Upgrading to GlassFish 4

Near to the start of the project I was asked by Andreas Hueni to check the compatibility of the current implementation of the system with glassfish 4 from their previous install of glassfish 3.1.2.2.  
After a couple of weeks of spike testing the graphical user interface design I decided that it would be a worthwhile venture to upgrade glassfish as it would allow me to find any bugs with the system as I was working throughout the project.

## Downloading and installing glassfish

1. Within the eclipse IDE select the help tab along the top of the menu bar.
2. Select eclipse marketplace.
3. Search for glassfish, then choose and download the tools for your glassfish version.
4. Select window in the menu bar and head to preferences.
5. Select server and choose runtime environments.
6. Add a new runtime environment glassfish 4 and click create new local server.
7. Set the JDK to 1.7.? (Whatever your most recent version is).
8. Runtime name should be set to GlassFish 4.
9. Choose a local directory to install your GlassFish server.
10. Click install server and let this complete.

## Updating GlassFish 4 to include JAX-RS 2.0

Glassfish 4 with the current implementation of SPECCHIO will not work due to its incompatibility with JAX-RS 2.0. Following the steps below the user will be able to upgrade the standard version of glassfish install with the newest version of JAX and glassfish (along with any minor updates that may come in from the time of writing)

1. Right click and run your new glassfish installation.
2. Head to the glassfish tab and click glassfish update centre.
3. This will then prompt you to install the update centre software through the command line of the eclipse launcher. (This process will take a long time to complete)
4. Right click the glassfish 4 implementation under the server tab within eclipse and select the glassfish tab and then select update centre again.
   1. If this does not work head to your installation folder of glassfish and double click on your bin folder, within this folder there will be a batch file called updatetool. Run this file and it will open a command window then type y when instructed to do so.
5. This new window that pops up will have application images in the left hand pane, within this tab click on the available updates tab.
6. Install all updates as this is not done during the download through eclipses marketplace. (Must ensure that the JAX-RS 2.0 update is within this list of updates)

## Configuring a MySQL Connection Pool in GlassFish

GlassFish provides a graphical administration interface at <http://localhost:4848>. You can also configure it by directly editing the file $GLASSFISH\_DOMAIN\_HOME/config/domain.xml.

1. Copy the MySQL JDBC connector (mysql-connector-java-5.x.x-bin.jar) into $GLASSFISH\_DOMAIN\_HOME/lib/ext and re-start GlassFish.
2. Create a JDBC connection pool using the Glassfish administration interface:
   1. Visit Resources > JDBC > JDBC Connection Pools.
   2. Click “New”.
   3. Set “Pool Name” to specchio\_web\_pool.
   4. Set “Resource Type” to javax.sql.DataSource.
   5. Set “Database Driver Vendor” to “MySQL”.
   6. Test using the “Ping” button on the connection pool’s main page
3. Check database connection information for specchio\_web\_pool:
   1. Visit Resource > JDBC > JDBC Connection Pools > specchio\_web\_pool > Additional Properties.
   2. If using a URL to configure the database information, set BOTH Url and URL to jdbc:mysql://localhost:3306/specchio.
   3. Otherwise, set serverName, databaseName and port appropriately.
   4. Check that the settings for user and password match those used in the database configuration scripts used when initialising the SPECCHIO database (above).
4. Enable “Match Connections” for specchio\_web\_pool (Resource > JDBC > JDBC Connection Pools > specchio\_web\_pool > Advanced)
5. Create a JDBC Resource using the Glassfish administration interface:
   1. Visit Resources > JDBC > JDBC Resources.
   2. Click “New”
   3. Set “JNDI Name” to jdbc/specchio.
   4. Set “Pool Name” to specchio\_web\_pool.
   5. “Description” can be anything.

## Configuring User Authentication

The following instructions assume that your SPECCHIO applicaton “jar” is installed in a folder called $SPECCHIO\_CLIENT\_HOME. If using Eclipse, $SPECCHIO\_CLIENT\_HOME is the root folder of the “SPECCHIO Web Client” work space.

1. Add your GlassFish instance’s public key to the SPECCHIO Web Client’s Java key store[[1]](#footnote-1):
   1. $ keytool -export -alias s1as -keystore $GLASSFISH\_DOMAIN\_HOME/config/keystore.jks -file glassfish.crt [no password required]
   2. $ keytool -import -alias <any string> -file glassfish.crt -keystore $SPECCHIO\_CLIENT\_HOME/specchio.keystore [use password "specchio"]
2. Set up JDBC Realm authentication in Glassfish (note that the labels on GlassFish’s realm configuration page are somewhat misleading):
   1. Visit Configurations > server\_config > Security > Realms.
   2. Click “New”.
   3. Set “Name” to specchioRealm.
   4. Set “Class Name” to com.sun.enterprise.security.auth.realm.jdbc.JDBCRealm.
   5. Set “JAAS Context” to jdbcRealm.
   6. Set “JNDI” to jdbc/specchio.
   7. Set “User Table” to specchio.specchio\_user.
   8. Set “User Name Column” to user.
   9. Set “Password Column” to password.
   10. Set “Group Table” to specchio.specchio\_user\_group. (This is actually called the “user group table” in other documentation.)
   11. Set “Group Table User Name Column” to user.
   12. Set “Group Name Column” to group\_name.
   13. Set “Database User” and “Database Password” to the SPECCHIO administrator’s username (usually sdb\_admin) and password.
   14. Set both “Digest Algorithm” and “Password Encryption Algorithm” to MD5.

## Installing the SPECCHIO Web Services and OAI Service

1. Deploy the SPECCHIO Web Service
   1. *Using Eclipse:* Eclipse deploys the application automatically if using the project checked out from Subversion; use the “Server” tab on the “Java EE” view to start and stop the GlassFish server.
   2. *From the command line:* sudo /opt/glassfish3/glassfish/bin/asadmin deploy --force specchio-services.war
   3. *Using GlassFish’s administrative interface*: Choose “Deploy an application” from the home page, then specify the “war” file as the “packaged file to be uploaded to the server”. Check the value for “context root” matches the path to be used by the client, usually specchio\_service.
2. Optional: Deploy the jOAI service:
   1. Unzip the joai-3.x.x.x.zip package, replacing the x’s with the appropriate version number.
   2. *From the command line*: sudo /opt/glassfish3/glassfish/bin/asadmin deploy --force joai-3.x.x.x/oai.war
   3. *Using GlassFish’s administrative interface*: Choose “Deploy an application” from the home page, then specify the joai-3.x.x.x/oai.war file as the “packaged file to be uploaded to the server”. Leave all other settings at their default values.

## Configuring the web application to include glassfish 4’s default implementation of servlets.

1. Within the eclipse client with the project installed navigate through the package explorer to SPECCHIO Web Application
   1. SPECCHIO Web Application
   2. WebContent
   3. WEB-INF
   4. Right click web.xml and open with text editor
2. Within this new window copy and paste the following lines of xml.
   1. *(This step may not be required and may come with the default installation of SPECCHIO from the git repository)*

<?xml version="1.0" encoding="UTF-8"?>

<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://java.sun.com/xml/ns/javaee" xmlns:web="http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd" id="WebApp\_ID" version="2.5">

<display-name>ch.specchio.service</display-name>

<context-param>

<param-name>ANDSXMLFileLocation</param-name>

<param-value>/tmp/specchio\_data/files/rifcs</param-value>

</context-param>

<servlet>

<servlet-name>Jersey</servlet-name>

<servlet-class>org.glassfish.jersey.servlet.ServletContainer</servlet-class>

<init-param>

<param-name>jersey.config.server.provider.packages</param-name>

<param-value>ch.specchio.services</param-value>

</init-param>

<init-param>

<param-name>au.org.ands.researchdata</param-name>

<param-value>enabled</param-value>

</init-param>

<init-param>

<param-name>END\_USER\_LICENSE</param-name>

<param-value>disabled</param-value>

</init-param>

<init-param>

<param-name>END\_USER\_LICENSE\_SHORT\_TEXT</param-name>

<param-value>YOU ACKNOWLEDGE THAT YOU HAVE READ THE PROGRAM LICENSE AGREEMENT, UNDERSTAND IT AND AGREE TO BE BOUND BY ITS TERMS AND CONDITIONS. YOU FURTHER AGREE THAT IT IS THE COMPLETE AND EXCLUSIVE STATEMENT OF THE AGREEMENT BETWEEN YOU AND UNIVERSITY OF ZURICH WHICH SUPERSEDES ANY PROPOSAL OR PRIOR AGREEMENT, ORAL OR WRITTEN, AND ANY OTHER COMMUNICATIONS RELATING TO THE SUBJECT MATTER OF THIS AGREEMENT.</param-value>

</init-param>

<init-param>

<param-name>END\_USER\_LICENSE\_URL</param-name>

<param-value>http://specchio.ch/SPECCHIO\_UZH\_License\_V1.pdf</param-value>

</init-param>

<load-on-startup>1</load-on-startup>

</servlet>

<servlet-mapping>

<servlet-name>Jersey</servlet-name>

<url-pattern>/\*</url-pattern>

</servlet-mapping>

<login-config>

<auth-method>BASIC</auth-method>

<realm-name>specchioRealm</realm-name>

</login-config>

<security-constraint>

<web-resource-collection>

<web-resource-name>SPECCHIO Web Services</web-resource-name>

<description/>

<url-pattern>/ands/\*</url-pattern>

<url-pattern>/browser/\*</url-pattern>

<url-pattern>/campaign/\*</url-pattern>

<url-pattern>/instrumentation/\*</url-pattern>

<url-pattern>/metadata/\*</url-pattern>

<url-pattern>/spectral\_file/\*</url-pattern>

<url-pattern>/spectrum/\*</url-pattern>

<url-pattern>/user/\*</url-pattern>

</web-resource-collection>

<auth-constraint>

<role-name>admin</role-name>

<role-name>user</role-name>

</auth-constraint>

</security-constraint>

</web-app>

1. [↑](#footnote-ref-1)