

Installing the Tuscany Plugin for Geronimo

This page is made obsolete by <http://cwiki.apache.org/confluence/display/TUSCANYWIKI/Installing+Tuscany+Plugin+for+Geronimo>.

The Tuscany Plugin for Geronimo is being developed and tested on Geronimo v2.1.3.

Installing on Geronimo Tomcat server

Follow the steps below to install the plugin on Geronimo v2.1.3 with Tomcat as the web container.

1. Edit <GERONIMO_HOME>\var\config\config.xml
 - Add the gbean xml-fragment given in Listing-1 under "org.apache.geronimo.configs/tomcat6-deployer/2.1.3/car" module.
 - Add the gbean xml-fragment given in Listing-2 under "org.apache.geronimo.configs/openejb-deployer/2.1.3/car" module.



If you are starting with a fresh Geronimo Tomcat 2.1.3 server, you can replace <GERONIMO_HOME>\var\config\config.xml with config.xml available at [this link](#)

2. Start Geronimo.
3. Download tuscany-tomcat-1.0-SNAPSHOT.car from <http://people.apache.org/repo/m2-snapshot-repository/org/apache/geronimo/plugins/tuscany-tomcat/1.0-SNAPSHOT/>
4. Deploy the plugin using the command "<GERONIMO_HOME>\bin\deploy install-plugin tuscany-tomcat-1.0-SNAPSHOT.car"

Listing-1:

```
<gbean name="TomcatWebBuilder">
  <reference name="ModuleBuilderExtensions">
    <pattern>
      <name>TuscanyModuleBuilderExtension</name>
    </pattern>
    <pattern>
      <name>PersistenceUnitBuilder</name>
    </pattern>
    <pattern>
      <name>JspModuleBuilderExtension</name>
    </pattern>
    <pattern>
      <name>MyFacesModuleBuilderExtension</name>
    </pattern>
  </reference>
</gbean>
```

Listing-2:

```
<gbean name="EJBBuilder">
  <reference name="ModuleBuilderExtensions">
    <pattern>
      <name>Axis2ModuleBuilderExtension</name>
    </pattern>
    <pattern>
      <name>AxisModuleBuilderExtension</name>
    </pattern>
    <pattern>
      <name>CorbaModuleBuilderExtension</name>
    </pattern>
    <pattern>
      <name>TuscanyModuleBuilderExtension</name>
    </pattern>
    <pattern>
      <name>PersistenceUnitBuilder</name>
    </pattern>
    <pattern>
      <name>CXFModuleBuilderExtension</name>
    </pattern>
  </reference>
</gbean>
```

Installing on Geronimo Jetty server

Follow the steps below to install the plugin on Geronimo v2.1.3 with Jetty as the web container.

1. Edit <GERONIMO_HOME>\var\config\config.xml
 - Add the gbean xml-fragment given in Listing-3 under "org.apache.geronimo.configs/jetty6-deployer/2.1.3/car" module.
 - Add the gbean xml-fragment given in Listing-2 under "org.apache.geronimo.configs/openejb-deployer/2.1.3/car" module.



If you are starting with a fresh Geronimo Jetty 2.1.3 server, you can replace <GERONIMO_HOME>\var\config\config.xml with config.xml available at [this link](#)

2. Start Geronimo.
3. Download tuscany-jetty-1.0-SNAPSHOT.car from <http://people.apache.org/repo/m2-snapshot-repository/org/apache/geronimo/plugins/tuscany-jetty/1.0-SNAPSHOT/>
4. Deploy the plugin using the command "<GERONIMO_HOME>\bin\deploy install-plugin tuscany-jetty-1.0-SNAPSHOT.car"

Listing-3:

```
<gbean name="JettyWebBuilder">
  <reference name="ModuleBuilderExtensions">
    <pattern>
      <name>TuscanyModuleBuilderExtension</name>
    </pattern>
    <pattern>
      <name>PersistenceUnitBuilder</name>
    </pattern>
    <pattern>
      <name>JspModuleBuilderExtension</name>
    </pattern>
    <pattern>
      <name>MyFacesModuleBuilderExtension</name>
    </pattern>
  </reference>
</gbean>
```

Building the plugin from source and installing the plugin



This section assumes that you have Subversion and Maven2 installed.

If you want to build the plugin from source and then install the plugin in Geronimo server, follow the steps below.

1. Checkout the plugin source from <http://svn.apache.org/repos/asf/geronimo/plugins/tuscany/trunk/>
2. Open a command window and change to the directory to which source is checked out.
3. Build the plugin using the command "mvn install"
4. Edit <GERONIMO_HOME>\var\config\config.xml
 - In Geronimo Tomcat v2.1.3 server
 - Add the gbean xml-fragment given in Listing-1 under "org.apache.geronimo.configs/tomcat6-deployer/2.1.3/car" module.
 - Add the gbean xml-fragment given in Listing-2 under "org.apache.geronimo.configs/openejb-deployer/2.1.3/car" module.
 - In Geronimo Jetty v2.1.3 server
 - Add the gbean xml-fragment given in Listing-3 under "org.apache.geronimo.configs/jetty6-deployer/2.1.3/car" module.
 - Add the gbean xml-fragment given in Listing-2 under "org.apache.geronimo.configs/openejb-deployer/2.1.3/car" module.
5. Start the Geronimo server.
6. Access <http://localhost:8080/console> and navigate to "Plugins" portlet.
7. Click "Add Repository" link under Install Geronimo Plugins section.
8. Enter the URL of your local maven2 repository and click "Add Repository" button.
 - For example, if your local maven2 repository directory is D:\m2repo use the URL file:D:/m2repo
9. In the "Repository" dropdown list, select your local maven2 repository added in the previous step and click "Show Plugins in selected repository" button.
10. Select "Geronimo :: Tuscany Plugin for Geronimo Tomcat" ("Geronimo :: Tuscany Plugin for Geronimo Jetty" in case of Geronimo Jetty server) and click **Install**.
11. In the plugin information page, click **Install** to complete the plugin installation.



If your plugin installation failed in an earlier attempt, make sure that "System Modules" portlet does not list "org.apache.geronimo.plugins/tuscany-tomcat/1.0-SNAPSHOT/car" or "org.apache.geronimo.plugins/tuscany-jetty/1.0-SNAPSHOT/car" configuration. If there is one such entry, uninstall it by clicking on "Uninstall" link.





If you want latest plugin and tuscany dependency jars to be installed during plugin installation, make sure you delete <GERONIMO_HOME>/repository/org/apache/geronimo/plugins/geronimo-tuscany* folders and <GERONIMO_HOME>/repository/org/apache/tuscany/sca folder are deleted before you proceed with the installation.

Running sample applications

Deploy SCA contribution

We have a sample sca contribution (helloworld-service) that has three Helloworld service components. The JSP, Servlet and EJB samples access the services provided by this contribution.

Follow the steps below to deploy the helloworld-service.

1. Download helloworld-service-1.0-SNAPSHOT.jar from <http://people.apache.org/repo/m2-snapshot-repository/org/apache/geronimo/plugins/helloworld-service/1.0-SNAPSHOT/>
2. Access <http://localhost:8080/console> and navigate to "Deploy New" portlet.
3. In the **Archive** field, select helloworld-service-1.0-SNAPSHOT.jar downloaded above.
4. Click **Install** to install the sca contribution.

Running Helloworld JSP sample

The sample web application helloworld-jsp has a JSP that defines an SCA reference using SCA JSP tag library. The reference is wired in web.composite. The reference is injected at runtime.

Follow the steps below to run the sample application.

1. Download helloworld-jsp-1.0-SNAPSHOT.war from <http://people.apache.org/repo/m2-snapshot-repository/org/apache/geronimo/plugins/helloworld-jsp/1.0-SNAPSHOT/>
2. Access <http://localhost:8080/console> and navigate to "Deploy New" portlet.
3. In the **Archive** field, select helloworld-jsp-1.0-SNAPSHOT.war downloaded above.
4. Click **Install** to install the web application.
5. Access <http://localhost:8080/helloworld-jsp/>

Running Helloworld Servlet sample

The sample web application helloworld-servlet has a HelloworldServlet with @Reference, @Property, @Context and @ComponentName annotations. The references are wired in web.composite. The annotated fields are injected at runtime.

Follow the steps below to run the sample application.

1. Download helloworld-servlet-1.0-SNAPSHOT.war from <http://people.apache.org/repo/m2-snapshot-repository/org/apache/geronimo/plugins/helloworld-servlet/1.0-SNAPSHOT/>
2. Access <http://localhost:8080/console> and navigate to "Deploy New" portlet.
3. In the **Archive** field, select helloworld-servlet-1.0-SNAPSHOT.war downloaded above.
4. Click **Install** to install the web application.
5. Access <http://localhost:8080/helloworld-servlet/>

Running Helloworld EJB sample

The sample EJB application helloworld-ejb has a Stateless session bean HelloworldServiceBean with @Reference, @Property, @Context and @ComponentName annotations. The references are wired in ejb-jar.composite. The annotated fields are injected at runtime. The sample web application helloworld-web has a HelloworldEjbServlet that invokes HelloworldServiceBean.

Follow the steps below to run the sample application.

1. Download helloworld-ejb-1.0-SNAPSHOT.jar from <http://people.apache.org/repo/m2-snapshot-repository/org/apache/geronimo/plugins/helloworld-ejb/1.0-SNAPSHOT/>
2. Download helloworld-web-1.0-SNAPSHOT.war from <http://people.apache.org/repo/m2-snapshot-repository/org/apache/geronimo/plugins/helloworld-web/1.0-SNAPSHOT/>
3. Access <http://localhost:8080/console> and navigate to "Deploy New" portlet.
4. In the **Archive** field, select helloworld-ejb-1.0-SNAPSHOT.jar downloaded above.
5. Click **Install** to install the ejb application.
6. In the **Archive** field, select helloworld-web-1.0-SNAPSHOT.war downloaded above.
7. Click **Install** to install the web application.
8. Access <http://localhost:8080/helloworld-web/>

Running Helloworld MDB sample

The sample EJB application helloworld-mdb-ejb has a message-driven-bean HelloworldMDB that listens on a JMS queue named NamesQueue. HelloworldMDB has @Reference, @Property, @Context and @ComponentName annotations. The references are wired in ejb-jar.composite. The annotated fields are injected at runtime. The sample web application helloworld-mdb-web has a HelloworldMdbServlet that sends the name entered in the webpage as a message to the NamesQueue. Output from HelloworldMDB can be seen in Geronimo console window.

Follow the steps below to run the sample application.

1. Download [jms-resources.xml](#)
2. Download helloworld-mdb-ejb-1.0-SNAPSHOT.jar from <http://people.apache.org/repo/m2-snapshot-repository/org/apache/geronimo/plugins/helloworld-mdb-ejb/1.0-SNAPSHOT/>
3. Download helloworld-mdb-web-1.0-SNAPSHOT.war from <http://people.apache.org/repo/m2-snapshot-repository/org/apache/geronimo/plugins/helloworld-mdb-web/1.0-SNAPSHOT/>
4. Access <http://localhost:8080/console> and navigate to "Deploy New" portlet.
5. In the **Archive** field, select <GERONIMO_HOME>\repository\org\apache\geronimo\modules\geronimo-activemq-ra\2.1.3\geronimo-activemq-ra-2.1.3.rar
6. In the **Plan** field, select jms-resources.xml downloaded above.
7. Click **Install** to complete creating JMS resources.
8. In the **Archive** field, select helloworld-mdb-ejb-1.0-SNAPSHOT.jar downloaded above.
9. Click **Install** to install the ejb application.
10. In the **Archive** field, select helloworld-mdb-web-1.0-SNAPSHOT.war downloaded above.
11. Click **Install** to install the web application.
12. Access <http://localhost:8080/helloworld-mdb-web/>

Running Helloworld Filter sample

The sample web application helloworld-filter has a filter HelloworldFilter with @Reference, @Property, @Context and @ComponentName annotations. The references are wired in web.composite. The annotated fields are injected at runtime.

Follow the steps below to run the sample application.

1. Download helloworld-filter-1.0-SNAPSHOT.war from <http://people.apache.org/repo/m2-snapshot-repository/org/apache/geronimo/plugins/helloworld-filter/1.0-SNAPSHOT/>
2. Access <http://localhost:8080/console> and navigate to "Deploy New" portlet.
3. In the **Archive** field, select helloworld-filter-1.0-SNAPSHOT.war downloaded above.
4. Click **Install** to install the web application.
5. Access <http://localhost:8080/helloworld-filter/>

Running Helloworld Event Listener sample

The sample web application helloworld-listener has a ServletContextListener, a ServletContextAttributeListener, a ServletRequestListener, a ServletRequestAttributeListener, an HttpSessionListener, and an HttpSessionAttributeListener with @Reference, @Property, @Context and @ComponentName annotations. The references are wired in web.composite. The annotated fields are injected at runtime.

Follow the steps below to run the sample application.

1. Download helloworld-listener-1.0-SNAPSHOT.war from <http://people.apache.org/repo/m2-snapshot-repository/org/apache/geronimo/plugins/helloworld-listener/1.0-SNAPSHOT/>
2. Access <http://localhost:8080/console> and navigate to "Deploy New" portlet.
3. In the **Archive** field, select helloworld-listener-1.0-SNAPSHOT.war downloaded above.
4. Click **Install** to install the web application.
5. Access <http://localhost:8080/helloworld-listener/>

Running Helloworld JSF sample

The sample JSF application helloworld-jsf has a controller class HelloWorldController with @Reference, @Property, @Context and @ComponentName annotations. The references are wired in web.composite. The annotated fields are injected at runtime.

Follow the steps below to run the sample application.

1. Download helloworld-jsf-1.0-SNAPSHOT.war from <http://people.apache.org/repo/m2-snapshot-repository/org/apache/geronimo/plugins/helloworld-jsf/1.0-SNAPSHOT/>
2. Access <http://localhost:8080/console> and navigate to "Deploy New" portlet.
3. In the **Archive** field, select helloworld-jsf-1.0-SNAPSHOT.war downloaded above.
4. Click **Install** to install the web application.
5. Access <http://localhost:8080/helloworld-jsf/>