

Setting up environment to program Web Services

1. Requirement

1. Hardware

- Laptop/Desktop
- Network card
- Minimum 256MB memory (512MB recommended)

2. Operating System

- Windows 2000/XP professional
- Linux

3. Web Services Software

- Java 2 SDK v1.4.1
- Apache Tomcat 4.1.24
- Apache AXIS 1.1
- Apache XML security 1.0.4

2. Installation

Apache Axis installation for Windows

This document details how to install Apache Axis as a Web Application under Apache Tomcat.

You will need to download and install the following software:

1. Java 2 SDK v1.4.1, <http://java.sun.com/j2se/1.4.1/download.html>
2. Apache Tomcat v4.1.24, <http://jakarta.apache.org/builds/jakarta-tomcat-4.0/release/v4.1.24/bin/jakarta-tomcat-4.1.24.exe>
3. XML Security v1.0.4, http://www.apache.org/dist/xml/security/java-library/xmlsecurity-bin-1_0_4.zip
4. Axis v1.1, http://ws.apache.org/axis/dist/1_1/axis-1_1.zip

1. Java 2 SDK

- Run the downloaded executable (j2sdk-1_4_1-windows-i586.exe) which will install the
- SDK in C:\j2sdk1.4.1. Set the JAVA_HOME environment variable to point to this directory as follows:
- Click on START->CONTROL PANEL->SYSTEM
- Click on the Advanced tab
- Click on the Environment Variables button
- Click on the New... button in the user variable section and enter the details
- Add the Java binaries to your PATH variable in the same way by setting a user variable called PATH with the value "%PATH%;C:\j2sdk1.4.1\bin"

2. Apache Tomcat

- Run the downloaded executable (jakarta-tomcat-4.1.24.exe), and assume the installation directory is C:\jakarta-tomcat-4.1.24.

- Edit C:\jakarta-tomcat-4.1.24\conf\tomcat-users.xml and create an "admin" and "manager" role as well as a user with both roles. The contents of the file should be similar to:


```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <role rolename="manager"/>
  <role rolename="admin"/>
  <user username="myuser" password="mypass"
    roles="admin,manager"/>
</tomcat-users>
```
- Start Tomcat by running C:\jakarta-tomcat-4.1.24\bin\startup.bat and test it by browsing <http://localhost:8080/>
- Stop Tomcat by running C:\jakarta-tomcat-4.1.24\bin\shutdown.bat.

3. XML Security

- Download and unzip http://www.apache.org/dist/xml/security/java-library/xmlsec-bin-1_0_4.zip
- Copy xml-sec.jar to C:\axis-1_1\lib\
- Set-up your CLASSPATH environment variable to including the following:


```
C:\axis-1_1\lib\xml-sec.jar;
```

4. Apache Axis

- Unzip the downloaded Axis archive to C: (this will create a directory C:\axis-1_1).
- Extract the file xmlsec.jar from the downloaded security archive to C:\axis-1_1\webapps\axis\WEB-INF\lib.
- Set-up your CLASSPATH environment variable to including the following:
 - The current working directory
 - All the AXIS jar files as found in C:\axis-1_1\lib
 - C:\jakarta-tomcat-4.1.24\common\lib\servlet.jar
- Your CLASSPATH should therefore look something like:


```
C:\axis-1_1\lib\axis.jar;
C:\axis-1_1\lib\axis-ant.jar;
C:\axis-1_1\lib\commons-discovery.jar;
C:\axis-1_1\lib\commons-logging.jar;
C:\axis-1_1\lib\jaxrpc.jar;
C:\axis-1_1\lib\log4j-1.2.8.jar;
C:\axis-1_1\lib\saaj.jar;
C:\axis-1_1\lib\wsdl4j.jar;
C:\axis-1_1\lib\xercesImpl.jar;
C:\axis-1_1\lib\xmlParserAPIs.jar;
C:\jakarta-tomcat-4.1.24\common\lib\servlet.jar;
C:\axis-1_1\lib\xml-sec.jar;
```
- Now tell Tomcat about your Axis web application by creating the file C:\jakarta-tomcat-4.1.24\webapps\axis.xml with the following content:


```
<Context path="/axis" docBase="C:\axis-1_1\webapps\axis" debug="0"
  privileged="true">
```

```
<Logger className="org.apache.catalina.logger.FileLogger"
  prefix="axis_log." suffix=".txt" timestamp="false"/>
</Context>
```

- Enable the Axis Admin servlet by editing C:\axis-1_1\webapps\axis\WEB-INF\web.xml and uncommenting the servlet mapping for /servlet/AdminServlet.
- You can now restart Tomcat and browse to <http://localhost:8080/axis>. Click on the [Validate](#) link to test that all libraries are accessible to Axis (both core and optional).
- Test your installation by click <http://localhost:8080/axis/happyaxis.jsp>
- Compare it to <http://geon05.sdsc.edu:8080/axis/happyaxis.jsp>

Deploy a Sample Web service packaged within Axis installations

Deploy one of the sample Web Services to test the system and to create the C:\axis-1_1\webapps\axis\WEB-INF\server-config.wsdd file. From C:\axis-1_1 issue the command (on one line):

```
java org.apache.axis.client.AdminClient -
lhttp://localhost:8080/axis/services/AdminService samples/stock/deploy.wsdd
```

This should return the following :

```
.- Processing file samples/stock/deploy.wsdd
.- <Admin>Done processing</Admin>
```

Finally test the service that's just been deployed by issuing the following command (on one line):

```
java samples.stock.GetQuote -lhttp://localhost:8080/axis/servlet/AxisServlet
user1 -wpass1 XXX This should return:
XXX: 55.25
```

A useful article can be found at <http://www.onjava.com/lpt/a/1578> for information on creating your first Web Service and client.

SOAP Monitor (This software allows you to see the soap input and output messages interaction between Web Service and Soap Client)

When you've deployed your first Web Service the server-config.wsdd file is created in C:\axis-1_1\webapps\axis\WEB-INF. If you want to use the SOAPMonitor then you'll have to compile it and manually edit this file to include service and handler declarations. Change directory to C:\axis-1_1\webapps\axis and compile SOAPMonitorApplet.java as follows:

```
javac SOAPMonitorApplet.java
```

Now edit C:\axis-1_1\webapps\axis\WEB-INF\server-config.wsdd and make the following additions:

Add `<handler type="soapmonitor"/>` as the first element under both the

<requestFlow> and <responseFlow> child elements of <globalConfiguration> so that you end up with the following basic structure which may include other elements:

```
<globalConfiguration>
  <parameter name=.../>
  <requestFlow>

    <handler type="soapmonitor"/>
  </requestFlow>
  <responseFlow>

    <handler type="soapmonitor"/>
  </responseFlow>
</globalConfiguration>
```

Then add the handler definition amongst the other handlers:

```
<handler name="soapmonitor"
  type="java:org.apache.axis.handlers.SOAPMonitorHandler"
  > <parameter name="wsdlURL"
  value="/axis/SOAPMonitorService-
  impl.wsdl"/> <parameter name="serviceName"
  value="SOAPMonitorService"/> <parameter name="namespace"
  value="http://tempuri.org/wsdl/2001/12/SOAPMonitorServiceimpl"/>
  <parameter name="portName" value="Demo"/> </handler>
```

Restart Tomcat and browse to <http://localhost:8080/axis/SOAPMonitor>. Run a Web Service client and the request and response SOAP messages will be captured and displayed.