



JSF Bridge v.1.0.3

Project Documentation

Table of Contents

1 JSF Bridge	
1.1 Summary	1
1.2 Guide	2

1.1 Summary

About Apache Portals JSF Bridge

The JSF Bridge allows [Java Server Faces](#) to be run as [JSR-168](#) compliant Portlets with very little change made to the JSF application.

Existing or new JSF Applications can be transparently deployed as Portlet Application or Web Application.

For examples of how to use the JSF bridge, please go to the [JSF Portlet Demo](#) application which J2's JSF Portlet Bridge.

J2's portlet bridge provides a generic JSF portlet that can be extended by your custom application or configured in *portlet.xml* in order to run a JSF application.

The `<init-param>` element of the *portlet.xml* can be configured to support multiple view when using *org.apache.portals.bridges.jsf.FacesPortlet* as a portlet class.

<code><init-param></code>	Description
ViewPage	The default page in view mode.
EditPage	The default page in edit mode.
HelpPage	The default page in help mode.
CustomPage	The default page in custom mode.

1.2 Guide

Jetspeed Simple JSF Portlet Guide

This guide provides a tutorial for creating a very simple JSF portlet with one template in the portlet view mode.

1. The Portlet Class

Create the file JSFSimplest.java in a directory called jsf-simplest/WEB-INF/classes:

```
public class JSFSimplest extends org.apache.portals.bridges.jsf.FacesPortlet
{
    public void doView(javax.portlet.RenderRequest request,
        javax.portlet.RenderResponse response)
        throws javax.portlet.PortletException, java.io.IOException
    {
        super.doView(request, response);
    }
}
```

Compile the class in the jsf-simplest/WEB-INF/classes directory using the command,

```
javac -cp
portlet-api-1.0.jar:portals-bridges-jsf-1.0.jar:portals-bridges-common-1.0.jar
JSFSimplest.java
```

2. The portlet.xml

Create the file portlet.xml in the jsf-simplest/WEB-INF directory.

```
<?xml version="1.0" encoding="UTF-8"?>
<portlet-app id="jsfsimplest" version="1.0">
  <portlet id="JSFSimplest">
    <portlet-name>JSFSimplestPortlet</portlet-name>
    <display-name>JSF Simplest Display Name</display-name>
    <portlet-class>JSFSimplest</portlet-class>
    <init-param>
      <name>ViewPage</name>
      <value>/WEB-INF/view/view.jsp</value>
    </init-param>
  </portlet>
</portlet-app>
```

```

    </init-param>
    <supports>
      <mime-type>text/html</mime-type>
      <portlet-mode>VIEW</portlet-mode>
    </supports>
    <supported-locale>en</supported-locale>
    <supported-locale>fr</supported-locale>
    <portlet-info>
      <title>JSF Simplest Title</title>
      <short-title>JSF Simplest Short Title</short-title>
    </portlet-info>
  </portlet>
</portlet-app>

```

3. The web.xml

Create the file web.xml in the jsf-simplest/WEB-INF directory.

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
    "http://java.sun.com/dtd/web-app_2_3.dtd">

<web-app>
  <display-name>JSF Simplest</display-name>
  <description>The world's simplest JSF portlet</description>

  <!-- Faces Config -->
  <context-param>
    <param-name>javax.faces.application.CONFIG_FILES</param-name>
    <param-value>/WEB-INF/faces-config.xml</param-value>
  </context-param>

  <!-- Faces Servlet -->
  <servlet>
    <servlet-name>Faces Servlet</servlet-name>
    <servlet-class>javax.faces.webapp.FacesServlet</servlet-class>
  </servlet>

  <!-- Faces extension mapping -->
  <servlet-mapping>
    <servlet-name>Faces Servlet</servlet-name>
    <url-pattern>*.jsf</url-pattern>
  </servlet-mapping>
</web-app>

```

4. The View

Create the view.jsp file in the jsf-simplest/WEB-INF/view directory.

```

<%@ taglib uri="http://java.sun.com/jsf/core" prefix="f" %>
<%@ taglib uri="/WEB-INF/tld/portlet.tld" prefix='portlet'%>

```

```
<portlet:defineObjects/>
<f:view>
  <h2>A JSF Portlet</h2>
  Hello
  <% if ( renderRequest.getUserPrincipal() == null ) { %>
    guest
  <% } else { %>
    <%=renderRequest.getUserPrincipal().getName()%>
  <% } %>
</f:view>
```

5. The faces-config.xml

Create a faces-config.xml file in the jsf-simplest/WEB-INF directory.

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE faces-config PUBLIC
  "-//Sun Microsystems, Inc.//DTD JavaServer Faces Config 1.1//EN"
  "http://java.sun.com/dtd/web-facesconfig_1_1.dtd">

<faces-config>

  <application>
    <locale-config>
      <default-locale>en</default-locale>
      <supported-locale>fr</supported-locale>
    </locale-config>
  </application>

</faces-config>
```

6. The Dependency JARs

Copy the dependencies to the WEB-INF/lib directory. These jars should be in your Maven repository. If so executing these commands in the lib directory will set up the dependencies for you.

```
ln -s ~/.maven/repository/commons-beanutils/jars/commons-beanutils-1.7.0.jar
ln -s ~/.maven/repository/commons-collections/jars/commons-collections-3.1.jar
ln -s ~/.maven/repository/commons-digester/jars/commons-digester-1.7.jar
ln -s ~/.maven/repository/myfaces/jars/myfaces-api-1.1.0.jar
ln -s ~/.maven/repository/myfaces/jars/myfaces-impl-1.1.0.jar
ln -s ~/.maven/repository/myfaces/jars/tomahawk-1.1.0.jar
ln -s
~/.maven/repository/org.apache.portals.bridges/jars/portals-bridges-jsf-1.0.jar
ln -s ~/.maven/repository/xerces/jars/xerces-2.4.0.jar
ln -s ~/.maven/repository/xml-apis/jars/xml-apis-2.0.2.jar
```

7. The WAR file

From the directory jsf-simplest combine the files above into a war file using the command,

```
jar cvf ../jsfsimplest.war .
```

8. Deploy the WAR file

Copy the war file to `$CATALINA_HOME/webapps/jetspeed/WEB-INF/deploy`. Jetspeed-2 will deploy the webapp.

9. The PSML

Create the PSML page using the Jetspeed portlet chooser. Login and click on the edit page icon. Click on the add portlet icon. Checkbox and add the JSFSimplestPortlet to your page. Your user must have the permission to edit pages. The user admin password admin has permission to edit all pages.