    {
        result.setLocation(template, t, t);
        return result;
    }
}

```

HashLiteral HashLiteral() :

```

{

```

```

Token begin, end;
Expression key, value;
ArrayList keys = new ArrayList();
ArrayList values = new ArrayList();
}
{
begin=<OPEN_BRACE>
[
key=Expression()
(<COMMA>|<COLON>)
value=Expression()
{
stringLiteralOnly(key);
keys.add(key);
values.add(value);
}
(
<COMMA>
key=Expression()
(<COMMA>|<COLON>)
value=Expression()
{
stringLiteralOnly(key);
keys.add(key);
values.add(value);
}
)*
]
end=<CLOSE_BRACE>
{
HashLiteral result = new HashLiteral(keys, values);
result.setLocation(template, begin, end);
return result;
}
}

/**
 * A production representing the ${...}
 * that outputs a variable.
 */
DollarVariable StringOutput() :
{
Expression exp;
Token begin, end;
}
{
begin=<OUTPUT_ESCAPE>
exp=Expression()

```

```

{
    notHashLiteral(exp, "scalar");
    notListLiteral(exp, "scalar");
    notBooleanLiteral(exp, "scalar");
}
end=<CLOSE_BRACE>
{
    DollarVariable result = new DollarVariable(exp, escapedExpression(exp));
    result.setLocation(template, begin, end);
    return result;
}
}

```

NumericalOutput NumericalOutput() :

```

{
    Expression exp;
    Token fmt = null, begin, end;
}
{
    begin=<NUMERICAL_ESCAPE>
    exp=Expression() { numberLiteralOnly(exp);}
    [
        <SEMICOLON>
        fmt=<ID>
    ]
    end=<CLOSE_BRACE>
    {
        NumericalOutput result;
        if (fmt != null) {
            int minFrac = -1; // -1 indicates that the value has not been set
            int maxFrac = -1;

            StringTokenizer st = new StringTokenizer(fmt.image, "mM", true);
            char type = '-';
            while (st.hasMoreTokens()) {
                String token = st.nextToken();
                try {
                    if (type != '-') {
                        switch (type) {
                            case 'm':
                                if (minFrac != -1) throw new ParseException("invalid formatting string", fmt.beginLine,
fmt.beginColumn);
                                minFrac = Integer.parseInt(token);
                                break;
                            case 'M':
                                if (maxFrac != -1) throw new ParseException("invalid formatting string", fmt.beginLine,
fmt.beginColumn);
                                maxFrac = Integer.parseInt(token);

```

```

        break;
    default:
        throw new ParseException();
    }
    type = '-';
} else if (token.equals("m")) {
    type = 'm';
} else if (token.equals("M")) {
    type = 'M';
} else {
    throw new ParseException();
}
}
catch (ParseException e) {
    String msg = getErrorStart(fmt)
        + "\nInvalid format specifier "
        + fmt.image;
    throw new ParseException(msg, fmt.beginLine, fmt.beginColumn);
}
catch (NumberFormatException e) {
    String msg = getErrorStart(fmt)
        + "\nInvalid number in the format specifier "
        + fmt.image;
    throw new ParseException(msg, fmt.beginLine, fmt.beginColumn);
}
}

if (maxFrac == -1) {
    if (minFrac == -1) {
        String msg = getErrorStart(fmt)
            + "\nInvalid format specification, at least one of m and M must be specified!";
        throw new ParseException(msg, fmt.beginLine, fmt.beginColumn);
    }
    maxFrac = minFrac;
} else if (minFrac == -1) {
    minFrac = 0;
}
if (minFrac > maxFrac) {
    String msg = getErrorStart(fmt)
        + "\nInvalid format specification, min cannot be greater than max!";
    throw new ParseException(msg, fmt.beginLine, fmt.beginColumn);
}
if (minFrac > 50 || maxFrac > 50) { // sanity check
    String msg = getErrorStart(fmt)
        + "\nCannot specify more than 50 fraction digits";
    throw new ParseException(msg, fmt.beginLine, fmt.beginColumn);
}
result = new NumericalOutput(exp, minFrac, maxFrac);

```

```

    } else { // if format != null
        result = new NumericalOutput(exp);
    }
    result.setLocation(template, begin, end);
    return result;
}
}

```

TemplateElement If() :

```

{
    Token start, end, t;
    Expression condition;
    TemplateElement block;
    IfBlock ifBlock;
    ConditionalBlock cblock;
}
{
    start=<IF>
    condition=Expression()
    <DIRECTIVE_END>
    block=OptionalBlock()
    {
        cblock = new ConditionalBlock(condition, block, true);
        cblock.setLocation(template, start, block);
        ifBlock = new IfBlock(cblock);
    }
(
    t=<ELSE_IF>
    condition=Expression()
    LooseDirectiveEnd()
    block=OptionalBlock()
    {
        cblock = new ConditionalBlock(condition, block, false);
        cblock.setLocation(template, t, block);
        ifBlock.addBlock(cblock);
    }
)*
[
    t=<ELSE>
    block=OptionalBlock()
    {
        cblock = new ConditionalBlock(null, block, false);
        cblock.setLocation(template, t, block);
        ifBlock.addBlock(cblock);
    }
]
end=<END_IF>
{

```

```

        ifBlock.setLocation(template, start, end);
        return ifBlock;
    }
}

```

AttemptBlock Attempt() :

```

{
    Token start, end;
    TemplateElement block, recoveryBlock;
}
{
    start=<ATTEMPT>
    block=OptionalBlock()
    recoveryBlock=Recover()
    (
        end=<END_RECOVER>
        |
        end=<END_ATTEMPT>
    )
    {
        AttemptBlock result = new AttemptBlock(block, recoveryBlock);
        result.setLocation(template, start, end);
        return result;
    }
}

```

RecoveryBlock Recover() :

```

{
    Token start;
    TemplateElement block;
}
{
    start=<RECOVER>
    block=OptionalBlock()
    {
        RecoveryBlock result = new RecoveryBlock(block);
        result.setLocation(template, start, block);
        return result;
    }
}

```

IteratorBlock List() :

```

{
    Expression exp;
    Token index, start, end;
    TemplateElement block;
}
{

```

```

start=<LIST> {++loopNesting;}
exp=Expression()
<AS>
index=<ID>
<DIRECTIVE_END>
block=OptionalBlock()
end=<END_LIST>
{
  --loopNesting;
  IteratorBlock result = new IteratorBlock(exp,
                                           index.image,
                                           block,
                                           false);
  result.setLocation(template, start, end);
  return result;
}
}

```

IteratorBlock ForEach() :

```

{
  Expression exp;
  Token index, start, end;
  TemplateElement block;
}
{
start=<FOREACH> {++loopNesting;}
index=<ID>
<IN>
exp=Expression()
<DIRECTIVE_END>
block=OptionalBlock()
end=<END_FOREACH>
{
  --loopNesting;
  IteratorBlock result = new IteratorBlock(exp,
                                           index.image,
                                           block,
                                           true);
  result.setLocation(template, start, end);
  return result;
}
}

```

VisitNode Visit() :

```

{
  Token start, end;
  Expression targetNode, namespaces=null;
}

```

```

{
  start=<VISIT>
  targetNode=Expression()
  [
    <USING>
    namespaces=Expression()
  ]
  end=LooseDirectiveEnd()
  {
    VisitNode result = new VisitNode(targetNode, namespaces);
    result.setLocation(template, start, end);
    return result;
  }
}

```

RecurseNode Recurse() :

```

{
  Token start, end = null;
  Expression node=null, namespaces=null;
}
{
  (
    start=<SIMPLE_RECURSE>
    |
    (
      start=<RECURSE>
      [
        node=Expression()
      ]
      [
        <USING>
        namespaces=Expression()
      ]
      end=LooseDirectiveEnd()
    )
  )
  {
    if (end == null) end = start;
    RecurseNode result = new RecurseNode(node, namespaces);
    result.setLocation(template, start, end);
    return result;
  }
}

```

FallbackInstruction FallBack() :

```

{
  Token tok;
}

```



```

{
tok=<FALLBACK>
{
if (!inMacro) {
    throw new ParseException(getErrorStart(tok)
        + "\nCannot fall back "
        + " outside a macro.",
        tok.beginLine, tok.beginColumn);

    }
FallbackInstruction result = new FallbackInstruction();
result.setLocation(template, tok, tok);
return result;
}
}

/**
 * Production used to break out of a loop or a switch block.
 */
BreakInstruction Break() :
{
    Token start;
}
{
start=<BREAK>
{
    if (loopNesting < 1 && switchNesting <1)
    {
        String msg = getErrorStart(start) + "\n"
            + start.image
            + " occurred outside a loop or a switch block.";
        throw new ParseException(msg, start.beginLine, start.beginColumn);
    }
    BreakInstruction result = new BreakInstruction();
    result.setLocation(template, start, start);
    return result;
}
}

/**
 * Production used to jump out of a macro.
 * The stop instruction terminates the rendering of the template.
 */
ReturnInstruction Return() :
{
    Token start, end=null;
    Expression exp = null;
}

```

```

{
(
start=<SIMPLE_RETURN>{end = start;}
|
start=<RETURN> exp=Expression() end=LooseDirectiveEnd()
)
{
if (inMacro) {
if (exp != null) {
String msg = getErrorStart(start)
+ "\nA macro cannot return a value";
throw new ParseException(msg, start.beginLine, start.beginColumn);
}
}
else if (inFunction) {
if (exp == null) {
String msg = getErrorStart(start)
+ "\nA function must return a value";
throw new ParseException(msg, start.beginLine, start.beginColumn);
}
}
else {
if (exp == null) {
String msg = getErrorStart(start)
+ "\nA return instruction can only occur inside a macro of function";
throw new ParseException(msg, start.beginLine, start.beginColumn);
}
}
ReturnInstruction result = new ReturnInstruction(exp);
result.setLocation(template, start, end);
return result;
}
}

```

StopInstruction Stop() :

```

{
Token start = null;
Expression exp = null;
}
{
(
start=<HALT>
|
start=<STOP> exp=Expression() LooseDirectiveEnd()
)
{
StopInstruction result = new StopInstruction(exp);
result.setLocation(template, start, start);
}
}

```

```

        return result;
    }
}

TemplateElement Nested() :
{
    Token t, end;
    ArrayList bodyParameters;
    BodyInstruction result = null;
}
{
    (
        (
            t=<SIMPLE_NESTED>
            {
                result = new BodyInstruction(null);
                result.setLocation(template, t, t);
            }
        )
        |
        (
            t=<NESTED>
            bodyParameters=PositionalArgs()
            end=LooseDirectiveEnd()
            {
                result = new BodyInstruction(bodyParameters);
                result.setLocation(template, t, end);
            }
        )
    )
    {
        if (!inMacro) {
            throw new ParseException(getErrorStart(t)
                + "\nCannot use a "
                + t.image
                + " instruction outside a macro.",
                t.beginLine, t.beginColumn);
        }
        return result;
    }
}

```

TemplateElement Flush() :

```

{
    Token t;
}
{
    t=<FLUSH>

```

```

{
    FlushInstruction result = new FlushInstruction();
    result.setLocation(template, t, t);
    return result;
}
}

```

TemplateElement Trim() :

```

{
    Token t;
    TrimInstruction result=null;
}
{
    (
        t=<TRIM> {result = new TrimInstruction(true, true);}
        |
        t=<LTRIM> {result = new TrimInstruction(true, false);}
        |
        t=<RTRIM> {result = new TrimInstruction(false, true);}
        |
        t=<NOTRIM> {result = new TrimInstruction(false, false);}
    )
    {
        result.setLocation(template, t, t);
        return result;
    }
}

```

TemplateElement Assign() :

```

{
    Token start, end;
    int scope;
    Token id=null;
    Expression nameExp, exp, nsExp=null;
    String varName;
    ArrayList assignments = new ArrayList();
    Assignment ass;
    TemplateElement block;
}
{
    (
        start=<ASSIGN> {scope = Assignment.NAMESPACE;}
        |
        start=<GLOBALASSIGN>{scope = Assignment.GLOBAL;}
        |
        start=<LOCALASSIGN> {scope = Assignment.LOCAL;}
    )
    {

```

```

scope = Assignment.LOCAL;
if (!inMacro && !inFunction) {
    String msg = getErrorStart(start)
        + "\nLocal variable assigned outside a macro.";
    throw new ParseException(msg, start.beginLine, start.beginColumn);
}
}
)
nameExp=IdentifierOrStringLiteral()
{
    varName = (nameExp instanceof StringLiteral) ? ((StringLiteral) nameExp).getAsString() : nameExp.toString();
}
((
<EQUALS>
exp=Expression()
{
    ass = new Assignment(varName, exp, scope);
    ass.setLocation(template, nameExp, exp);
    assignments.add(ass);
}
(
LOOKAHEAD([<COMMA>](<ID>|<STRING_LITERAL>)<EQUALS>)
[<COMMA>]
nameExp=IdentifierOrStringLiteral()
{
    varName = (nameExp instanceof StringLiteral) ? ((StringLiteral) nameExp).getAsString() :
nameExp.toString();
}
<EQUALS>
exp=Expression()
{
    ass = new Assignment(varName, exp, scope);
    ass.setLocation(template, nameExp, exp);
    assignments.add(ass);
}
)*
[
id=<IN>
nsExp=Expression() {if (scope != Assignment.NAMESPACE) throw new ParseException(getErrorStart(id) +
"\nCannot assign to namespace here.", id.beginLine, id.beginColumn);}
]
end=LooseDirectiveEnd()
{
    AssignmentInstruction ai = new AssignmentInstruction(scope);
    for (int i = 0; i < assignments.size(); i++) {
        ai.addAssignment((Assignment) assignments.get(i));
    }
    ai.setNamespaceExp(nsExp);
}

```

```

        ai.setLocation(template, start, end);
        return ai;
    }
)
|
(
[
    id=<IN>
    nsExp=Expression() {if (scope != Assignment.NAMESPACE) throw new ParseException(getErrorStart(id) +
"\nCannot assign to namespace here.", id.beginLine, id.beginColumn);}
]
<DIRECTIVE_END>
block=OptionalBlock()
(
    end=<END_LOCAL> {if (scope != Assignment.LOCAL) throw new ParseException(getErrorStart(end) +
"\nMismatched assignment tags.", end.beginLine, end.beginColumn);}
|
    end=<END_ASSIGN> {if (scope != Assignment.NAMESPACE) throw new
ParseException(getErrorStart(end) + "\nMismatched assignment tags.", end.beginLine, end.beginColumn);}
|
    end=<END_GLOBAL> {if (scope != Assignment.GLOBAL) throw new ParseException(getErrorStart(end) +
"\nMismatched assignment tags", end.beginLine, end.beginColumn);}
)
{
    BlockAssignment ba = new BlockAssignment(block, varName, scope, nsExp);
    ba.setLocation(template, start, end);
    return ba;
}
))
}

```

Include Include() :

```

{
    Expression nameExp;
    Token att, start, end;
    Expression exp, parseExp = null, encodingExp = null;
}
{
    start=<INCLUDE>
    nameExp=Expression()
    [<SEMICOLON>]
    (
        att=<ID>
        <EQUALS>
        exp=Expression()
        {
            String attString = att.image;
            if (attString.equalsIgnoreCase("parse")) {

```

```

        parseExp = exp;
    }
    else if (attString.equalsIgnoreCase("encoding")) {
        encodingExp = exp;
    }
    else {
        String msg = getErrorStart(att)
            + "\nexpecting parse= or encoding= to be specified.";
        throw new ParseException(msg, att.beginLine, att.beginColumn);
    }
}
)*
end=LooseDirectiveEnd()
{
    Include result = new Include(template, nameExp, encodingExp, parseExp);
    result.setLocation(template, start, end);
    return result;
}
}

```

LibraryLoad Import() :

```

{
    Token start, end, ns;
    Expression nameExp;
}
{
    start=<IMPORT>
    nameExp=Expression()
    <AS>
    ns=<ID>
    end=LooseDirectiveEnd()
    {
        LibraryLoad result = new LibraryLoad(template, nameExp, ns.image);
        result.setLocation(template, start, end);
        template.addImport(result);
        return result;
    }
}

```

Macro Macro() :

```

{
    Token arg, start, end;
    Expression nameExp;
    String name;
    ArrayList argNames = new ArrayList();
    HashMap args = new HashMap();
    ArrayList defNames = new ArrayList();
    Expression defValue=null;
}

```

```

TemplateElement block;
boolean isFunction = false, hasDefaults=false;
boolean isCatchAll = false;
String catchAll = null;
}
{
(
  start=<MACRO>
  |
  start=<FUNCTION> {isFunction = true;}
)
{
  if (inMacro || inFunction) {
    throw new ParseException(getErrorStart(start)
      + "\nMacros cannot be nested.", start.beginLine, start.endLine);
  }
  if (isFunction) inFunction = true; else inMacro = true;
}
nameExp=IdentifierOrStringLiteral()
{
  name = (nameExp instanceof StringLiteral) ? ((StringLiteral) nameExp).getAsString() : nameExp.toString();
}
[<OPEN_PAREN>]
(
  arg=<ID> {defValue = null;}
  [<ELLIPSIS> { isCatchAll = true; }]
  [
    <EQUALS>
    defValue=Expression()
    {
      defNames.add(arg.image);
    }
  ]
  hasDefaults = true;
  [<COMMA>]
)
{
  if (catchAll != null) {
    throw new ParseException(getErrorStart(arg)
      + "\nThere may only be one \"catch-all\" parameter in a macro declaration, "
      + "and it must be the last parameter.", arg.beginLine, arg.endLine);
  }
  if (isCatchAll) {
    if (defValue != null) {
      throw new ParseException(getErrorStart(arg)
        + "\n\"Catch-all\" macro parameter may not have a default value.",
        arg.beginLine, arg.endLine);
    }
    catchAll = arg.image;
  }
}

```



```

    } else {
        argNames.add(arg.image);
        if (hasDefaults && defValue == null) {
            throw new ParseException(getErrorStart(arg)
                + "\nIn a macro declaration, parameters without a default value "
                + "must all occur before the parameters with default values.",
                arg.beginLine, arg.endLine);
        }
        args.put(arg.image, defValue);
    }
}
)*
[<CLOSE_PAREN>]
<DIRECTIVE_END>
block=OptionalBlock()
(
    end=<END_MACRO> { if(isFunction) throw new ParseException(getErrorStart(start) + "\nExpected function
end tag here.", start.beginLine, start.endLine); }
|
    end=<END_FUNCTION> { if(!isFunction) throw new ParseException(getErrorStart(start) + "\nExpected
macro end tag here.", start.beginLine, start.endLine); }
)
{
    inMacro = inFunction = false;
    Macro result = new Macro(name, argNames, args, block);
    result.setCatchAll(catchAll);
    result.isFunction = isFunction;
    result.setLocation(template, start, end);
    template.addMacro(result);
    return result;
}
}

```

CompressedBlock Compress() :

```

{
    TemplateElement block;
    Token start, end;
}
{
    start=<COMPRESS>
    block=OptionalBlock()
    end=<END_COMPRESS>
    {
        CompressedBlock cb = new CompressedBlock(block);
        cb.setLocation(template, start, end);
        return cb;
    }
}

```

```
}
```

TemplateElement UnifiedMacroTransform() :

```
{
    Token start=null, end, t;
    HashMap namedArgs = null;
    ArrayList positionalArgs = null, bodyParameters = null;
    String directiveName = null;
    TemplateElement nestedBlock = null;
    Expression exp;
}
{
    start=<UNIFIED_CALL>
    exp=Expression()
    {
        if (exp instanceof Identifier || (exp instanceof Dot && ((Dot) exp).onlyHasIdentifiers())) {
            directiveName = exp.getCanonicalForm();
        }
    }
    [<TERMINATING_WHITESPACE>]
    (
        LOOKAHEAD(<ID><EQUALS>)
        namedArgs = NamedArgs()
        |
        positionalArgs=PositionalArgs()
    )
    [
        <SEMICOLON>{bodyParameters = new ArrayList();}
        [
            [<TERMINATING_WHITESPACE>]t=<ID> {bodyParameters.add(t.image);}
            (
                [<TERMINATING_WHITESPACE><COMMA>
                [<TERMINATING_WHITESPACE>t = <ID>{bodyParameters.add(t.image);}
            )*
        ]
    ]
    (
        end=<EMPTY_DIRECTIVE_END>
        |
        (
            <DIRECTIVE_END>
            nestedBlock=OptionalBlock()
            end=<UNIFIED_CALL_END>
        )
    {
        String s = end.image.substring(3);
        s = s.substring(0, s.length() -1).trim();
        if (s.length() >0 && !s.equals(directiveName)) {
            String msg = getErrorStart(end);
```

```

        if (directiveName == null) {
            throw new ParseException(msg + "\nExpecting </@>", end.beginLine, end.beginColumn);
        }
        else {
            throw new ParseException(msg + "\nExpecting </@> or </@" + directiveName + ">", end.beginLine,
end.beginColumn);
        }
    }
}
)
)
{
    TemplateElement result = (positionalArgs != null) ? new UnifiedCall(exp, positionalArgs, nestedBlock,
bodyParameters)
        : new UnifiedCall(exp, namedArgs, nestedBlock, bodyParameters);
    result.setLocation(template, start, end);
    return result;
}
}

```

TemplateElement Call() :

```

{
    Token start, end, id;
    HashMap namedArgs = null;
    ArrayList positionalArgs = null;
    String macroName= null;
}
{
    start=<CALL>
    id=<ID> {macroName = id.image;}
    (
        LOOKAHEAD(<ID><EQUALS>)
        namedArgs=NamedArgs()
        |
        (
            [
                LOOKAHEAD(<OPEN_PAREN>)
                <OPEN_PAREN>
            ]
            positionalArgs=PositionalArgs()
            [<CLOSE_PAREN>]
        )
    )
    end=LooseDirectiveEnd()
    {
        UnifiedCall result = null;
        if (positionalArgs != null) {
            result = new UnifiedCall(new Identifier(macroName), positionalArgs, null, null);
        }
    }
}

```

```

    }
    else {
        result = new UnifiedCall(new Identifier(macroName), namedArgs, null, null);
    }
    result.legacySyntax = true;
    result.setLocation(template, start, end);
    return result;
}
}

```

HashMap NamedArgs() :

```

{
    HashMap result = new HashMap();
    Token t;
    Expression exp;
}
{
    (
        t=<ID>
        <EQUALS>
        {
            token_source.SwitchTo(token_source.NAMED_PARAMETER_EXPRESSION);
            token_source.inInvocation = true;
        }
        exp=Expression()
        {
            result.put(t.image, exp);
        }
    )+
    {
        token_source.inInvocation = false;
        return result;
    }
}

```

ArrayList PositionalArgs() :

```

{
    ArrayList result = new ArrayList();
    Expression arg;
}
{
    [
        arg=Expression() {result.add(arg);}
        (
            [<COMMA>]
            arg=Expression() {result.add(arg);}
        )*
    ]
}

```

```
{
    return result;
}
}
```

Comment Comment() :

```
{
    Token start, end;
    StringBuffer buf = new StringBuffer();
}
{
    (
        start=<COMMENT>
        |
        start=<TERSE_COMMENT>
    )
    end=UnparsedContent(buf)
    {
        Comment result = new Comment(buf.toString());
        result.setLocation(template, start, end);
        return result;
    }
}
```

TextBlock NoParse() :

```
{
    Token start, end;
    StringBuffer buf = new StringBuffer();
}
{
    start=<NOPARSE>
    end=UnparsedContent(buf)
    {
        TextBlock result = new TextBlock(buf.toString(), true);
        result.setLocation(template, start, end);
        return result;
    }
}
```

TransformBlock Transform() :

```
{
    Token start, end, argName;
    Expression exp, argExp;
    TemplateElement content = null;
    HashMap args = null;
}
```

```

{
  start=<TRANSFORM>
  exp=Expression()
  [<SEMICOLON>]
  (
    argName=<ID>
    <EQUALS>
    argExp = Expression()
    {
      if (args == null) args = new HashMap();
      args.put(argName.image, argExp);
    }
  )*
  (
    end=<EMPTY_DIRECTIVE_END>
    |
    (<DIRECTIVE_END>content=OptionalBlock()end=<END_TRANSFORM>)
  )
  {
    TransformBlock result = new TransformBlock(exp, args, content);
    result.setLocation(template, start, end);
    return result;
  }
}

```

SwitchBlock Switch() :

```

{
  SwitchBlock switchBlock;
  Case caseIns;
  Expression switchExp;
  Token start, end;
  boolean defaultFound = false;
}
{
  start=<SWITCH>
  switchExp=Expression()
  <DIRECTIVE_END>
  {
    ++switchNesting;
    switchBlock = new SwitchBlock(switchExp);
  }
  (
    LOOKAHEAD(2)
    caseIns=Case()
    {
      if (caseIns.isDefault) {
        if (defaultFound) {
          String msg = getErrorStart(start)

```

```

        + "\nYou can only have one default case in a switch statement";
        throw new ParseException(msg, start.beginLine, start.beginColumn);
    }
    defaultFound = true;
}
    switchBlock.addCase(caseIns);
}
)*
[<WHITESPACE>]
end=<END_SWITCH>
{
    --switchNesting;
    switchBlock.setLocation(template, start, end);
    return switchBlock;
}
}

```

Case Case() :

```

{
    Expression exp = null;
    TemplateElement block;
    boolean isDefault = false;
    Token start;
}
{
[<WHITESPACE>]
(
    start=<CASE>exp=Expression()<DIRECTIVE_END>
    |
    start=<DEFAULT>{isDefault = true;}
)
block=OptionalBlock()
{
    Case result = new Case(exp, block, isDefault);
    result.setLocation(template, start, block);
    return result;
}
}

```

EscapeBlock Escape() :

```

{
    Token variable, start, end;
    Expression escapeExpr;
    TemplateElement content;
}
{
    start=<ESCAPE>
    variable=<ID>

```

```

<AS>
escapeExpr=Expression()
<DIRECTIVE_END>
{
    EscapeBlock result = new EscapeBlock(variable.image, escapeExpr, escapedExpression(escapeExpr));
    escapes.addFirst(result);
}
content=OptionalBlock()
{
    result.setContent(content);
    escapes.removeFirst();
}
end=<END_ESCAPE>
{
    result.setLocation(template, start, end);
    return result;
}
}

```

```

NoEscapeBlock NoEscape() :
{
    Token start, end;
    TemplateElement content;
}
{
    start=<NOESCAPE>
    {
        if(escapes.isEmpty()) {
            String msg = getErrorStart(start)
                + "\nnoescape with no matching escape encountered.";
            throw new ParseException(msg, start.beginLine, start.beginColumn);
        }
        Object escape = escapes.removeFirst();
    }
    content=OptionalBlock()
    end=<END_NOESCAPE>
    {
        escapes.addFirst(escape);
        NoEscapeBlock result = new NoEscapeBlock(content);
        result.setLocation(template, start, end);
        return result;
    }
}
}

```

```

/**
 * Production to terminate potentially empty elements. Either a ">" or "/>"
 */

```


Token LooseDirectiveEnd() :

```
{
    Token t;
}
{
    (
        t=<DIRECTIVE_END>
        |
        t=<EMPTY_DIRECTIVE_END>
    )
    {
        return t;
    }
}
```

PropertySetting Setting() :

```
{
    Token start, end, key;
    Expression value;
}
{
    start=<SETTING>
    key=<ID>
    <EQUALS>
    value=Expression()
    end=LooseDirectiveEnd()
    {
        PropertySetting result = new PropertySetting(key.image, value);
        result.setLocation(template, start, end);
        return result;
    }
}
```

/**

* A production for FreeMarker directives.

*/

TemplateElement FreemarkerDirective() :

```
{
    TemplateElement tp;
}
{
    (
        tp=If()
        |
        tp=List()
        |
        tp=ForEach()
        |
    )
}
```

```
tp=Assign()  
|  
tp=Include()  
|  
tp=Import()  
|  
tp=Macro()  
|  
tp=Compress()  
|  
tp=UnifiedMacroTransform()  
|  
tp=Call()  
|  
tp=Comment()  
|  
tp=NoParse()  
|  
tp=Transform()  
|  
tp=Switch()  
|  
tp=Setting()  
|  
tp=Break()  
|  
tp=Return()  
|  
tp=Stop()  
|  
tp=Flush()  
|  
tp=Trim()  
|  
tp=Nested()  
|  
tp=Escape()  
|  
tp=NoEscape()  
|  
tp=Visit()  
|  
tp=Recurse()  
|  
tp=FallBack()  
|  
tp=Attempt()  
)
```

```

    {
        return tp;
    }
}

/**
 * Production for a block of raw text
 * i.e. text that contains no
 * FreeMarker directives.
 */

TextBlock PCData() :
{
    StringBuffer buf = new StringBuffer();
    Token t=null, start=null, prevToken = null;
}
{
    (
        LOOKAHEAD(<WHITESPACE>|<PRINTABLE_CHARS>|<FALSE_ALERT>)
        (
            {prevToken = t;}
            t=<WHITESPACE>
        |
            t=<PRINTABLE_CHARS>
        |
            t=<FALSE_ALERT>
        )
        {
            buf.append(t.image);
            if (start == null) start = t;
            {if (prevToken != null) prevToken.next = null;}
        }
    )+
    {
        if (stripText && contentNesting == 1)
            return TextBlock.EMPTY_BLOCK;

        TextBlock result = new TextBlock(buf.toString(), false);
        result.setLocation(template, start, t);
        return result;
    }
}

/**
 * Production for dealing with unparsed content,
 * i.e. what is inside a comment or noparse tag.
 * It returns the ending token. The content
 * of the tag is put in buf.

```

```
*/
```

```
Token UnparsedContent(StringBuffer buf) :
```

```
{
    Token t;
}
{
    ((t=<KEEP_GOING> | t=<MAYBE_END> | t=<TERSE_COMMENT_END> |
t=<LONE_LESS_THAN_OR_DASH>)
    {
        buf.append(t.image);
    })+
    {
        buf.setLength(buf.length() - t.image.length());
        return t;
    }
}
```

```
TemplateElement Content() :
```

```
{
    MixedContent nodes = new MixedContent();
    TemplateElement elem, begin=null;
    contentNesting++;
}
{
    (
        LOOKAHEAD(1) // Just tells javacc that we know what we're doing.
        (
            elem=PCData()
            |
            elem=StringOutput()
            |
            elem=NumericalOutput()
            |
            elem=FreemarkerDirective()
        )
        {
            if (begin == null) {
                begin = elem;
            }
            nodes.addElement(elem);
        }
    )+
    {
        contentNesting--;
        nodes.setLocation(template, begin, elem);
    }
return nodes;
}
```

```

}

/**
 * A production freemarker text that may contain
 * ${...} and #{...} but no directives.
 */

TemplateElement FreeMarkerText() :
{
    MixedContent nodes = new MixedContent();
    TemplateElement elem, begin = null;
}
{
    (
        (
            elem=PCData()
            |
            elem=StringOutput()
            |
            elem=NumericalOutput()
        )
        {
            if (begin == null) {
                begin = elem;
            }
            nodes.addElement(elem);
        }
    )+
    {
        nodes.setLocation(template, begin, elem);
        return nodes;
    }
}

/**
 * A production for a block of optional content.
 * Returns an empty Text block if there is no
 * content.
 */

TemplateElement OptionalBlock() :
{
    TemplateElement tp = TextBlock.EMPTY_BLOCK;
}
{
    [
        LOOKAHEAD(1) // has no effect but to get rid of a spurious warning.
        tp=Content()
    ]
}

```

```

    ]
    {
        return tp;
    }
}

void HeaderElement() :
{
    Token key;
    Expression exp = null;
}
{
    [<WHITESPACE>]
    (
        <TRIVIAL_FTL_HEADER>
        |
        (
            <FTL_HEADER>
            (
                key=<ID>
                <EQUALS>
                exp=Expression()
            {
                String ks = key.image;
                TemplateModel value = null;
                try {
                    value = exp.getAsTemplateModel(null);
                } catch (Exception e) {
                    throw new ParseException("Could not evaluate expression: "
                        + exp.getCanonicalForm() + " "
                        + exp.getStartLocation()
                        + "\nUnderlying cause: " +
                        e.getMessage(), exp);
                }
            }
            String vs = null;
            if (value instanceof TemplateScalarModel) {
                try {
                    vs = ((TemplateScalarModel) exp).getAsString();
                } catch (TemplateModelException tme) {}
            }
            if (template != null) {
                if (ks.equalsIgnoreCase("encoding")) {
                    if (vs == null) {
                        throw new ParseException("expecting encoding string here: "
                            + exp.getStartLocation(), exp);
                    }
                }
                String encoding = template.getEncoding();
                if (encoding != null && !encoding.equals(vs)) {

```

```

        throw new Template.WrongEncodingException(vs);
    }
}
else if (ks.equalsIgnoreCase("STRIP_WHITESPACE")) {
    this.stripWhitespace = getBoolean(exp);
}
else if (ks.equalsIgnoreCase("STRIP_TEXT")) {
    this.stripText = getBoolean(exp);
}
else if (ks.equalsIgnoreCase("STRICT_SYNTAX")) {
    this.token_source.strictEscapeSyntax = getBoolean(exp);
}
else if (ks.equalsIgnoreCase("ns_prefixes")) {
    if (!(value instanceof TemplateHashModelEx)) {
        throw new ParseException("Expecting a hash of prefixes to namespace URI's here. " +
exp.getStartLocation(), exp);
    }
    TemplateHashModelEx prefixMap = (TemplateHashModelEx) value;
    try {
        TemplateCollectionModel keys = prefixMap.keys();
        for (TemplateModelIterator it = keys.iterator(); it.hasNext();) {
            String prefix = ((TemplateScalarModel) it.next()).getAsString();
            TemplateModel valueModel = prefixMap.get(prefix);
            if (!(valueModel instanceof TemplateScalarModel)) {
                throw new ParseException("Non-string value in prefix to namespace hash. " +
exp.getStartLocation(), exp);
            }
            String nsURI = ((TemplateScalarModel) valueModel).getAsString();
            try {
                template.addPrefixNSMapping(prefix, nsURI);
            } catch (IllegalArgumentException iae) {
                throw new ParseException(iae.getMessage() + " " + exp.getStartLocation(), exp);
            }
        }
    } catch (TemplateModelException tme) {
    }
}
else if (ks.equalsIgnoreCase("attributes")) {
    if (!(value instanceof TemplateHashModelEx)) {
        throw new ParseException("Expecting a hash of attribute names to values here. " +
exp.getStartLocation(), exp);
    }
    TemplateHashModelEx attributeMap = (TemplateHashModelEx) value;
    try {
        TemplateCollectionModel keys = attributeMap.keys();
        for (TemplateModelIterator it = keys.iterator(); it.hasNext();) {
            String attName = ((TemplateScalarModel) it.next()).getAsString();
            Object attValue = DeepUnwrap.unwrap(attributeMap.get(attName));

```

```

        template.setCustomAttribute(attName, attValue);
    }
} catch (TemplateModelException tme) {
}
}
else {
    throw new ParseException("Unknown FTL header parameter: " + key.image,
        key.beginLine, key.beginColumn);
}
}
}
)*
)
LooseDirectiveEnd()
)
}

```

Map ParamList() :

```

{
    Identifier id;
    Expression exp;
    Map result = new HashMap();
}
{
    (
        id=Identifier()
        <EQUALS>
        exp=Expression() {result.put(id.toString(), exp);}
        [<COMMA>]
    )+
    {
        return result;
    }
}

```

/**

* Root production to be used when parsing

* an entire file.

*/

TemplateElement Root() :

```

{
    TemplateElement doc;
}
{
    [
        LOOKAHEAD([<WHITESPACE>])(<TRIVIAL_FTL_HEADER>|<FTL_HEADER>))
        HeaderElement()
    ]
}

```



```

]
doc=OptionalBlock()
<EOF>
{
  doc.setParentRecursively(null);
  return doc.postParseCleanup(stripWhitespace);
}
}

```

Found in path(s):

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/core/FMParser.jj

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

/*

* Copyright (c) 2005 The Visigoth Software Society. All rights

* reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

*

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

*

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in

* the documentation and/or other materials provided with the

* distribution.

*

* 3. The end-user documentation included with the redistribution, if

* any, must include the following acknowledgement:

* "This product includes software developed by the

* Visigoth Software Society (<http://www.visigoths.org/>)."

* Alternately, this acknowledgement may appear in the software itself,

* if and wherever such third-party acknowledgements normally appear.

*

* 4. Neither the name "FreeMarker", "Visigoth", nor any of the names of the

* project contributors may be used to endorse or promote products derived

* from this software without prior written permission. For written

* permission, please contact visigoths@visigoths.org.

*

* 5. Products derived from this software may not be called "FreeMarker" or "Visigoth"

* nor may "FreeMarker" or "Visigoth" appear in their names

* without prior written permission of the Visigoth Software Society.

*

* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED

* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE

* DISCLAIMED. IN NO EVENT SHALL THE VISIGOTH SOFTWARE SOCIETY OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

* =====

*

* This software consists of voluntary contributions made by many
* individuals on behalf of the Visigoth Software Society. For more
* information on the Visigoth Software Society, please see
* <http://www.visigoths.org/>
*/

Found in path(s):

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-
jar/freemarker/cache/StringTemplateLoader.java

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

Copyright 2003-2007 Sun Microsystems, Inc. All rights reserved.

The contents of this file are subject to the terms of either the
GNU General Public License Version 2 only ("GPL") or the Common
Development and Distribution License("CDDL") (collectively, the
"License"). You may not use this file except in compliance with
particular file as subject to the "Classpath" exception as
provided by Sun in the GPL Version 2 section of the License file

Found in path(s):

* /opt/cola/permits/1173793086_1623767916.82/0/freemarker-2-3-13-sources-jar/freemarker/ext/jsp/web-
jsptaglibrary_2_1.xsd

1.3 activemq 5.8.0

1.3.1 Available under license :

ActiveMQ :: All JAR bundle

Copyright 2005-2013 The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (<http://www.apache.org/>).

Apache License

Version 2.0, January 2004

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

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

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

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

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

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

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

1.4 commons-logging 1.1.1

1.4.1 Available under license :

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a

copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

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

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

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

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

or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

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

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

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

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of

this License, without any additional terms or conditions.

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

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

END OF TERMS AND CONDITIONS

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

To apply the Apache License to your work, attach the following

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

Copyright [yyyy] [name of copyright owner]

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

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

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

```
// -----  
// NOTICE file corresponding to the section 4d of The Apache License,  
// Version 2.0, in this case for Commons Logging  
// -----
```

Commons Logging
Copyright 2001-2007 The Apache Software Foundation

This product includes/uses software(s) developed by 'an unknown organization'

- Unnamed - avalon-framework:avalon-framework:jar:4.1.3
- Unnamed - log4j:log4j:jar:1.2.12
- Unnamed - logkit:logkit:jar:1.0.1

1.5 protobuf-java 2.6.0

1.6 hawtbuf 1.9

1.7 mysql-jdbc 8.0.11

1.7.1 Available under license :

Found license 'General Public License 2.0' in '* Copyright (c) 2012, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with * Does this plugin require the connection itself to be confidential boolean requiresConfidentiality();'

Found license 'General Public License 2.0' in '* Copyright (c) 2012, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with public boolean requiresConfidentiality() {'

Found license 'General Public License 2.0' in '* Copyright (c) 2010, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with'

Found license 'General Public License 2.0' in '* Copyright (c) 2015, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with * Check if given plugin requires confidentiality, but connection is without SSL private void checkConfidentiality(AuthenticationPlugin<?> plugin) { if (plugin.requiresConfidentiality() && !this.protocol.getSocketConnection().isSSEstablished()) { checkConfidentiality(plugin); checkConfidentiality(plugin); checkConfidentiality(plugin);'

Found license 'General Public License 2.0' in '* Copyright (c) 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with'

Found license 'General Public License 2.0' in '* Copyright (c) 2004, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with'

Found license 'General Public License 2.0' in '* Copyright (c) 2016, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with'

Found license 'General Public License 2.0' in '* Copyright (c) 2017, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with'

Found license 'General Public License 2.0' in '* Copyright (c) 2015, 2017, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with'

Found license 'General Public License 2.0' in '* Copyright (c) 2013, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with'

Found license 'General Public License 2.0' in '* Copyright (c) 2014, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with'

Found license 'General Public License 2.0' in '* Copyright (c) 2012, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with'

Found license 'General Public License 2.0' in '* Copyright (c) 2015, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a

copy of the GNU General Public License along with'

Found license 'General Public License 2.0' in '* Copyright (c) 2002, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with'

Found license 'General Public License 2.0' in '* Copyright (c) 2011, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with'

Found license 'General Public License 2.0' in '* Copyright (c) 2002, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with * The ASF licenses this file to You under the Apache License, Version 2.0 * (the "License"); you may not use this file except in compliance with * the License. You may obtain a copy of the License at * <http://www.apache.org/licenses/LICENSE-2.0> * distributed under the License is distributed on an "AS IS" BASIS,'

Found license 'General Public License 2.0' in '* Copyright (c) 2005, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with'

Found license 'General Public License 2.0' in '* Copyright (c) 2007, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with'

Found license 'General Public License 2.0' in '* Copyright (c) 2008, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with'

Found license 'General Public License 2.0' in '* Copyright (c) 2009, 2018, Oracle and/or its affiliates. All rights reserved. * This program is free software; you can redistribute it and/or modify it under * the terms of the GNU

General Public License, version 2.0, as published by the * limited to OpenSSL) that is licensed under separate terms, as designated in a * This program is distributed in the hope that it will be useful, but WITHOUT * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS * FOR A PARTICULAR PURPOSE. See the GNU General Public License, version 2.0, * for more details. * You should have received a copy of the GNU General Public License along with'

1.8 spring-test 2.5.6

1.8.1 Available under license :

Apache License

Version 2.0, January 2004

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

as of the date such litigation is filed.

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

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

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

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

Notwithstanding the above, nothing herein shall supersede or modify

the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

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

END OF TERMS AND CONDITIONS

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

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include

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

Copyright [yyyy] [name of copyright owner]

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

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

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

1.9 hibernate-validator 5.2.0.Final

1.9.1 Available under license :

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

```
/*  
* Hibernate Validator, declare and validate application constraints  
*  
* License: Apache License, Version 2.0  
* See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE-2.0>.  
*/
```

Found in path(s):

```
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/cfg/context/ConstraintMappingContextImplBase.java  
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/xml/GroupConversionBuilder.java  
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/size/SizeValidatorForArrayOfFloat.java  
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/size/package-info.java  
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/MinValidatorForNumber.java  
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/context/CrossParameterTarget.java  
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/spi/cfg/ConstraintMappingContributor.java
```

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/aggregated/rule/ParallelMethodsMustNotDefineParameterConstraints.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/raw/AbstractConstrainedElement.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/privilegedactions/Unmarshal.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/messageinterpolation/parser/TokenCollector.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/messageinterpolation/el/RootResolver.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/package-info.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/groups/DefaultValidationOrder.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/aggregated/ParameterMetaData.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/past/PastValidatorForChronoZonedDateTime.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/logging/LoggerFactory.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/ValidatorContextImpl.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/hv/Mod10CheckValidator.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/hv/Mod11CheckValidator.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/hv/ScriptAssertValidator.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/valuehandling/package-info.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/privilegedactions/GetResource.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/defs/NotEmptyDef.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/scriptengine/ScriptEvaluator.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/context/ConstructorConstraintMappingContext.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/context/ReturnValueConstraintMappingContext.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/privilegedactions/NewJaxbContext.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/privilegedactions/SetAccessibility.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/ExecutableHelper.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-

jar/org/hibernate/validator/cfg/context/AnnotationProcessingOptions.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/ValidatorFactoryImpl.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/past/PastValidatorForReadableInstant.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/ValidatorImpl.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/cfg/context/TypeConstraintMappingContextImpl.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/hv/br/CNPJValidator.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/messageinterpolation/parser/ELState.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/context/ParameterTarget.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/spi/group/DefaultGroupSequenceProvider.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/future/FutureValidatorForOffsetDateTime.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/defs/AssertTrueDef.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/metadata/aggregated/GroupConversionHelper.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/defs/EmailDef.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/MessageInterpolatorContext.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/size/SizeValidatorForArrayOfInt.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/defs/FutureDef.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/util/privilegedactions/GetDeclaredConstructors.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/messageinterpolation/parser/EscapedState.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/messageinterpolation/FormatterWrapper.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/constraints/br/CNPJ.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/util/classhierarchy/Filter.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/defs/CreditCardNumberDef.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/constraintvalidators/RegexpURLValidator.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/defs/MinDef.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-

jar/org/hibernate/validator/internal/engine/messageinterpolation/parser/Token.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/cfg/context/ReturnValueConstraintMappingContextImpl.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/cfg/context/ConstructorTarget.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/util/privilegedactions/GetDeclaredField.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/metadata/raw/ConstrainedExecutable.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/spi/time/TimeProvider.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/spi/constraintdefinition/package-info.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/util/ModUtil.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/cfg/context/DefaultConstraintMapping.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/engine/messageinterpolation/TermResolver.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/cfg/defs/DigitsDef.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/metadata/aggregated/rule/OverridingMethodMustNotAlterParameterConstraints.j
 ava
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/engine/valuehandling/JavaFXPropertyValueUnwrapper.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/engine/messageinterpolation/parser/MessageDescriptorFormatException.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/resourceloading/AggregateResourceBundleLocator.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/constraintvalidators/hv/EmailValidator.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/metadata/BeanMetaDataManager.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/constraintvalidators/hv/LengthValidator.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/metadata/core/ConstraintHelper.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/constraintvalidators/hv/ModCheckBase.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/cfg/context/PropertyConstraintMappingContextImpl.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/metadata/descriptor/PropertyDescriptorImpl.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/xml/ConstrainedParameterBuilder.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/metadata/raw/ConstrainedElement.java

- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/DecimalMinValidatorForCharSequence.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/defs/NotNullDef.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/size/SizeValidatorForArraysOfLong.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/HibernateValidatorFactory.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/DecimalMaxValidatorForNumber.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/parameternameprovider/package-info.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/aggregated/UnconstrainedEntityMetaDataSingleton.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/defs/ScriptAssertDef.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/DigitsValidatorForCharSequence.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/provider/MetaDataProviderKeyedByClassName.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/descriptor/ParameterDescriptorImpl.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/HibernateValidatorConfiguration.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/xml/ConstrainedTypeBuilder.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/future/FutureValidatorForInstant.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/valuehandling/package-info.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/constraints/EAN.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/groups/ValidationOrderGenerator.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/GenericConstraintDef.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/parameternameprovider/ReflectionParameterNameProvider.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/context/ParameterConstraintMappingContext.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/aggregated/MetaDataBuilder.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/constraintvalidation/ConstraintValidatorContextImpl.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/constraintvalidation/ConstraintValidatorManager.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/valuehandling/TypeResolverBasedValueUnwrapper.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/future/package-info.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/privilegedactions/GetResources.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/messageinterpolation/parser/BeginState.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/descriptor/GroupConversionDescriptorImpl.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/context/MethodTarget.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/cfg/context/ConfiguredConstraint.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/privilegedactions/NewInstance.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/messageinterpolation/parser/InterpolationTermState.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/ValueContext.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/logging/Messages.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/messageinterpolation/parser/TokenIterator.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/future/FutureValidatorForDate.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/privilegedactions/GetClassLoader.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/constraintdefinition/ConstraintDefinitionBuilderImpl.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/ServiceLoaderBasedConstraintDefinitionContributor.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/constraints/br/CPF.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/raw/ConfigurationSource.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/constraintvalidation/ConstraintTree.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/hv/EANValidator.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/resolver/CachingTraversableResolverForSingleValidation.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/messageinterpolation/HibernateMessageInterpolatorContext.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/AssertTrueValidator.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/resolver/JPATraversableResolver.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/size/SizeValidatorForArraysOfBoolean.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/messageinterpolation/parser/ParserState.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/raw/ConstrainedField.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/StringHelper.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/xml/ResourceLoaderHelper.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/aggregated/ConstraintMetaData.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/context/MethodConstraintMappingContext.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/privilegedactions/GetResolvedMemberMethods.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/xml/ValidationXmlParser.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/descriptor/ExecutableDescriptorImpl.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/valuehandling/OptionalValueUnwrapper.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/hv/LuhnCheckValidator.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/privilegedactions/GetConstraintValidatorList.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/messageinterpolation/el/MapBasedFunctionMapper.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/future/FutureValidatorForReadablePartial.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/ConstraintDef.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/constraints/Length.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/xml/XmlParserHelper.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/constraints/ModCheck.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/groups/Sequence.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/messageinterpolation/el/SimpleELContext.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/defs/PastDef.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/cfg/context/ParameterConstraintMappingContextImpl.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/size/SizeValidatorForArrayOfByte.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/defs/AssertFalseDef.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/resourceloading/CachingResourceBundleLocator.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/aggregated/rule/ReturnValueMayOnlyBeMarkedOnceAsCascadedPerHierarchyLine.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/DecimalMinValidatorForNumber.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/ConstraintMapping.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/size/SizeValidatorForMap.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/past/PastValidatorForDate.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/context/PropertyTarget.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/defs/MaxDef.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/aggregated/PropertyMetaData.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/MinValidatorForCharSequence.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/defs/RangeDef.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/context/GroupConversionTargetContext.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/core/MetaConstraint.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/descriptor/ReturnValueDescriptorImpl.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/facets/Cascadable.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/cfg/context/CrossParameterConstraintMappingContextImpl.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/constraints/SafeHtml.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/resourceloading/PlatformResourceBundleLocator.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/context/TypeConstraintMappingContext.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/defs/Mod10CheckDef.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/defs/DecimalMinDef.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/size/SizeValidatorForCharSequence.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/context/ReturnValueTarget.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-

jar/org/hibernate/validator/internal/util/logging/Log.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/engine/messageinterpolation/parser/MessageState.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/spi/resourceloading/ResourceBundleLocator.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/metadata/aggregated/BeanMetaDataImpl.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/metadata/core/AnnotationProcessingOptions.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/constraintvalidators/bv/past/PastValidatorForCalendar.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/metadata/facets/Validatable.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/constraintvalidators/bv/past/PastValidatorForReadablePartial.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/constraints/CreditCardNumber.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/util/privilegedactions/GetAnnotationParameter.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/constraintvalidators/bv/size/SizeValidatorForArraysOfPrimitives.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/util/privilegedactions/GetMethods.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/engine/groups/ValidationOrder.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/constraintvalidators/bv/future/FutureValidatorForReadableInstant.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/constraintvalidation/HibernateConstraintValidatorContext.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/engine/ValidationContext.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/metadata/aggregated/rule/VoidMethodsMustNotBeReturnValueConstrained.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/parameternameprovider/ParanamerParameterNameProvider.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/util/privilegedactions/NewSchema.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/HibernateValidator.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/util/privilegedactions/GetDeclaredMethods.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/util/privilegedactions/GetMethod.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/engine/DefaultParameterNameProvider.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/constraints/Email.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-

jar/org/hibernate/validator/internal/metadata/aggregated/ValidatableParametersMetaData.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/util/annotationfactory/AnnotationProxy.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/context/TypeTarget.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/metadata/descriptor/ElementDescriptorImpl.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/messageinterpolation/LocalizedMessage.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/defs/LengthDef.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/context/Cascadable.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/constraintdefinition/ConstraintDefinitionBuilderContextImpl.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/metadata/raw/ConstrainedType.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/util/ReflectionHelper.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/metadata/aggregated/ExecutableMetaData.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/spi/constraintdefinition/ConstraintDefinitionContributor.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/messageinterpolation/InterpolationTermType.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/defs/EANDef.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/defs/URLDef.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/util/CollectionHelper.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/xml/ConstrainedExecutableBuilder.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/metadata/aggregated/AbstractConstraintMetaData.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/hv/NotBlankValidator.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/constraints/Range.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/constraintdefinition/ConstraintDefinitionContribution.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/ConstraintViolationImpl.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/groups/Group.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/size/SizeValidatorForArrayOfDouble.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-

jar/org/hibernate/validator/internal/engine/time/DefaultTimeProvider.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/cfg/defs/DecimalMaxDef.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/engine/messageinterpolation/el/MapBasedVariableMapper.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/metadata/raw/ConstrainedParameter.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/engine/valuehandling/UnwrapMode.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/util/privilegedactions/GetDeclaredFields.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/cfg/defs/SizeDef.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/constraintvalidators/bv/PatternValidator.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/metadata/aggregated/BeanMetaData.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/cfg/defs/Mod11CheckDef.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/cfg/context/CrossParameterConstraintMappingContext.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/metadata/aggregated/rule/MethodConfigurationRule.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/constraintvalidators/bv/size/SizeValidatorForCollection.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/xml/ValidationBootstrapParameters.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/constraintvalidators/hv/package-info.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/xml/BootstrapConfigurationImpl.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/util/privilegedactions/LoadClass.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/constraintvalidators/hv/br/CPFValidator.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/constraintvalidators/bv/future/FutureValidatorForChronoZonedDateTime.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/engine/path/NodeImpl.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/util/privilegedactions/GetDeclaredConstructor.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/xml/XmlMappingParser.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/constraints/ScriptAssert.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
 jar/org/hibernate/validator/internal/metadata/provider/TypeAnnotationAwareMetaDataProvider.java
 * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-

jar/org/hibernate/validator/HibernateValidatorContext.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/metadata/location/ConstraintLocation.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/metadata/provider/ProgrammaticMetaDataProvider.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/util/classhierarchy/ClassHierarchyHelper.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/resourceloading/DelegatingResourceBundleLocator.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/resolver/DefaultTraversableResolver.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/defs/PatternDef.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/NullValidator.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/past/PastValidatorForInstant.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/xml/ClassLoadingHelper.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/cfg/context/ExecutableConstraintMappingContextImpl.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/MaxValidatorForCharSequence.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/metadata/core/ConstraintOrigin.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/spi/valuehandling/package-info.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/util/annotationfactory/AnnotationFactory.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/constraints/CompositionType.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/messageinterpolation/EITermResolver.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/util/annotationfactory/AnnotationDescriptor.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/NotNullValidator.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/ConfigurationImpl.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/xml/MetaConstraintBuilder.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/util/privilegedactions/GetMethodFromPropertyName.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/defs/NotBlankDef.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/constraints/LuhnCheck.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-

jar/org/hibernate/validator/internal/constraintvalidators/bv/DecimalMaxValidatorForCharSequence.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/hv/SafeHtmlValidator.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/constraints/NotEmpty.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/cfg/context/CascadableConstraintMappingContextImplBase.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/util/privilegedactions/ConstructorInstance.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/util/classhierarchy/Filters.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/messageinterpolation/InterpolationTerm.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/future/FutureValidatorForCalendar.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/messageinterpolation/ParameterMessageInterpolator.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/size/SizeValidatorForArraysOfChar.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/size/SizeValidatorForArray.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/metadata/descriptor/CrossParameterDescriptorImpl.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/constraints/URL.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/engine/constraintvalidation/ConstraintValidatorFactoryImpl.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/hv/URLValidator.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/spi/valuehandling/ValidatedValueUnwrapper.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/valuehandling/UnwrapValidatedValue.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/hv/ModCheckValidator.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/util/IdentitySet.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/metadata/raw/ExecutableElement.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/util/TypeResolutionHelper.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/metadata/aggregated/rule/ParallelMethodsMustNotDefineGroupConversionForC
ascadedReturnValue.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/defs/NullDef.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/size/SizeValidatorForArraysOfShort.java

- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/provider/MetaDataProvider.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/descriptor/BeanDescriptorImpl.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/path/PathImpl.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/IgnoreJava6Requirement.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/xml/ConstrainedFieldBuilder.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/context/Unwrapable.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/AssertFalseValidator.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/context/Constrainable.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/Version.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/constraintvalidation/ConstraintViolationCreationContext.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/path/PropertyNode.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/privilegedactions/GetDeclaredMethod.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/messageinterpolation/ResourceBundleMessageInterpolator.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/Contracts.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/DigitsValidatorForNumber.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/messageinterpolation/ParameterTermResolver.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/group/GroupSequenceProvider.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/aggregated/ReturnValueMetaData.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/constraints/ConstraintComposition.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/constraints/Mod11Check.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/xml/ConstrainedGetterBuilder.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/constraints/br/TituloEleitoral.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/past/package-info.java
- * /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/raw/BeanConfiguration.java

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/descriptor/ConstraintDescriptorImpl.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/hv/ScriptAssertContext.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/past/PastValidatorForOffsetDateTime.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/scriptengine/ScriptEvaluatorFactory.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/MaxValidatorForNumber.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/context/PropertyConstraintMappingContext.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/cfg/context/GroupConversionTargetContextImpl.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/path/package-info.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/hv/ParameterScriptAssertValidator.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/constraints/Mod10Check.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/constraints/NotBlank.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/messageinterpolation/AbstractMessageInterpolator.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/provider/XmlMetaDataProvider.java
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/core/AnnotationProcessingOptionsImpl.java

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

/*

* Copyright 2009 IIZUKA Software Technologies Ltd

*

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

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

* You may obtain a copy of the License at

*

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

*

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

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

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

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-

jar/org/hibernate/validator/internal/util/TypeHelper.java

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

```
/*
 * Hibernate Validator, declare and validate application constraints
 *
 * License: Apache License, Version 2.0
 * See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE-2.0>.
 */
/**
 * Retrieves constraint related meta data for the parameters of the given
 * executable.
 *
 * @param executable The executable of interest.
 *
 * @return A list with parameter meta data for the given executable.
 */
```

Found in path(s):

```
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/metadata/provider/AnnotationMetaDataProvider.java
No license file was found, but licenses were detected in source scan.
```

```
~ Hibernate Validator, declare and validate application constraints
~
~ License: Apache License, Version 2.0
~ See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE
```

Found in path(s):

```
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/constraints/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/metadata/raw/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/defs/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/util/annotationfactory/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/cfg/context/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/META-
INF/validation-configuration-1.1.xsd
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/overview.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-jar/META-
INF/validation-mapping-1.0.xsd
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/group/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-
```

jar/org/hibernate/validator/internal/util/classhierarchy/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/provider/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/messageinterpolation/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/aggregated/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/path/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/scriptengine/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/core/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/xml/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/messageinterpolation/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/constraintvalidation/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/resolver/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/constraintvalidation/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/messageinterpolation/parser/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/constraints/br/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/spi/resourceloading/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/privilegedactions/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/resourceloading/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/cfg/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/cfg/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/location/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/groups/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/descriptor/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/cfg/context/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-

```

jar/org/hibernate/validator/internal/util/logging/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/META-INF/validation-mapping-1.1.xsd
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/spi/group/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/facets/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/metadata/aggregated/rule/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/META-INF/validation-configuration-1.0.xsd
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/util/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/internal/engine/messageinterpolation/el/package.html
* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/0/hibernate-validator-5-2-0-final-sources-jar/org/hibernate/validator/package.html

```

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

```

/*
* Hibernate Validator, declare and validate application constraints
*
* License: Apache License, Version 2.0
* See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE-2.0>.
*/
/**
* <p>
* A method-level constraint, that evaluates a script expression against the
* annotated method or constructor. This constraint can be used to implement
* validation routines that depend on several parameters of the annotated
* executable.
* </p>
* <p>
* Script expressions can be written in any scripting or expression language,
* for which a <a href="http://jcp.org/en/jsr/detail?id=223">JSR 223</a>
* ("Scripting for the Java<sup>TM</sup> Platform") compatible engine can be
* found on the classpath. To refer to a parameter within the scripting
* expression, use its name as obtained by the active
* { @link javax.validation.ParameterNameProvider}. By default, { @code arg0}, { @code arg1 } etc.
* will be used as parameter names.
* </p>
* <p>
* The following listing shows an example using the JavaScript engine which
* comes with the JDK:
* </p>
* <pre>

```

```

* { @code @ParameterScriptAssert(script = "arg0.before(arg1)", lang = "javascript")
* public void createEvent(Date start, Date end) { ... }
* }
* </pre>
* <p>
* Can be specified on any method or constructor.
* </p>
*
* @author Gunnar Morling
*/

```

Found in path(s):

```

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/constraints/ParameterScriptAssert.java

```

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

```

/*
* Hibernate Validator, declare and validate application constraints
*
* License: Apache License, Version 2.0
* See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE-2.0>.
*/
/*
* Written by Doug Lea with assistance from members of JCP JSR-166
* Expert Group and released to the public domain, as explained at
* http://creativecommons.org/licenses/publicdomain
*/

```

Found in path(s):

```

* /opt/ws_local/PERMITS_SQL/1069394861_1594653358.43/hibernate-validator-5-2-0-final-sources-
jar/org/hibernate/validator/internal/util/ConcurrentReferenceHashMap.java

```

1.10 struts 2.1.6

1.10.1 Available under license :

```

/* =====
* The OpenSymphony Software License, Version 1.1
*
* (this license is derived and fully compatible with the Apache Software
* License - see http://www.apache.org/LICENSE.txt)
*
* Copyright (c) 2001-2004 The OpenSymphony Group. All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*

```

- * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- *
 - * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
 - *
 - * 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
 - * "This product includes software developed by the OpenSymphony Group (<http://www.opensymphony.com/>)."
 - * Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
 - * 4. The names "OpenSymphony" and "The OpenSymphony Group" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact license@opensymphony.com.
 - * 5. Products derived from this software may not be called "OpenSymphony" or "XWork", nor may "OpenSymphony" or "XWork" appear in their name, without prior written permission of the OpenSymphony Group.
 - * THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 - * =====
 - * /

FreeMarker 1.x was released under the LGPL license. Later, by community consensus, we have switched over to a BSD-style license. As of FreeMarker 2.2pre1, the original author, Benjamin Geer, has relinquished the copyright in behalf of Visigoth Software Society. The current copyright holder is the Visigoth Software Society.

Copyright (c) 2003 The Visigoth Software Society. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. The end-user documentation included with the redistribution, if any, must include the following acknowledgement:
"This product includes software developed by the
Visigoth Software Society (<http://www.visigoths.org/>)."
Alternately, this acknowledgement may appear in the software itself, if and wherever such third-party acknowledgements normally appear.
3. Neither the name "FreeMarker", "Visigoth", nor any of the names of the project contributors may be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact visigoths@visigoths.org.
4. Products derived from this software may not be called "FreeMarker" or "Visigoth" nor may "FreeMarker" or "Visigoth" appear in their names without prior written permission of the Visigoth Software Society.

THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE VISIGOTH SOFTWARE SOCIETY OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Visigoth Software Society. For more information on the Visigoth Software Society, please see <http://www.visigoths.org/>

Apache License

Version 2.0, January 2004

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

designated in writing by the copyright owner as "Not a Contribution."

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

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must

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

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

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

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

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

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly

negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

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

/* =====

- * The OpenSymphony Software License, Version 1.1
- *
- * (this license is derived and fully compatible with the Apache Software License - see <http://www.apache.org/LICENSE.txt>)
- *
- * Copyright (c) 2001-2004 The OpenSymphony Group. All rights reserved.
- *
- * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
- *
- * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- *
- * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- *
- * 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
 - * "This product includes software developed by the
 - * OpenSymphony Group (<http://www.opensymphony.com/>)."
 - * Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
- *
- * 4. The names "OpenSymphony" and "The OpenSymphony Group" must not be used to endorse or promote products derived from this

* software without prior written permission. For written
 * permission, please contact license@opensymphony.com .
 *
 * 5. Products derived from this software may not be called "OpenSymphony"
 * or "OGNL", nor may "OpenSymphony" or "OGNL" appear in their
 * name, without prior written permission of the OpenSymphony Group.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
 * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
 * OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 * =====
 */

Apache Struts
 Copyright 2000-2007 The Apache Software Foundation

This product includes software developed by
 The Apache Software Foundation (<http://www.apache.org/>).
 domTT (<http://www.mojavelinux.com/projects/domtooltip/>).

The binary distributions includes the following third party software:
 FreeMarker (<http://freemarker.org/>).
 OGNL (<http://www.opensymphony.com/ognl/>).
 XWork (<http://www.opensymphony.com/xwork/>).

1.11 commons-io 1.3.2

1.11.1 Available under license :

Apache Jakarta Commons IO
 Copyright 2001-2007 The Apache Software Foundation

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"

means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

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

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

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

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

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

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and

attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

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

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

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

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

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

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

appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.12 activemq-broker 5.8.0

1.12.1 Available under license :

ActiveMQ :: Broker
Copyright 2005-2013 The Apache Software Foundation

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

and conversions to other media types.

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

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

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

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

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

1.13 jboss-logging 3.2.1.Final

1.13.1 Available under license :

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

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

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

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

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

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

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

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

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

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

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

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

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

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,

within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

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

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

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

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

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

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

other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

1.14 geronimojmspec 1.1.1

1.14.1 Available under license :

Apache Geronimo

Copyright 2003-2008 The Apache Software Foundation

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

the Work and Derivative Works thereof.

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

1.15 bean-validation-api 1.1.0.Final

1.16 junit 3.8.1

1.16.1 Available under license :

Common Public License - v 1.0

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

1. DEFINITIONS

"Contribution" means:

- a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
- b) in the case of each subsequent Contributor:

- i) changes to the Program, and

- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

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

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

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

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

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

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

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

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

- a) it complies with the terms and conditions of this Agreement; and
- b) its license agreement:
 - i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
 - ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
 - iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

- a) it must be made available under this Agreement; and
- b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

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

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

5. NO WARRANTY

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

6. DISCLAIMER OF LIABILITY

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

7. GENERAL

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

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

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

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

1.17 ognl 2.6.11

1.17.1 Available under license :

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

```
//-----  
// Copyright (c) 1998-2004, Drew Davidson and Luke Blanshard  
// All rights reserved.  
//  
// Redistribution and use in source and binary forms, with or without  
// modification, are permitted provided that the following conditions are  
// met:  
//  
// Redistributions of source code must retain the above copyright notice,  
// this list of conditions and the following disclaimer.  
// Redistributions in binary form must reproduce the above copyright  
// notice, this list of conditions and the following disclaimer in the  
// documentation and/or other materials provided with the distribution.  
// Neither the name of the Drew Davidson nor the names of its contributors  
// may be used to endorse or promote products derived from this software  
// without specific prior written permission.  
//  
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS  
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT  
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS  
// FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE  
// COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,  
// INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,  
// BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS  
// OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED  
// AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,  
// OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF  
// THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH  
// DAMAGE.  
//-----  
  
/*  
* This file defines the syntax of OGNL, the Object-Graph Navigation Language. This  
* language was devised by Drew Davidson, who called it Key-Value Coding Language. Luke  
* Blanshard then made up the new name and reimplemented it using ANTLR, refining and  
* polishing the language a bit on the way. Drew maintained the system for a couple of  
* years; then Luke converted the ANTLR grammar to JavaCC, to eliminate the run-time  
* dependency on ANTLR.  
*  
* See package.html for a description of the language.  
*/
```

```

options {
    // Parser options
    LOOKAHEAD      = 1;
    STATIC         = false;
    JAVA_UNICODE_ESCAPE = true;
    UNICODE_INPUT  = true;

    // Tree options
    MULTI         = true;
    NODE_DEFAULT_VOID = true;
}

PARSER_BEGIN(OgnlParser)

package ognl;

import java.math.*;

/**
 * OgnlParser is a JavaCC parser class; it translates OGNL expressions into abstract
 * syntax trees (ASTs) that can then be interpreted by the getValue and setValue methods.
 */
public class OgnlParser
{
}

PARSER_END(OgnlParser)

/**
 * This is the top-level construct of OGNL.
 */
Node topLevelExpression() : {}
{
    expression() <EOF> { return jjtree.rootNode(); }
}

// sequence (level 14)
void expression() : {}
{
    assignmentExpression() ( "," assignmentExpression() #Sequence(2) )*
}

// assignment expression (level 13)
void assignmentExpression() : {}
{

```

```

conditionalTestExpression() [ "=" assignmentExpression() #Assign(2) ]
}

// conditional test (level 12)
void conditionalTestExpression() : {}
{
    logicalOrExpression()
    [ "?" conditionalTestExpression() ":" conditionalTestExpression() #Test(3) ]
}

// logical or (||) (level 11)
void logicalOrExpression() : {}
{
    logicalAndExpression() ( "|" "or" ) logicalAndExpression() #Or(2) *
}

// logical and (&&) (level 10)
void logicalAndExpression() : {}
{
    inclusiveOrExpression() ( "&&" "and" ) inclusiveOrExpression() #And(2) *
}

// bitwise or non-short-circuiting or (|) (level 9)
void inclusiveOrExpression() : {}
{
    exclusiveOrExpression() ( "|" "bor" ) exclusiveOrExpression() #BitOr(2) *
}

// exclusive or (^) (level 8)
void exclusiveOrExpression() : {}
{
    andExpression() ( "^" "xor" ) andExpression() #Xor(2) *
}

// bitwise or non-short-circuiting and (&) (level 7)
void andExpression() : {}
{
    equalityExpression() ( "&" "band" ) equalityExpression() #BitAnd(2) *
}

// equality/inequality (==/!=) (level 6)
void equalityExpression() : {}
{

```

```

relationalExpression()
(
    ("==" | "eq") relationalExpression() #Eq(2)
    |
    ("!=" | "neq") relationalExpression() #NotEq(2)
)*
}

// boolean relational expressions (level 5)
void relationalExpression() : { }
{
    shiftExpression()
    (
       ("<" | "lt") shiftExpression() #Less(2)
        |
       (">" | "gt") shiftExpression() #Greater(2)
        |
       ("<=" | "lte") shiftExpression() #LessEq(2)
        |
       (">=" | "gte") shiftExpression() #GreaterEq(2)
        |
        "in" shiftExpression() #In(2)
        |
        "not" "in" shiftExpression() #NotIn(2)
    )*
}

// bit shift expressions (level 4)
void shiftExpression() : { }
{
    additiveExpression()
    (
       ("<<" | "shl") additiveExpression() #ShiftLeft(2)
        |
       (">>" | "shr") additiveExpression() #ShiftRight(2)
        |
       (">>>" | "ushr") additiveExpression() #UnsignedShiftRight(2)
    )*
}

// binary addition/subtraction (level 3)
void additiveExpression() : { }
{
    multiplicativeExpression()
    (

```

```

    "+" multiplicativeExpression() #Add(2)
    |
    "-" multiplicativeExpression() #Subtract(2)
  )*
}

// multiplication/division/remainder (level 2)
void multiplicativeExpression() : { }
{
  unaryExpression()
  (
    "*" unaryExpression() #Multiply(2)
    |
    "/" unaryExpression() #Divide(2)
    |
    "%" unaryExpression() #Remainder(2)
  )*
}

// unary (level 1)
void unaryExpression() : {
  StringBuffer sb;
  Token t;
  ASTInstanceof ionode;
}
{
  (
    "-" unaryExpression() #Negate(1)
    |
    "+" unaryExpression() // Just leave it there
    |
    "~" unaryExpression() #BitNegate(1)
    |
    ("!" | "not") unaryExpression() #Not(1)
    |
    navigationChain()
    [
      "instanceof"
      t = <IDENT> { sb = new StringBuffer(t.image); ionode = jjtThis; } #Instanceof(1)
      ( "." t = <IDENT>      { sb.append('.').append( t.image ); }
      )*                    { ionode.setTargetType( new String(sb) ); }
    ]
  )
}

```

// navigation chain: property references, method calls, projections, selections, etc.

```

void navigationChain() : {}
{
    primaryExpression()
    ( "."
      ( /* Prevent the "eval" ambiguity from issuing a warning; see discussion below. */
        ( LOOKAHEAD(2) methodCall() | propertyName() )
          // Also handle "{", which requires a lookahead of 2.
        | ( LOOKAHEAD(2) projection() | selection() )
        | "(" expression() ")"
        ) #Chain(2)

      | index() #Chain(2)

      | "(" expression() ")" #Eval(2)

        /* Using parentheses to indicate evaluation of the current
           object makes this language ambiguous, because the
           expression "ident(args)" could be seen as a single
           method call or as a property name followed by an
           evaluation. We always put the method call first and
           turn off the ambiguity warning; we always want to
           interpret this as a method call. */

      ) *
    }

void primaryExpression() : {
    Token t;
    String className = null;
}
{
    (
      (<CHAR_LITERAL> | <BACK_CHAR_LITERAL> | <STRING_LITERAL> | <INT_LITERAL> |
      <FLT_LITERAL>)
        { jjtThis.setValue( token_source.literalValue ); } #Const(0)
      |
      "true"
        { jjtThis.setValue( Boolean.TRUE ); } #Const(0)
      |
      "false"
        { jjtThis.setValue( Boolean.FALSE ); } #Const(0)
      |
      "null" #Const(0) // Null is the default value in an ASTConst
      |
      LOOKAHEAD(2) "#this"
        { jjtThis.setName( "this" ); } #ThisVarRef(0)
      |
      LOOKAHEAD(2) "#root"
        { jjtThis.setName( "root" ); } #RootVarRef(0)
      |
      LOOKAHEAD(2) "#" t=<IDENT>
        { jjtThis.setName( t.image ); } #VarRef(0)
    )
}

```

```

|
LOOKAHEAD(2) ":" "[" expression() "]" { jjtThis.setValue( jjtThis.jjtGetChild(0) ); } #Const(1)
|
staticReference()
|
LOOKAHEAD(2) constructorCall()
|
// Prevent the "eval" ambiguity from issuing a warning; see discussion elsewhere.
( LOOKAHEAD(2) methodCall() | propertyName() )
|
index()
|
 "(" expression() )"
|
"[" [assignmentExpression() ("," assignmentExpression())*] #List "]"
|
LOOKAHEAD(2) ( "#" (className=classReference())? "{" [keyValueExpression() (","
keyValueExpression())*] { jjtThis.setClassName(className); } "}" ) #Map
)
}

void keyValueExpression() : {
{
( assignmentExpression() (":" assignmentExpression())? ) #KeyValue
}

void staticReference() : {
String className = "java.lang.Math";
Token t;
}
{
className=classReference()
(// Prevent the "eval" ambiguity from issuing a warning; see discussion elsewhere.
LOOKAHEAD(2)
staticMethodCall( className )
|
t=<IDENT> { jjtThis.init( className, t.image ); } #StaticField(0)
)
}

String classReference(): {
String result = "java.lang.Math";
}
{
"@ ( result=className() )? @" { return result; }
}

String className(): {

```



```

Token t;
StringBuffer result;
}
{
t=<IDENT>      { result = new StringBuffer( t.image ); }
( "." t=<IDENT> { result.append('.').append( t.image ); }
)*           { return new String(result); }
}

void constructorCall() #Ctor : {
String className;
Token t;
StringBuffer sb;
}
{
"new" className=className()
(
LOOKAHEAD(2) (
"(" [ assignmentExpression() ( "," assignmentExpression() )* ] ")"
{
jjtThis.setClassName(className);
}
)
|
LOOKAHEAD(2) (
"[" "]" "{" [assignmentExpression() ( "," assignmentExpression() )* ] #List "]"
{
jjtThis.setClassName(className);
jjtThis.setArray(true);
}
)
|
LOOKAHEAD(2) (
"[" assignmentExpression() "]"
{
jjtThis.setClassName(className);
jjtThis.setArray(true);
}
)
)
}

void propertyName() #Property : {
Token t;
}
{
t=<IDENT> { jjtThis.setValue( t.image ); } #Const
}

```

```

void staticMethodCall( String className ) #StaticMethod : {
    Token t;
}
{
    t=<IDENT> "(" [ assignmentExpression() ( "," assignmentExpression() )* ] ")"
        { jjtThis.init( className, t.image ); }
}

void methodCall() #Method : {
    Token t;
}
{
    t=<IDENT> "(" [ assignmentExpression() ( "," assignmentExpression() )* ] ")"
        { jjtThis.setMethodName( t.image ); }
}

/**
 * Apply an expression to all elements of a collection, creating a new collection
 * as the result.
 */
void projection() #Project : {}
{
    "{" expression() "}"
}

void selection() : {}
{
    LOOKAHEAD(2) selectAll()
    |
    LOOKAHEAD(2) selectFirst()
    |
    LOOKAHEAD(2) selectLast()
}

/**
 * Apply a boolean expression to all elements of a collection, creating a new collection
 * containing those elements for which the expression returned true.
 */
void selectAll() #Select : {}
{
    "{" "?" expression() "}"
}

/**
 * Apply a boolean expression to all elements of a collection, creating a new collection
 * containing those elements for the first element for which the expression returned true.
 */

```

```

void selectFirst() #SelectFirst : {}
{
    "{" "^" expression() "}"
}

/**
 * Apply a boolean expression to all elements of a collection, creating a new collection
 * containing those elements for the first element for which the expression returned true.
 */
void selectLast() #SelectLast : {}
{
    "{" "$" expression() "}"
}

void index() #Property : {}
{
    "[" expression() "]" { jjtThis.setIndexedAccess(true); }
|
<DYNAMIC_SUBSCRIPT> { jjtThis.setValue( token_source.literalValue ); } #Const
{
    jjtThis.setIndexedAccess(true);
}
}

// LEXER PRODUCTIONS

TOKEN_MGR_DECLS:
{
    /** Holds the last value computed by a constant token. */
    Object literalValue;
    /** Holds the last character escaped or in a character literal. */
    private char charValue;
    /** Holds char literal start token. */
    private char charLiteralStartQuote;
    /** Holds the last string literal parsed. */
    private StringBuffer stringBuffer;

    /** Converts an escape sequence into a character value. */
    private char escapeChar()
    {
        int ofs = stringBuffer.length() - 1;
        switch ( stringBuffer.charAt(ofs) ) {
            case 'n': return '\n';
            case 'r': return '\r';
            case 't': return '\t';
            case 'b': return '\b';
            case 'f': return '\f';
            case '\\': return '\\';
        }
    }
}

```

```

        case '\': return '\\';
        case '\"': return '\"';
    }

    // Otherwise, it's an octal number. Find the backslash and convert.
    while ( image.charAt(--ofs) != '\\' )
    {
    }
    int value = 0;
    while ( ++ofs < image.length() )
        value = (value << 3) | (image.charAt(ofs) - '0');
    return (char) value;
}

private Object makeInt()
{
    Object result;
    String s = image.toString();
    int base = 10;

    if ( s.charAt(0) == '0' )
        base = (s.length() > 1 && (s.charAt(1) == 'x' || s.charAt(1) == 'X'))? 16 : 8;
    if ( base == 16 )
        s = s.substring(2); // Trim the 0x off the front
    switch ( s.charAt(s.length()-1) ) {
        case 'l': case 'L':
            result = Long.valueOf( s.substring(0,s.length()-1), base );
            break;

        case 'h': case 'H':
            result = new BigInteger( s.substring(0,s.length()-1), base );
            break;

        default:
            result = Integer.valueOf( s, base );
            break;
    }
    return result;
}

private Object makeFloat()
{
    String s = image.toString();
    switch ( s.charAt(s.length()-1) ) {
        case 'f': case 'F':
            return Float.valueOf( s );

        case 'b': case 'B':
            return new BigDecimal( s.substring(0,s.length()-1) );
    }
}

```

```

        case 'd': case 'D':
        default:
            return Double.valueOf( s );
    }
}
}

// Whitespace -- ignored
SKIP:
{ " " | "\t" | "\f" | "\r" | "\n" }

// An identifier.
TOKEN:
{
    < IDENT: <LETTER> (<LETTER>|<DIGIT>)* >
|
    < #LETTER: [
        "\u0024",
        "\u0041"-"\u005a",
        "\u005f",
        "\u0061"-"\u007a",
        "\u00c0"-"\u00d6",
        "\u00d8"-"\u00f6",
        "\u00f8"-"\u00ff",
        "\u0100"-"\u1fff",
        "\u3040"-"\u318f",
        "\u3300"-"\u337f",
        "\u3400"-"\u3d2d",
        "\u4e00"-"\u9fff",
        "\uf900"-"\ufaff"
    ] >
|
    < #DIGIT:
    [
        "\u0030"-"\u0039",
        "\u0660"-"\u0669",
        "\u06f0"-"\u06f9",
        "\u0966"-"\u096f",
        "\u09e6"-"\u09ef",
        "\u0a66"-"\u0a6f",
        "\u0ae6"-"\u0aef",
        "\u0b66"-"\u0b6f",
        "\u0be7"-"\u0bef",
        "\u0c66"-"\u0c6f",
        "\u0ce6"-"\u0cef",
        "\u0d66"-"\u0d6f",
        "\u0e50"-"\u0e59",

```

```

        "\u0ed0" - "\u0ed9",
        "\u1040" - "\u1049"
    ] >
}

/**
 * Token for "dynamic subscripts", which are one of: [^], [], [$], and [*]. The
 * appropriate constant from the DynamicSubscript class is stored in the token manager's
 * "value" field.
 */
TOKEN:
{
    < DYNAMIC_SUBSCRIPT: "[" [ "^", "|", "$", "*" ] "]" >
    {
        switch (image.charAt(1)) {
            case '^': literalValue = DynamicSubscript.first; break;
            case '|': literalValue = DynamicSubscript.mid; break;
            case '$': literalValue = DynamicSubscript.last; break;
            case '*': literalValue = DynamicSubscript.all; break;
        }
    }
}

/**
 * Character and string literals, whose object value is stored in the token manager's
 * "literalValue" field.
 */
MORE:
{
    "\"" : WithinBackCharLiteral
|
    """" { stringBuffer = new StringBuffer(); } : WithinCharLiteral
|
    "\"" { stringBuffer = new StringBuffer(); } : WithinStringLiteral
}

<WithinCharLiteral> MORE:
{
    < ESC: "\\" ( ["n", "r", "t", "b", "f", "\\", "", "\\", "\"]
        | (["0" - "3"])? ["0" - "7"] (["0" - "7"])?
        )
    >
    { charValue = escapeChar(); stringBuffer.append(charValue); }
|
    < (~["\"", "\"]) >
    { charValue = image.charAt( image.length() - 1 ); stringBuffer.append(charValue); }
}

```

<WithinCharLiteral> TOKEN:

```
{
  < CHAR_LITERAL: """">
  {
    if (stringBuffer.length() == 1) {
      literalValue = new Character( charValue );
    } else {
      literalValue = new String( stringBuffer );
    }
  }
  : DEFAULT
}
```

<WithinBackCharLiteral> MORE:

```
{
  < BACK_CHAR_ESC: <ESC> >
  { charValue = escapeChar(); }
|
  < (~["", "\\"]>
  { charValue = image.charAt( image.length()-1 ); }
}
```

<WithinBackCharLiteral> TOKEN:

```
{
  < BACK_CHAR_LITERAL: """">
  { literalValue = new Character( charValue ); }; DEFAULT
}
```

<WithinStringLiteral> MORE:

```
{
  < STRING_ESC: <ESC> >
  { stringBuffer.append( escapeChar() ); }
|
  < (~["", "\\"]>
  { stringBuffer.append( image.charAt(image.length()-1) ); }
}
```

<WithinStringLiteral> TOKEN:

```
{
  <STRING_LITERAL: """">
  { literalValue = new String( stringBuffer ); }
  : DEFAULT
}
```

/**

* Integer or real Numeric literal, whose object value is stored in the token manager's

* "literalValue" field.

*/

TOKEN:

```
{
  < INT_LITERAL:
    ("0" ("0"- "7")* | ["1"- "9"] ("0"- "9")* | "0" ["x", "X"] ("0"- "9", "a"- "f", "A"- "F")+ )
    (["l", "L", "h", "H"])?
  >
  { literalValue =
    makeInt(); }
|
  < FLT_LITERAL:
    (<DEC_FLT> (<EXPONENT>)? (<FLT_SUFF>)?
    | <DEC_DIGITS> <EXPONENT> (<FLT_SUFF>)?
    | <DEC_DIGITS> <FLT_SUFF>
    )
  >
  { literalValue = makeFloat(); }

| < #DEC_FLT: ("0"- "9")+ "." ("0"- "9")* | "." ("0"- "9")+ >
| < #DEC_DIGITS: ("0"- "9")+ >
| < #EXPONENT: ["e", "E"] (["+", "-"])? ("0"- "9")+ >
| < #FLT_SUFF: ["d", "D", "f", "F", "b", "B"] >
}
```

Found in path(s):

* /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ognl.jjt

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

2004, Drew Davidson and Luke Blanshard

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Drew Davidson nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,

INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):

* /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/package.html

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

```
/*@bgen(jjtree) Generated By:JJTree: Do not edit this line. src/java/ognl/ognl.jj */
/*@egen*///-----
// Copyright (c) 1998-2004, Drew Davidson and Luke Blanshard
// All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// met:
//
// Redistributions of source code must retain the above copyright notice,
// this list of conditions and the following disclaimer.
// Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// Neither the name of the Drew Davidson nor the names of its contributors
// may be used to endorse or promote products derived from this software
// without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
// FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
// COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
// INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
// BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
// OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
// AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
// OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
// THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
// DAMAGE.
//-----

/*
 * This file defines the syntax of OGNL, the Object-Graph Navigation Language. This
 * language was devised by Drew Davidson, who called it Key-Value Coding Language. Luke
 * Blanshard then made up the new name and reimplemented it using ANTLR, refining and
```

```

* polishing the language a bit on the way. Drew maintained the system for a couple of
* years; then Luke converted the ANTLR grammar to JavaCC, to eliminate the run-time
* dependency on ANTLR.
*
* See package.html for a description of the language.
*/

```

```

options {
    // Parser options
    LOOKAHEAD      = 1;
    STATIC         = false;
    JAVA_UNICODE_ESCAPE = true;
    UNICODE_INPUT  = true;
}

```

```

PARSER_BEGIN(OgnlParser)

```

```

package ognl;

```

```

import java.math.*;

```

```

/**

```

```

* OgnlParser is a JavaCC parser class; it translates OGNL expressions into abstract
* syntax trees (ASTs) that can then be interpreted by the getValue and setValue methods.
*/

```

```

public class OgnlParser/*@bgen(jjtree)*/implements OgnlParserTreeConstants/*@egen*/
{/*@bgen(jjtree)*/
    protected JJTOgnlParserState jjtree = new JJTOgnlParserState();

```

```

/*@egen*/
}

```

```

PARSER_END(OgnlParser)

```

```

/**

```

```

* This is the top-level construct of OGNL.
*/

```

```

Node topLevelExpression() : {}
{
    expression() <EOF> { return jjtree.rootNode(); }
}

```

```

// sequence (level 14)

```

```

void expression() : {}
{

```

```

assignmentExpression() ( " , /*@bgen(jjtree) #Sequence( 2) */
    {
        ASTSequence jjtn001 = new ASTSequence(JJTSEQUENCE);
        boolean jjtc001 = true;
        jjtree.openNodeScope(jjtn001);
    }
    try {
/*@egen*/ assignmentExpression()/*@bgen(jjtree)*/
        } catch (Throwable jjte001) {
            if (jjtc001) {
                jjtree.clearNodeScope(jjtn001);
                jjtc001 = false;
            } else {
                jjtree.popNode();
            }
            if (jjte001 instanceof RuntimeException) {
                throw (RuntimeException)jjte001;
            }
            if (jjte001 instanceof ParseException) {
                throw (ParseException)jjte001;
            }
            throw (Error)jjte001;
        } finally {
            if (jjtc001) {
                jjtree.closeNodeScope(jjtn001, 2);
            }
        }
/*@egen*/ )*
    }

```

// assignment expression (level 13)

```
void assignmentExpression() : {}
```

```
{
```

```
conditionalTestExpression() [ "="/*@bgen(jjtree) #Assign( 2) */
```

```

    {
        ASTAssign jjtn001 = new ASTAssign(JJTASSIGN);
        boolean jjtc001 = true;
        jjtree.openNodeScope(jjtn001);
    }
    try {
/*@egen*/ assignmentExpression()/*@bgen(jjtree)*/
        } catch (Throwable jjte001) {
            if (jjtc001) {
                jjtree.clearNodeScope(jjtn001);
                jjtc001 = false;
            } else {
                jjtree.popNode();
            }
        }

```

```

        if (jjte001 instanceof RuntimeException) {
            throw (RuntimeException)jjte001;
        }
        if (jjte001 instanceof ParseException) {
            throw (ParseException)jjte001;
        }
        throw (Error)jjte001;
    } finally {
        if (jjtc001) {
            jjtree.closeNodeScope(jjtn001, 2);
        }
    }
}

/*@egen*/ ]
}

// conditional test (level 12)
void conditionalTestExpression() : {}
{
    logicalOrExpression()
        [ "?" conditionalTestExpression() ":"/*@bgen(jjtree) #Test( 3) */
            {
                ASTTest jjtn001 = new ASTTest(JJTTEST);
                boolean jjtc001 = true;
                jjtree.openNodeScope(jjtn001);
            }
            try {
/*@egen*/ conditionalTestExpression()/*@bgen(jjtree)*/
                } catch (Throwable jjte001) {
                    if (jjtc001) {
                        jjtree.clearNodeScope(jjtn001);
                        jjtc001 = false;
                    } else {
                        jjtree.popNode();
                    }
                }
                if (jjte001 instanceof RuntimeException) {
                    throw (RuntimeException)jjte001;
                }
                if (jjte001 instanceof ParseException) {
                    throw (ParseException)jjte001;
                }
                throw (Error)jjte001;
            } finally {
                if (jjtc001) {
                    jjtree.closeNodeScope(jjtn001, 3);
                }
            }
        }
}

/*@egen*/ ]
}

```

```

// logical or (||) (level 11)
void logicalOrExpression() : { }
{
    logicalAndExpression() (("||" | "or")/*@bgen(jjtree) #Or( 2) */
        {
            ASTOr jjtn001 = new ASTOr(JJTOR);
            boolean jjtc001 = true;
            jjtree.openNodeScope(jjtn001);
        }
        try {
/*@egen*/ logicalAndExpression()/*@bgen(jjtree)*/
            } catch (Throwable jjte001) {
                if (jjtc001) {
                    jjtree.clearNodeScope(jjtn001);
                    jjtc001 = false;
                } else {
                    jjtree.popNode();
                }
                if (jjte001 instanceof RuntimeException) {
                    throw (RuntimeException)jjte001;
                }
                if (jjte001 instanceof ParseException) {
                    throw (ParseException)jjte001;
                }
                throw (Error)jjte001;
            } finally {
                if (jjtc001) {
                    jjtree.closeNodeScope(jjtn001, 2);
                }
            }
/*@egen*/ )*
        }

// logical and (&&) (level 10)
void logicalAndExpression() : { }
{
    inclusiveOrExpression() ("&&" | "and")/*@bgen(jjtree) #And( 2) */
        {
            ASTAnd jjtn001 = new ASTAnd(JJTAND);
            boolean jjtc001 = true;
            jjtree.openNodeScope(jjtn001);
        }
        try {
/*@egen*/ inclusiveOrExpression()/*@bgen(jjtree)*/
            } catch (Throwable jjte001) {
                if (jjtc001) {

```

```

        jjtree.clearNodeScope(jjtn001);
        jjtc001 = false;
    } else {
        jjtree.popNode();
    }
    if (jjte001 instanceof RuntimeException) {
        throw (RuntimeException)jjte001;
    }
    if (jjte001 instanceof ParseException) {
        throw (ParseException)jjte001;
    }
    throw (Error)jjte001;
} finally {
    if (jjtc001) {
        jjtree.closeNodeScope(jjtn001, 2);
    }
}
/*@egen*/ )*
}

// bitwise or non-short-circuiting or (|) (level 9)
void inclusiveOrExpression() : {}
{
    exclusiveOrExpression(("|" | "bor")/*@bgen(jjtree) #BitOr( 2) */
        {
            ASTBitOr jjtn001 = new ASTBitOr(JJTBITOR);
            boolean jjtc001 = true;
            jjtree.openNodeScope(jjtn001);
        }
        try {
/*@egen*/ exclusiveOrExpression()/*@bgen(jjtree)*/
        } catch (Throwable jjte001) {
            if (jjtc001) {
                jjtree.clearNodeScope(jjtn001);
                jjtc001 = false;
            } else {
                jjtree.popNode();
            }
            if (jjte001 instanceof RuntimeException) {
                throw (RuntimeException)jjte001;
            }
            if (jjte001 instanceof ParseException) {
                throw (ParseException)jjte001;
            }
            throw (Error)jjte001;
        } finally {
            if (jjtc001) {

```

```

        jjtree.closeNodeScope(jjtn001, 2);
    }
}

/*@egen*/)*
}

// exclusive or (^) (level 8)
void exclusiveOrExpression() : {}
{
    andExpression() (("^^" | "xor")/*@bgen(jjtree) #Xor( 2) */
        {
            ASTXor jjtn001 = new ASTXor(JJTXOR);
            boolean jjtc001 = true;
            jjtree.openNodeScope(jjtn001);
        }
        try {
/*@egen*/ andExpression()/*@bgen(jjtree)*/
            } catch (Throwable jjte001) {
                if (jjtc001) {
                    jjtree.clearNodeScope(jjtn001);
                    jjtc001 = false;
                } else {
                    jjtree.popNode();
                }
                if (jjte001 instanceof RuntimeException) {
                    throw (RuntimeException)jjte001;
                }
                if (jjte001 instanceof ParseException) {
                    throw (ParseException)jjte001;
                }
                throw (Error)jjte001;
            } finally {
                if (jjtc001) {
                    jjtree.closeNodeScope(jjtn001, 2);
                }
            }
        }
/*@egen*/)*
}

// bitwise or non-short-circuiting and (&) (level 7)
void andExpression() : {}
{
    equalityExpression() ((" & " | "band")/*@bgen(jjtree) #BitAnd( 2) */
        {
            ASTBitAnd jjtn001 = new ASTBitAnd(JJTBITAND);
            boolean jjtc001 = true;

```

```

        jjtree.openNodeScope(jjtn001);
    }
    try {
/*@egen*/ equalityExpression()/*@bgen(jjtree)*/
    } catch (Throwable jjte001) {
        if (jjtc001) {
            jjtree.clearNodeScope(jjtn001);
            jjtc001 = false;
        } else {
            jjtree.popNode();
        }
        if (jjte001 instanceof RuntimeException) {
            throw (RuntimeException)jjte001;
        }
        if (jjte001 instanceof ParseException) {
            throw (ParseException)jjte001;
        }
        throw (Error)jjte001;
    } finally {
        if (jjtc001) {
            jjtree.closeNodeScope(jjtn001, 2);
        }
    }
/*@egen*/ )*
}

```

```

// equality/inequality (==/!=) (level 6)
void equalityExpression() : {}
{
    relationalExpression()
    (
        ("==" | "eq")/*@bgen(jjtree) #Eq( 2) */
        {
            ASTEq jjtn001 = new ASTEq(JJTEQ);
            boolean jjtc001 = true;
            jjtree.openNodeScope(jjtn001);
        }
        try {
/*@egen*/ relationalExpression()/*@bgen(jjtree)*/
    } catch (Throwable jjte001) {
        if (jjtc001) {
            jjtree.clearNodeScope(jjtn001);
            jjtc001 = false;
        } else {
            jjtree.popNode();
        }
        if (jjte001 instanceof RuntimeException) {

```



```

        throw (RuntimeException)jjte001;
    }
    if (jjte001 instanceof ParseException) {
        throw (ParseException)jjte001;
    }
    throw (Error)jjte001;
} finally {
    if (jjtc001) {
        jjtree.closeNodeScope(jjtn001, 2);
    }
}
/*@egen*/
|
    ("!=" | "neq")/*@bgen(jjtree) #NotEq( 2) */
    {
        ASTNotEq jjtn002 = new ASTNotEq(JJTNOTEQ);
        boolean jjtc002 = true;
        jjtree.openNodeScope(jjtn002);
    }
    try {
/*@egen*/ relationalExpression()/*@bgen(jjtree)*/
    } catch (Throwable jjte002) {
        if (jjtc002) {
            jjtree.clearNodeScope(jjtn002);
            jjtc002 = false;
        } else {
            jjtree.popNode();
        }
        if (jjte002 instanceof RuntimeException) {
            throw (RuntimeException)jjte002;
        }
        if (jjte002 instanceof ParseException) {
            throw (ParseException)jjte002;
        }
        throw (Error)jjte002;
    } finally {
        if (jjtc002) {
            jjtree.closeNodeScope(jjtn002, 2);
        }
    }
/*@egen*/
    )*
}

```

```

// boolean relational expressions (level 5)
void relationalExpression() : { }
{

```

```

shiftExpression()
(
 ("<" | "lt")/*@bgen(jjtree) #Less( 2) */
  {
    ASTLess jjtn001 = new ASTLess(JJTLESS);
    boolean jjtc001 = true;
    jjtree.openNodeScope(jjtn001);
  }
  try {
/*@egen*/ shiftExpression()/*@bgen(jjtree)*/
  } catch (Throwable jjte001) {
    if (jjtc001) {
      jjtree.clearNodeScope(jjtn001);
      jjtc001 = false;
    } else {
      jjtree.popNode();
    }
    if (jjte001 instanceof RuntimeException) {
      throw (RuntimeException)jjte001;
    }
    if (jjte001 instanceof ParseException) {
      throw (ParseException)jjte001;
    }
    throw (Error)jjte001;
  } finally {
    if (jjtc001) {
      jjtree.closeNodeScope(jjtn001, 2);
    }
  }
/*@egen*/
|
 (">" | "gt")/*@bgen(jjtree) #Greater( 2) */
  {
    ASTGreater jjtn002 = new ASTGreater(JJTGREATER);
    boolean jjtc002 = true;
    jjtree.openNodeScope(jjtn002);
  }
  try {
/*@egen*/ shiftExpression()/*@bgen(jjtree)*/
  } catch (Throwable jjte002) {
    if (jjtc002) {
      jjtree.clearNodeScope(jjtn002);
      jjtc002 = false;
    } else {
      jjtree.popNode();
    }
    if (jjte002 instanceof RuntimeException) {
      throw (RuntimeException)jjte002;
    }
  }

```

```

    }
    if (jjte002 instanceof ParseException) {
        throw (ParseException)jjte002;
    }
    throw (Error)jjte002;
} finally {
    if (jjtc002) {
        jjtree.closeNodeScope(jjtn002, 2);
    }
}
}
/*@egen*/
|
("<=" | "lte")/*@bgen(jjtree) #LessEq( 2) */
{
    ASTLessEq jjtn003 = new ASTLessEq(JJTLESSEQ);
    boolean jjtc003 = true;
    jjtree.openNodeScope(jjtn003);
}
try {
/*@egen*/ shiftExpression()/*@bgen(jjtree)*/
} catch (Throwable jjte003) {
    if (jjtc003) {
        jjtree.clearNodeScope(jjtn003);
        jjtc003 = false;
    } else {
        jjtree.popNode();
    }
}
if (jjte003 instanceof RuntimeException) {
    throw (RuntimeException)jjte003;
}
if (jjte003 instanceof ParseException) {
    throw (ParseException)jjte003;
}
throw (Error)jjte003;
} finally {
    if (jjtc003) {
        jjtree.closeNodeScope(jjtn003, 2);
    }
}
}
/*@egen*/
|
(">=" | "gte")/*@bgen(jjtree) #GreaterEq( 2) */
{
    ASTGreaterEq jjtn004 = new ASTGreaterEq(JJTGREATEREQ);
    boolean jjtc004 = true;
    jjtree.openNodeScope(jjtn004);
}
try {

```

```

/*@egen*/ shiftExpression()/*@bgen(jjtree)*/
    } catch (Throwable jjte004) {
    if (jjtc004) {
        jjtree.clearNodeScope(jjtn004);
        jjtc004 = false;
    } else {
        jjtree.popNode();
    }
    if (jjte004 instanceof RuntimeException) {
        throw (RuntimeException)jjte004;
    }
    if (jjte004 instanceof ParseException) {
        throw (ParseException)jjte004;
    }
    throw (Error)jjte004;
} finally {
    if (jjtc004) {
        jjtree.closeNodeScope(jjtn004, 2);
    }
}

```

```
/*@egen*/
```

```
|
```

```
"in"/*@bgen(jjtree) #In( 2) */
```

```

{
    ASTIn jjtn005 = new ASTIn(JJTIN);
    boolean jjtc005 = true;
    jjtree.openNodeScope(jjtn005);
}
try {
/*@egen*/ shiftExpression()/*@bgen(jjtree)*/
    } catch (Throwable jjte005) {
    if (jjtc005) {
        jjtree.clearNodeScope(jjtn005);
        jjtc005 = false;
    } else {
        jjtree.popNode();
    }
    if (jjte005 instanceof RuntimeException) {
        throw (RuntimeException)jjte005;
    }
    if (jjte005 instanceof ParseException) {
        throw (ParseException)jjte005;
    }
    throw (Error)jjte005;
} finally {
    if (jjtc005) {
        jjtree.closeNodeScope(jjtn005, 2);
    }
}

```

```

    }
/*@egen*/
|
    "not" "in"/*@bgen(jjtree) #NotIn( 2) */
    {
        ASTNotIn jjtn006 = new ASTNotIn(JJTNOTIN);
        boolean jjtc006 = true;
        jjtree.openNodeScope(jjtn006);
    }
    try {
/*@egen*/ shiftExpression()/*@bgen(jjtree)*/
    } catch (Throwable jjte006) {
        if (jjtc006) {
            jjtree.clearNodeScope(jjtn006);
            jjtc006 = false;
        } else {
            jjtree.popNode();
        }
        if (jjte006 instanceof RuntimeException) {
            throw (RuntimeException)jjte006;
        }
        if (jjte006 instanceof ParseException) {
            throw (ParseException)jjte006;
        }
        throw (Error)jjte006;
    } finally {
        if (jjtc006) {
            jjtree.closeNodeScope(jjtn006, 2);
        }
    }
/*@egen*/
    )*
}

// bit shift expressions (level 4)
void shiftExpression() : { }
{
    additiveExpression()
    (
        ("<<" | "shl")/*@bgen(jjtree) #ShiftLeft( 2) */
        {
            ASTShiftLeft jjtn001 = new ASTShiftLeft(JJTSHIFTLEFT);
            boolean jjtc001 = true;
            jjtree.openNodeScope(jjtn001);
        }
        try {
/*@egen*/ additiveExpression()/*@bgen(jjtree)*/

```

```

    } catch (Throwable jjte001) {
    if (jjtc001) {
        jjtree.clearNodeScope(jjtn001);
        jjtc001 = false;
    } else {
        jjtree.popNode();
    }
    if (jjte001 instanceof RuntimeException) {
        throw (RuntimeException)jjte001;
    }
    if (jjte001 instanceof ParseException) {
        throw (ParseException)jjte001;
    }
    throw (Error)jjte001;
    } finally {
    if (jjtc001) {
        jjtree.closeNodeScope(jjtn001, 2);
    }
    }
/*@egen*/
|
    (">>" | "shr")/*@bgen(jjtree) #ShiftRight( 2) */
    {
        ASTShiftRight jjtn002 = new ASTShiftRight(JJTSHIFTRIGHT);
        boolean jjtc002 = true;
        jjtree.openNodeScope(jjtn002);
    }
    try {
/*@egen*/ additiveExpression()/*@bgen(jjtree)*/
    } catch (Throwable jjte002) {
    if (jjtc002) {
        jjtree.clearNodeScope(jjtn002);
        jjtc002 = false;
    } else {
        jjtree.popNode();
    }
    if (jjte002 instanceof RuntimeException) {
        throw (RuntimeException)jjte002;
    }
    if (jjte002 instanceof ParseException) {
        throw (ParseException)jjte002;
    }
    throw (Error)jjte002;
    } finally {
    if (jjtc002) {
        jjtree.closeNodeScope(jjtn002, 2);
    }
    }
}

```

```

/*@egen*/
|
  (">>>" | "ushr")/*@bgen(jjtree) #UnsignedShiftRight( 2) */
  {
    ASTUnsignedShiftRight jjtn003 = new ASTUnsignedShiftRight(JTUNSIGNEDSHIFTRIGHT);
    boolean jjtc003 = true;
    jjtree.openNodeScope(jjtn003);
  }
  try {
/*@egen*/ additiveExpression()/*@bgen(jjtree)*/
    } catch (Throwable jjte003) {
      if (jjtc003) {
        jjtree.clearNodeScope(jjtn003);
        jjtc003 = false;
      } else {
        jjtree.popNode();
      }
      if (jjte003 instanceof RuntimeException) {
        throw (RuntimeException)jjte003;
      }
      if (jjte003 instanceof ParseException) {
        throw (ParseException)jjte003;
      }
      throw (Error)jjte003;
    } finally {
      if (jjtc003) {
        jjtree.closeNodeScope(jjtn003, 2);
      }
    }
}
/*@egen*/
)*
}

```

```

// binary addition/subtraction (level 3)
void additiveExpression() : {}
{
  multiplicativeExpression()
  (
    "+"/*@bgen(jjtree) #Add( 2) */
    {
      ASTAdd jjtn001 = new ASTAdd(JTADD);
      boolean jjtc001 = true;
      jjtree.openNodeScope(jjtn001);
    }
    try {
/*@egen*/ multiplicativeExpression()/*@bgen(jjtree)*/
      } catch (Throwable jjte001) {

```

```

    if (jjtc001) {
        jjtree.clearNodeScope(jjtn001);
        jjtc001 = false;
    } else {
        jjtree.popNode();
    }
    if (jjte001 instanceof RuntimeException) {
        throw (RuntimeException)jjte001;
    }
    if (jjte001 instanceof ParseException) {
        throw (ParseException)jjte001;
    }
    throw (Error)jjte001;
} finally {
    if (jjtc001) {
        jjtree.closeNodeScope(jjtn001, 2);
    }
}
/*@egen*/
|
    "-"/*@bgen(jjtree) #Subtract( 2) */
    {
        ASTSubtract jjtn002 = new ASTSubtract(JJTSUBTRACT);
        boolean jjtc002 = true;
        jjtree.openNodeScope(jjtn002);
    }
    try {
/*@egen*/ multiplicativeExpression()/*@bgen(jjtree)*/
    } catch (Throwable jjte002) {
        if (jjtc002) {
            jjtree.clearNodeScope(jjtn002);
            jjtc002 = false;
        } else {
            jjtree.popNode();
        }
        if (jjte002 instanceof RuntimeException) {
            throw (RuntimeException)jjte002;
        }
        if (jjte002 instanceof ParseException) {
            throw (ParseException)jjte002;
        }
        throw (Error)jjte002;
    } finally {
        if (jjtc002) {
            jjtree.closeNodeScope(jjtn002, 2);
        }
    }
}
/*@egen*/

```



```

)*
}

// multiplication/division/remainder (level 2)
void multiplicativeExpression() : {}
{
    unaryExpression()
    (
        ""/*@bgen(jjtree) #Multiply( 2) */
        {
            ASTMultiply jjtn001 = new ASTMultiply(JJTMULTIPLY);
            boolean jjtc001 = true;
            jjtree.openNodeScope(jjtn001);
        }
        try {
/*@egen*/ unaryExpression()/*@bgen(jjtree)*/
        } catch (Throwable jjte001) {
            if (jjtc001) {
                jjtree.clearNodeScope(jjtn001);
                jjtc001 = false;
            } else {
                jjtree.popNode();
            }
            if (jjte001 instanceof RuntimeException) {
                throw (RuntimeException)jjte001;
            }
            if (jjte001 instanceof ParseException) {
                throw (ParseException)jjte001;
            }
            throw (Error)jjte001;
        } finally {
            if (jjtc001) {
                jjtree.closeNodeScope(jjtn001, 2);
            }
        }
/*@egen*/
        |
        ""/*@bgen(jjtree) #Divide( 2) */
        {
            ASTDivide jjtn002 = new ASTDivide(JJTDIVIDE);
            boolean jjtc002 = true;
            jjtree.openNodeScope(jjtn002);
        }
        try {
/*@egen*/ unaryExpression()/*@bgen(jjtree)*/
        } catch (Throwable jjte002) {
            if (jjtc002) {

```

```

        jjtree.clearNodeScope(jjtn002);
        jjtc002 = false;
    } else {
        jjtree.popNode();
    }
    if (jjte002 instanceof RuntimeException) {
        throw (RuntimeException)jjte002;
    }
    if (jjte002 instanceof ParseException) {
        throw (ParseException)jjte002;
    }
    throw (Error)jjte002;
} finally {
    if (jjtc002) {
        jjtree.closeNodeScope(jjtn002, 2);
    }
}
/*@egen*/
|
"% "/*@bgen(jjtree) #Remainder( 2) */
{
    ASTRemainder jjtn003 = new ASTRemainder(JJTREMAINDER);
    boolean jjtc003 = true;
    jjtree.openNodeScope(jjtn003);
}
try {
/*@egen*/ unaryExpression()/*@bgen(jjtree)*/
} catch (Throwable jjte003) {
    if (jjtc003) {
        jjtree.clearNodeScope(jjtn003);
        jjtc003 = false;
    } else {
        jjtree.popNode();
    }
    if (jjte003 instanceof RuntimeException) {
        throw (RuntimeException)jjte003;
    }
    if (jjte003 instanceof ParseException) {
        throw (ParseException)jjte003;
    }
    throw (Error)jjte003;
} finally {
    if (jjtc003) {
        jjtree.closeNodeScope(jjtn003, 2);
    }
}
/*@egen*/
)*

```

```

}

// unary (level 1)
void unaryExpression() : {
    StringBuffer sb;
    Token t;
    ASTInstanceof ionode;
}
{
    (
        "-"/*@bgen(jjtree) #Negate( 1) */
        {
            ASTNegate jjtn001 = new ASTNegate(JJTNEGATE);
            boolean jjtc001 = true;
            jjtree.openNodeScope(jjtn001);
        }
        try {
/*@egen*/ unaryExpression()/*@bgen(jjtree)*/
        } catch (Throwable jjte001) {
            if (jjtc001) {
                jjtree.clearNodeScope(jjtn001);
                jjtc001 = false;
            } else {
                jjtree.popNode();
            }
            if (jjte001 instanceof RuntimeException) {
                throw (RuntimeException)jjte001;
            }
            if (jjte001 instanceof ParseException) {
                throw (ParseException)jjte001;
            }
            throw (Error)jjte001;
        } finally {
            if (jjtc001) {
                jjtree.closeNodeScope(jjtn001, 1);
            }
        }
/*@egen*/
        |
        "+" unaryExpression() // Just leave it there
        |
        "~"/*@bgen(jjtree) #BitNegate( 1) */
        {
            ASTBitNegate jjtn002 = new ASTBitNegate(JJTBITNEGATE);
            boolean jjtc002 = true;
            jjtree.openNodeScope(jjtn002);
        }
        try {

```

```

/*@egen*/ unaryExpression()/*@bgen(jjtree)*/
    } catch (Throwable jjte002) {
        if (jjtc002) {
            jjtree.clearNodeScope(jjtn002);
            jjtc002 = false;
        } else {
            jjtree.popNode();
        }
        if (jjte002 instanceof RuntimeException) {
            throw (RuntimeException)jjte002;
        }
        if (jjte002 instanceof ParseException) {
            throw (ParseException)jjte002;
        }
        throw (Error)jjte002;
    } finally {
        if (jjtc002) {
            jjtree.closeNodeScope(jjtn002, 1);
        }
    }
/*@egen*/
|
("!" | "not")/*@bgen(jjtree) #Not( 1) */
    {
        ASTNot jjtn003 = new ASTNot(JJTNOT);
        boolean jjtc003 = true;
        jjtree.openNodeScope(jjtn003);
    }
    try {
/*@egen*/ unaryExpression()/*@bgen(jjtree)*/
        } catch (Throwable jjte003) {
            if (jjtc003) {
                jjtree.clearNodeScope(jjtn003);
                jjtc003 = false;
            } else {
                jjtree.popNode();
            }
            if (jjte003 instanceof RuntimeException) {
                throw (RuntimeException)jjte003;
            }
            if (jjte003 instanceof ParseException) {
                throw (ParseException)jjte003;
            }
            throw (Error)jjte003;
        } finally {
            if (jjtc003) {
                jjtree.closeNodeScope(jjtn003, 1);
            }
        }
    }

```

```

    }
/*@egen*/
|
  navigationChain()
  [
    "instanceof"
    t = <IDENT>/*@bgen(jjtree) #Instanceof( 1) */
      {
        ASTInstanceof jjtn004 = new ASTInstanceof(JJTINSTANCEOF);
        boolean jjtc004 = true;
        jjtree.openNodeScope(jjtn004);
      }
      try {
/*@egen*//*@bgen(jjtree)*/
      {
        jjtree.closeNodeScope(jjtn004, 1);
        jjtc004 = false;
      }
/*@egen*/ { sb = new StringBuffer(t.image); ionode = jjtn004; }/*@bgen(jjtree)*/
      } finally {
        if (jjtc004) {
          jjtree.closeNodeScope(jjtn004, 1);
        }
      }
}
/*@egen*/
  ( "." t = <IDENT>      { sb.append('.').append( t.image ); }
  )*                    { ionode.setTargetType( new String(sb) ); }
  ]
)
}

// navigation chain: property references, method calls, projections, selections, etc.
void navigationChain() : {}
{
  primaryExpression()
  ( "."/*@bgen(jjtree) #Chain( 2) */
  {
    ASTChain jjtn001 = new ASTChain(JJTCHAIN);
    boolean jjtc001 = true;
    jjtree.openNodeScope(jjtn001);
  }
  try {
/*@egen*/
    ( /* Prevent the "eval" ambiguity from issuing a warning; see discussion below. */
      ( LOOKAHEAD(2) methodCall() | propertyName() )
      // Also handle "{", which requires a lookahead of 2.
      | ( LOOKAHEAD(2) projection() | selection() )

```

```

| "(" expression() ")"
)/*@bgen(jjtree)*/
} catch (Throwable jjte001) {
    if (jjtc001) {
        jjtree.clearNodeScope(jjtn001);
        jjtc001 = false;
    } else {
        jjtree.popNode();
    }
    if (jjte001 instanceof RuntimeException) {
        throw (RuntimeException)jjte001;
    }
    if (jjte001 instanceof ParseException) {
        throw (ParseException)jjte001;
    }
    throw (Error)jjte001;
} finally {
    if (jjtc001) {
        jjtree.closeNodeScope(jjtn001, 2);
    }
}
/*@egen*/

/*@bgen(jjtree) #Chain( 2) */
{
    ASTChain jjtn002 = new ASTChain(JJTCHAIN);
    boolean jjtc002 = true;
    jjtree.openNodeScope(jjtn002);
}
try {
/*@egen*/ index()/*@bgen(jjtree)*/
} catch (Throwable jjte002) {
    if (jjtc002) {
        jjtree.clearNodeScope(jjtn002);
        jjtc002 = false;
    } else {
        jjtree.popNode();
    }
    if (jjte002 instanceof RuntimeException) {
        throw (RuntimeException)jjte002;
    }
    if (jjte002 instanceof ParseException) {
        throw (ParseException)jjte002;
    }
    throw (Error)jjte002;
} finally {
    if (jjtc002) {
        jjtree.closeNodeScope(jjtn002, 2);
    }
}

```

```

    }
    }
/*@egen*/

| "(" expression()/*@bgen(jjtree) #Eval( 2) */
    {
        ASTEval jjtn003 = new ASTEval(JJTEVAL);
        boolean jjtc003 = true;
        jjtree.openNodeScope(jjtn003);
    }
    try {
/*@egen*/ ")"/*@bgen(jjtree)*/
    } finally {
        if (jjtc003) {
            jjtree.closeNodeScope(jjtn003, 2);
        }
    }
}
/*@egen*/

```

/* Using parentheses to indicate evaluation of the current object makes this language ambiguous, because the expression "ident(args)" could be seen as a single method call or as a property name followed by an evaluation. We always put the method call first and turn off the ambiguity warning; we always want to interpret this as a method call. */

```

)*
}

```

```

void primaryExpression() : {
    Token t;
    String className = null;
}
{
    (
        (<CHAR_LITERAL> | <BACK_CHAR_LITERAL> | <STRING_LITERAL> | <INT_LITERAL> |
        <FLT_LITERAL>)*/*@bgen(jjtree) #Const( 0) */
        {
            ASTConst jjtn001 = new ASTConst(JJTCONST);
            boolean jjtc001 = true;
            jjtree.openNodeScope(jjtn001);
        }
        try {
/*@egen*//*@bgen(jjtree)*/
        {
            jjtree.closeNodeScope(jjtn001, 0);
        }
    }
}

```

```

        jjtc001 = false;
    }

/*@egen*/
    { jjtn001.setValue( token_source.literalValue ); }/*@bgen(jjtree)*/
    } finally {
        if (jjtc001) {
            jjtree.closeNodeScope(jjtn001, 0);
        }
    }
}

/*@egen*/
|
    "true"/*@bgen(jjtree) #Const( 0) */
    {
        ASTConst jjtn002 = new ASTConst(JJTCONST);
        boolean jjtc002 = true;
        jjtree.openNodeScope(jjtn002);
    }
    try {
/*@egen*//*@bgen(jjtree)*/
        {
            jjtree.closeNodeScope(jjtn002, 0);
            jjtc002 = false;
        }
/*@egen*/
        { jjtn002.setValue( Boolean.TRUE ); }/*@bgen(jjtree)*/
    } finally {
        if (jjtc002) {
            jjtree.closeNodeScope(jjtn002, 0);
        }
    }
}

/*@egen*/
|
    "false"/*@bgen(jjtree) #Const( 0) */
    {
        ASTConst jjtn003 = new ASTConst(JJTCONST);
        boolean jjtc003 = true;
        jjtree.openNodeScope(jjtn003);
    }
    try {
/*@egen*//*@bgen(jjtree)*/
        {
            jjtree.closeNodeScope(jjtn003, 0);
            jjtc003 = false;
        }
/*@egen*/
        { jjtn003.setValue( Boolean.FALSE ); }/*@bgen(jjtree)*/
    } finally {
        if (jjtc003) {
            jjtree.closeNodeScope(jjtn003, 0);
        }
    }
}

```



```

    }
/*@egen*/
/*@bgen(jjtree) #Const( 0) */
{
    ASTConst jjtn004 = new ASTConst(JJTCONST);
    boolean jjtc004 = true;
    jjtree.openNodeScope(jjtn004);
}
try {
/*@egen*/
    "null"/*@bgen(jjtree)*/
} finally {
    if (jjtc004) {
        jjtree.closeNodeScope(jjtn004, 0);
    }
}
/*@egen*/          // Null is the default value in an ASTConst
|
    LOOKAHEAD(2) "#this"/*@bgen(jjtree) #ThisVarRef( 0) */
        {
            ASTThisVarRef jjtn005 = new ASTThisVarRef(JJTTHISVARREF);
            boolean jjtc005 = true;
            jjtree.openNodeScope(jjtn005);
        }
        try {
/*@egen*//*@bgen(jjtree)*/
            {
                jjtree.closeNodeScope(jjtn005, 0);
                jjtc005 = false;
            }
/*@egen*/          { jjtn005.setName( "this" ); }/*@bgen(jjtree)*/
        } finally {
            if (jjtc005) {
                jjtree.closeNodeScope(jjtn005, 0);
            }
        }
}
/*@egen*/
|
    LOOKAHEAD(2) "#root"/*@bgen(jjtree) #RootVarRef( 0) */
        {
            ASTRootVarRef jjtn006 = new ASTRootVarRef(JJTROOTVARREF);
            boolean jjtc006 = true;
            jjtree.openNodeScope(jjtn006);
        }
        try {
/*@egen*//*@bgen(jjtree)*/
            {
                jjtree.closeNodeScope(jjtn006, 0);
            }
}

```

```

        jjtc006 = false;
    }
/*@egen*/    { jjtn006.setName( "root" ); }/*@bgen(jjtree)*/
    } finally {
        if (jjtc006) {
            jjtree.closeNodeScope(jjtn006, 0);
        }
    }
/*@egen*/
|
    LOOKAHEAD(2) "#" t<=IDENT>/*@bgen(jjtree) #VarRef( 0) */
    {
        ASTVarRef jjtn007 = new ASTVarRef(JJTVARREF);
        boolean jjtc007 = true;
        jjtree.openNodeScope(jjtn007);
    }
    try {
/*@egen*//*@bgen(jjtree)*/
    {
        jjtree.closeNodeScope(jjtn007, 0);
        jjtc007 = false;
    }
/*@egen*/    { jjtn007.setName( t.image ); }/*@bgen(jjtree)*/
    } finally {
        if (jjtc007) {
            jjtree.closeNodeScope(jjtn007, 0);
        }
    }
/*@egen*/
|
    LOOKAHEAD(2) ":" "[" expression() "]"/*@bgen(jjtree) #Const( 1) */
    {
        ASTConst jjtn008 = new ASTConst(JJTCONST);
        boolean jjtc008 = true;
        jjtree.openNodeScope(jjtn008);
    }
    try {
/*@egen*//*@bgen(jjtree)*/
    {
        jjtree.closeNodeScope(jjtn008, 1);
        jjtc008 = false;
    }
/*@egen*/    { jjtn008.setValue( jjtn008.jjtGetChild(0) ); }/*@bgen(jjtree)*/
    } finally {
        if (jjtc008) {
            jjtree.closeNodeScope(jjtn008, 1);
        }
    }
}

```

```

/*@egen*/
|
|   staticReference()
|
|   LOOKAHEAD(2) constructorCall()
|
|   // Prevent the "eval" ambiguity from issuing a warning; see discussion elsewhere.
|   ( LOOKAHEAD(2) methodCall() | propertyName() )
|
|   index()
|
|   "(" expression() )"
|
|   "{"/*@bgen(jjtree) List */
|       {
|           ASTList jjtn009 = new ASTList(JJTLIST);
|           boolean jjtc009 = true;
|           jjtree.openNodeScope(jjtn009);
|       }
|       try {
/*@egen*/ [assignmentExpression() ("," assignmentExpression())/*@bgen(jjtree)*/
|           } catch (Throwable jjte009) {
|               if (jjtc009) {
|                   jjtree.clearNodeScope(jjtn009);
|                   jjtc009 = false;
|               } else {
|                   jjtree.popNode();
|               }
|               if (jjte009 instanceof RuntimeException) {
|                   throw (RuntimeException)jjte009;
|               }
|               if (jjte009 instanceof ParseException) {
|                   throw (ParseException)jjte009;
|               }
|               throw (Error)jjte009;
|           } finally {
|               if (jjtc009) {
|                   jjtree.closeNodeScope(jjtn009, true);
|               }
|           }
/*@egen*/ "}"
|
|   LOOKAHEAD(2)/*@bgen(jjtree) Map */
|       {
|           ASTMap jjtn010 = new ASTMap(JJTMAP);
|           boolean jjtc010 = true;
|           jjtree.openNodeScope(jjtn010);
|       }

```

```

        try {
/*@egen*/ ( "#" (className=classReference())? "{" [keyValueExpression() ("," keyValueExpression())*] {
jjtn010.setClassName(className); } "}" )/*@bgen(jjtree)*/
        } catch (Throwable jjte010) {
            if (jjtc010) {
                jjtree.clearNodeScope(jjtn010);
                jjtc010 = false;
            } else {
                jjtree.popNode();
            }
            if (jjte010 instanceof RuntimeException) {
                throw (RuntimeException)jjte010;
            }
            if (jjte010 instanceof ParseException) {
                throw (ParseException)jjte010;
            }
            throw (Error)jjte010;
        } finally {
            if (jjtc010) {
                jjtree.closeNodeScope(jjtn010, true);
            }
        }
/*@egen*/
    )
}

```

```

void keyValueExpression() : { }
/*@bgen(jjtree) KeyValue */
{
    ASTKeyValue jjtn001 = new ASTKeyValue(JJTKEYVALUE);
    boolean jjtc001 = true;
    jjtree.openNodeScope(jjtn001);
}
try {
/*@egen*/
    ( assignmentExpression() (":" assignmentExpression())? )/*@bgen(jjtree)*/
} catch (Throwable jjte001) {
    if (jjtc001) {
        jjtree.clearNodeScope(jjtn001);
        jjtc001 = false;
    } else {
        jjtree.popNode();
    }
    if (jjte001 instanceof RuntimeException) {
        throw (RuntimeException)jjte001;
    }
    if (jjte001 instanceof ParseException) {
        throw (ParseException)jjte001;
    }
}

```

```

    }
    throw (Error)jjte001;
} finally {
    if (jjtc001) {
        jjtree.closeNodeScope(jjtn001, true);
    }
}
/*@egen*/
}

void staticReference() : {
    String className = "java.lang.Math";
    Token t;
}
{
    className=classReference()
    ( // Prevent the "eval" ambiguity from issuing a warning; see discussion elsewhere.
    LOOKAHEAD(2)
    staticMethodCall( className )
    |
    t=<IDENT>/*@bgen(jjtree) #StaticField( 0) */
        {
            ASTStaticField jjtn001 = new ASTStaticField(JJTSTATICFIELD);
            boolean jjtc001 = true;
            jjtree.openNodeScope(jjtn001);
        }
        try {
/*@egen*//*@bgen(jjtree)*/
            {
                jjtree.closeNodeScope(jjtn001, 0);
                jjtc001 = false;
            }
/*@egen*/
            { jjtn001.init( className, t.image ); }/*@bgen(jjtree)*/
        } finally {
            if (jjtc001) {
                jjtree.closeNodeScope(jjtn001, 0);
            }
        }
/*@egen*/
    )
}

String classReference(): {
    String result = "java.lang.Math";
}
{
    "@ ( result=className() )? "@" { return result; }
}
}

```

```

String className(): {
    Token t;
    StringBuffer result;
}
{
    t=<IDENT>          { result = new StringBuffer( t.image ); }
    ( "." t=<IDENT>    { result.append('.').append( t.image ); }
)*
    { return new String(result); }
}

void constructorCall() : { /*@bgen(jjtree) Ctor */
    ASTCtor jjtn000 = new ASTCtor(JJTCTOR);
    boolean jjtc000 = true;
    jjtree.openNodeScope(jjtn000);
/*@egen*/
    String className;
    Token t;
    StringBuffer sb;
}
{ /*@bgen(jjtree) Ctor */
    try {
/*@egen*/
        "new" className=className()
        (
            LOOKAHEAD(2) (
                (" [ assignmentExpression() ( "," assignmentExpression() )* ] ") /*@bgen(jjtree)*/
                {
                    jjtree.closeNodeScope(jjtn000, true);
                    jjtc000 = false;
                }
/*@egen*/
                {
                    jjtn000.setClassName(className);
                }
            )
            |
            LOOKAHEAD(2) (
                "[" "]" "{ /*@bgen(jjtree) List */
                {
                    ASTList jjtn001 = new ASTList(JJTLIST);
                    boolean jjtc001 = true;
                    jjtree.openNodeScope(jjtn001);
                }
                try {
/*@egen*/ [assignmentExpression() ("," assignmentExpression())*] /*@bgen(jjtree)*/
                } catch (Throwable jjtc001) {
                    if (jjtc001) {

```

```

        jjtree.clearNodeScope(jjtn001);
        jjtc001 = false;
    } else {
        jjtree.popNode();
    }
    if (jjte001 instanceof RuntimeException) {
        throw (RuntimeException)jjte001;
    }
    if (jjte001 instanceof ParseException) {
        throw (ParseException)jjte001;
    }
    throw (Error)jjte001;
} finally {
    if (jjtc001) {
        jjtree.closeNodeScope(jjtn001, true);
    }
}
}
}

/*@egen*/ "}"/*@bgen(jjtree)*/
{
    jjtree.closeNodeScope(jjtn000, true);
    jjtc000 = false;
}

/*@egen*/
{
    jjtn000.setClassName(className);
    jjtn000.setArray(true);
}

)
|
LOOKAHEAD(2) (
    "[" assignmentExpression() "]"/*@bgen(jjtree)*/
    {
        jjtree.closeNodeScope(jjtn000, true);
        jjtc000 = false;
    }
}

/*@egen*/
{
    jjtn000.setClassName(className);
    jjtn000.setArray(true);
}

)
)/*@bgen(jjtree)*/
} catch (Throwable jjte000) {
    if (jjtc000) {
        jjtree.clearNodeScope(jjtn000);
        jjtc000 = false;
    } else {
        jjtree.popNode();
    }
}

```

```

    }
    if (jjte000 instanceof RuntimeException) {
        throw (RuntimeException)jjte000;
    }
    if (jjte000 instanceof ParseException) {
        throw (ParseException)jjte000;
    }
    throw (Error)jjte000;
} finally {
    if (jjtc000) {
        jjtree.closeNodeScope(jjtn000, true);
    }
}
/*@egen*/
}

void propertyName()      : {/*@bgen(jjtree) Property */
    ASTProperty jjtn000 = new ASTProperty(JJTPROPERTY);
    boolean jjtc000 = true;
    jjtree.openNodeScope(jjtn000);
/*@egen*/
    Token t;
}
{/*@bgen(jjtree) Property */
    try {
/*@egen*/
        t=<IDENT>/*@bgen(jjtree) Const */
        {
            ASTConst jjtn001 = new ASTConst(JJTCONST);
            boolean jjtc001 = true;
            jjtree.openNodeScope(jjtn001);
        }
        try {
/*@egen*//*@bgen(jjtree)*/
            {
                jjtree.closeNodeScope(jjtn001, true);
                jjtc001 = false;
            }
/*@egen*/ { jjtn001.setValue( t.image ); }/*@bgen(jjtree)*/
        } finally {
            if (jjtc001) {
                jjtree.closeNodeScope(jjtn001, true);
            }
        }
/*@egen*//*@bgen(jjtree)*/
    } finally {
        if (jjtc000) {
            jjtree.closeNodeScope(jjtn000, true);
        }
    }
}

```



```

    }
    }
    /*@egen*/
}

void staticMethodCall( String className )      : {/*@bgen(jjtree) StaticMethod */
    ASTStaticMethod jjtn000 = new ASTStaticMethod(JJTSTATICMETHOD);
    boolean jjtc000 = true;
    jjtree.openNodeScope(jjtn000);
    /*@egen*/
    Token t;
}
{/*@bgen(jjtree) StaticMethod */
    try {
    /*@egen*/
        t=<IDENT> "(" [ assignmentExpression() ( "," assignmentExpression() )* ] ")"/*@bgen(jjtree)*/
            {
                jjtree.closeNodeScope(jjtn000, true);
                jjtc000 = false;
            }
    /*@egen*/
        { jjtn000.init( className, t.image ); }/*@bgen(jjtree)*/
    } catch (Throwable jjte000) {
        if (jjtc000) {
            jjtree.clearNodeScope(jjtn000);
            jjtc000 = false;
        } else {
            jjtree.popNode();
        }
        if (jjte000 instanceof RuntimeException) {
            throw (RuntimeException)jjte000;
        }
        if (jjte000 instanceof ParseException) {
            throw (ParseException)jjte000;
        }
        throw (Error)jjte000;
    } finally {
        if (jjtc000) {
            jjtree.closeNodeScope(jjtn000, true);
        }
    }
}
/*@egen*/
}

void methodCall()      : {/*@bgen(jjtree) Method */
    ASTMethod jjtn000 = new ASTMethod(JJTMETHOD);
    boolean jjtc000 = true;
    jjtree.openNodeScope(jjtn000);

```

```

/*@egen*/
    Token t;
}
/*@bgen(jjtree) Method */
    try {
/*@egen*/
        t=<IDENT> "(" [ assignmentExpression() ( "," assignmentExpression() )* ] )"/**@bgen(jjtree)*/
            {
                jjtree.closeNodeScope(jjtn000, true);
                jjtc000 = false;
            }
/*@egen*/
                { jjtn000.setMethodName( t.image ); }/**@bgen(jjtree)*/
    } catch (Throwable jjte000) {
        if (jjtc000) {
            jjtree.clearNodeScope(jjtn000);
            jjtc000 = false;
        } else {
            jjtree.popNode();
        }
        if (jjte000 instanceof RuntimeException) {
            throw (RuntimeException)jjte000;
        }
        if (jjte000 instanceof ParseException) {
            throw (ParseException)jjte000;
        }
        throw (Error)jjte000;
    } finally {
        if (jjtc000) {
            jjtree.closeNodeScope(jjtn000, true);
        }
    }
/*@egen*/
}

/**
 * Apply an expression to all elements of a collection, creating a new collection
 * as the result.
 */
void projection()      : {/**@bgen(jjtree) Project */
    ASTProject jjtn000 = new ASTProject(JJTPROJECT);
    boolean jjtc000 = true;
    jjtree.openNodeScope(jjtn000);
/*@egen*/}
/*@bgen(jjtree) Project */
    try {
/*@egen*/
        "{" expression() }"/**@bgen(jjtree)*/

```

```

    } catch (Throwable jjte000) {
    if (jjtc000) {
        jjtree.clearNodeScope(jjtn000);
        jjtc000 = false;
    } else {
        jjtree.popNode();
    }
    if (jjte000 instanceof RuntimeException) {
        throw (RuntimeException)jjte000;
    }
    if (jjte000 instanceof ParseException) {
        throw (ParseException)jjte000;
    }
    throw (Error)jjte000;
    } finally {
    if (jjtc000) {
        jjtree.closeNodeScope(jjtn000, true);
    }
    }
    /*@egen*/
}

void selection() : {}
{
    LOOKAHEAD(2) selectAll()
    |
    LOOKAHEAD(2) selectFirst()
    |
    LOOKAHEAD(2) selectLast()
}

/**
 * Apply a boolean expression to all elements of a collection, creating a new collection
 * containing those elements for which the expression returned true.
 */
void selectAll() : {/*@bgen(jjtree) Select */
    ASTSelect jjtn000 = new ASTSelect(JJTSELECT);
    boolean jjtc000 = true;
    jjtree.openNodeScope(jjtn000);
    /*@egen*/}
    {/*@bgen(jjtree) Select */
    try {
    /*@egen*/
    "{" "?" expression() "}"/*@bgen(jjtree)*/
    } catch (Throwable jjte000) {
    if (jjtc000) {
        jjtree.clearNodeScope(jjtn000);
        jjtc000 = false;

```

```

    } else {
        jjtree.popNode();
    }
    if (jjte000 instanceof RuntimeException) {
        throw (RuntimeException)jjte000;
    }
    if (jjte000 instanceof ParseException) {
        throw (ParseException)jjte000;
    }
    throw (Error)jjte000;
} finally {
    if (jjtc000) {
        jjtree.closeNodeScope(jjtn000, true);
    }
}
/*@egen*/
}

/**
 * Apply a boolean expression to all elements of a collection, creating a new collection
 * containing those elements for the first element for which the expression returned true.
 */
void selectFirst() : {/*@bgen(jjtree) SelectFirst */
    ASTSelectFirst jjtn000 = new ASTSelectFirst(JJTSELECTFIRST);
    boolean jjtc000 = true;
    jjtree.openNodeScope(jjtn000);
    /*@egen*/}
    {/*@bgen(jjtree) SelectFirst */
        try {
            /*@egen*/
            "{" "^" expression() "}"/*@bgen(jjtree)*/
        } catch (Throwable jjte000) {
            if (jjtc000) {
                jjtree.clearNodeScope(jjtn000);
                jjtc000 = false;
            } else {
                jjtree.popNode();
            }
            if (jjte000 instanceof RuntimeException) {
                throw (RuntimeException)jjte000;
            }
            if (jjte000 instanceof ParseException) {
                throw (ParseException)jjte000;
            }
            throw (Error)jjte000;
        } finally {
            if (jjtc000) {
                jjtree.closeNodeScope(jjtn000, true);
            }
        }
    }
}

```

```

    }
  }
  /*@egen*/
}

/**
 * Apply a boolean expression to all elements of a collection, creating a new collection
 * containing those elements for the first element for which the expression returned true.
 */
void selectLast()      : {/*@bgen(jjtree) SelectLast */
  ASTSelectLast jjtn000 = new ASTSelectLast(JJTSELECTLAST);
  boolean jjtc000 = true;
  jjtree.openNodeScope(jjtn000);
  /*@egen*/
  {/*@bgen(jjtree) SelectLast */
    try {
  /*@egen*/
    "{" "$" expression() "}"/*@bgen(jjtree)*/
    } catch (Throwable jjte000) {
      if (jjtc000) {
        jjtree.clearNodeScope(jjtn000);
        jjtc000 = false;
      } else {
        jjtree.popNode();
      }
      if (jjte000 instanceof RuntimeException) {
        throw (RuntimeException)jjte000;
      }
      if (jjte000 instanceof ParseException) {
        throw (ParseException)jjte000;
      }
      throw (Error)jjte000;
    } finally {
      if (jjtc000) {
        jjtree.closeNodeScope(jjtn000, true);
      }
    }
  }
  /*@egen*/
}

void index()          : {/*@bgen(jjtree) Property */
  ASTProperty jjtn000 = new ASTProperty(JJTPROPERTY);
  boolean jjtc000 = true;
  jjtree.openNodeScope(jjtn000);
  /*@egen*/
  {/*@bgen(jjtree) Property */
    try {
  /*@egen*/

```

```

    "[" expression( "]"/*@bgen(jjtree)*/
        {
            jjtree.closeNodeScope(jjtn000, true);
            jjtc000 = false;
        }
/*@egen*/ { jjtn000.setIndexedAccess(true); }
|
<DYNAMIC_SUBSCRIPT>/*@bgen(jjtree) Const */
    {
        ASTConst jjtn001 = new ASTConst(JJTCONST);
        boolean jjtc001 = true;
        jjtree.openNodeScope(jjtn001);
    }
    try {
/*@egen*//*@bgen(jjtree)*/
        {
            jjtree.closeNodeScope(jjtn001, true);
            jjtc001 = false;
        }
/*@egen*/ { jjtn001.setValue( token_source.literalValue ); }/*@bgen(jjtree)*/
        } finally {
            if (jjtc001) {
                jjtree.closeNodeScope(jjtn001, true);
            }
        }
/*@egen*//*@bgen(jjtree)*/
    {
        jjtree.closeNodeScope(jjtn000, true);
        jjtc000 = false;
    }
/*@egen*/
    {
        jjtn000.setIndexedAccess(true);
/*@bgen(jjtree)*/
    } catch (Throwable jjte000) {
        if (jjtc000) {
            jjtree.clearNodeScope(jjtn000);
            jjtc000 = false;
        } else {
            jjtree.popNode();
        }
        if (jjte000 instanceof RuntimeException) {
            throw (RuntimeException)jjte000;
        }
        if (jjte000 instanceof ParseException) {
            throw (ParseException)jjte000;
        }
        throw (Error)jjte000;
    }

```

```

    } finally {
        if (jjtc000) {
            jjtree.closeNodeScope(jjtn000, true);
        }
    }
}
/*@egen*/
}

// LEXER PRODUCTIONS

TOKEN_MGR_DECLS:
{
    /** Holds the last value computed by a constant token. */
    Object literalValue;
    /** Holds the last character escaped or in a character literal. */
    private char charValue;
    /** Holds char literal start token. */
    private char charLiteralStartQuote;
    /** Holds the last string literal parsed. */
    private StringBuffer stringBuffer;

    /** Converts an escape sequence into a character value. */
    private char escapeChar()
    {
        int ofs = image.length() - 1;
        switch ( image.charAt(ofs) ) {
            case 'n': return '\n';
            case 'r': return '\r';
            case 't': return '\t';
            case 'b': return '\b';
            case 'f': return '\f';
            case '\\': return '\\';
            case '\"': return '\"';
            case '\"': return '\"';
        }

        // Otherwise, it's an octal number. Find the backslash and convert.
        while ( image.charAt(--ofs) != '\\' )
            {}
        int value = 0;
        while ( ++ofs < image.length() )
            value = (value << 3) | (image.charAt(ofs) - '0');
        return (char) value;
    }

    private Object makeInt()
    {
        Object result;

```

```

String s = image.toString();
int base = 10;

if ( s.charAt(0) == '0' )
    base = (s.length() > 1 && (s.charAt(1) == 'x' || s.charAt(1) == 'X'))? 16 : 8;
if ( base == 16 )
    s = s.substring(2); // Trim the 0x off the front
switch ( s.charAt(s.length()-1) ) {
    case 'l': case 'L':
        result = Long.valueOf( s.substring(0,s.length()-1), base );
        break;

    case 'h': case 'H':
        result = new BigInteger( s.substring(0,s.length()-1), base );
        break;

    default:
        result = Integer.valueOf( s, base );
        break;
}
return result;
}

private Object makeFloat()
{
    String s = image.toString();
    switch ( s.charAt(s.length()-1) ) {
        case 'f': case 'F':
            return Float.valueOf( s );

        case 'b': case 'B':
            return new BigDecimal( s.substring(0,s.length()-1) );

        case 'd': case 'D':
        default:
            return Double.valueOf( s );
    }
}

// Whitespace -- ignored
SKIP:
{ " " | "\t" | "\f" | "\r" | "\n" }

// An identifier.
TOKEN:
{
    < IDENT: <LETTER> (<LETTER>|<DIGIT>)* >

```



```

|
< #LETTER: [
    "$",
    "A"- "Z",
    "_",
    "a"- "z",
    "\u00c0"- "\u00d6",
    "\u00d8"- "\u00f6",
    "\u00f8"- "\u00ff",
    "\u0100"- "\u1fff",
    "\u3040"- "\u318f",
    "\u3300"- "\u337f",
    "\u3400"- "\u3d2d",
    "\u4e00"- "\u9fff",
    "\uf900"- "\ufaff"
] >
|
< #DIGIT:
[
    "0"- "9",
    "\u0660"- "\u0669",
    "\u06f0"- "\u06f9",
    "\u0966"- "\u096f",
    "\u09e6"- "\u09ef",
    "\u0a66"- "\u0a6f",
    "\u0ae6"- "\u0aef",
    "\u0b66"- "\u0b6f",
    "\u0be7"- "\u0bef",
    "\u0c66"- "\u0c6f",
    "\u0ce6"- "\u0cef",
    "\u0d66"- "\u0d6f",
    "\u0e50"- "\u0e59",
    "\u0ed0"- "\u0ed9",
    "\u1040"- "\u1049"
] >
}

/**
 * Token for "dynamic subscripts", which are one of: [^], [], [$], and [*]. The
 * appropriate constant from the DynamicSubscript class is stored in the token manager's
 * "value" field.
 */
TOKEN:
{
< DYNAMIC_SUBSCRIPT: "[" [ "^", "|", "$", "*" ] "]" >
    {
        switch (image.charAt(1)) {
            case '^': literalValue = DynamicSubscript.first; break;

```

```

        case '|': literalValue = DynamicSubscript.mid; break;
        case '$': literalValue = DynamicSubscript.last; break;
        case '*': literalValue = DynamicSubscript.all; break;
    }
}

/**
 * Character and string literals, whose object value is stored in the token manager's
 * "literalValue" field.
 */
MORE:
{
    "\"" : WithinBackCharLiteral
|
    "\"" { stringBuffer = new StringBuffer(); }: WithinCharLiteral
|
    "\"" { stringBuffer = new StringBuffer(); }: WithinStringLiteral
}

<WithinCharLiteral> MORE:
{
    < ESC: "\\\" ( [\"n\", \"r\", \"t\", \"b\", \"f\", \"\\\", \"\", \"^\", \"'\"]
        | ([\"0\"-\"3\"])? [\"0\"-\"7\"] ([\"0\"-\"7\"])?
        )
    >
    { charValue = escapeChar(); stringBuffer.append(charValue); }
|
    < (~[\"'\", \"\\\"] ) >
    { charValue = image.charAt( image.length()-1 ); stringBuffer.append(charValue); }
}

<WithinCharLiteral> TOKEN:
{
    < CHAR_LITERAL: \"\">
    {
        if (stringBuffer.length() == 1) {
            literalValue = new Character( charValue );
        } else {
            literalValue = new String( stringBuffer );
        }
    }
    : DEFAULT
}

<WithinBackCharLiteral> MORE:
{
    < BACK_CHAR_ESC: <ESC> >

```

```

        { charValue = escapeChar(); }
    |
    < (~["\"", "\\"]) >
        { charValue = image.charAt( image.length()-1 ); }
    }

<WithinBackCharLiteral> TOKEN:
{
    < BACK_CHAR_LITERAL: "\"" >
        { literalValue = new Character( charValue ); } : DEFAULT
}

<WithinStringLiteral> MORE:
{
    < STRING_ESC: <ESC> >
        { stringBuffer.append( escapeChar() ); }
    |
    < (~["\"", "\\"]) >
        { stringBuffer.append( image.charAt(image.length()-1) ); }
}

<WithinStringLiteral> TOKEN:
{
    <STRING_LITERAL: "\"\" >
        { literalValue = new String( stringBuffer ); }
        : DEFAULT
}

/**
 * Integer or real Numeric literal, whose object value is stored in the token manager's
 * "literalValue" field.
 */
TOKEN:
{
    < INT_LITERAL:
        ( "0" ([ "0"-"7" ])* | [ "1"-"9" ] ([ "0"-"9" ])* | "0" [ "x", "X" ] ([ "0"-"9", "a"-"f", "A"-"F" ])+ )
        ([ "l", "L", "h", "H" ])?
    >
        { literalValue =
            makeInt(); }
    |
    < FLT_LITERAL:
        ( <DEC_FLT> (<EXPONENT>)? (<FLT_SUFF>)?
          | <DEC_DIGITS> <EXPONENT> (<FLT_SUFF>)?
          | <DEC_DIGITS> <FLT_SUFF>
        )
    >
        { literalValue = makeFloat(); }
}

```

```
| < #DEC_FLT: ([0-9])+ "." ([0-9])* | "." ([0-9])+ >
| < #DEC_DIGITS: ([0-9])+ >
| < #EXPONENT: ["e","E"] ([+","-"])? ([0-9])+ >
| < #FLT_SUFF: ["d","D","f","F","b","B"] >
}
```

Found in path(s):

* /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ognl.jj

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

```
// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// Redistributions of source code must retain the above copyright notice,
// this list of conditions and the following disclaimer.
// Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// Neither the name of the Drew Davidson nor the names of its contributors
// without specific prior written permission.
```

Found in path(s):

* /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/OgnlRuntime.java

* /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/OgnlInvokePermission.java

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

```
// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// Redistributions of source code must retain the above copyright notice,
// this list of conditions and the following disclaimer.
// Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// Neither the name of the Drew Davidson nor the names of its contributors
// without specific prior written permission.
```

Found in path(s):

* /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ObjectMethodAccessor.java

* /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ElementsAccessor.java

* /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTEq.java

* /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/CollectionElementsAccessor.java

* /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/SimpleNode.java

* /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTEval.java

* /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTLessEq.java

* /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTProperty.java

* /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ObjectPropertyAccessor.java

- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTBitNegate.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTBitAnd.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTUnsignedShiftRight.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTMultiply.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ObjectIndexedPropertyDescriptor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ObjectArrayPool.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/DefaultMemberAccess.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTAnd.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTLess.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTDivide.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTShiftRight.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTBitOr.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTKeyValue.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTVarRef.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTProject.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTIn.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTStaticField.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTNot.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/Evaluation.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTRootVarRef.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ListPropertyAccessor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTGreater.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/MethodFailedException.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTChain.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/TypeConverter.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ObjectNullHandler.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/EnumerationElementsAccessor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTSelectLast.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/EnumerationIterator.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/NumericTypes.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/NumberElementsAccessor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTMap.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/DefaultClassResolver.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTOr.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTXor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTNotIn.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ClassResolver.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ArrayElementsAccessor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTThisVarRef.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/IntHashMap.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/MapPropertyAccessor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/EnumerationPropertyAccessor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/NoSuchPropertyException.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTTest.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/MethodAccessor.java

- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/OgnlContext.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ArrayPropertyAccessor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTGreaterEq.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTNotEq.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ExpressionSyntaxException.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/PropertyAccessor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ExpressionNode.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/SetPropertyAccessor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/IteratorElementsAccessor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTAssign.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTSelectFirst.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTSequence.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/OgnlException.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTConst.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/IteratorPropertyAccessor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/IteratorEnumeration.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/DefaultTypeConverter.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/OgnlOps.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/MemberAccess.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ObjectElementsAccessor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTNegate.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/InappropriateExpressionException.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/MapElementsAccessor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTMethod.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTShiftLeft.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTAdd.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/NullHandler.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTSelect.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/DynamicSubscript.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTSubtract.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTRemainder.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTStaticMethod.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTList.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTCtor.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/EvaluationPool.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/Ognl.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/ASTInstanceOf.java
- * /opt/cola/permits/1136662877_1613850795.93/0/ognl-2-6-11-sources-2-jar/ognl/Node.java

1.18 activemq-client 5.8.0

1.18.1 Available under license :

ActiveMQ :: Client

Copyright 2005-2013 The Apache Software Foundation

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

the Work and Derivative Works thereof.

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

1.19 xwork 2.1.2

1.19.1 Available under license :

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

```
/*  
* Copyright (c) 2002-2003 by OpenSymphony  
* All rights reserved.  
*/
```

Found in path(s):

```
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-  
jar/com/opensymphony/xwork2/util/FileManager.java  
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-  
jar/com/opensymphony/xwork2/config/impl/MockConfiguration.java  
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-  
jar/com/opensymphony/xwork2/util/ClassLoaderUtil.java  
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-  
jar/com/opensymphony/xwork2/util/TextUtils.java
```

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

```
/*  
* $Id: WildcardHelper.java 1063 2006-07-10 00:30:29Z mrdon $  
*  
* Copyright 2003-2004 The Apache Software Foundation.  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,
```

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/util/PatternMatcher.java

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

/*

* \$Id: ActionConfigMatcher.java 1833 2008-06-21 09:29:39Z rainerh \$

*

* Copyright 2003,2004 The Apache Software Foundation.

*

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

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

* You may obtain a copy of the License at

*

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

*

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

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

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

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/config/impl/ActionConfigMatcher.java

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

/*

* Copyright (c) 2002-2006 by OpenSymphony

* All rights reserved.

*/

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/ActionChainResult.java

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

/**

* Licensed to the Apache Software Foundation (ASF) under one or more

* contributor license agreements. See the NOTICE file distributed with

* this work for additional information regarding copyright ownership.

* The ASF licenses this file to You under the Apache License, Version 2.0

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

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/util/finder/UrlSet.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/util/finder/ClassFinder.java
No license file was found, but licenses were detected in source scan.

/**

* Copyright (C) 2006 Google Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/inject/package-info.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/inject/ConstructionContext.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/inject/util/ReferenceType.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/inject/util/FinalizableSoftReference.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/inject/Scope.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/inject/Context.java

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/Factory.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/util/Strings.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/ContainerBuilder.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/util/FinalizableReference.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/InternalFactory.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/util/ReferenceCache.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/ContainerImpl.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/ExternalContext.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/util/FinalizableReferenceQueue.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/Scoped.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/Container.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/DependencyException.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/util/FinalizableWeakReference.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/Key.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/util/Function.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/Inject.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/util/FinalizablePhantomReference.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/InternalContext.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/inject/util/ReferenceMap.java
No license file was found, but licenses were detected in source scan.

/*
* \$Id\$
*
* Copyright 2003-2004 The Apache Software Foundation.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at

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

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/util/ClearableValueStack.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2002-2007 by OpenSymphony
* All rights reserved.
*/

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/ModelDriven.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/interceptor/DefaultWorkflowInterceptor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/DefaultActionProxyFactory.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/validator/ValidationInterceptor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/validator/FieldValidator.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/util/ValueStackFactory.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/interceptor/PrepareInterceptor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/interceptor/PreResultListener.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/ActionInvocation.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/interceptor/NoParameters.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/util/ValueStack.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/interceptor/ExceptionHandler.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/ActionProxy.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-

jar/com/opensymphony/xwork2/ValidationAware.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/Result.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/Preparable.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/ActionProxyFactory.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/Action.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/interceptor/MethodFilterInterceptorUtil.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/interceptor/ConversionErrorInterceptor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/ognl/XWorkTypeConverterWrapper.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/validator/ActionValidatorManager.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/XWorkException.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/validator/DefaultActionValidatorManager.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/interceptor/ChainingInterceptor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/interceptor/ParametersInterceptor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/interceptor/ParameterRemoverInterceptor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/interceptor/LoggingInterceptor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/interceptor/ParameterNameAware.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/ognl/OgnlTypeConverterWrapper.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/ResourceBundleTextProvider.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/validator/ShortCircuitableValidator.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/UnknownHandler.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/interceptor/ParameterFilterInterceptor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/conversion/impl/XWorkBasicConverter.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/TextProviderFactory.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/validator/Validator.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-

jar/com/opensymphony/xwork2/conversion/ObjectTypeDeterminer.java

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

//All rights reserved.

//Redistribution and use in source and binary forms, with or without

//modification, are permitted provided that the following conditions are

//Redistributions of source code must retain the above copyright notice,

//this list of conditions and the following disclaimer.

//Redistributions in binary form must reproduce the above copyright

//notice, this list of conditions and the following disclaimer in the

//documentation and/or other materials provided with the distribution.

//Neither the name of the Drew Davidson nor the names of its contributors

//without specific prior written permission.

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-

jar/com/opensymphony/xwork2/conversion/NullHandler.java

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

/*

* Copyright 1999-2005 The Apache Software Foundation.

*

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

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

* You may obtain a copy of the License at

*

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

*

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

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

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

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-

jar/com/opensymphony/xwork2/util/DomHelper.java

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

/*

* Copyright (c) 2002-2006 by OpenSymphony

* All rights reserved.

*/

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-

jar/com/opensymphony/xwork2/ognl/OgnlValueStackFactory.java

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/CompoundRoot.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/TextProviderSupport.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/ConfigurationManager.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/RegexFieldValidator.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/interceptor/PrefixMethodInvocationUtil.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/entities/ExceptionMappingConfig.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/Validations.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/interceptor/AliasInterceptor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/metadata/ExpressionValidatorDescription.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/interceptor/MethodFilterInterceptor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/ValidatorContext.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/interceptor/TimerInterceptor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/DefaultUnknownHandlerManager.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/providers/InterceptorBuilder.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/interceptor/annotations/After.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/EmailValidator.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/entities/ActionConfig.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/InvalidMetadataException.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ActionContext.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/RuntimeConfiguration.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/DelegatingValidatorContext.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/metadata/URLValidatorDescription.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/XWorkTestCase.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/entities/InterceptorConfig.java

- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/Unchainable.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/DoubleRangeFieldValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/impl/DefaultConfiguration.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/ExpressionValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ActionSupport.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/NamedVariablePatternMatcher.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/interceptor/annotations/BeforeResult.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/metadata/RequiredStringValidatorDescription.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/metadata/RequiredFieldValidatorDescription.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/TextParseUtil.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/metadata/VisitorFieldValidatorDescription.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/ConversionErrorFieldValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/conversion/impl/AnnotationXWorkConverter.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/ArrayUtils.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/logging/LoggerUtils.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/interceptor/annotations/AnnotationWorkflowInterceptor.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/ValidatorType.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/UrlValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/conversion/metadata/ConversionDescription.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/mock/MockInterceptor.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/ConversionErrorFieldValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/interceptor/AbstractInterceptor.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/ShortRangeFieldValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/metadata/ConversionErrorFieldValidatorDescription.java

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/ConfigurationException.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/conversion/impl/EnumTypeConverter.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/ValidationParameter.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/DefaultTextProvider.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ognl/accessor/XWorkEnumerationAccessor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/RepopulateConversionErrorFieldValidatorSupport.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/reflection/ReflectionContextState.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/spring/SpringObjectFactory.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/DefaultActionProxy.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/conversion/TypeConversionException.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/metadata/StringLengthFieldValidatorDescription.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/logging/commons/CommonsLogger.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/URLValidator.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/logging/Logger.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/StringLengthFieldValidator.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/EmailValidator.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/conversion/impl/DefaultObjectTypeDeterminer.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ognl/accessor/CompoundRootAccessor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/metadata/EmailValidatorDescription.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/impl/NamespaceMatcher.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/conversion/impl/XWorkConverter.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/ReferenceResolverException.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/entities/ResultConfig.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/ExpressionValidator.java

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/DefaultValidatorFileParser.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ActionEventListener.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/conversion/impl/InstantiatingNullHandler.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/StringLengthFieldValidator.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/entities/UnknownHandlerConfig.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/CustomValidator.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/conversion/annotations/Conversion.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/entities/Parameterizable.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/LocalizedTextUtil.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/AnnotationValidationConfigurationBuilder.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/conversion/annotations/ConversionType.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/VisitorFieldValidator.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/mock/MockActionInvocation.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/interceptor/StaticParametersInterceptor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/metadata/ValidatorDescription.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/mock/MockActionProxy.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/LongRangeFieldValidator.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/Element.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/metadata/FieldExpressionValidatorDescription.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/mock/MockResult.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/ContainerProvider.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/DefaultActionInvocation.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/FieldExpressionValidator.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/IntRangeFieldValidator.java

- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ognl/accessor/XWorkListPropertyAccessor.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/impl/NamespaceMatch.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ognl/accessor/ObjectProxyPropertyAccessor.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/KeyProperty.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/conversion/annotations/ConversionRule.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ognl/accessor/XWorkMapPropertyAccessor.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/RequiredFieldValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/XWorkList.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/PackageProvider.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/RequiredFieldValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/entities/ResultTypeConfig.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/XWork.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ObjectFactory.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/logging/LoggerFactory.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/Validation.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/providers/XmlHelper.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/CreateIfNull.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/providers/XmlConfigurationProvider.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/interceptor/ScopedModelDrivenInterceptor.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/entities/InterceptorMapping.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/AbstractRangeValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/interceptor/ScopedModelDriven.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/metadata/DoubleRangeFieldValidatorDescription.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/ValidationException.java

- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ognl/accessor/XWorkMethodAccessor.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/XWorkTestCaseHelper.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/VisitorFieldValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ognl/OgnlUtil.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/Validateable.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/LocaleProvider.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/DateRangeFieldValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/interceptor/I18nInterceptor.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/interceptor/annotations/Before.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ValidationAwareSupport.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/spring/interceptor/ActionAutowiringInterceptor.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/TestNGXWorkTestCase.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ognl/accessor/XWorkObjectPropertyAccessor.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/FieldExpressionValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/ConfigurationUtil.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ognl/SecurityMemberAccess.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/IntRangeFieldValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/entities/InterceptorStackConfig.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/metadata/IntRangeFieldValidatorDescription.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/FieldValidatorSupport.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ognl/ObjectProxy.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/RequiredStringValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/DefaultValidatorFactory.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/logging/commons/CommonsLoggerFactory.java

- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/conversion/annotations/TypeConversion.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/DateRangeFieldValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/entities/PackageConfig.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/metadata/DateRangeFieldValidatorDescription.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/TextProvider.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/interceptor/ModelDrivenInterceptor.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/AnnotationActionValidatorManager.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ognl/OgnlValueStack.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/conversion/impl/GenericsObjectTypeDeterminer.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/ValidatorConfig.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/XWorkMessages.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/validators/ValidatorSupport.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/Key.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/AnnotationUtils.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/ognl/accessor/XWorkIteratorPropertyAccessor.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/UnknownHandlerManager.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/annotations/DoubleRangeFieldValidator.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/logging/jdk/JdkLogger.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/ConfigurationProvider.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/validator/metadata/AbstractFieldValidatorDescription.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/spring/SpringProxyableObjectFactory.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/MockActionInvocation.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/config/entities/InterceptorListHolder.java
- * /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/interceptor/ExceptionMappingInterceptor.java


```
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/util/logging/jdk/JdkLoggerFactory.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/config/Configuration.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/mock/MockObjectTypeDeterminer.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/ognl/accessor/XWorkCollectionPropertyAccessor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/validator/validators/RequiredStringValidator.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/interceptor/Interceptor.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/validator/annotations/RegexFieldValidator.java
No license file was found, but licenses were detected in source scan.
```

```
# All rights reserved.
```

```
Found in path(s):
```

```
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/xwork-
messages.properties
No license file was found, but licenses were detected in source scan.
```

```
/*
```

```
* $Id: ActionConfigMatcher.java 1630 2007-10-05 15:10:08Z mrdon $
*
* Copyright 2003,2004 The Apache Software Foundation.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

```
Found in path(s):
```

```
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/config/impl/AbstractMatcher.java
No license file was found, but licenses were detected in source scan.
```

```
/* Copyright 2005-2006 Tim Fennell
*
```

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

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/util/ResolverUtil.java

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

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// Redistributions of source code must retain the above copyright notice,
// this list of conditions and the following disclaimer.
// Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// Neither the name of the Drew Davidson nor the names of its contributors
// without specific prior written permission.

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/conversion/TypeConverter.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/conversion/impl/DefaultTypeConverter.java

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

/*
* \$Id: WildcardHelper.java 1833 2008-06-21 09:29:39Z rainerh \$
*
* Copyright 2003-2004 The Apache Software Foundation.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/util/WildcardHelper.java
No license file was found, but licenses were detected in source scan.

/*

* \$Id: \$
*
* Copyright 2003-2004 The Apache Software Foundation.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/util/ClassPathFinder.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright 2005 The Apache Software Foundation.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/location/LocationImpl.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/location/Locatable.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/location/Location.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/location/LocationAttributes.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/location/LocationUtils.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/location/Located.java

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

/*

* Copyright (c) 2002-2003, Atlassian Software Systems Pty Ltd All rights reserved.

*

* Redistribution and use in source and binary forms, with or without modification,
* are permitted provided that the following conditions are met:

*

* * Redistributions of source code must retain the above copyright notice,

* this list of conditions and the following disclaimer.

* * Redistributions in binary form must reproduce the above copyright notice,

* this list of conditions and the following disclaimer in the documentation and/or

* other materials provided with the distribution.

* * Neither the name of Atlassian Software Systems Pty Ltd nor the names of

* its contributors may be used to endorse or promote products derived from this

* software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND

* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED

* WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE

* DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR

* ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

* (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;

* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON

* ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS

* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/profiling/ProfilingTimerBean.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-

jar/com/opensymphony/xwork2/util/profiling/UtilTimerStack.java
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/util/profiling/ObjectProfiler.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (c) 2002-2008 by OpenSymphony  
* All rights reserved.  
*/
```

Found in path(s):
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/interceptor/annotations/InputConfig.java
No license file was found, but licenses were detected in source scan.

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

Found in path(s):
* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-
jar/com/opensymphony/xwork2/util/PropertiesReader.java
No license file was found, but licenses were detected in source scan.

```
/**  
* Licensed to the Apache Software Foundation (ASF) under one or more  
* contributor license agreements. See the NOTICE file distributed with  
* this work for additional information regarding copyright ownership.  
* The ASF licenses this file to You under the Apache License, Version 2.0  
* (the "License"); you may not use this file except in compliance with  
* the License. You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software
```

- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

* /opt/cola/permits/1196577984_1629933505.28/0/xwork-2-1-2-sources-jar/com/opensymphony/xwork2/util/finder/ResourceFinder.java

1.20 commons-fileupload 1.2.1

1.20.1 Available under license :

Apache License

Version 2.0, January 2004

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a

cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

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

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

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

5. Submission of Contributions. Unless You explicitly state otherwise,

any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

Apache Commons FileUpload

Copyright 2002-2008 The Apache Software Foundation

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

1.21 slf4j 1.7.21

1.21.1 Available under license :

Copyright (c) 2004-2014 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,

EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.22 commons-lang3 3.4

1.22.1 Available under license :

<OWNER> = Regents of the University of California

<ORGANIZATION> = University of California, Berkeley

<YEAR> = 1998

In the original BSD license, both occurrences of the phrase "COPYRIGHT HOLDERS AND CONTRIBUTORS" in the disclaimer read "REGENTS AND CONTRIBUTORS".

Here is the license template:

Copyright (c) <YEAR>, <OWNER>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the <ORGANIZATION> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.23 geronimoj2eemanagement 1.0.1

1.23.1 Available under license :

Apache Geronimo

Copyright 2003-2008 The Apache Software Foundation

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

and conversions to other media types.

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

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

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

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

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

1.24 classmate 1.1.0

1.24.1 Available under license :

Java ClassMate library was originally written by Tatu Saloranta (tatu.saloranta@iki.fi)

Other developers who have contributed code are:

* Brian Langel

This copy of Java ClassMate library is licensed under Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

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

1.25 activemq-openwire-legacy 5.8.0

1.25.1 Available under license :

ActiveMQ :: Openwire Legacy Support
Copyright 2005-2013 The Apache Software Foundation

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

"Derivative Works" shall mean any work, whether in Source or Object

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a

file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

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

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

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

©2021 Cisco Systems, Inc. All rights reserved.