

Apache Axis

Axis scenario

[Back to DRLVM Test Tracking](#)

Summary

[Status](#)

[Building Axis scenario](#)

[Running Axis scenario](#)

[Old Information](#)

Status

Axis scenario is included into the regular automated snapshot testing.

[Back to Summary](#)

Building Axis scenario

Here is the sequence you need to path through to run Axis from BTI from scratch on your system:

1) check-out bti infrastructure:

```
cd <bti_folder>
svn co https://svn.apache.org/repos/asf/harmony/enhanced/buildtest/trunk/infra
```

2) '<bti_folder>/infra/adaptors' directory will contain all testing suites - installing all of them might take signifcant amount of time, so you may just remove all the directories except 'axis2_app' one.

3) modify '<bti_folder>/infra/framework.local.properties' and set the following property there:

```
drlvm.parameters.shared.jre.dir=<path to the jre under test>
```

[Back to Summary](#)

Running Axis scenario

To run the resulted stuff:

```
cd <bti_folder>
ant install
ant setup
ant -Dtest.suites=axis2_app run
```

Results could be observed at <bti_folder>/infra/build/results/axis2_app folder. But if the result of last commands was 'BUILD SUCCESSFUL' then the scenario has passed.

[Back to Summary](#)

Old Information

Update: 2nd Nov 2006 Debug on IBM VME

Failures

test.utils.cache.TestJavaClass@testGetMethod	reflect and check the synchronized keyword
--	--

test.utils.cache.TestJavaMethod@testGetMethodWithVectorMethods	reflect and check the synchronized keyword
test.utils.TestJavaUtils@testXmlNameToJava Failure	reflect and check the synchronized keyword

Errors

test.encoding.TestString2@testFrenchAccents	UnicodeLittle
test.encoding.TestString2@testFrenchAccents2	
test.encoding.TestString2@testGermanUmlauts	
test.encoding.TestString2@testSimpleString	
test.encoding.TestString2@testStringWithApostrophes	
test.encoding.TestString2@testStringWithEntities	
test.encoding.TestString2@testStringWithLeadingAndTrailingSpaces	
test.encoding.TestString2@testStringWithRawEntities	
test.encoding.TestString2@testWelcomeUnicode	
test.encoding.TestString2@testWelcomeUnicode2	
test.encoding.TestString2@testWelcomeUnicode3	
test.encoding.TestString2@testWhitespace	
test.saaj.TestMessageProperty@testEncodingPropertyUTF16	

Update: 27th Oct 2006

I've tried the unit tests of Axis 1.4 final with Harmony and IBM VME.

- you should have a web server running on the localhost at port 8080. If your server is on a different port, replace references to 8080 to your own port number.
- download the axis src package , copy the \$axis/webapps/axis directory to tomcat/webapps
- adds xml-apis.jar and xercesImpl.jar to the AXISCLASSPATH so that Axis can find the parser
setClassPath :
set AXIS_HOME=c:\axis
set AXIS_LIB=%AXIS_HOME%\lib
set AXISCLASSPATH=%AXIS_LIB%\axis.jar;%AXIS_LIB%\commons-discovery.jar;
%AXIS_LIB%\commons-logging.jar;%AXIS_LIB%\jaxrpc.jar;%AXIS_LIB%\saaj.jar;
%AXIS_LIB%\log4j-1.2.8.jar;%AXIS_LIB%\xml-apis.jar;%AXIS_LIB%\xercesImpl.jar
- start the web server
- Navigate to <http://127.0.0.1:8080/axis/>, you should now see an Apache-Axis start page. If you do not, then the webapp is not actually installed, or the appserver is not running.
- Validate Axis with happyaxis : <http://localhost:8080/axis/happyaxis.jsp> You must not proceed until all needed libraries can be found, and this validation page is happy.
- run : ant (compile) , ant junit, see the %axis%/docs/install.html for more info.

the result is,

Click the number for detail

ClassLib	OS	VM	SVN reversion	Tests	Failure s	Error s	Success	Time(s)
RI1.5.07	winxp sp2	NA	NA	718	0	0	100%	69.840
Harmony	winxp sp2	IBM	468122	718	3	14	97.63%	74.739

Update: 22nd June 2006

As of 22nd June 2006, the Apache Axis welcome page now runs with Harmony and the IBM VME. The validation page does complain about the compiler /tools.jar but I think it should be possible to work around this since Tomcat 5.5 supports the Eclipse JDT Compiler.

Update: 8th June 2006

List of classes still required as of 8th June 2006:

```
java/awt/AWTPermission
java/awt/MenuContainer
java/awt/Panel
java/awt/datatransfer/DataFlavor
java/awt/datatransfer/Transferable
java/awt/image/ImageObserver
javax.accessibility/Accessible
```

I generated a list of required classes for Axis by taking axis-bin-1_3.tar.gz and installing it under tomcat 5.0.30. Using a browser, I enabled the admin service and clicked on all of the links on the Axis welcome page.

I then compared the resulting list of 397 JRE classes against those found in the jars generated by a build of Classlib r386322. 361 classes were found. The list of classes still required are:

```
java/awt/AWTPermission
java/awt/Component
java/awt/Container
java/awt/MenuContainer
java/awt/Panel
java/awt/datatransfer/DataFlavor
java/awt/datatransfer/Transferable
java/awt/image/ImageObserver
java/beans/BeanDescriptor
java/beans/BeanInfo
java/beans/EventSetDescriptor
java/beans/FeatureDescriptor
java/beans/IndexedPropertyDescriptor
java/beans/IntrospectionException
java/beans/Introspector
java/beans/MethodDescriptor
java/beans/PropertyChangeEvent
java/beans/PropertyChangeListener
java/beans/PropertyChangeListenerProxy
java/beans/PropertyChangeSupport
java/beans/PropertyDescriptor
java/beans/PropertyVetoException
java/beans/SimpleBeanInfo
java/rmi/RemoteException
java/util/logging/ConsoleHandler
java/util/logging/ErrorHandler
java/util/logging/Formatter
java/util/logging/Handler
java/util/logging/Level
java/util/logging/LogManager
java/util/logging/LogRecord
java/util/logging/Logger
java/util/logging/LoggingPermission
java/util/logging/SimpleFormatter
java/util/logging/StreamHandler
javax.accessibility/Accessible
```

[Back to Summary](#) [Back to DRLVM Test Tracking](#)