

Testing WSIF

1 Building the unit tests

TODO

2 Setting up your environment to run the tests

TODO

3 Running the tests

WSIF has a number of unit tests found in the tests directory. These use Junit (currently Junit 3.7). WSIF unit tests are loosely themed and test function in multiple providers. For instance `jms.JmsTest` tests `jms:address`, `jms:property` and `jms:propertyValue` tags across the `SoapJms`, `AxisJms` and `NativeJms` providers. You can either run a specific unit test or run all of them by running `util.WSIFTestRunner`. `wsif.test.properties` has `wsif.test.components=ejb=on,jms=on, remotewsdL=on,jndi=on,fix=on,async=on` which allows you to switch off areas across all unit tests. So because I rarely change the EJB provider, I run with `ejb=off` so when I run `WSIFTestRunner` none of the EJB tests get run. This can be dangerous and it is generally better to run all the unit tests. Individual unit tests can call `TestUtilities.areWeTesting("ejb")` for instance. There are various listeners needed to run the unit tests (`JMS2HTTPBridge`, `JMSAsyncListener`, `NativeJMSRequestListener`) and unit tests automatically start and stop the listeners that they need. All the tests need to find WSDL and other files on the local filesystem. This is done by looking for the WSIF `src`, `samples` and `tests` directories under the directory specified by the `wsif.path` property in `wsif.test.properties`. So I have `wsif.path=c:\\wsif`. The setup required to run the unit tests is based on the various providers being used.

For Soap and Axis over Http, find all files called `DeploymentDescriptor.xml` in the `samples` and `tests` directories and deploy them to your favourite web server. The web server must be running on your localhost using port 8080. In the future `localhost:8080` should be a property loaded from `wsif.test.propeerties`. The level of soap or axis used on your server does not have to be the same as the level used by WSIF. I use soap 2.2 on tomcat so I issue `java org.apache.soap.server.ServiceManagerClient http://localhost:8080/soap/servlet/rpcrouter deploy "samples\\services\\addressbook\\DeploymentDescriptor.xml"`

There is no setup to do to run tests under the Java provider.

To run tests using the EJB provider, ... todo ???

There are three providers that are enabled for Jms : Soap, Axis and NativeJms. Asynchronous support is currently only supported over Jms, so to test out async you need Jms. Obviously you need a Jms implementation. WSIF uses JNDI to lookup Jms queues and queue connection factories. Topics are not supported at the moment. The batch files `samples\wsifjmssetup.bat` and `samples\wsifjms tidy.bat` create and delete the JNDI definitions for the Jms queues and queue connection factory needed to run the samples and some of the unit tests. I use IBM MQSeries as my Jms implementation so the batch files `samples\wsifmqsetup.bat` and `samples\wsifmq tidy.bat` create and delete the MQSeries queues needed to run the samples and some of the unit tests. The batch files `tests\wsiftestjmssetup.bat` and `tests\wsiftestmqsetup.bat` create the JNDI definitions and the MQSeries queues needed to run the unit tests. No code in WSIF or in the unit tests is MQ-specific, except for these mq batch files and `WSIFJMSFinderForMq`. The unit tests also assume the local queue manager is being used. The unit tests also assume that the JNDI database is on the filesystem in `\JNDI-Directory`.

WSIF provides a `JMS2HTTPBridge` which takes soap messages from a Jms queue, routes them to a url and sends the response message back to a `replyTo` queue. The `JMS2HTTPBridge` is intended as a test tool only. So unit tests that test `SoapJms` and `AxisJms` start the `JMS2HTTPBridge` automatically. `NativeJms` requests are served by the `NativeJMSRequestListener` which implements the backend service for our unit tests.

Specifying `remotewsdl=on` in `wsif.test.components` will enable the tests in the `wsdl` package to be run. These tests test out loading `wsdl` from a url, from a jar file and test out imports. So you need `wsdl\ClassLoaderTest.jar` on your classpath, and `http://localhost:8080/wsdl/AddressBook.wsdl` needs to be available.

The intention is for all unit tests to be run by `WSIFTestRunner` so everyone runs them all automatically. Unfortunately there are a few tests which cannot be run in this way (or rather I could not make them run automatically). An example of this is the `translated.messages` tests, because these need to set the locale, and I could not reset the locale multiple times in the same test.

The intention is for everyone to make sure that all the unit tests are kept working all the time, so no regressions find their way in. If you add in new code (or even fixes) into WSIF, please add in new unit tests, or extend existing tests to cover the new code you've added. Otherwise tomorrow someone may unknowingly break your code. If you add in new unit tests please add them to `WSIFTestRunner` so that everyone will automatically run them as well. I guess when you make a change that requires a service to be redeployed or a queue definition to be

Testing WSIF

added or changed, please notify everyone on axis-dev so that everyone can update their setup.