JBoss Portlet Tools User Guide

Version: 1.1.0.GA

1. Introduction 1
1.1. What is JBoss Portal and Portlet Tools? 1
1.2. Key Features of JBoss Portlet Tools 1
1.3. Requirements and Installation 1
2. JBoss Portlet Tools Tasks 3
2.1. Creating and Deploying a Java Portlet 3
2.1.1. Creating a Web Project with JBoss Portlet Capabilities 3
2.1.2. Adding a Java Portlet to a Web Project 11
2.1.3. Deploying a Portlet to JBoss Portal 14
2.2. Creating and Deploying a JSF Portlet 17
2.2.1. Creating a JSF Project with JBoss Portlet Capabilities 17
2.2.2. Adding a JSF Portlet to the Project and Deploying It to JBoss Portal 24
2.3. Creating and Deploying a Seam Portlet 26
2.3.1. Creating a Seam Project with JBoss Portlet Capabilities 26
2.3.2. Adding a Seam Portlet to the Project and Deploying It to JBoss Portal 31
3. Reference 35
3.1. JBoss Portlet Descriptors 35
3.2. Wizards 35
3.2.1. Java Portlet Wizard 35
3.2.2. JSF/Seam Portlet Wizard 41
3.3. JBoss Portlet Preferences 46
4. Summary 47
4.1. Other Relevant Resources on the Topic 47

Introduction

Starting from 3.0.0.Alpha1 version, JBoss set of plugins includes tools for supporting JBoss Portal and JSR-186/JSR-286 portlets. Thus, this guide is intended to explain you how to get started and manage JBoss Portlet Tools.

1.1. What is JBoss Portal and Portlet Tools?

JBoss Portal provides an open source platform for hosting and serving a portal's Web interface, publishing and managing its content, and customizing its experience. It supports a wide range of features, including standard portlets, single sign-on, clustering, and internationalization. More about JBoss Portal you can find visiting its home page at: *http://www.jboss.org/jbossportal/*.

With JBoss Portlet Tools you can easily create a Java, JSF and Seam portlet and deploy it to JBoss Portal.

1.2. Key Features of JBoss Portlet Tools

JBoss Portlet Tools supports the *JSR-168 Portlet Specification (Portlet 1.0)* [http://www.jcp.org/en/jsr/detail?id=168] and *JSR-286 Portlet Specification (Portlet 2.0)* [http://www.jcp.org/en/jsr/detail?id=286] and works with *JBoss Portlet Bridge* [http://www.jboss.org/portletbridge/] for supporting Portlets in JSF/Seam applications. To enable these features, you need to add the JBoss Portlet facet to a new or an existing web project (see *JBoss Portlet Tools Tasks*). The project could be a non-WTP project, but you should know that if it does not have the proper portlet API jar's the generated classes will have compile errors.

The next table lists key features supplied by JBoss Portlet Tools.

Table 1.1. Key Features of JBoss Portlet Tools

Feature	Benefit	Chapter
JBoss Portlet facets	JBoss Portlet Tools provides the Java, JSF and Seam Portlet facets that could be enabled for a Web project.	*JBoss Portlet Tools Tasks*
Java Portlet wizard	The wizard helps create a JSR-186/JSR-286 compliant portlets	*Java Portlet Wizard*
JSF/Seam Portlet wizard	This wizard helps create JSF/Seam portlets	*JSF/Seam Portlet Wizard*

1.3. Requirements and Installation

JBoss Portlet Tools needs the following minimal requirements:

- Eclipse 3.4.2/Ganymede
- JBoss Tools 3.0.0.Alpha1 (for higher versions of JBoss Tools make sure you use the *compatible version of Eclipse* [http://www.jboss.org/community/wiki/MatrixofsupportedplatformsruntimesandtechnologiesinJBossToolsJBDS])
- JDK 5 or higher
- JBoss Portal + JBoss AS bundle from: *http://www.jboss.org/jbossportal/download/index.html*

Note:

JBoss Portlet Tools has been tested using 2.6.5 and 2.7.0.ALPHA versions of JBoss Portal. Remember, if you want to use Portlet version 2.0 (JSR-268), you have to choose JBoss Portal 2.7.

- JBoss Portlet Bridge (optionally) from: *http://www.jboss.org/portletbridge/downloads.html*

JBoss Portlet Tools is included as part of JBoss Tools project. Refer to *http://www.jboss.org/tools/download/installation.html* for installation methods of JBoss Tools.

JBoss Portlet Tools Tasks

2.1. Creating and Deploying a Java Portlet

This chapter shows how to create a Dynamic Web Project, add a Java Portlet to it and deploy it to the JBoss Portal.

2.1.1. Creating a Web Project with JBoss Portlet Capabilities

Fallow the next procedure to create a Web project with JBoss Portlet capabilities pointed to the JBoss Portal runtime.

1. Go to *File > New > Dynamic Web Project* if you are in the Web perspective or *File > New > Other > Web > Dynamic Web Project* in any other perspective. The New Dynamic Web Project wizard appears.

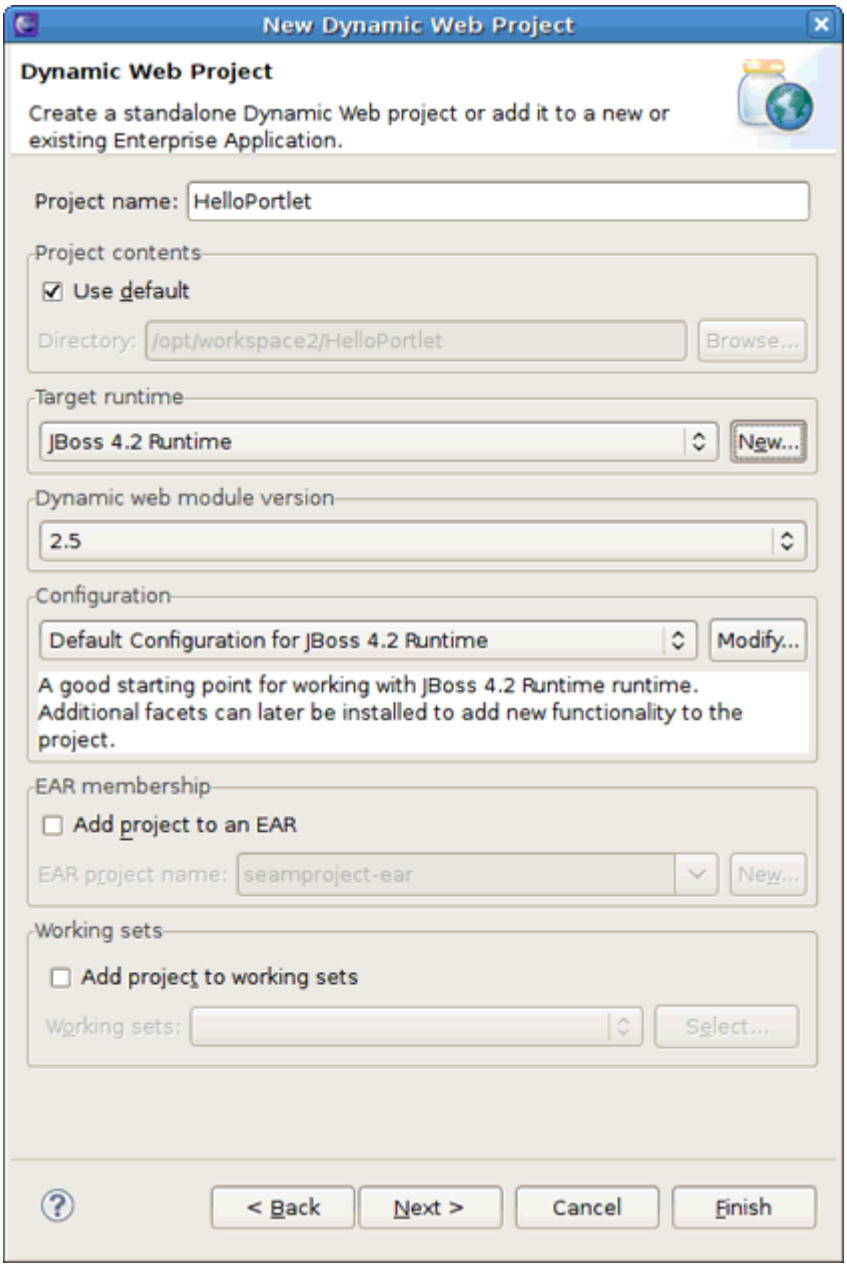

Figure 2.1. New Dynamic Web Project

2. Specify the name of the project.

3. Click *New* in the *Target Runtime* area to create a JBoss Portal runtime. Choose *JBoss Community > JBoss 4.2 Runtime* and select the *Create a new local server* check box below. Click *Next*.

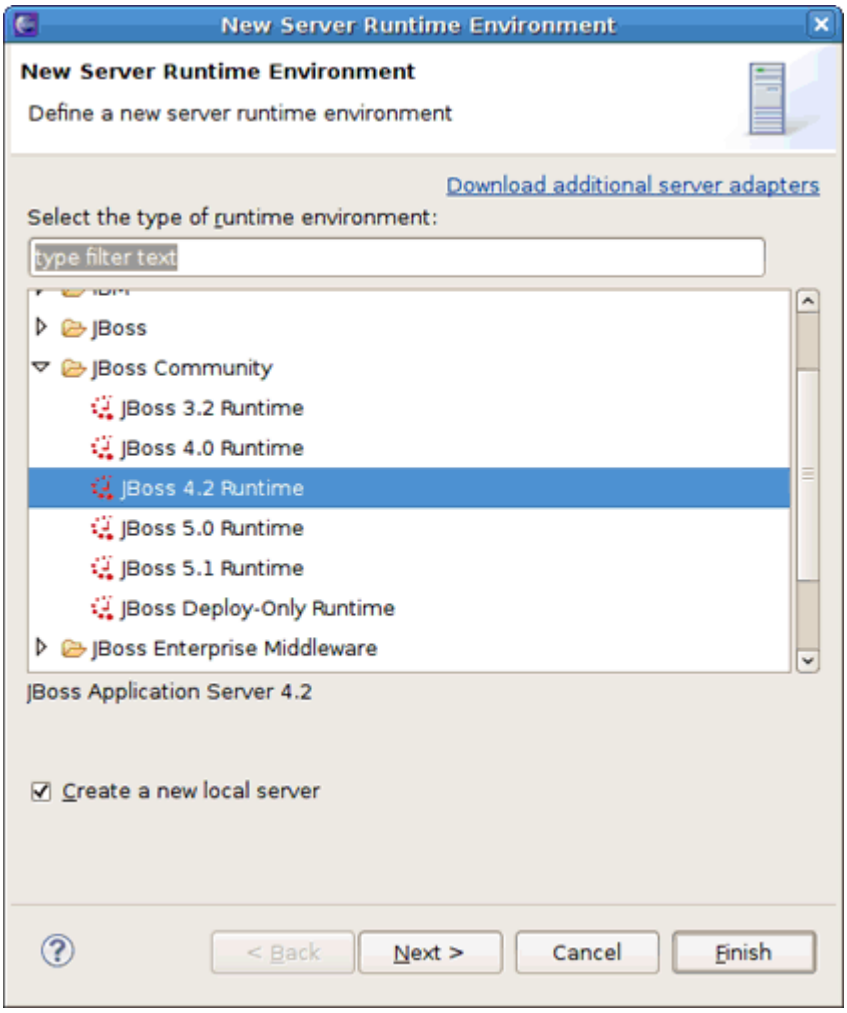

Figure 2.2. New Server Runtime

4. The New Server Runtime Environment wizard appears. In the *Name* field, type *JBoss Portal 2.7 Runtime*, and then use the *Browse* button to point to the location of JBoss Portal + JBoss AS extracted. Click *Next* to proceed.

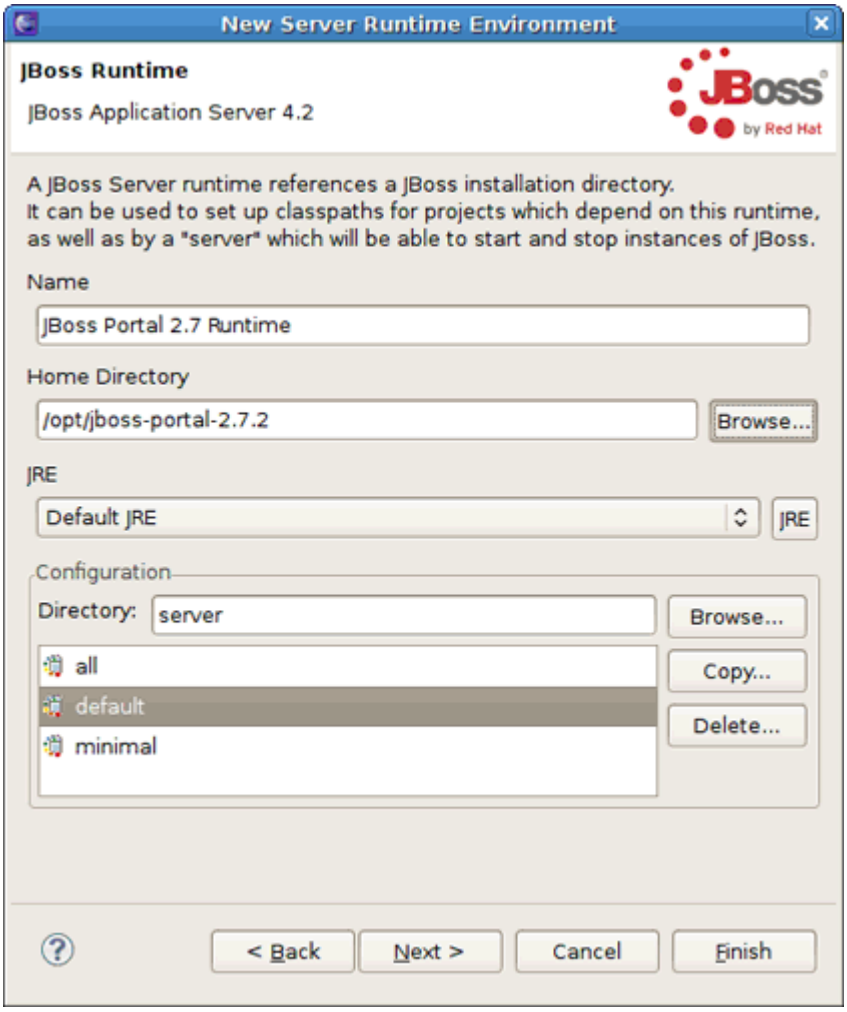

Figure 2.3. New Server Runtime Environment

5. At this point a new JBoss Server instance will be created. On the next page you can verify the runtime information defined. If something is incorrect, press *Back* to return to the previous wizard page. Click *Finish*.

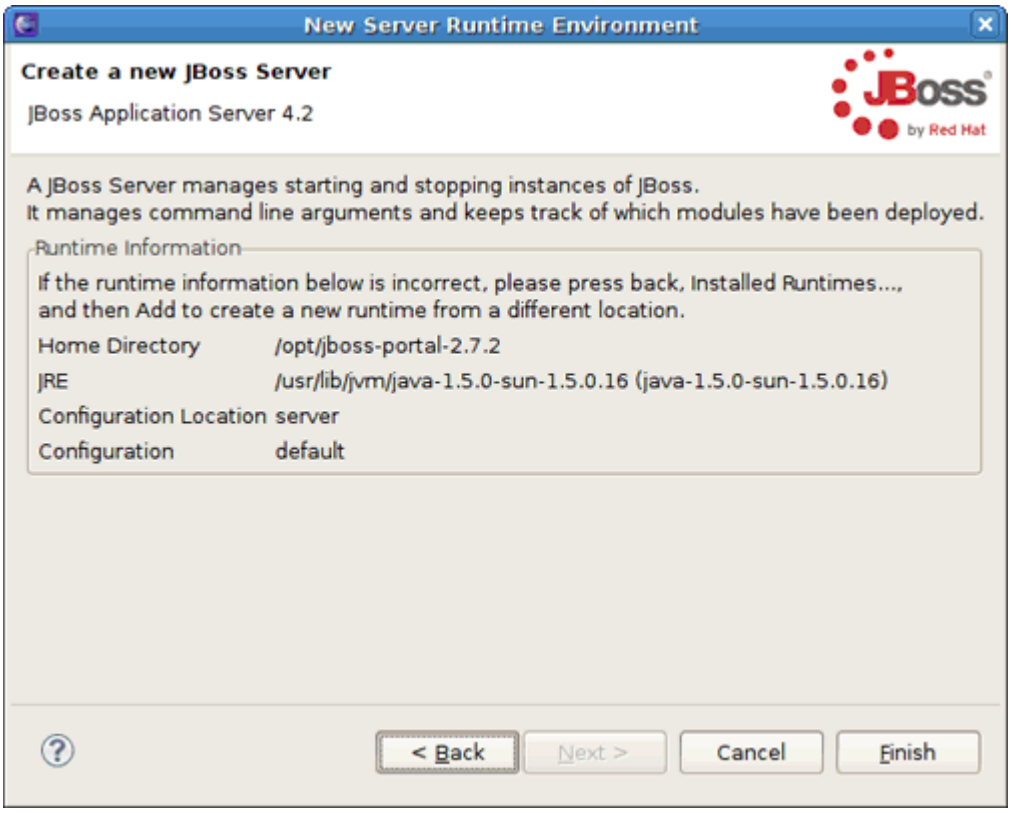

Figure 2.4. Target Server

6. Click *Modify* in the *Configuration* area to enable a portlet facet for the project.

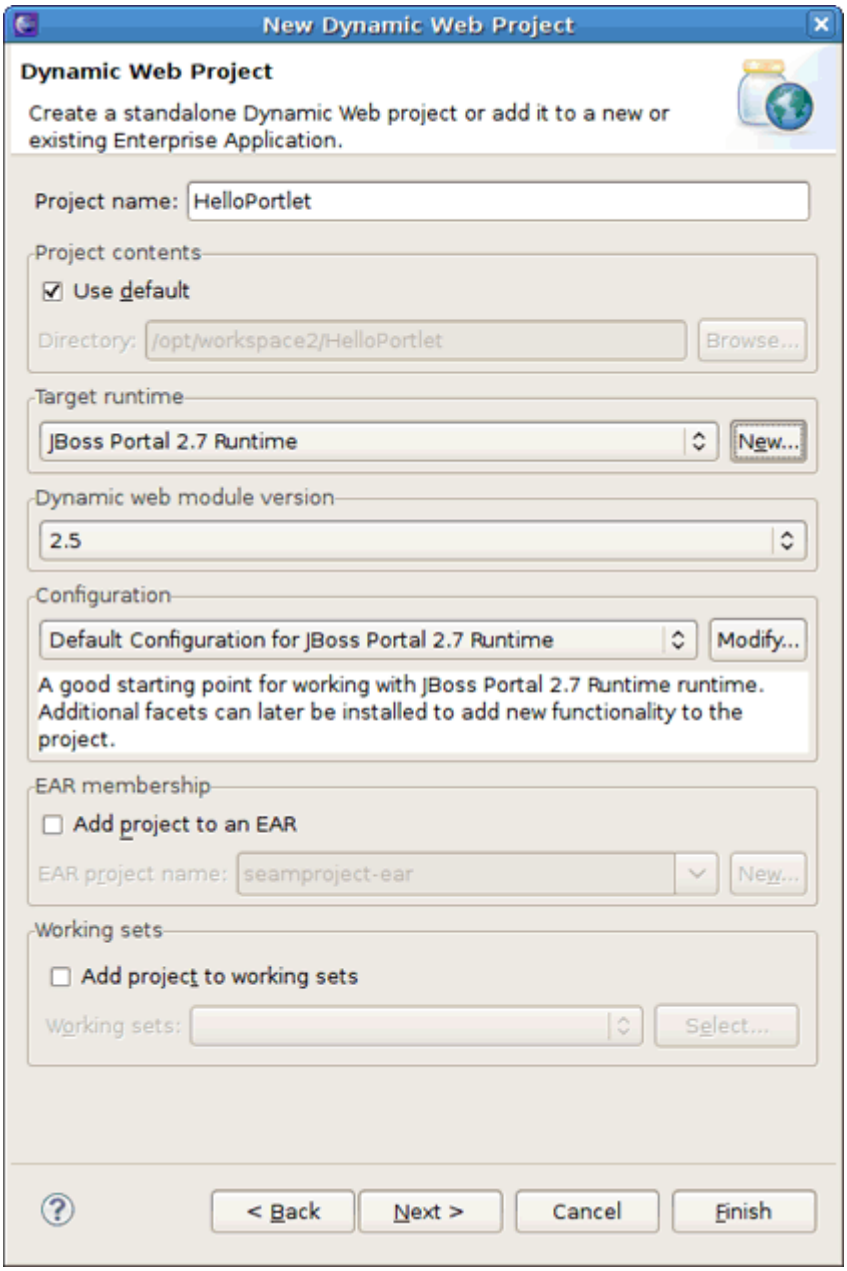

Figure 2.5. Setting the Project Configuration

7. In the Project Facets dialog, check *JBoss Core Portlet* and press *OK* .

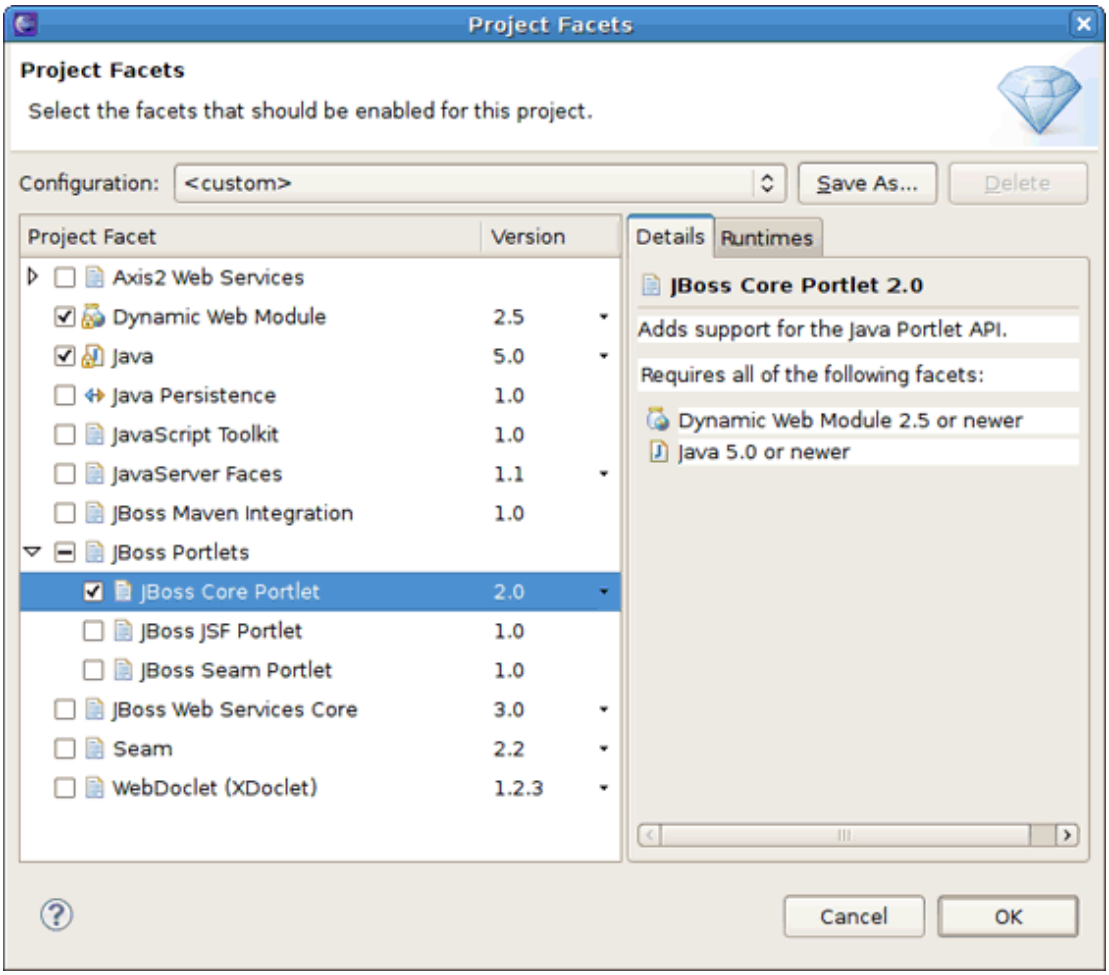

Figure 2.6. Enabling the JBoss Core Portlet Facet

Note:

If the portlet libraries aren't available in the runtime you targeted, *JBoss Portlets* facets will be hidden on this page.

To make them always visible no matter what the runtime is set, you should enable the appropriate option in *JBoss Portlet Preferences*.

8. The Java and Web Module pages are for configuring Java and Web modules in the project. Here all values are set, so leave everything as it is.

9. The last wizard page will ask you to add JBoss Portlet capabilities to the project. Select *Portlet Target Runtime Provider* and click *Finish* to complete the project creation.

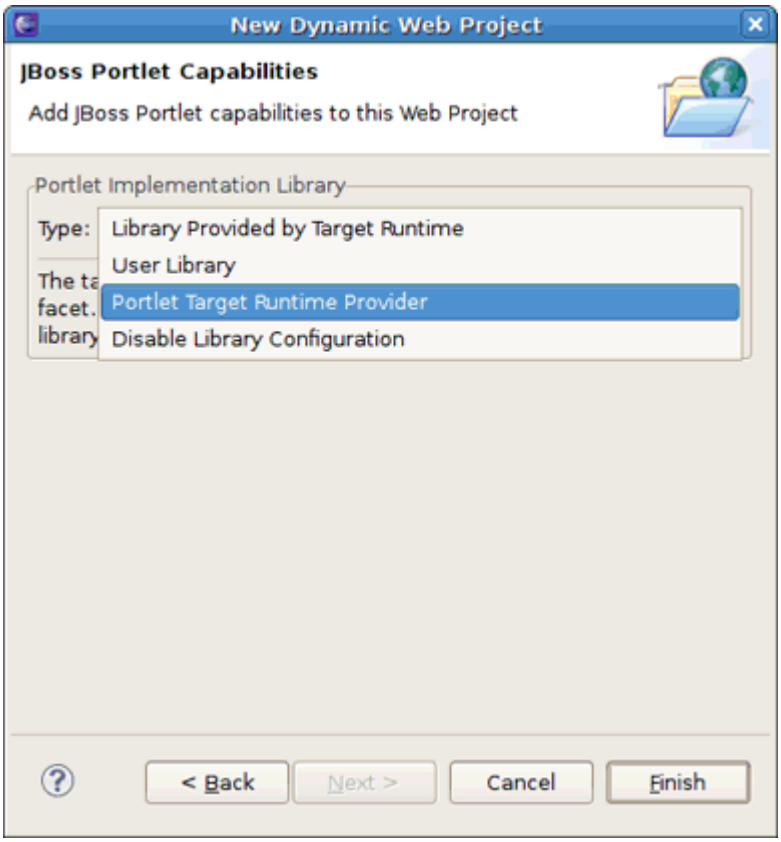

Figure 2.7. Including the Portlet Libraries Provided by Target Runtime

Note:

All types available under *Portlet Implementation Library* are fully described in the Wiki article at: http://www.jboss.org/community/wiki/PortletFacetLibraries.

As the result, JBoss Portlet Tools adds *JBoss Core Portlet* facet to the project, creates an empty `portlet.xml` file and adds the JBoss Portlet library to the project classpath.

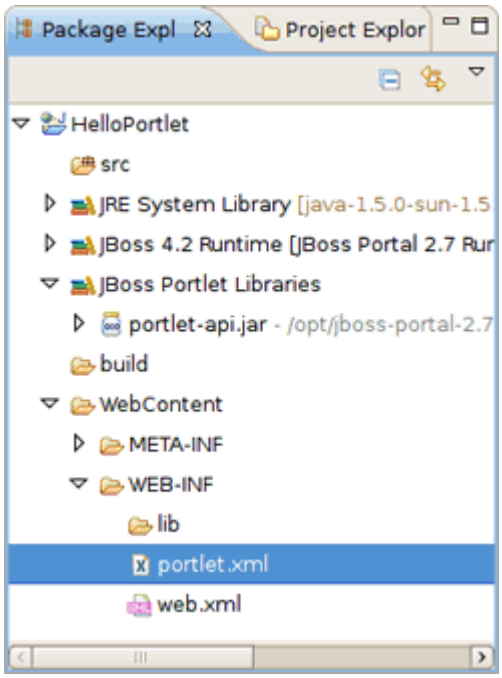

Figure 2.8. Portlet Project Structure Overview

2.1.2. Adding a Java Portlet to a Web Project

At the previous section you have created a web project with JBoss Portlet capabilities. The next steps will show you how to add a new Java portlet to the project.

1. Right-click the project and follow to *New > Other > JBoss Tools Web > Portlet > Java Portlet*.

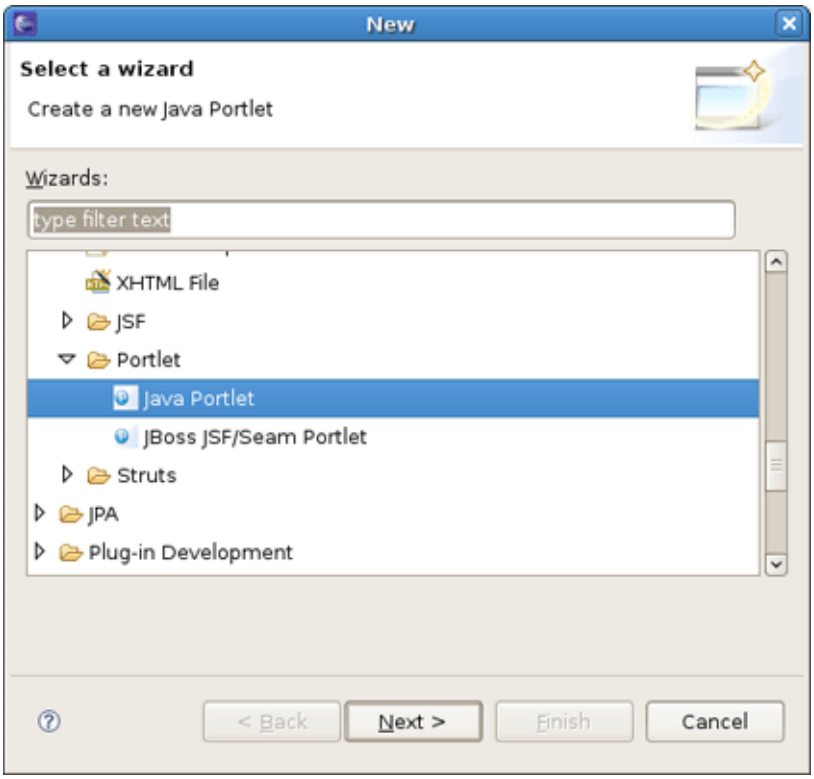

Figure 2.9. New Java Portlet

2. The Create Portlet wizard starts (for information about the wizard options, see *"Java Portlet Wizard"* in the guide reference). The wizard fills in the *Project* and *Source Folder* fields for you. You should specify a Java package and a class name (for instance, *org.example* and *TestPortlet*). Then click *Next* .

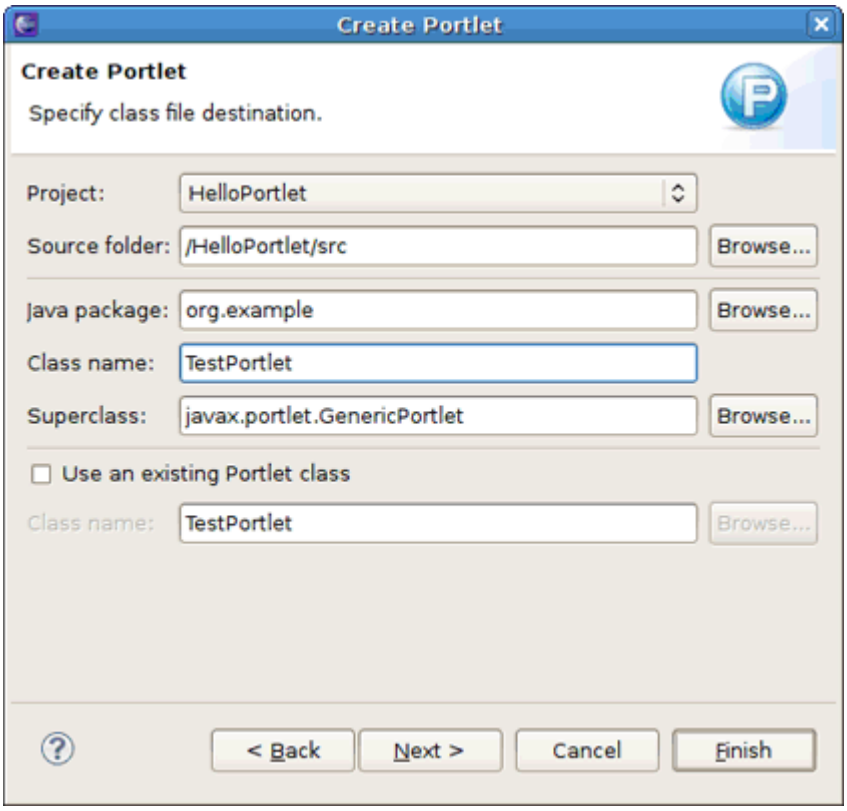

Figure 2.10. Specifying Class File Destination

3. You may leave the next three pages with default values, on the last one hit *Finish*.

Once a Java portlet is created, new resources are added to the project structure: a Java portlet class (`TestPortlet.java`), `default-object.xml` and `portlet-instances.xml` files and the `portlet.xml` descriptor is updated as well.

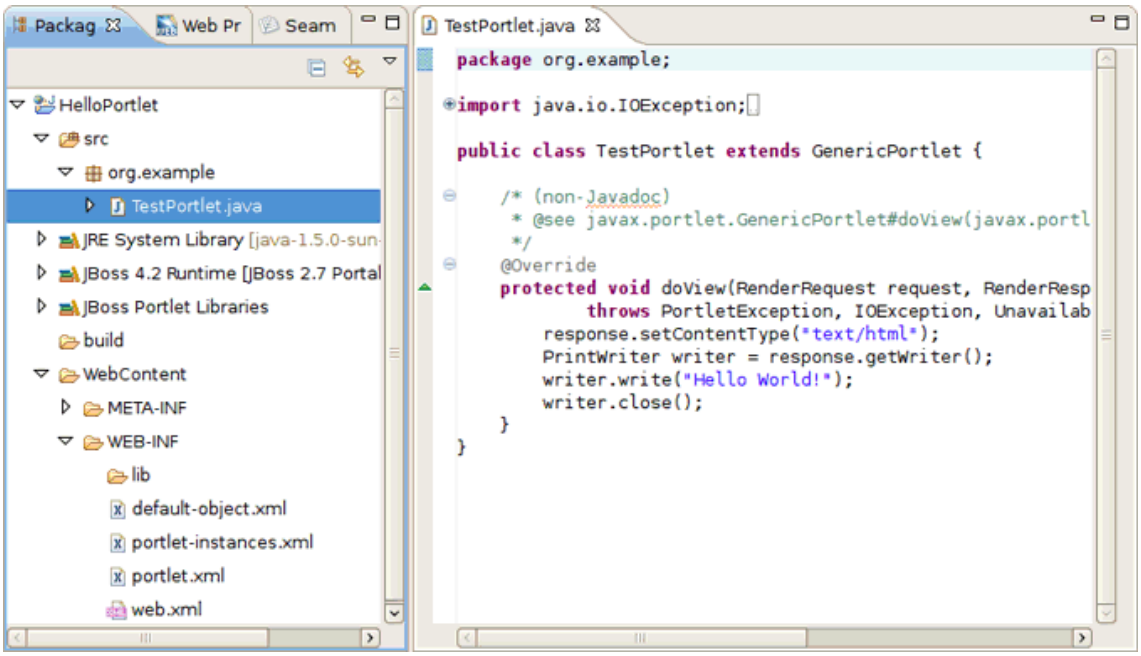

Figure 2.11. Structure of the Project with a Java Portlet

Thus, now the project is ready to be built and deployed.

2.1.3. Deploying a Portlet to JBoss Portal

You can deploy a portlet project in the way you deploy any other web application.

1. Right-click the project and do *Run As > Run On Server.* The Run On Server wizard starts.

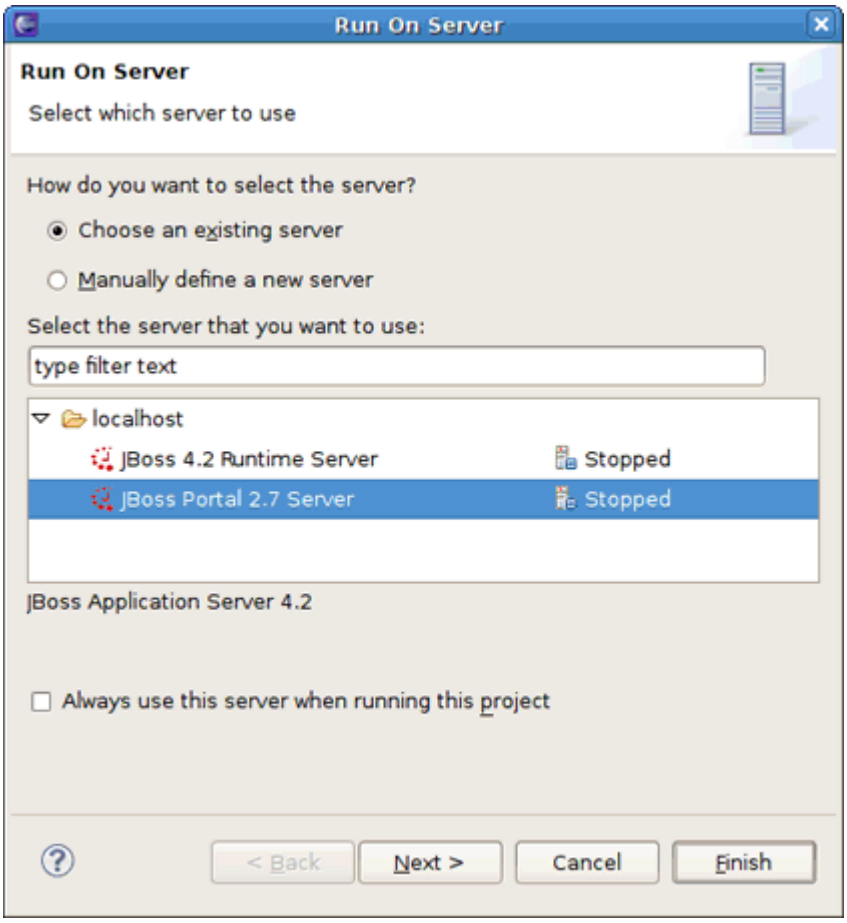

Figure 2.12. Running the Java Portlet on Server

2. Select *JBoss Portal 2.7 Server* created before and click *Next*.

3. On the *Add and Remove* page move the created project to the right and click *Finish*.

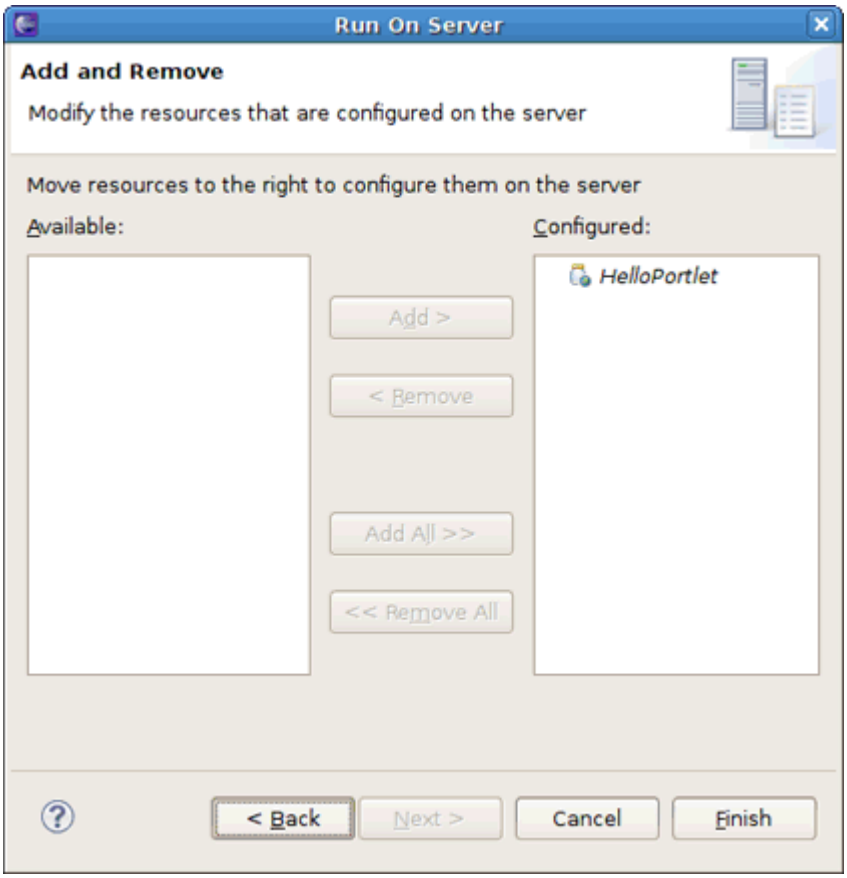

Figure 2.13. Deploying the Portlet Application

It will deploy the portlet application and start JBoss AS.

4. Use the *http://localhost:8080/portal/portal/default/default* URL to see your portlet in the browser.

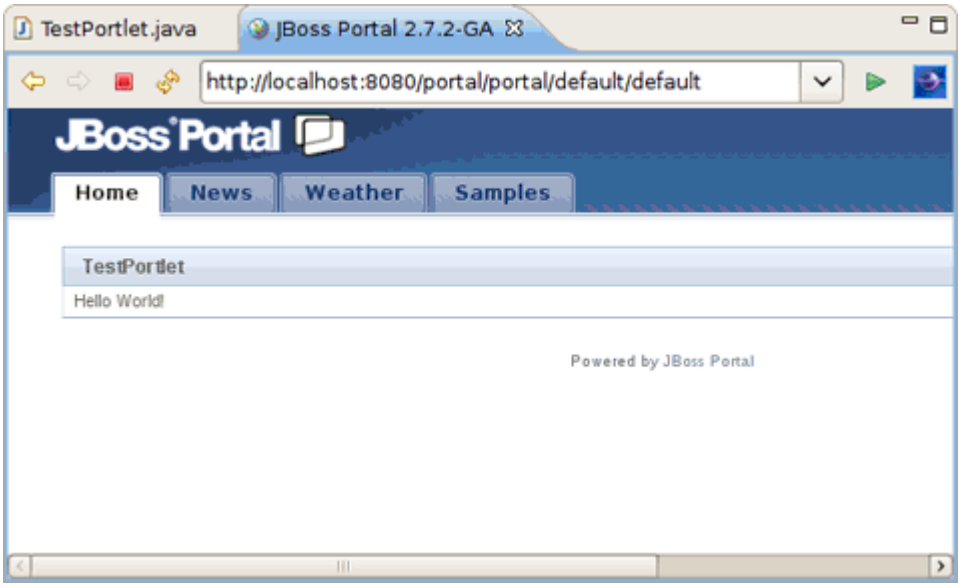

Figure 2.14. Java Portlet in the Browser Window

2.2. Creating and Deploying a JSF Portlet

This chapter will explain how you can configure a JSF portlet within a JSF project with JBoss Portlet capabilities and then deploy it to JBoss Portal.

2.2.1. Creating a JSF Project with JBoss Portlet Capabilities

You can create a JSF project with JBoss Portlet capabilities in two ways:

1. Create a dynamic Web project with the *JavaServer Faces* and *JBoss JSF Portlet* facets enabled (*the steps to proceed*)
2. Or create a JSF project using the wizard JBoss JSF Tools provides, then enable JSF and JBoss Portlet facets and add JBoss Portlet capabilities (*the steps to proceed*)

Refer to the further sections for the procedures on how to do this.

2.2.1.1. Creating a Dynamic Web Project with the JBoss Portlet Capabilities

The basic steps to create a dynamic Web project with the JBoss Portlet capabilities are as follows:

1. Start the *Dynamic Web Project* wizard navigating to *File > New > Other > Web > Dynamic Web Project*.

Tip:

You can also select the Java EE perspective and then go to *File > New > Dynamic Web Project*.

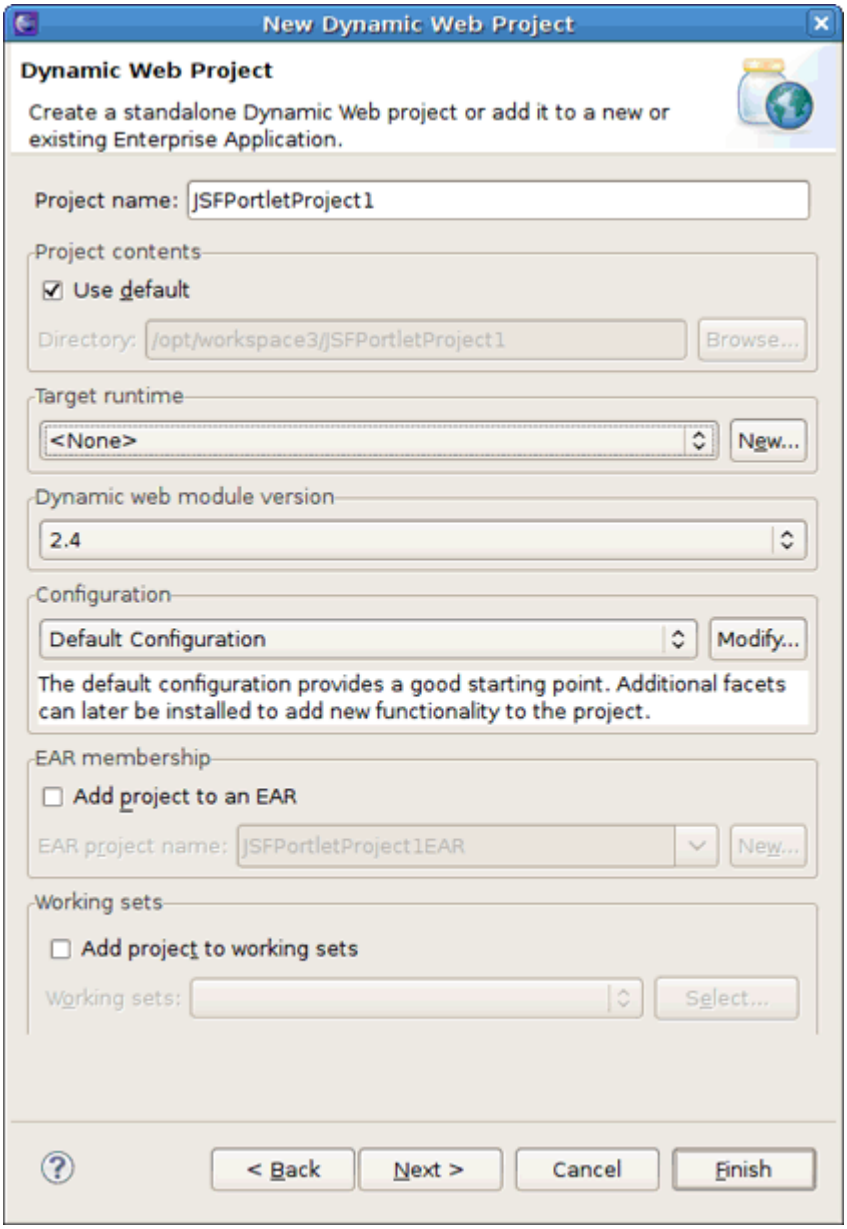

Figure 2.15. Setting the JSF Configuration

2. Specify the project name and set the target runtime to JBoss Portal by following the points 3, 4 and 5 in the *"Creating a Web Project with JBoss Portlet Capabilities"* procedure.

3. In the *Configuration* area, click *Modify* and enable *JavaServer Faces*, *JBoss Core Portlet* and *JBoss JSF Portlet* facets. Hit *OK*.

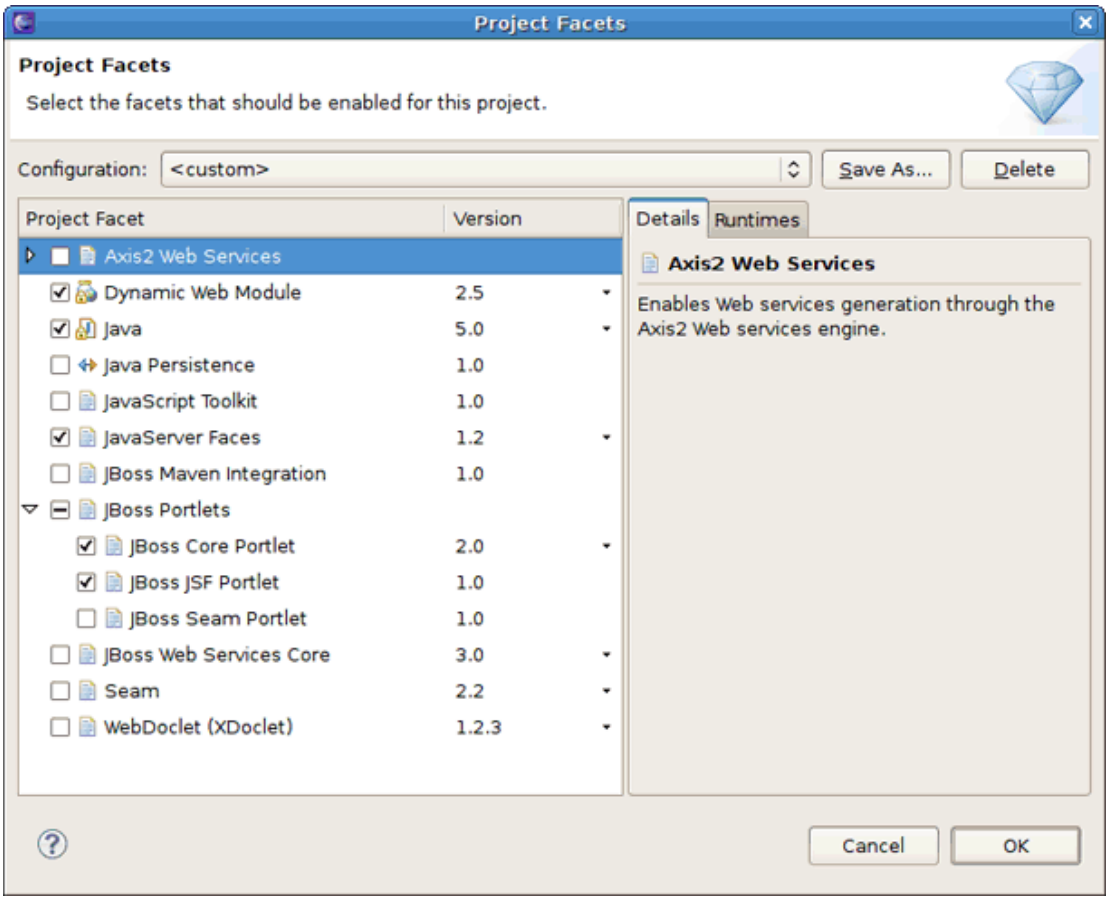

Figure 2.16. Enabling the Portlet Facets

4. You may leave the next two wizard pages with defaults, just press *Next* to proceed.

5. On the JBoss Portlet Capabilities page, select *Portlet Target Runtime Provider* and click *Next*.

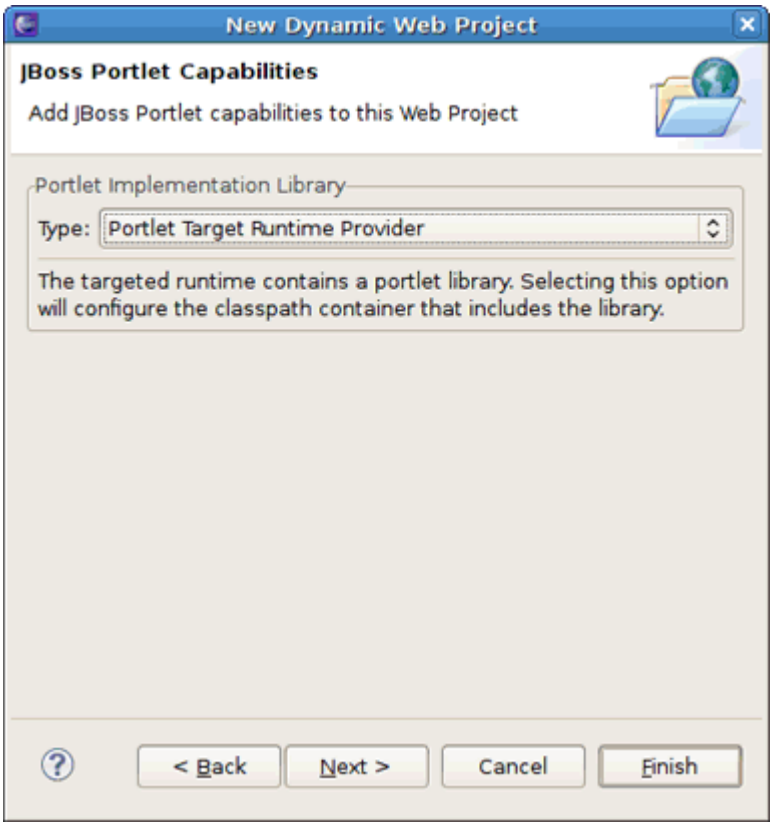

Figure 2.17. JBoss Portlet Capabilities

Tip:

All types of the portlet implementation library are described more closely in the wiki article at: *http://www.jboss.org/community/wiki/PortletFacetLibraries*.

6. Next wizard page is for configuring JSF capabilities. You can leave everything as it is here.

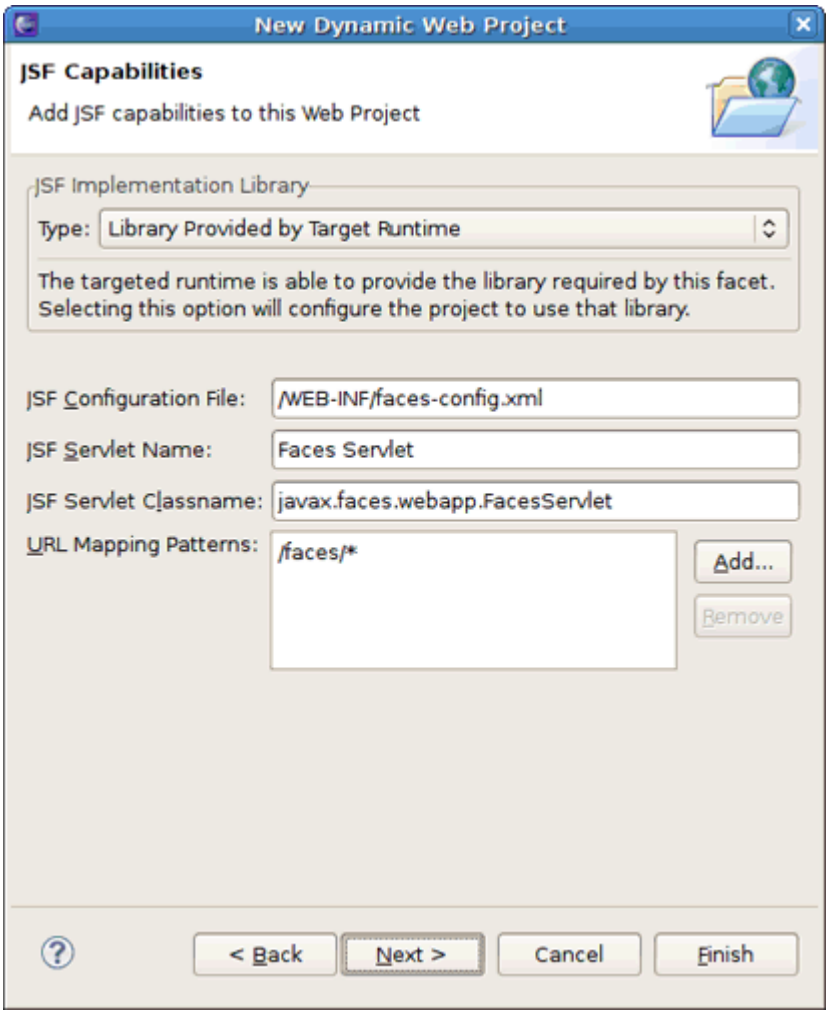

Figure 2.18. JSF Capabilities

7. On the JBoss JSF Portlet Capabilities page, select *JSF Portlet Target Runtime Provider* as the JSF portlet implementation library. It will copy Portlet Bridge libraries from the server runtime to the project classpath.

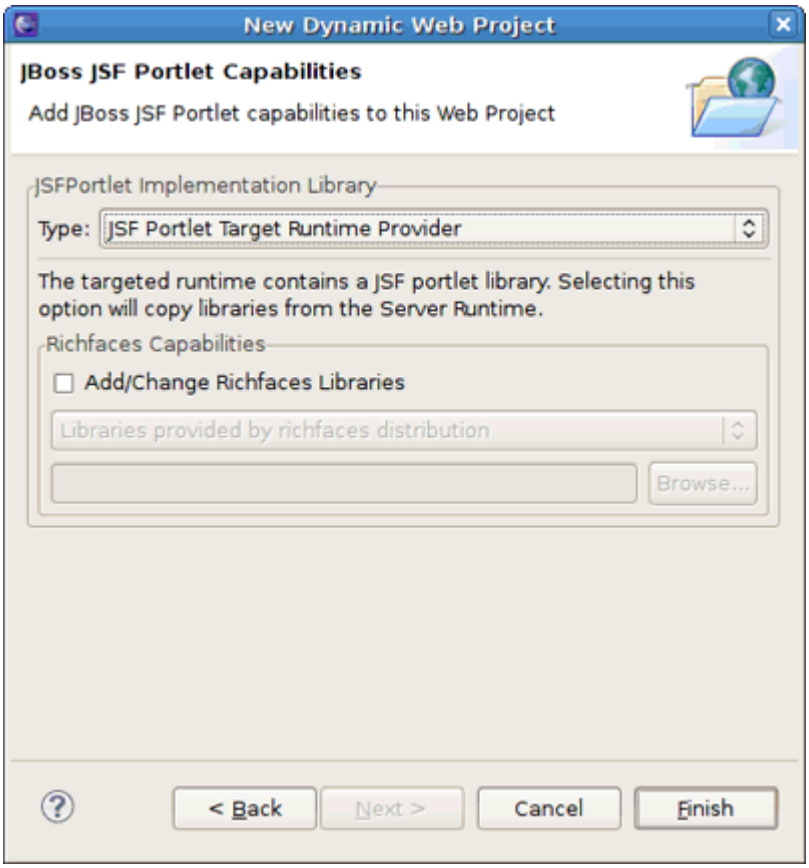

Figure 2.19. JBoss JSF PortletCapabilities

Tip:

On this page, it is also possible to add the RichFaces libraries from the RichFaces distribution by checking *Add/Change Richfaces Libraries*.

Note:

You can select the *JSF Portletbridge Runtime Provider* type. Then it is necessary to set the home of the Portlet Bridge distribution.

For information about all the JSF Portlet facet library providers, refer to the wiki article at: http://community.jboss.org/wiki/PortletFacetLibraries.

8. Click *Finish*. The project will be created in the workbench.

2.2.1.2. Creating a JSF Project and adding the JBoss Portlet Capabilities

How to organize a JSF project you can find out in the *JSF Tools User Guide* [../../jsf_tools_ref_guide/html_single/index.html#new_jsf_project]. Just remember to point the target runtime to JBoss Portal directory location (see how it is done for a dynamic Web project: steps 3, 4 and 5 in the *"Creating a Web Project with JBoss Portlet Capabilities"* procedure).

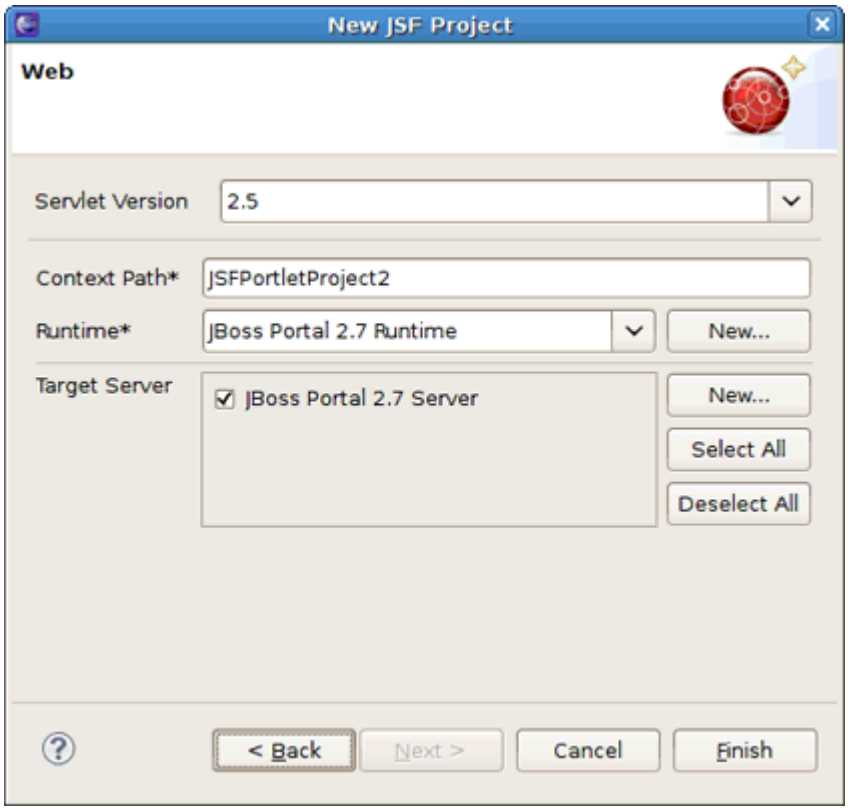

Figure 2.20. Creating New JSF Project

To add the JBoss Portlet capabilities to the created JSF project you should complete the next steps:

1. Right-click the project and click *Properties* to open the project Properties sheet. Select Project Facets on the left and enable the *JavaServer Faces*, *JBoss Core Portlet* and *JBoss JSF Portlet* facets.

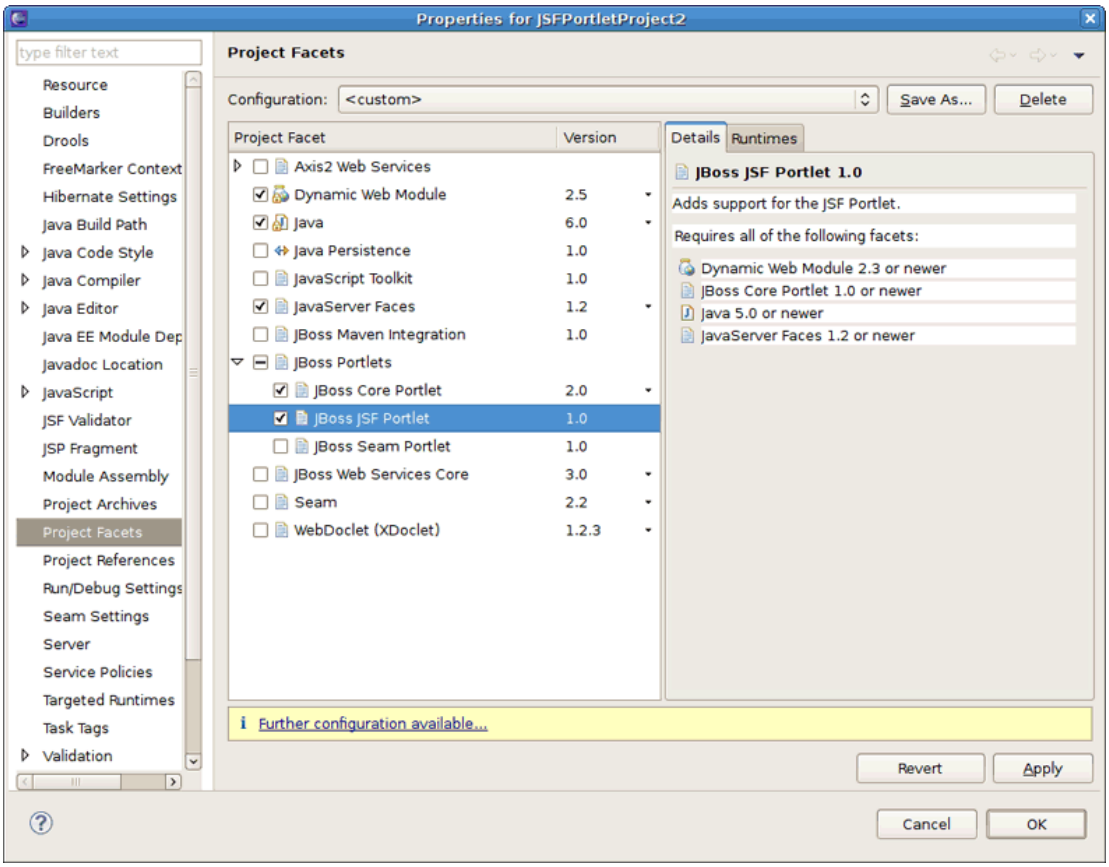

Figure 2.21. Project Facets

2. Notice, *"Further configuration available..."* warning appears at the bottom of the screen. It means that next you should configure the JSF Portlet capabilities in the project. Click the link. The Modify Faceted Project wizard appears.

3. Complete all wizard pages the same way as described in the steps 5, 6, 7 of the *"Creating a Dynamic Web Project with the JBoss Portlet Capabilities"* procedure. Hit *OK*.

4. To apply the changes click *Apply* and then *OK* to close the project Properties sheet.

2.2.2. Adding a JSF Portlet to the Project and Deploying It to JBoss Portal

The previous section has shown how to create a JSF project with JBoss Portlet and JSF Portlet capabilities enabled. Use the following procedure to add a JSF portlet to the created project and deploy it to JBoss Portal.

1. Call the Create Portlet wizard by navigating to *New > Other > JBoss Tools Web > Portlet > JSF/Seam Portlet* on the context menu of the project (for information about the wizard options, see *"JSF/Seam Portlet Wizard"* in the guide reference).

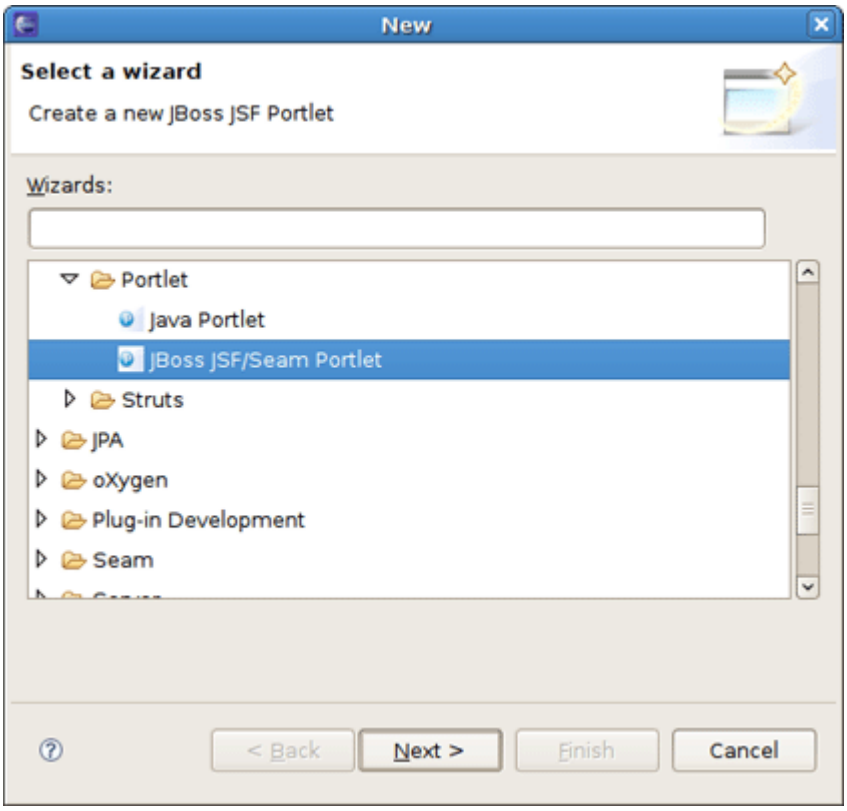

Figure 2.22. Calling New Portlet Wizard

Tip:

As you see, the JSF and Seam Portlet wizards are merged into one. We have set the JSF configuration for the project, so the default values in this wizard will be set as for a JSF portlet.

2. Leave all the wizard pages with default values by clicking the *Next* button, on the last one click *Finish* to complete the JSF portlet creation.

3. Complete the steps described in the *"Deploying a Portlet to JBoss Portal"* procedure to deploy a JSF portlet to JBoss Portal. Just use the other URL to see it in the browser: *http://localhost:8080/portal/portal/default/JSFPortlet*.

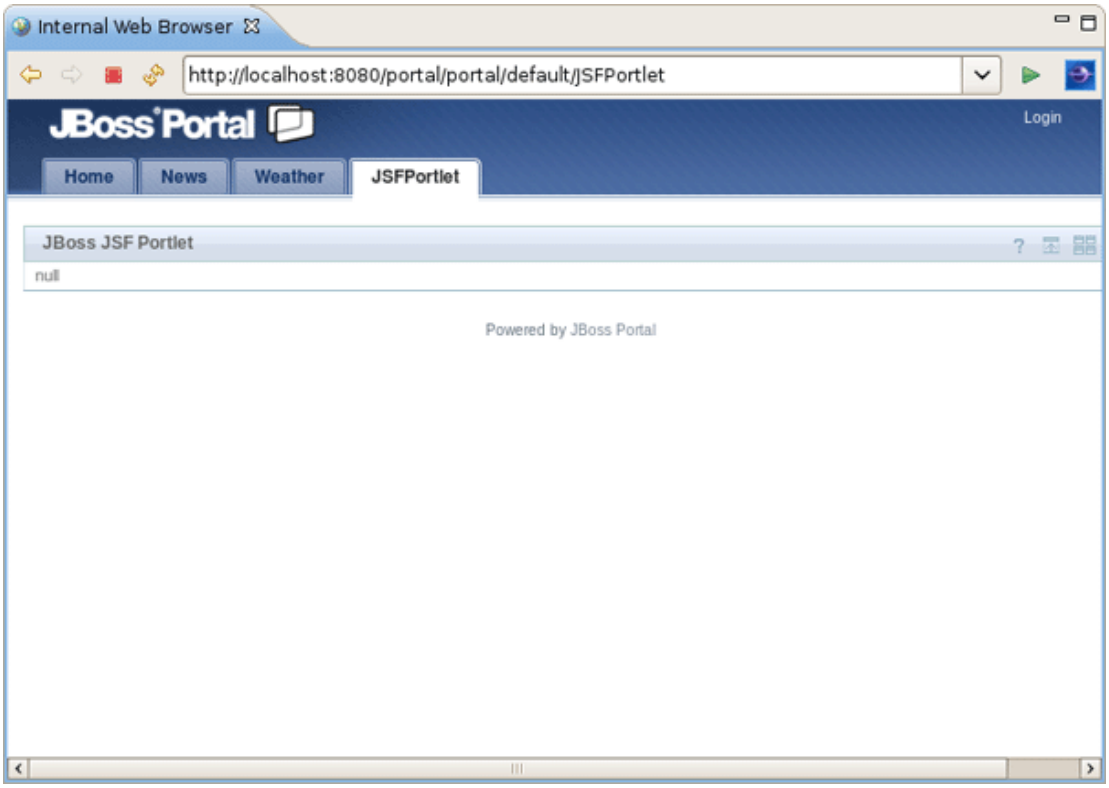

Figure 2.23. JSF Portlet in the Browser

2.3. Creating and Deploying a Seam Portlet

This chapter covers the steps on how to configure a Seam portlet within a Seam project with the help of the JBoss Portlet Tools features.

2.3.1. Creating a Seam Project with JBoss Portlet Capabilities

Follow one of two ways of creating a Seam project with JBoss Portlet capabilities enabled:

1. Create a dynamic Web project with the *Seam* and *JBoss Portlets* facets enabled (*the steps to proceed*)
2. Or create a Seam project with the JBoss Seam portlet configuration using the wizard JBoss Seam Tools provides and follow all the wizard steps to enable JBoss Portlet capabilities (*the steps to proceed*)

2.3.1.1. Creating a Dynamic Web Project with Seam and JBoss Portlet Capabilities

To create a dynamic Web project with Seam and JBoss Portlet capabilities you should take the following steps:

1. In the *File* menu, go to *New > Other > Web > Dynamic Web Project*. The New Dynamic Web Project wizard starts.

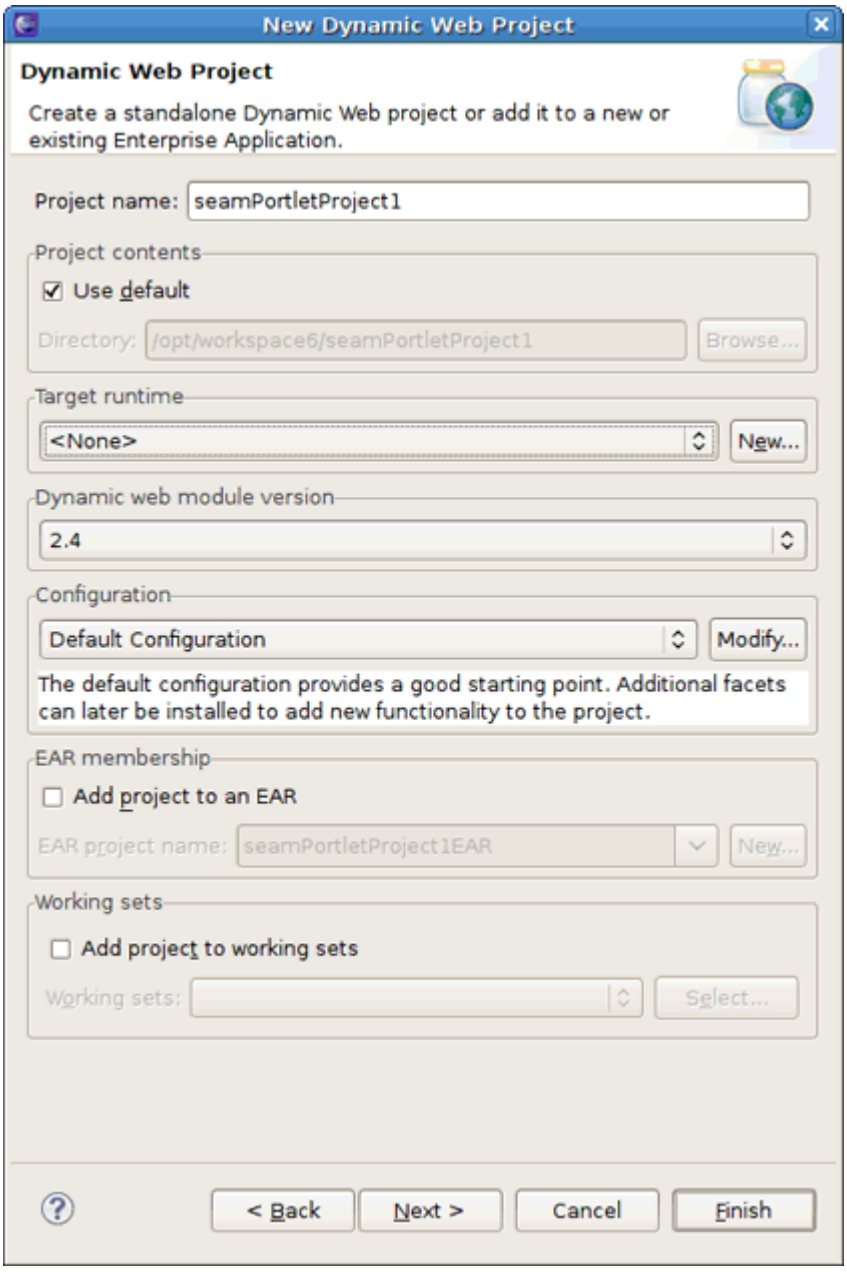

Figure 2.24. Creating a New Dynamic Web Project

2. Give the project a name and follow the steps 3, 4, 5 of the *"Creating a Web Project with JBoss Portlet Capabilities"* procedure to set the target runtime.

3. In the *Configuration* area of the first wizard page, select *JBoss Seam Portlet Project v2.0*. It will add needed facets to the project.

Tip:

If you now click the *Modify* button, you should see the *JavaServer Faces*, *Jboss Portlets* and *Seam* facets enabled.

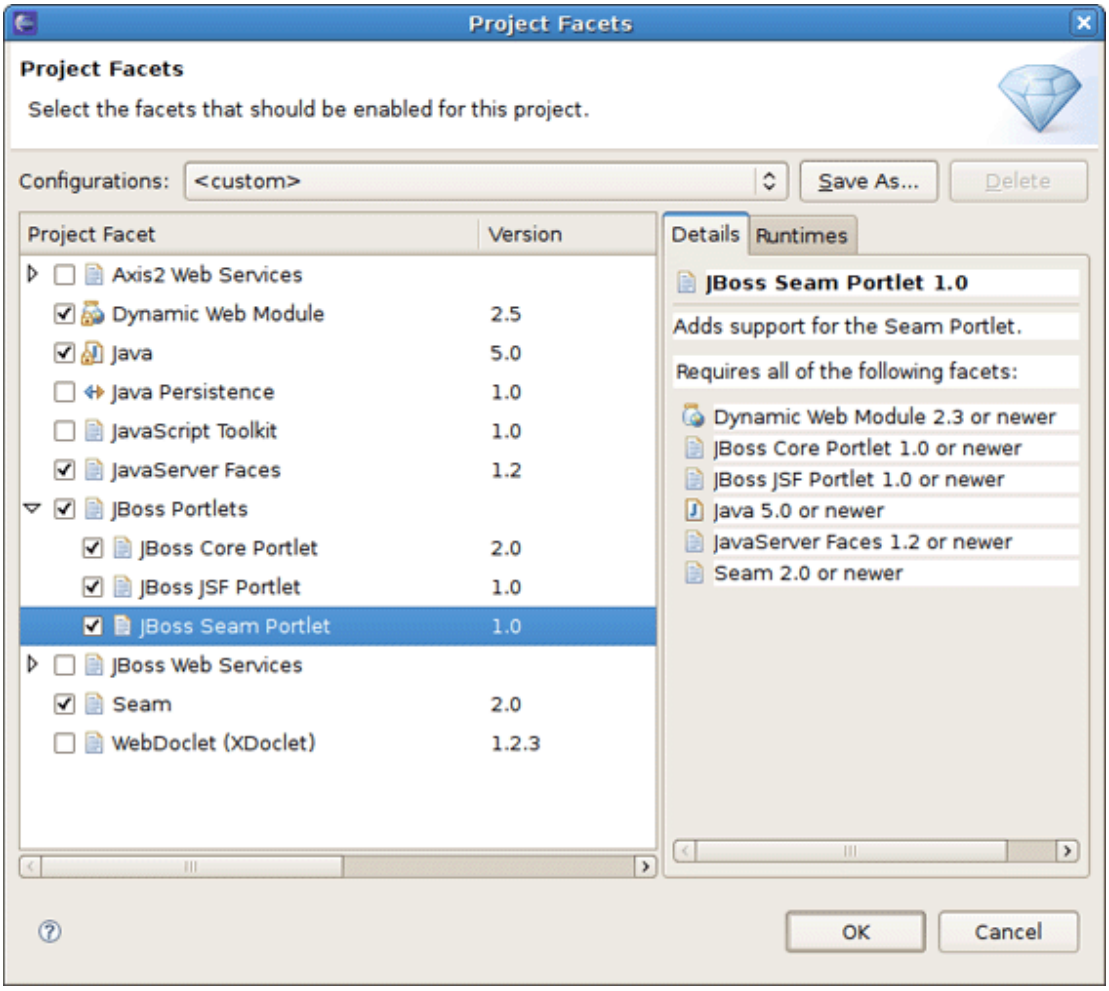

Figure 2.25. Project Facets

4. The next two pages are for adjusting the project Java and Web modules. They include the default values, so you can skip them by pressing *Next*.

5. On the Jboss Portlet Capabilities page, select *Portlet Target Runtime Provider* as the portlet implementation library.

Tip:

All types of the portlet implementation library are described more closely in the wiki article at: *http://www.jboss.org/community/wiki/PortletFacetLibraries*.

6. The next page is for configuring JSF capabilities. As it contains the default values, just click *Next* to proceed.

7. On the Seam Facet page, set a Seam runtime and a connection profile.

Tip:

For details about how to set a Seam runtime and a connection profile, see *"Configure Seam Facet Settings"* [../../seam/html_single/index.html#seamFacet] in the Chapter 2, *"Seam Dev Tools Reference Guide"* [../../seam/html_single/index.html].

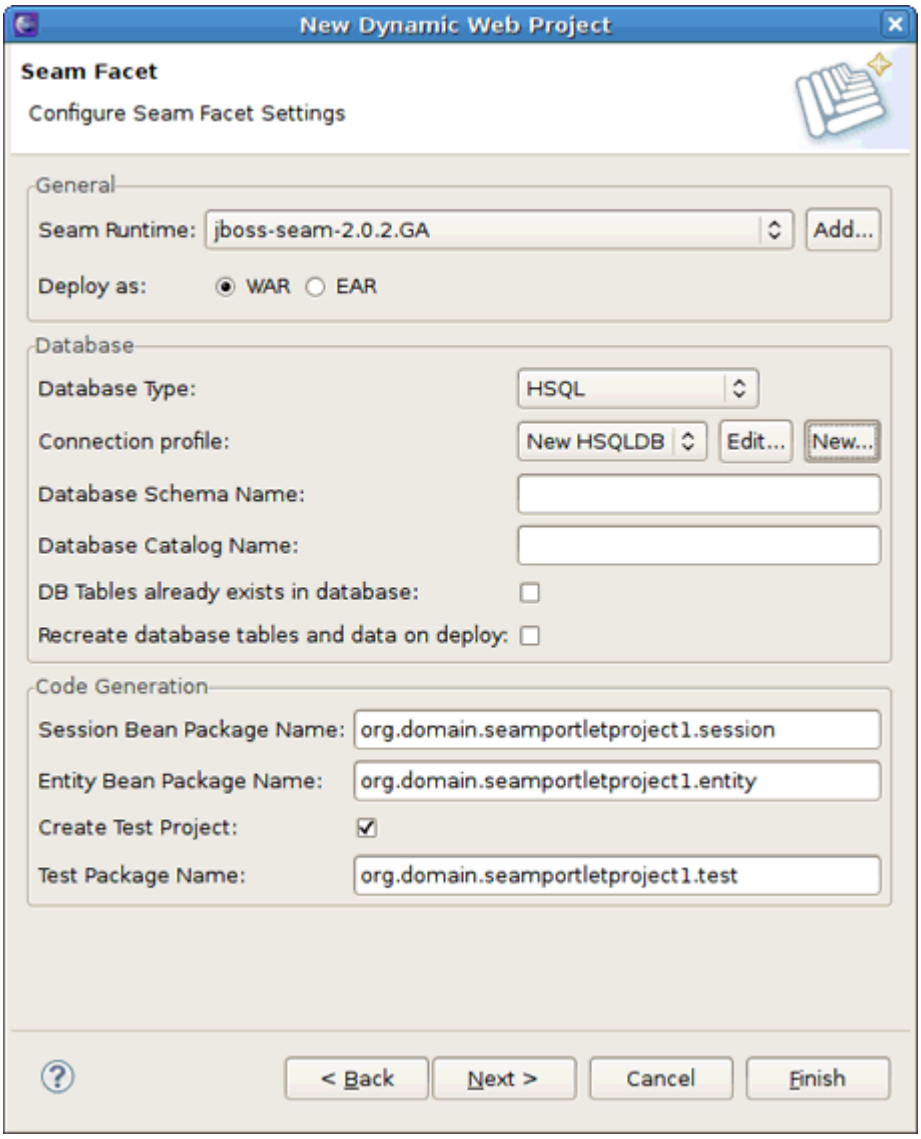

Figure 2.26. Seam Facet Settings

8. On the JBoss JSF Portlet Capabilities page, select the *JSF Portlet Target Runtime Provider* type of the implementation library.

Tip:

If you have JBoss Portlet Bridge downloaded, alternatively you can select *JSF Portletbridge Runtime Provider* and then provide a path to the JBoss Portlet Bridge distribution.

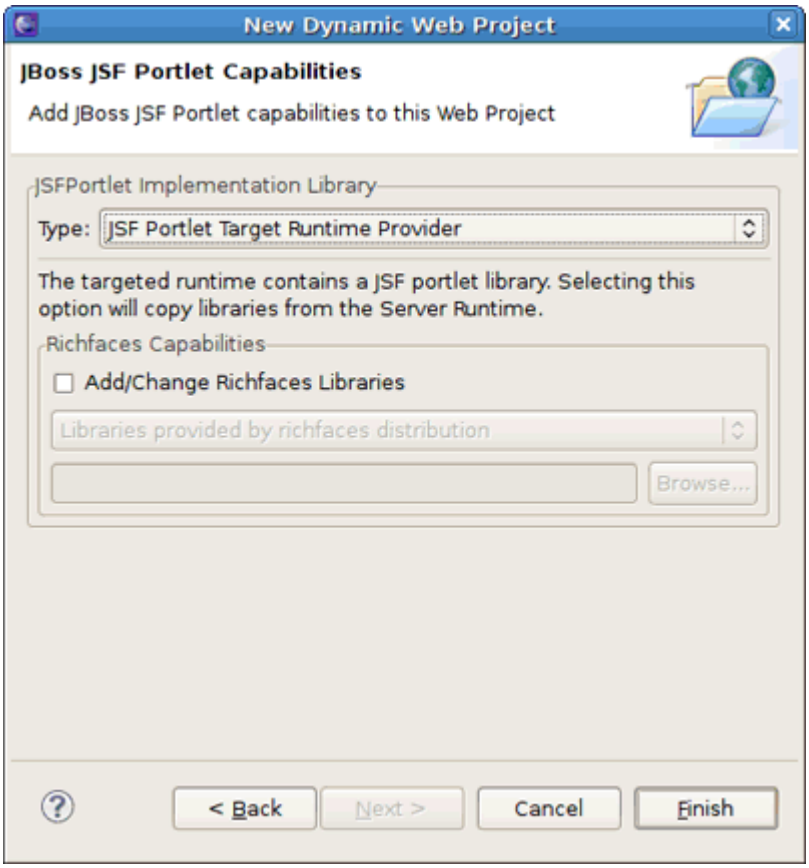

Figure 2.27. Setting Portlet Bridge Provider

It will add JBoss Portlet Bridge libraries to the project classpath.

9. Hit *Finish* to complete the project creation.

2.3.1.2. Creating a Seam Project with JBoss Portlet Capabilities

The steps to create a Seam project with JBoss Portlet capabilities are as follows:

1. In the *File* menu, go to *New > Other > Seam > Seam Web Project*. The New Seam Project wizard appears.
2. Next steps are the same as in the *"Creating a Dynamic Web Project with Seam and JBoss Portlet Capabilities"* procedure starting from the step 2.

2.3.2. Adding a Seam Portlet to the Project and Deploying It to JBoss Portal

The previous section has shown you how to create a Web project with Seam and JBoss Portlet capabilities. Now you can create a Seam portlet and deploy it to JBoss Portal by following the next procedure:

1. In the *File* or context menu of the project go to *New > Other > JBoss Tools Web > Portlet > JSF/Seam Portlet*.

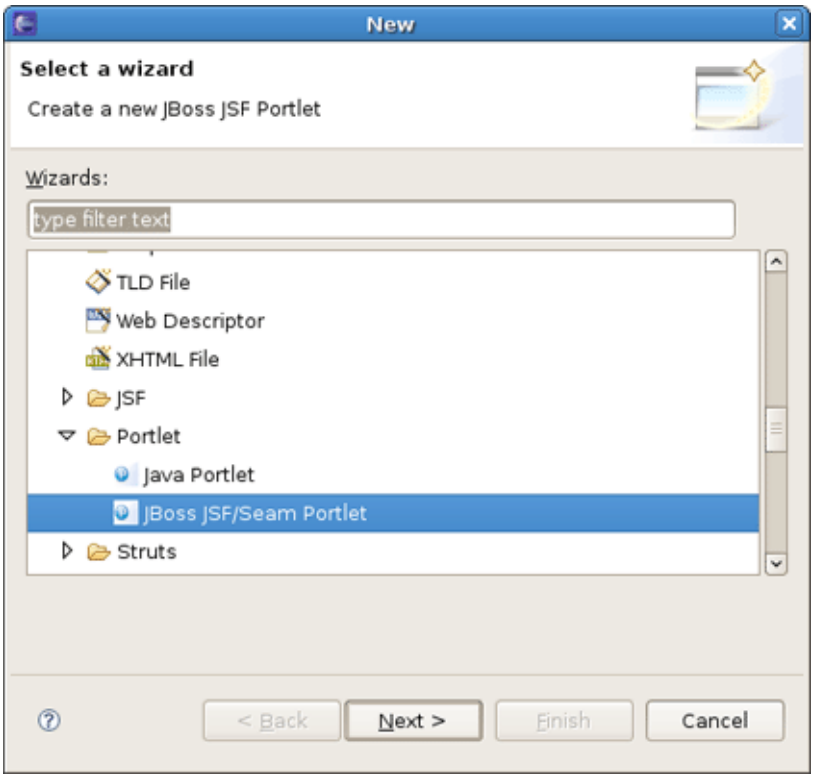

Figure 2.28. Calling the Create Portlet Wizard

2. The Create Portlet wizard starts (for more information about wizard options, see *"JSF/Seam Portlet Wizard"* in the guide reference). As the Seam configuration is set for the project, the wizard puts the values as for Seam portlet.

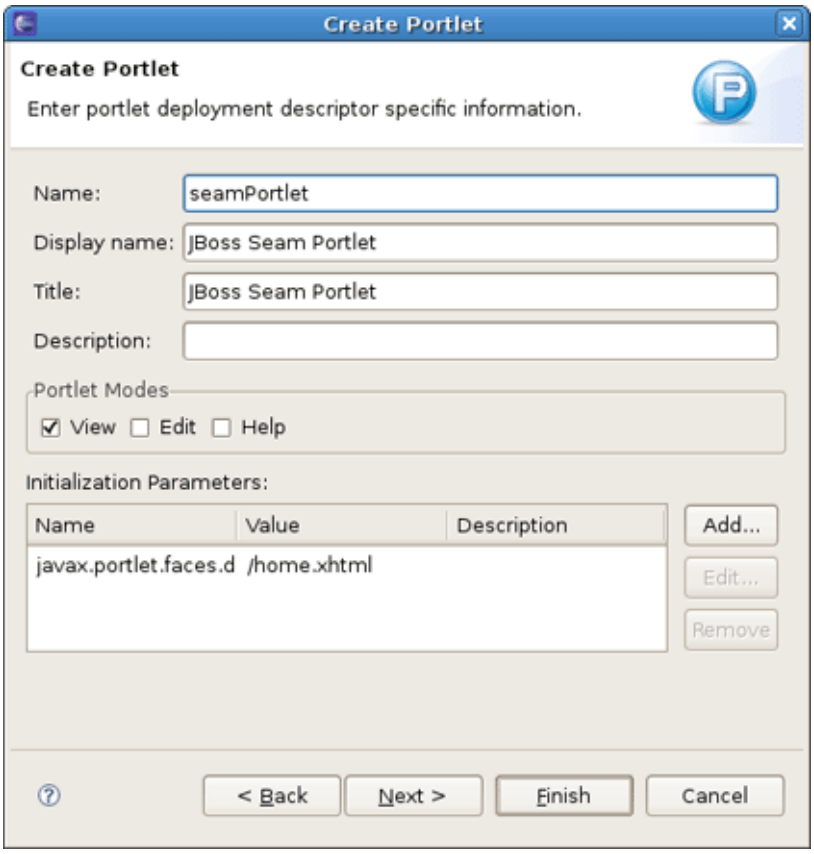

Figure 2.29. Creating Seam Portlet

3. Next two pages are filled out with default values, just press *Next* to proceed. On the last one click *Finish* to complete.

4. To deploy and run the portlet on JBoss Portal complete the steps described in the *"Deploying a Portlet to JBoss Portal"* procedure. Just use the following URL to see it in the browser: *http://localhost:8080/portal/portal/default/SeamPortlet*.

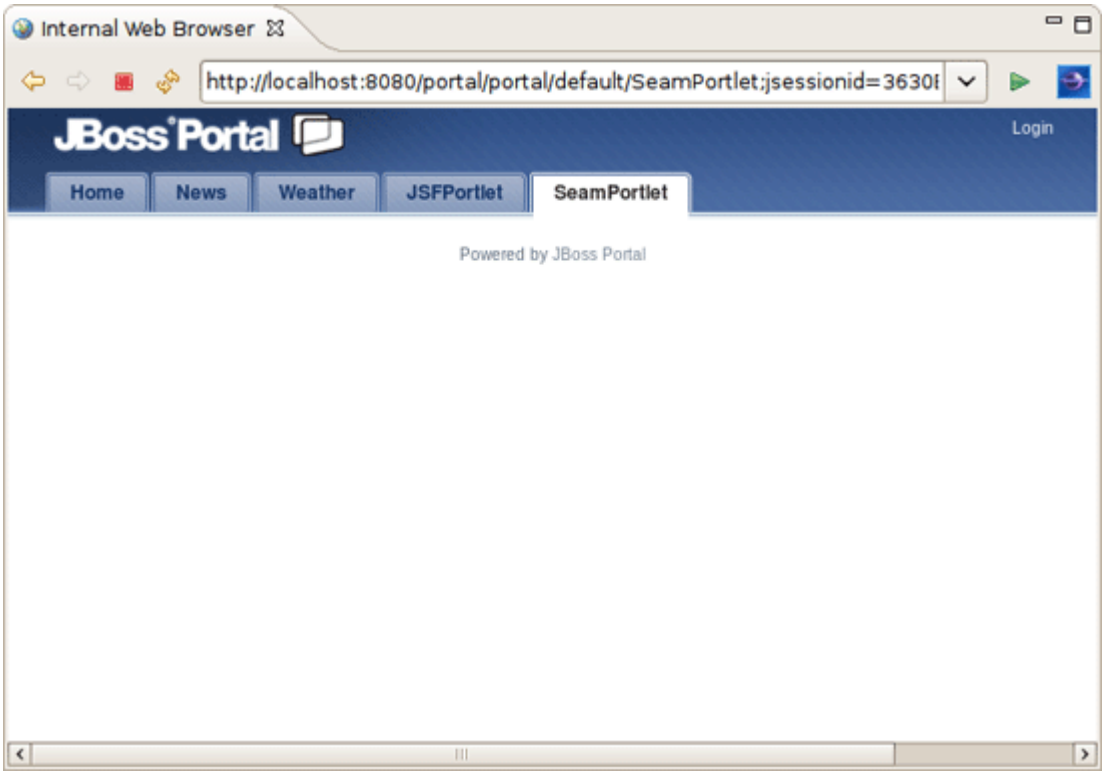

Figure 2.30. Seam Portlet in the Browser Window

Reference

This chapter includes detailed reference information about JBoss Portlet Tools.

3.1. JBoss Portlet Descriptors

In order to a web project with portlets could be deployed to JBoss Portal, it should include certain descriptors. The next table lists and describes these configuration files.

Table 3.1. JBoss Portlet Descriptors

Name	Description
portlet.xml	The mandatory portlet descriptor defined by the *JSR-168 Portlet Specification* [http://www.jcp.org/en/jsr/detail?id=168]. It is used during deployment.
portlet-instances.xml	The optional JBoss Portal specific descriptor. It allows to create a portlet instance from a portlet class definition.
*-object.xml	The optional JBoss Portal specific descriptor. It helps define the layout of the portal.
jboss-portlet.xml	The optional JBoss Portal specific descriptor. It allows to use JBoss specific functionality within a portlet application.

For more detail information about each descriptor, see *"Portlet Descriptors"* [http://docs.jboss.org/jbportal/v2.7.1/referenceGuide/html_single/#descriptors_portlet] in Chapter 6, *"JBoss Portal Reference Guide"* [http://docs.jboss.org/jbportal/v2.7.1/referenceGuide/html_single/].

3.2. Wizards

This section describes the JBoss Portlet Tools wizards:

- *Java Portlet wizard*
- *JSF/Seam Portlet wizard*

3.2.1. Java Portlet Wizard

The Java Portlet wizard helps you create a new Java portlet (1.0 or 2.0). It is based on the WTP's Create Servlet wizard, but adapted to JBoss Portlet. You can call it from the *File* or context menu of your project by following to *New > Other > JBoss Tools Web > Portlet > Java Portlet*.

The wizard consists of the four pages:

1. The first one includes the next options to adjust:

Table 3.2. Java Portlet Wizard. First Page Options.

Option	Description	Default
Project	Select the project to add a new Java portlet	The project selected when the wizard has been started
Source folder	The path to the source folder relative to the selected project	The source folder of the project selected in the *Project* field
Java package	Enter a package to contain a new portlet. Either type a valid package name or click *Browse* to select a package via the dialog box.	<blank>
Class name	Type a name for a new Java portlet class	<blank>
Superclass	Type or click *Browse* to select a superclass for a Java portlet class	javax.portlet.GenericPortlet
Use an existing Portlet class	Select this check box if you want to use an existing Java portlet class. In the appeared *Class name* field type or click *Browse* to select an existing portlet class.	Cleared

2. The second wizard page is for specifying modifiers, interfaces to implement and method stubs to generate.

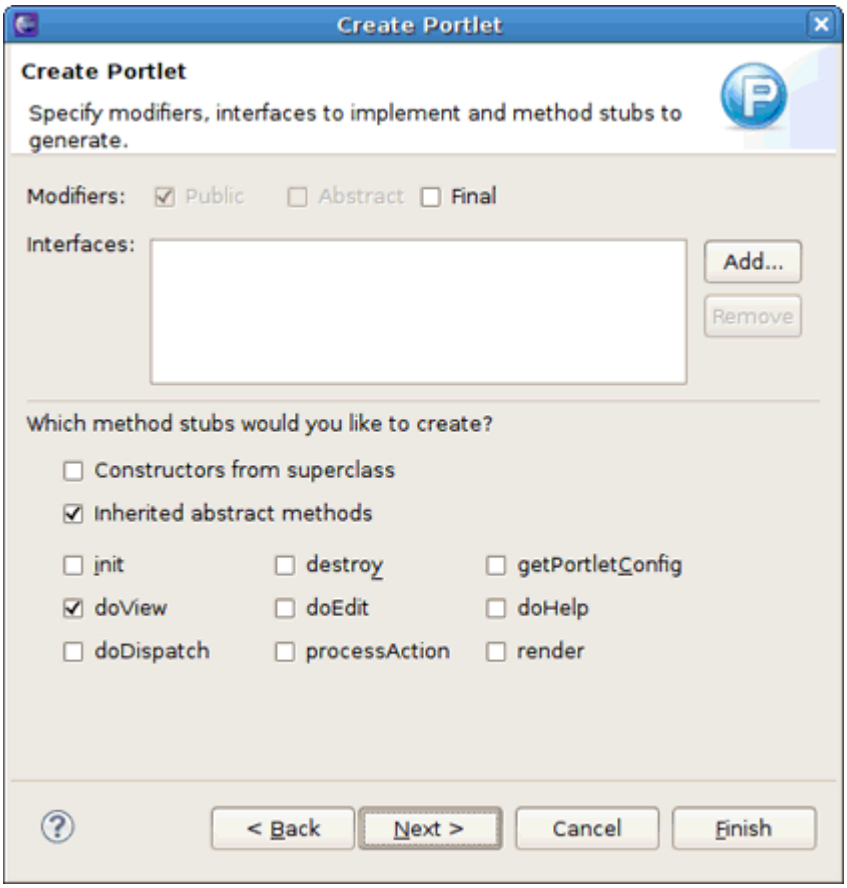

Figure 3.1. Second Page of the Java Portlet Wizard

The descriptions of the options listed on this page are in the following table.

Table 3.3. Java Portlet Wizard. Second Page Options.

Option	Description	Default
Modifiers	Select one or more access modifiers for a new Java portlet class: • Public • Abstract • Final	Public
Interfaces	Click *Add* to choose interfaces that a new class implements	<blank>
Which method stubs would you like to create?	Choose which methods from superclass to override in the Java portlet class: • Constructors from superclass	Inherited abstract methods, doView

Option	Description	Default
	• Inherited abstract methods • init • doView • doDispatch • destroy • doEdit • processAction • getPortletConfig • doHelp • render	

3. The third page is for adjusting the portlet deployment descriptor settings.

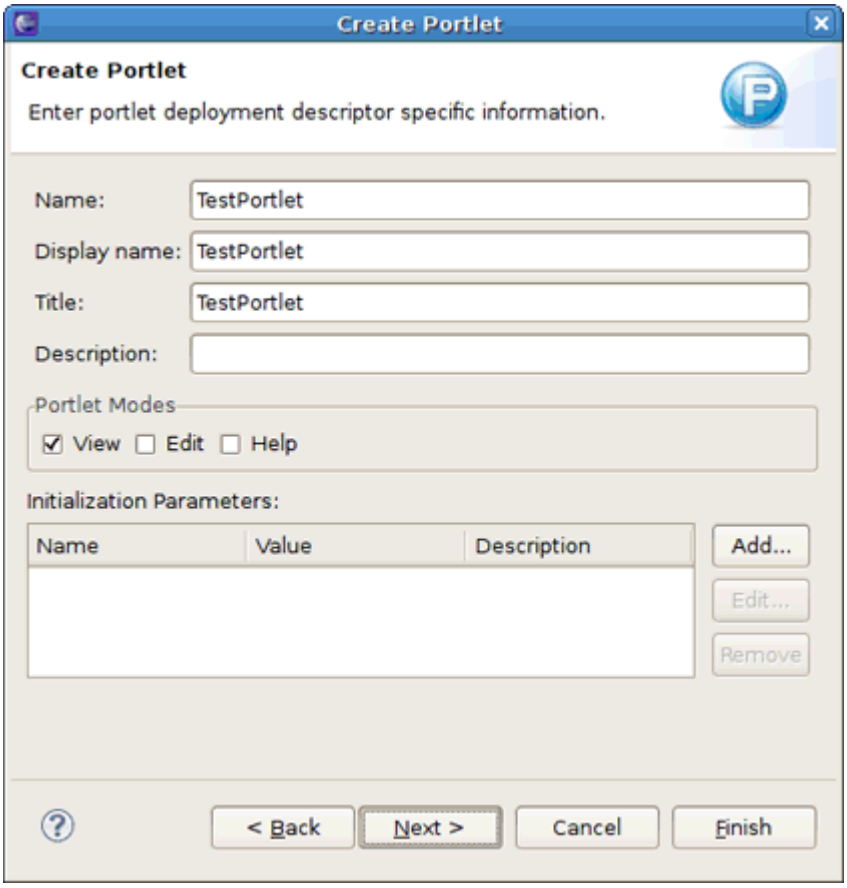

Figure 3.2. Third Page of the Java Portlet Wizard

Table 3.4. Java Portlet Wizard. Third Page Options.

Option	Description	Default
Name	Enter the portlet name. It does not have to be the portlet class name.	The name of the portlet class
Display name	Enter the display name of the Java portlet	The name of the portlet class
Title	Enter the title of the portlet	The name of the portlet class
Description	Enter the description of the portlet's function	<blank>
Portlet Modes	Select one or more of three modes defined by the JSR-286 specification: • *View* - generates markup reflecting the current state of the portlet; • *Edit* - allows customizing the behavior of the portlet; • *Help* - provides an information on how to use the portlet.	View
Initialization Parameters	Specify initialization parameters to create an initial state inside your portlet class. Click *Add* and fill in the *Name*, *Value* and *Description* (optional) fields to set a parameter.	<blank>

4. The last wizard page helps configure the JBoss Portlet specific descriptors.

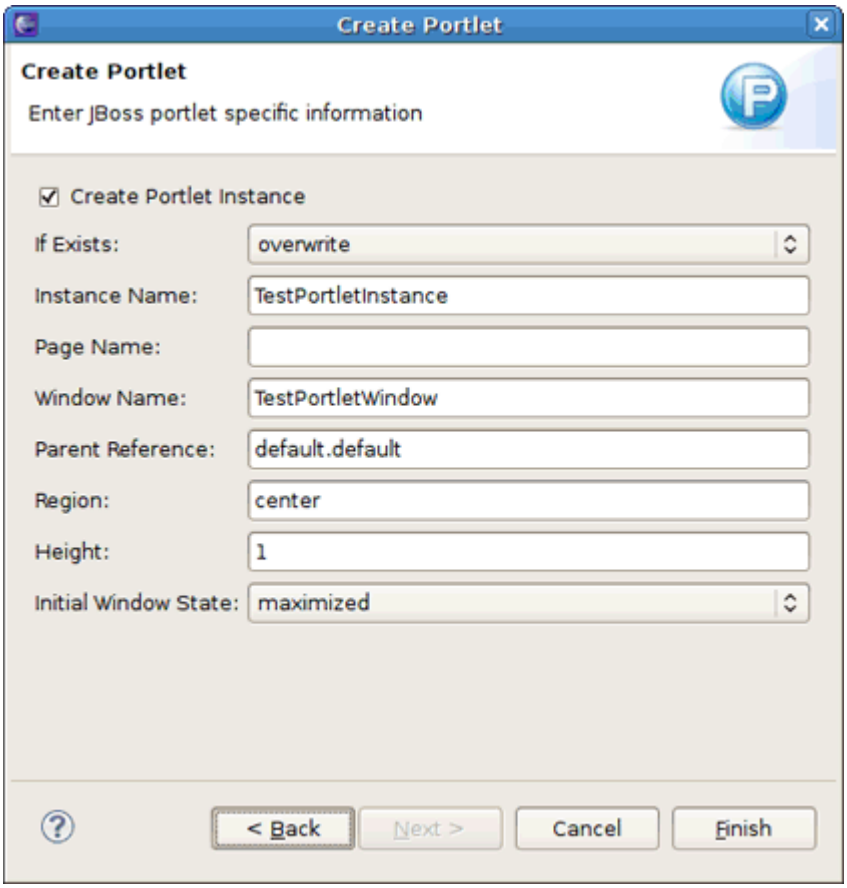

Figure 3.3. Last Page of the Java Portlet Wizard

Table 3.5. Java Portlet Wizard. Last Page Options.

Option	Description	Default
Create Portlet Instance	Clear the *Create Portlet Instance* check box, if you don't wont the wizard to create the deployment elements in the `default-object.xml` and `portlet-instances.xml` files	Selected
If Exists	Select one of two options: • *overwrite* - destroys the existing object and creates a new one based on the content of the deployment • *keep* - maintains the existing object deployment or creates a new one if it does not exist	overwrite
Instance Name	Enter a name of the portlet instance that the portlet window represents	[Portlet class name]Instance
Page Name	If set, the portlet page is created with the name defined	<blank>

Option	Description	Default
Window Name	Enter a name of the portlet window	[Portlet class name]Window
Parent Reference	Define a reference to the parent object. The syntax for this element is `[portal-instance].[portal-page]`. For example, `default.default` means that the window appears on the default page, in the default portal.	default.default
Region	Specify the region where the window should appear on the page	center
Height	Define the height of the window	1
Initial Window State	Set the window state indicator by selecting one of three states defined by the JSR-168 specification: • *normal* - a portlet shares this page with other portlets • *minimized* - a portlet may show very little information or none at all • *maximized* - a portlet may be the only portlet displayed on this page	maximized

3.2.2. JSF/Seam Portlet Wizard

The JSF/Seam Portlet wizard, that is merged into one JSF and Seam Portlet wizards, helps you create a JSF/Seam portlet that uses the JBoss Portlet Bridge libraries. The default values in this wizard depend on the configuration set for the project. For instance, if you set the JSF configuration, the default values in this wizard will be as for a JSF portlet.

The JSF/Seam Portlet wizard could be started from the *File* or context menu of your project by following to *New > Other > JBoss Tools Web > Portlet > JSF/Seam Portlet.*

The wizard includes three pages to adjust JSF/Seam portlet setting:

1. The first wizard page asks to select a project to add JSF/Seam portlet and set a class file destination.

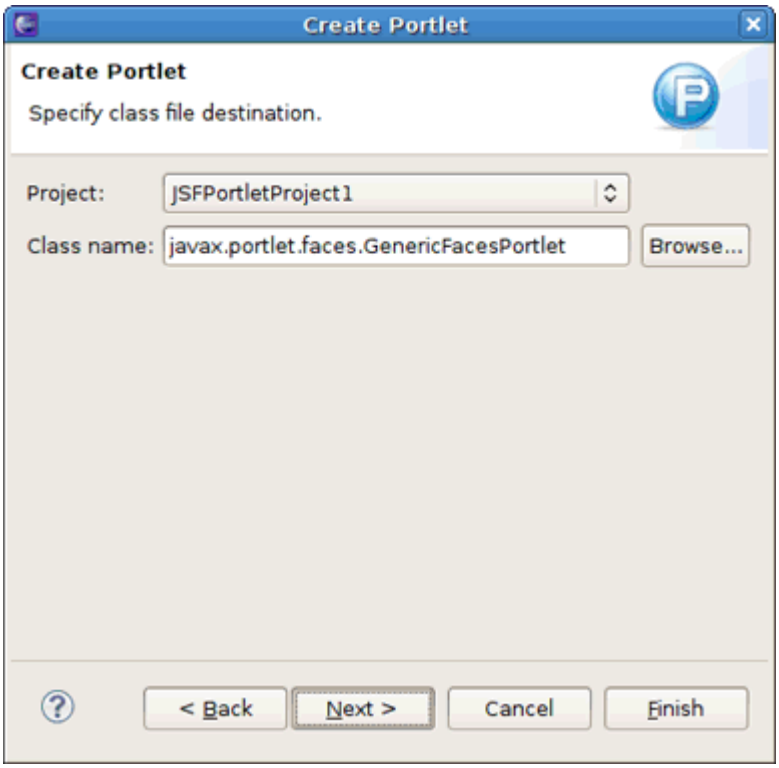

Figure 3.4. First Page of the JSF Portlet Wizard

Table 3.6. JSF/Seam Portlet Wizard. First Page Options.

Option	Description	Default
Project	Select the project to add a new JSF/Seam portlet	The project selected when the wizard has been started
Class name	Type or click *Browse* to select a class for a JSF/Seam portlet	javax.portlet.faces.GenericFacesPort

2. The second page allows to configure a portlet deployment descriptors.

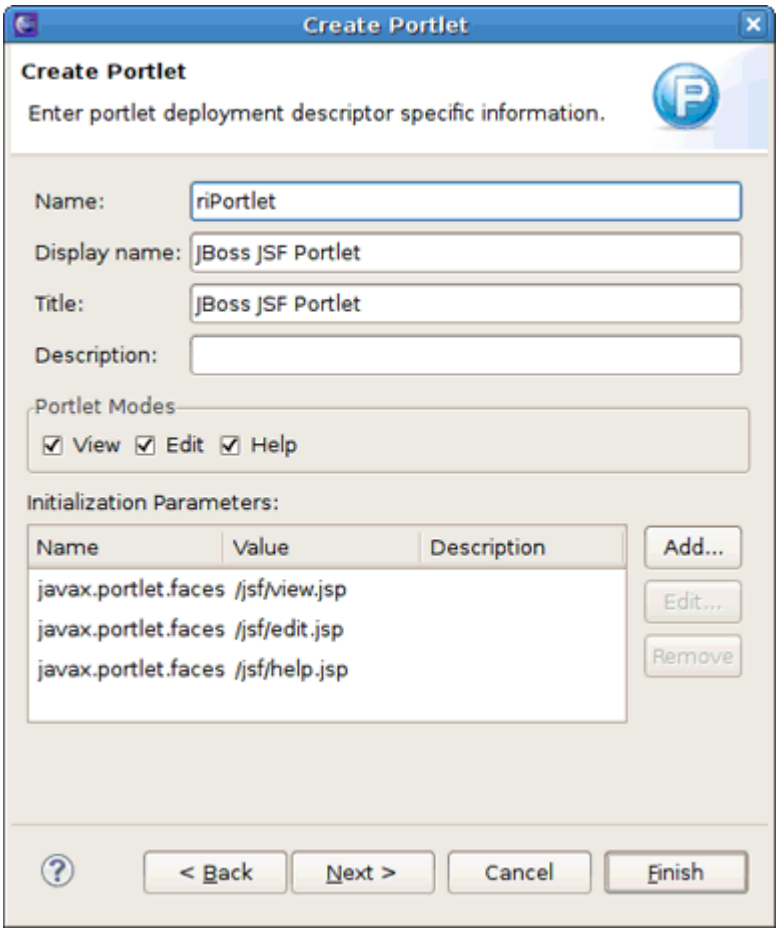

Figure 3.5. Second Page of the JSF Portlet Wizard

Table 3.7. JSF/Seam Portlet Wizard. Second Page Options.

Option	Description	Default (JSF / Seam)
Name	Enter the portlet name. It does not have to be the portlet class name.	riPortlet / seamPortlet
Display name	Enter the display name of the Java portlet	JBoss JSF Portlet / JBoss Seam Portlet
Title	Enter the title of the portlet	JBoss JSF Portlet / JBoss Seam Portlet
Description	Enter the description of the portlet's function	<blank>
Portlet Modes	Select one or more of three modes defined by the JSR-286 specification:	View, Edit, Help / View

Option	Description	Default (JSF / Seam)
	• *View* - generates markup reflecting the current state of the portlet • *Edit* - allows customizing the behavior of the portlet • *Help* - provides an information on how to use the portlet	
Initialization Parameters	Specify initialization parameters to create an initial state inside your portlet class. Click *Add* and fill in the *Name*, *Value* and *Description* (optional) fields to set a parameter.	javax.portlet.faces.defaultViewId.view javax.portlet.faces.defaultViewId.edit javax.portlet.faces.defaultViewId.help javax.portlet.faces.defaultViewId.view

3. The last wizard page helps you configure the JBoss Portlet specific information.

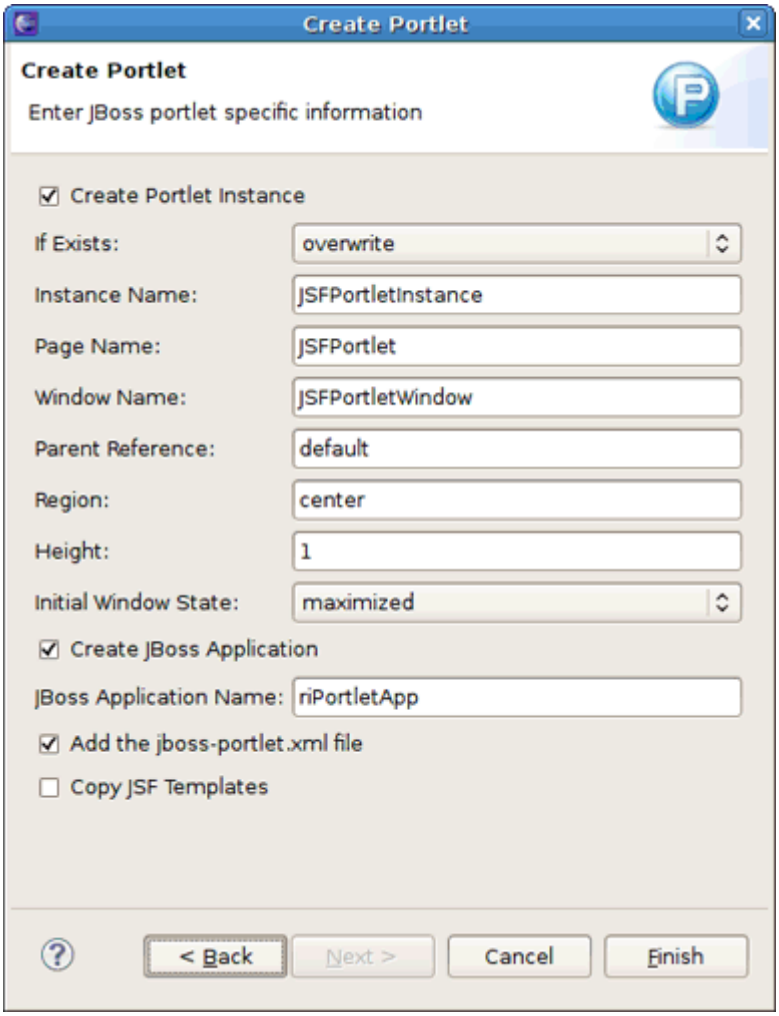

Figure 3.6. Third Page of the JSF Portlet Wizard

Table 3.8. JSF/Seam Portlet Wizard. Third Page Options.

Option	Description	Default (JSF / Seam)
Create Portlet Instance	Clear the *Create Portlet Instance* check box, if you don't wont the wizard to create the deployment elements in the `default-object.xml` and `portlet-instances.xml` files	Selected
If Exists	Select one of two options: • *overwrite* - destroys the existing object and creates a new one based on the content of the deployment • *keep* - maintains the existing object deployment or creates a new one if it does not exist	overwrite
Instance Name	Enter a name of the portlet instance that the portlet window represents	JSFPortletInstance / SeamPortletInstance
Page Name	If set, the portlet page is created with the name defined	JSFPortlet / SeamPortlet
Window Name	Enter a name of the portlet window	JSFPortletWindow / SeamPortletWindow
Parent Reference	Define a reference to the parent object. For example, `default` means that the page appears in the default portal.	default
Region	Specify the region where the window should appear on the page	center
Height	Define the height of the window	1
Initial Window State	Set the window state indicator by selecting one of three states defined by the JSR-168 specification: • *normal* - a portlet shares this page with other portlets • *minimized* - a portlet may show very little information or none at all • *maximized* - a portlet may be the only portlet displayed on this page	maximized
Create JBoss Application	Select to add the `jboss-app.xml` descriptor.	Selected

Option	Description	Default (JSF / Seam)
	riPortletApp / seamPortletApp - the default name of the JBoss application. You can change it by typing a new one in the *JBoss Application Name* field.	
Add the jboss-portlet.xml file	Select to create the `jboss-portlet.xml` file	Selected
Copy JSF Templates	Select to add the folder with JSF templates (`edit.jsp`, `help.jsp`, `view.jsp`) to the project	Cleared

3.3. JBoss Portlet Preferences

To open JBoss Portlet Preferences property sheet, go to *Window > Preferences* and then *JBoss Tools > JBoss Portlet*.

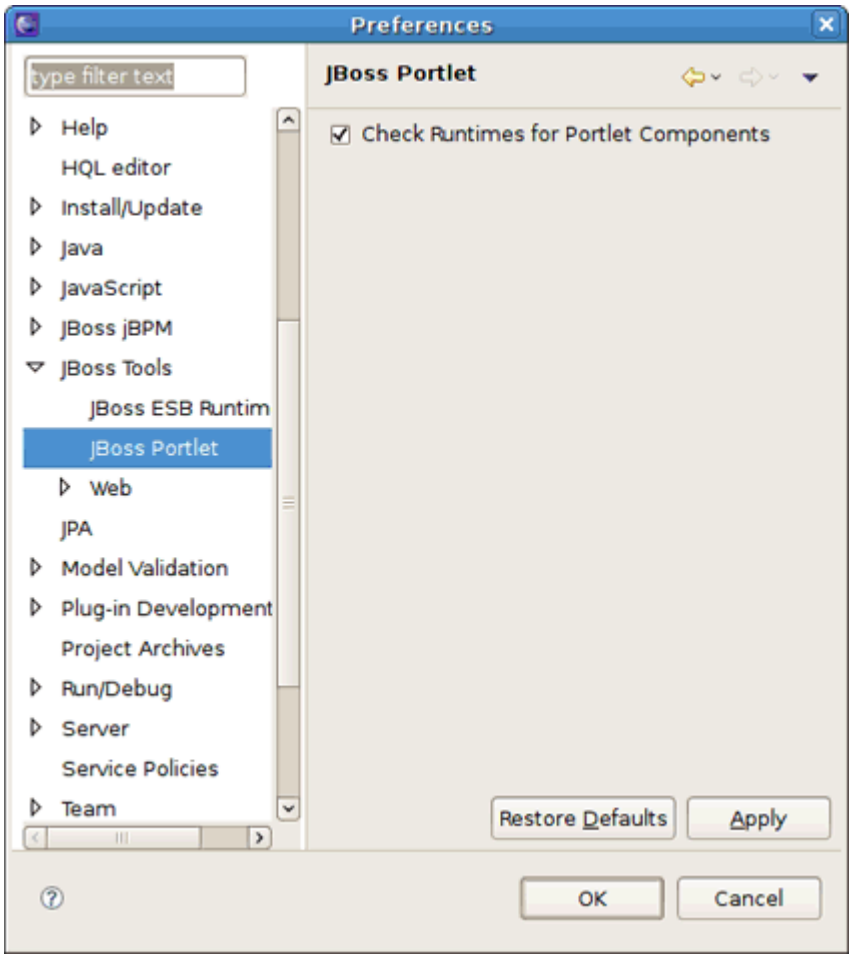

Figure 3.7. JBoss Runtime Preferences

The *Check runtimes for Portlet Components* option makes the JBoss Portlet functionality always visible no matter what the runtime is set. By default the option is checked.

Summary

In conclusion, with this document you could easily get started with JBoss Portlet Tools. The above chapters walked you through the steps on how to organize a web project with JBoss Portlet capabilities enabled, create a Java portlet, JSF or Seam portlet and deploy them to JBoss Portal. The document also includes the reference of JBoss Portlet Tools features.

If you have questions or suggestions concerned both the documentation and tools behavior, you are welcome to *JBoss Tools Users forum* [http://www.jboss.com/index.html?module=bb&op=viewforum&f=201].

Please, use *Jira* [https://jira.jboss.org/jira/browse/JBDS] to report bugs and requests on documentation.

4.1. Other Relevant Resources on the Topic

- JBoss Portal home page: *http://www.jboss.org/jbossportal/*
- JBoss Portal Wiki at: *http://wiki.jboss.org/wiki/JBossPortal*
- JBoss Portal documentation set at: *http://docs.jboss.org/jbportal/*
- JBoss Tools home page at: *http://www.jboss.org/tools/*
- The latest JBossTools/JBDS documentation builds at: *http://download.jboss.org/jbosstools/nightly-docs/*
- JBoss Developer Studio/JBoss Tools release documentation you can find at *http://docs.jboss.org/tools* [http://docs.jboss.org/tools/] in the corresponding release directory.