# **Getting Started Guide**

Version: 3.3.0.M5

1. Installation Instructions	1
1.1. Installing JBoss Tools Plugins	. 1
1.2. Usage Reporting	10
1.2.1. Collected usage information guide	11
2. Manage JBoss AS with JBoss Tools	13
2.1. How to Manage JBoss AS with JBoss Tools	13
2.1.1. Starting JBoss Server	15
2.1.2. Stopping JBoss Server	16
2.1.3. Server Container Preferences	17
2.2. How to Use Your Own JBoss AS Instance with JBoss Developer Studio	19
2.2.1. JBoss AS Installation	19
2.2.2. Adding and Configuring JBoss Server	19
3. Write Your First Project with JBoss Developer Studio	29
3.1. Create a Seam Application	29
3.1.1. Start Development Database	29
3.1.2. Create and deploy Seam Web Project	31
3.1.3. Start JBoss Application Server	47
3.1.4. Workshop Project Code Overview	53
3.2. Seam Action Development	54
3.2.1. Create a New Seam Action	55
3.2.2. Test Seam Action	56
3.2.3. Modify Seam Action User Interface	58
3.3. Declarative Security	61
3.3.1. Edit Login Authentication Logic	61
3.3.2. Secure Seam Page Component	62
3.4. Browsing Workshop Database	64
3.4.1. Database Connectivity Setup	65
3.4.2. Browse Workshop Database	65
3.5. Database Programming	67
3.5.1. Reverse Engineer CRUD from a Running Database	67
3.5.2. Use Hibernate Tools to Query Data via JPA	71
3.5.3. Use Hibernate Tools to visualize the Data Model	78
3.6. Rich Components	81
3.6.1. Add a Richfaces component to the CRUD Application	. 81
4. Developing a simple JSP web application	85
4.1. Setting Up the Project	85
4.2. Creating JSP Page	87
4.2.1. Editing a JSP Page	89
4.2.2. web.xml file	90
4.2.3. Deploying the project	92
4.2.4. JSP Page Preview	
4.2.5. Launch JSP Project	95
5. RAD development of a simple JSF application	
5.1. Setting up the project	

5.2. Creating JSP Pages	100
5.3. Creating Transition between two views	103
5.4. Creating Resource File	105
5.5. Creating Java Bean	109
5.6. Editing faces-config.xml File	114
5.7. Editing the JSP View Files	115
5.7.1. Editing inputnumber.jsp page	115
5.7.2. Editing success.jsp page	126
5.8. Creating index.jsp page	128
5.9. Running the Application	129
6. Project Examples	135
6.1. User Sites	137
6.2. Downloading a Project Example	137
6.3. Quick Fixes	144
7. FAQ	151
7.1. What should I do if the Visual Page Editor does not start under Linux	151
7.2. Visual Editor starts OK, but the Missing Natures dialog appears	152
7.3. Do I need to have JBoss Server installed to run JBoss Developer Studio?	153
7.4. I have an existing Seam 1.2.1 project. Can I migrate or import the project into a	
JBoss Developer Studio Seam project?	153
7.5. I have an existing Struts or JSF project. Can I open the project in JBoss Developer	
Studio?	153
7.6. Can I import a WAR file?	153
7.7. Is it possible to increase the performance of Eclipse after installing your product?. 1	54
7.8. How can I add my own tag library to the JBoss Tools Palette?	154
7.9. How to get Code Assist for Seam specific resources in an externally generated	
project?	154
7.10. How to import an example Seam project from jboss-eap directory?	154
7.11. Is a cross-platform project import possible for JBoss Developer Studio?	154
8. Further Reading	157

# **Installation Instructions**

# **1.1. Installing JBoss Tools Plugins**

The JBoss Tools plugins can be installed in Eclipse from the JBoss.org update site. JBoss Tools 3.2 requires Eclipse 3.6, which can be downloaded from the *Eclipse web site* [http:// www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/helios/].

To install the JBoss Tools plugins start Eclipse and select  $\textbf{Help} \rightarrow \textbf{Install New Software...}$ 

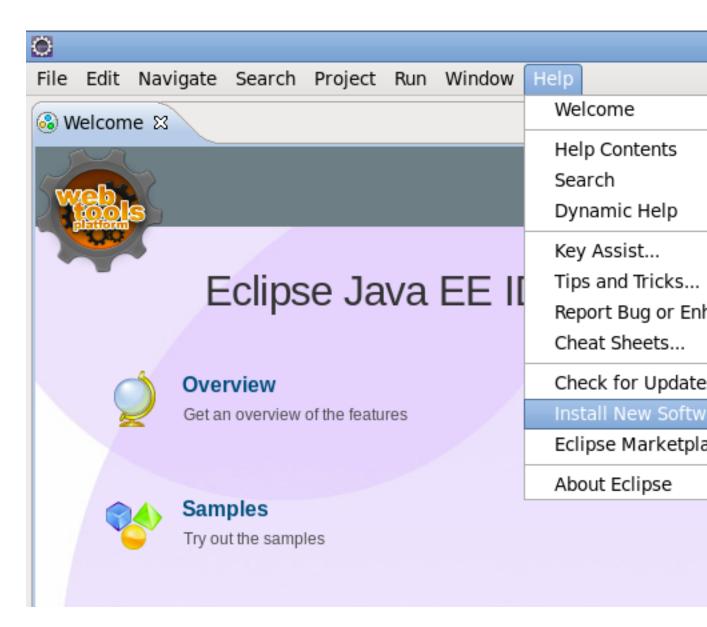


Figure 1.1. Install New Software

Click the Add... button.

	Install
Available Software	
Select a site or enter the location of a site.	
Work with: <sup>®</sup> type or select a site	
Find more s	software by working
type filter text	
Name	
① There is no site selected.	
Select All Deselect All	
Details	
Show only the latest versions of available software	🗌 Hide item
Group items by category	What is <u>alr</u>
Contact all update sites during install to find required	Isottware
Figure 1.2. Install Dialog	
?	< Back

This will display the **Add Repository** dialog. Enter **JBoss.org Tools** in the **Name** field, and *http://download.jboss.org/jbosstools/updates/JBossTools-3.2.0.GA*/ in the **Location** field. Click the **OK** button to save the changes and close the dialog.

۲	Add Repository	
Name:	JBoss.org Tools	Local.
Location:	http://download.jboss.org/jbosstools/updates/JBossToo	Archive
?	Cancel	ОК

#### Figure 1.3. Add Repository

The **JBoss.org Tools** site will be selected in the **Work with** drop down list, and after a moment the list of plugins that are included in the JBoss Tools package will be listed. From this list you can individually select the desired plugin, or select the **All JBoss Tools 3.2.0** option to install all the plugins.

	Install
Available Software	
Check the items that you wish to install.	
Work with: JBoss.org Tools - http://download.jboss.org/j	bosstools/updates/
F	ind more software
type filter text	
Name	
All JBoss Tools 3.2.0	
Application Development	
Image: Business Intelligence, Reporting and Charting	
Cloud	
Data Services	
Maven Support	
SOA Development	
Test and Performance	
Web and Java EE Development	
Select All Deselect All 26 items selected	
Details	
Contains ALL the plugins that are available from	
JBoss Tools except those related to integrat	tion with 3rd party
plugins. Selecting this category will give yo	
Figure 1.4. Available Software	
Show only the latest versions of available software	🗆 Hide
Group items by category	What
Contact all update sites during install to find required s	software

Click the **Next** button to calculate the system requirements and dependencies (this may take a little while). You will then be given an opportunity to review the plugins that will be installed.

# Ô

Install

## Install Details

Review the items to be installed.

Name	Version
Common tools for jBPM 3 and jBPM 4	4.4.0.v201
Context and Dependency Injection Tools	1.1.0.v201
🖗 FreeMarker IDE	1.1.0.v201
🖗 Hibernate Tools	3.4.0.v201
🖗 JBoss Archives Tools	3.2.0.v201
🖗 JBoss Drools Core	5.2.0.v201
🖗 JBoss Drools Guvnor	5.2.0.v201
🖗 JBoss Drools Task	5.2.0.v201
🖗 JBoss ESB Tools	1.3.0.v201
🖗 JBoss Portlet	1.1.0.v201
🖗 JBoss Runtime Detection Core	1.0.0.v201
🖗 JBoss Tools Community Project Examples	1.1.0.v201
🖗 JBoss Tools RichFaces	3.2.0.v201
🖗 JBoss Tools Usage Reporting	1.0.0.v201
Boss WebServices Tools	1.1.0.v201
	III
Cine, Halmann	

#### Size: Unknown

Details-

?

## Figure 1.5. Installation review

< Back

Click the **Next** button to install the selected plugins. You will be prompted to accept the various license agreements that cover the plugins that are to be installed. Review the licenses, select the **I accept the terms of the license agreements** option, and click the **Finish** button to install the plugins.

# Ö

# Install

# **Review Licenses**

Licenses must be reviewed and accepted before the software can be install

	Licenses: L		
	⊳	Eclipse Foundation Software User Agreement	$\left \right $
	⊳	Licensed under the Apache License, Version 2.0(the "License");	
	⊳	Red Hat, Inc. licenses these features and plugins to you under certain o	
Fi	igur	re 1.6. License review	
			l

Wait while the plugins are downloaded and installed.

۲	Installing Software
8	Installing Software
Fetching	org.eclipse.pde.core_3.6.2.v201de.core_3.6.2.v20110210_r36
🗆 Alwa	ys run in background
	Cancel Details >> Run in

#### Figure 1.7. Installing Software

You may be prompted with a warning informing you that you are attempting to install unsigned content. Click the **OK** button to continue.

0	Security Warning
	Warning: You are installing software that contains unsigned conte authenticity or validity of this software cannot be established. Do continue with the installation?
	OK Details >>

#### Figure 1.8. Unsigned Software Warning

You will then have to restart Eclipse to apply the new plugins. Click the **Restart Now** button to restart Eclipse.

	Software Updates		
?	You will need to restart Eclipse for the installation changes to take may try to apply the changes without restarting, but this may cau		
	Not Now Apply Changes Now Re		

#### Figure 1.9. Restart Eclipse

The plugin is now installed and ready to use.

## **1.2. Usage Reporting**

JBoss Tools now includes a usage plug-in that anonymously reports information back to JBoss. The plug-in is not enabled by default. To enable, click the **Yes** button.

0	JBoss Tools Us	age	$\mathbf{X}$		
Will you allow JBoss Tools team to receive anonymous usage statistics for this Eclipse instance with JBoss Tools ( <u>info</u> )?					
Yes No					

#### Figure 1.10. Usage plug-in pop-up

Once enabled, the plug-in will remain active until turned off. To turn the active plug-in off, navigate to Window  $\rightarrow$  Preferences  $\rightarrow$  JBoss Tools  $\rightarrow$  Usage Reporting.

The gathered data allows JBoss to see how the tools are being used and where they are being used geographically. Currently we are looking into the operating systems being used, screen resolution and how often the tooling environment is started. In the future geographic information will assist in focusing translation resources to areas where the developer environment is most used.

The plug-in uses Google Analytics to track and report data by acting as if you were visiting the site http://jboss.org/tools/usage/. To view the type of information being collected, refer to Section 1.2.1, "Collected usage information guide".

To view the source code of the usage plug-in visit *http://anonsvn.jboss.org/repos/jbosstools/trunk/usage/*.

#### 1.2.1. Collected usage information guide

Below you will find an outline of the information that is reported and the Goolge Analytics fields that are used to gather this information.

Version

The **Content** field has been modified to report the installed JBoss Developer Studio version. Sample returned values include: jbdevstudio-linux-gtk-x86\_64-4.0.0.v201009301221R-H20-Beta1.jar and jbdevstudio-linux-gtk-3.0.2.v201009161622R-H138-GA.jar.

Installed components

The **Keyword** field has been modified to report the installed JBoss Developer Studio components. Sample returned values include: JBoss AS, Drools, Teiid and ModeShape.

Visitor type

The Visitor type field reports if the current user is new or returning.

Language

The **Language** field reports the localized language the product is being used in. Sample returned values include: en-US, de-DE and fr-FR.

Location fields

The location fields report the geographical location where the product is being used based on the continent, country and city. Sample returned values include: Europe (continent), Germany (country) and Munich (city).

#### Eclipse interface and version

The **Browser** field has been modified to report the Eclipse interface and version being used. Sample returned values include: JBoss Developer Studio: 3.0.0 and JBoss Developer Studio: 3.0.1.

**Operating System** 

The **Operating System** field reports the Operating System and its version that the product is running on (with Linux distribution version reporting conducted through the **User Defined Value** field). Sample returned values include: Linux, Macintosh 10.4, Macintosh 10.6, Windows XP and Windows 7.

Linux distribution version

The **User Defined Value** field reports the distribution and version of Linux, if one is being used as the Operating System. Sample returned values include: Red Hat Enterprise Linux 5.4 and Fedora 13.

#### Screen colors

The **Screen colors** field reports the color depth being used. Sample returned values include: 32-bit and 24-bit.

#### Screen resolution

The **Screen resolution** field reports the resolution being used. Sample returned values include: 2048x1536 and 1920x1080.

#### Java version

The **Flash version** field has been modified to report the Java version used. Sample returned values include: 1.6.0\_20 and 1.5.0\_9.

#### Connection speed

The **Connection speed** field reports the type of internet connection being used. Sample returned values include: T1, Cable and DSL.

# Manage JBoss AS with JBoss Tools

In this chapter we'll focus more on how to operate the JBoss AS from JBoss Tools.

JBoss Tools can be used to operate a wide range of JBoss Application Server versions, which can be downloaded from the *JBoss Application Server* [http://www.jboss.org/jbossas/downloads.html] website.

# 2.1. How to Manage JBoss AS with JBoss Tools

This section covers the basics of working with the JBoss Server supported directly by JBoss Developer Studio via bundled AS plug-in. The server points to the JBoss Enterprise Application Platform Runtime shipped with JBoss Developer Studio.

۲		Preferences	
ty	pe filter text 🛛 🔏	Server Runtime Envir	onments
⊳	General	Add, remove, or edit ser	ver runtime e
⊳	Ant		
⊳	Data Management	Server runtime e <u>n</u> vironn	1
⊳	Drools	Name	Туре
	Drools Task	🕻 JBoss 5.1 Runtime	JBoss 5.1 R
	FreeMarker Editor		
	Guvnor		
⊳	Help		
	HQL editor		
⊳	Install/Update		
⊳	Java		
⊳	Java EE		
⊳	Java Persistence		
⊳	JavaScript		
⊳	JBoss jBPM		
⊳	JBoss Tools		
	ModeShape		
⊳	Plug-in Development		
	Project Archives		
	Remote Systems		
⊳	Run/Debug		
$\neg$	Server		
	Audio		
	Default Filesets		
	Launching		
Figu	re 2.1. JBoss Enterprise Application Pla Runtime Environments	atform Runtime	
⊳	Spring		
⊳	Team		
⊳	Teiid Designer		

To read more about AS plug-in, refer to the Server Manager guide.

#### 2.1.1. Starting JBoss Server

Starting JBoss Server is quite simple. JBoss Developer Studio allows you to control its behavior with the help of a special toolbar, where you could start it in a regular or debug mode, stop it or restart it.

 To launch the server click the green-with-white-arrow icon in the Servers view or right click server name in this view and click the Start button. If this view is not open, select Window → Show View → Other → Server → Servers

8	Problems 🙆 Tas	sks 💿 Seam Compo	nents 👭 Servers	5 23
$\nabla$	🖕 JBoss 5.1 Run	time Server [Stoppe	d]	
	🕨 🖹 XML Config	guration		
	🗟 Filesets			

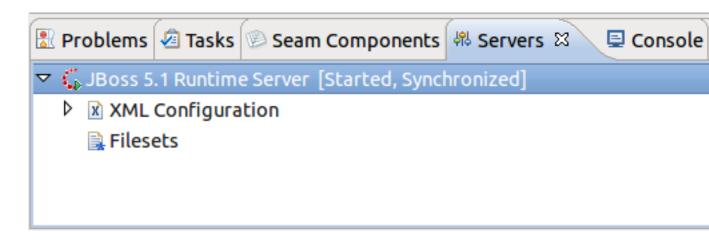
#### Figure 2.2. Starting from Icon

While launching, server output is written to the Console view:

Problems 🧟 Tasks	📨 Seam Components 👫 Servers 💷 Console 😂
JBoss 5.1 Runtime Serv	ver [JBoss Application Server Startup Configuration] /
07:57:12,238 INF0	[Http11Protocol] Initializing Coyote HTTP/1.
07:57:12,239 INF0	[AjpProtocol] Initializing Coyote AJP/1.3 on
07:57:12,260 INF0	[StandardService] Starting service jboss.web
07:57:12,263 INF0	[StandardEngine] Starting Servlet Engine: JB
07:57:12,320 INF0	[Catalina] Server startup in 80 ms
07:57:12,344 INF0	[TomcatDeployment] deploy, ctxPath=/jbossws
07:57:13,953 INF0	[TomcatDeployment]quired license terms exist
07:57:14,013 INF0	[TomcatDeployment]quired license terms exist
07:57:14,472 INF0	[RARDeployment] Re
07.57.14 407 THEO	[PAPDonloymont] Pc
<u> </u>	

#### Figure 2.3. Console Output

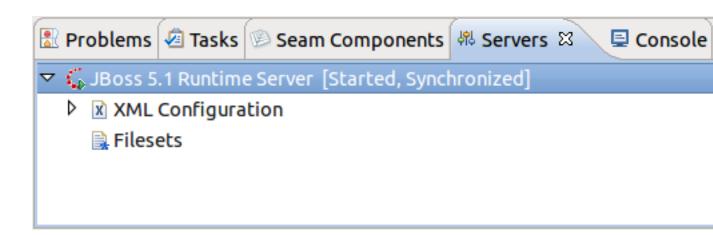
When the server is started you should see *Started* in the square brackets right next its name in the Servers view.



#### Figure 2.4. Server is Started

#### 2.1.2. Stopping JBoss Server

To stop the server, click the **Stop** button icon in Servers or right click the server name and press **Stop**.



#### Figure 2.5. Stopping Server

When the server is stopped you will see *Stopped* in the square brackets next to its name.

#### 2.1.3. Server Container Preferences

You can control how JBoss Developer Studio interacts with server containers in the Server editor. Double-click the server to open it in the editor.

🕻 JBoss 5.1 Runtime Server		
B Overview		
General Information Specify the host name and	d other common settings.	
Server name:	JBoss 5.1 Runtime Server	
Host name:	localhost	
Runtime Environment:	JBoss 5.1 Runtime	\$
Open launch configurati	ion	
- Server Behaviour		
Local	0	
- JMX Login Credential	s	
Set the JMX login and pa This is used by all JMX co	ssword for your server. mmands, and during server shutdown.	
User Name admin		
Password admin		
Overview Deployment		

Figure 2.6. Server Overview

Here you can specify some common settings: host name, server name, runtime as well as settings related to publishing, timeouts and server ports.

# 2.2. How to Use Your Own JBoss AS Instance with JBoss Developer Studio

Although JBoss Developer Studio works closely with JBoss EAP 5 we do not ultimately tie you to any particular server for deployment. There are some servers that Studio supports directly (via the bundled Eclipse WTP plug-ins). In this section we discuss how to manage self-installed JBoss AS. Suppose you want to deploy the application to JBoss 4.2.3 server. First of all you need to install it.

#### 2.2.1. JBoss AS Installation

 Download the binary package of JBoss AS, e.g. JBoss 4.2.3 and save it on your computer: http://labs.jboss.com/jbossas/downloads

It does not matter where on your system you install JBoss server.



#### Note:

The installation of JBoss server into a directory that has a name containing spaces provokes problems in some situations with Sun-based VMs. Try to avoid using installation folders that contain spaces in their names.

There is no requirement for root access to run JBoss Server on UNIX/Linux systems because none of the default ports are within the 0-1023 privileged port range.

• After you have the binary archive you want to install, use the JDK jar tool (or any other ZIP extraction tool) to extract the jboss-4.2.3.GA.zip archive contents into a location of your choice. The jboss-4.2.3.GA.tgz archive is a gzipped tar file that requires a gnutar compatible tar which can handle the long pathnames in the archive. The extraction process will create a jboss-4.2.3.GA directory.

#### 2.2.2. Adding and Configuring JBoss Server

Now we should add the just installed server into server manager in JBoss Developer Studio.

- Select the Servers view by selecting Window  $\rightarrow$  Show View  $\rightarrow$  Other  $\rightarrow$  Server  $\rightarrow$  Servers.
- Right click anywhere in this view and select  $\textbf{New} \rightarrow \textbf{Server}.$
- \* Select JBoss Community  $\rightarrow$  JBoss 4.2 Server

<b>9</b>	New Server
Define a New Server	
Choose the type of server to o	reate
Coloct the converture	Download additional se
Select the server type:	
type filter text	
JBoss AS 4.0	
JBoss AS 4.2	
JBoss AS 5.0	
JBoss AS 5.1	
JBoss AS 6.0	
👂 🗁 JBoss Enterprise Middle	eware
ObjectWeb	
JBoss Application Server 4.2	
Server's <u>h</u> ost name:	localhost
Server's <u>n</u> oschame.	localitosc
Server na <u>m</u> e:	JBoss 4.2 Runtime Server
? < <u>B</u> a	ick <u>N</u> ext > Cancel

Figure 2.7. Selecting Server Name and Server Type



#### Note:

Now in the New Server wizard there is a separation between the .org servers (the *JBoss Community* category) and product server that comes bundled with JBoss EAP (the *JBoss Enterprise Middleware* category).

- To create a new runtime, which Jboss AS 4.2 matches to, click the Next button
- In the next step you need to specify the location of the Server and define JRE to be used.

<b>9</b>	New Server
JBoss Runtime	
JBoss Application Se	rver 4.2
It can be used to set u	ne references a JBoss installation directory. Ip classpaths for projects which depend on this runti " which will be able to start and stop instances of JB
Name	
JBoss 4.2 Runtime	
Home Directory	
/redhat/apps/jboss	-4.2.3.GA
JRE	
java-6-sun-1.6.0.22	
Configuration	
Directory: server	
🤹 default	
i minimal	
🧌 all	
(?)	< <u>B</u> ack <u>N</u> ext > Cancel

Figure 2.8. Defining JBoss Runtime



#### Note:

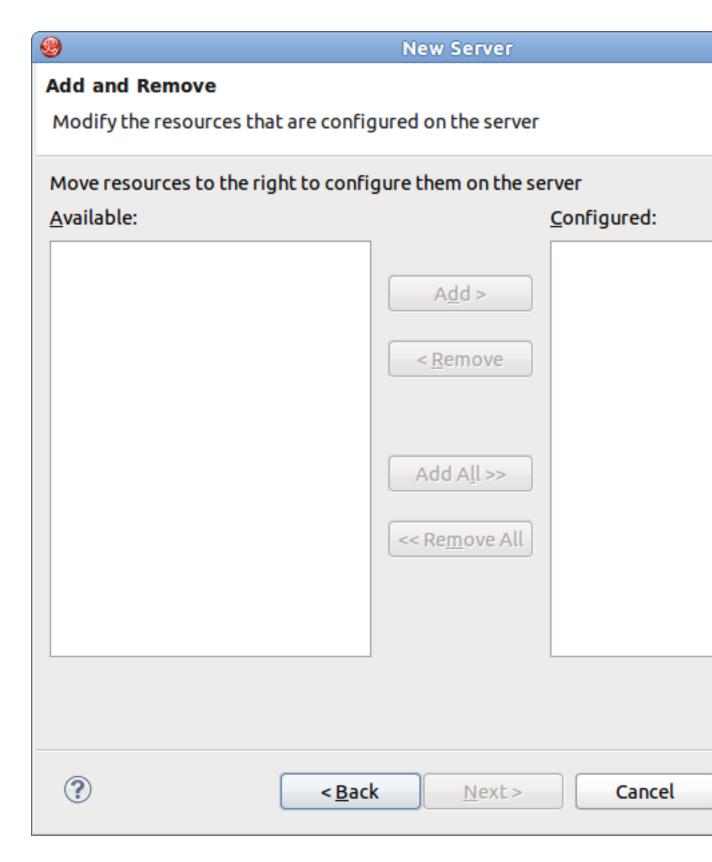
When adding a new server you will need to specify what JRE to use. It is important to set this value to a full JDK, not JRE. Again, you need a full JDK to run Web applications, JRE will not be enough.

• In the next dialog verify the specified information and if something is unfair go back and correct it

<b>8</b>	New Server	
Create a new JBoss Se	erver	
JBoss Application Serve	r 4.2	
_	starting and stopping instances of JBoss.	
Runtime Information	e arguments and keeps track of which modules ha	
	ion below is incorrect, please press back, Installed	
	a new runtime from a different location.	
Home Directory	/redhat/apps/jboss-4.2.3.GA	
JRE /usr/lib/jvm/java-6-sun-1.6.0.22 (java-6-sun-1		
Configuration Location		
Configuration	default	
Server Behaviour		
Local	<	
(?)	< Back Next > Cancel	
?	< <u>Back Next</u> > Cancel	

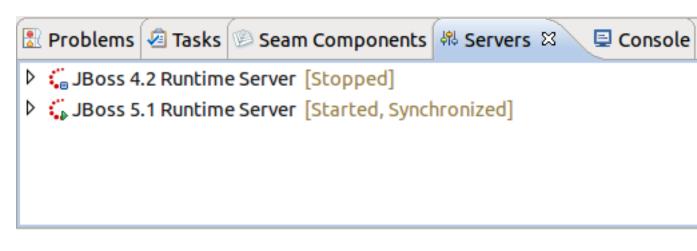
Figure 2.9. JBoss Runtime Summary

• In the last wizard's dialog modify the projects that are configured on the server and click the **Finish** button.



#### Figure 2.10. Configuring Projects

A new JBoss Server should now appear in the Servers view.



#### Figure 2.11. New JBoss Server

Now, we are ready to create the first web application.

# Write Your First Project with JBoss Developer Studio

This chapter is a set of hands-on labs. You get step-by-step information about how JBoss Developer Studio can be used during the development process.

## 3.1. Create a Seam Application

In this section you will learn how to create a Seam project in JBoss Developer Studio, how to start the server and what structure your project has after it is created.

#### 3.1.1. Start Development Database

Before opening the JBoss Developer studio you need to download and start the *Workshop Database* [http://docs.jboss.org/tools/resources/GSG\_database.zip].

To start the database just run ./runDBServer.sh or runDBServer.bat from the database directory.

The end result should be a console window that looks like:

Terminal - matthew@matthew-desktop: /redhat/p
<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>T</u> erminal <u>G</u> o <u>H</u> elp
<pre>matthew@matthew-desktop:/redhat/projects/database\$ matthew@matthew-desktop:/redhat/projects/database\$ Starting database engine [Server@19b49e6]: [Thread[main,5,main]]: checkRunni [Server@19b49e6]: [Thread[main,5,main]]: checkRunni [Server@19b49e6]: Startup sequence initiated from m [Server@19b49e6]: Loaded properties from [/redhat/p ver.properties] [Server@19b49e6]: Initiating startup sequence [Server@19b49e6]: Server socket opened successfully</pre>
<pre>[Server@19b49e6]: Database [index=0, id=0, db=file: ssfully in 503 ms.</pre>
[Server@19b49e6]: Startup sequence completed in 539 [Server@19b49e6]: 2011-01-14 08:36:09.262 HSQLDB se [Server@19b49e6]: To close normally, connect and ex [Server@19b49e6]: From command line, use [Ctrl]+[C]

Figure 3.1. Starting the Database



#### 3.1.2. Create and deploy Seam Web Project

Minimize the terminal window and run JBoss Developer Studio from Applications Menu or from the desktop icon.

First you will see the Workspace Launcher. Change the default workspace location if it's needed. Click the **OK** button.

<b>9</b>	Workspace Launcher
Select a wor	rkspace
	oper Studio stores your projects in a folder called a workspace rkspace folder to use for this session.
<u>W</u> orkspace:	/redhat/workspaces/workspace-jbds4/
Use this a	s the default and do not ask again
	Cancel

#### Figure 3.2. Workspace Launcher Dialog

After startup, you see the welcome page. You could read how to work with welcome pages in *previous* chapter. Now select **Create New...** icon and then press on **Create Seam Project** link.

The New Seam Project wizard is started. You need to enter a name (e.g., "workshop") and a location for your new project. The wizard has an option for selecting the actual Server (and not just WTP runtime) that will be used for the project. This allows the wizard to correctly identify where the destination folder for the required datasource and driver libraries.

Seam Project		
Seam Web Project		
Create standalone Seam Web Project		
Project na <u>m</u> e: workshop		
Project location		
✓ Use <u>d</u> efault location		
Location: /redhat/workspaces/workspace-jbds4/workshop		
Target r <u>u</u> ntime		
jboss-eap Runtime	Ne	v
Dynamic web module <u>v</u> ersion		
2.5		
Target Server		
jboss-eap		
Configuration		
Dynamic Web Project with Seam 2.2	•	
Configures a Dynamic Web application to use Seam v2.2		
Figure 3.3. New Seam Project Wizard		



All settings are already specified here, you can just modify the Configuration. Click on the **Modify...** button to configure your custom facet options:

( <u>@</u>	Project Facets
Project Facets	
	t
Select the facets that should be enabled for this proj	ecc.
Configuration: Dynamic Web Project with Seam 2.2	
Project Facet	Version
Axis2 Web Services	
CDI (Contexts and Dependency Injection)	1.0
CXF 2.x Web Services	1.0
🗹 👼 Dynamic Web Module	2.5
🗹 🛃 Java	1.5
🗌 📄 JavaScript	1.0
✓ JavaServer Faces	1.2
JAX-RS (REST Web Services)	1.1
JBoss Portlets	
🗌 🏄 JBoss Web Services Core	3.0
□ ↔ JPA	2.0
🗹 🗎 Seam	2.2
🗌 📄 WebDoclet (XDoclet)	1.2.3
?	
Figure 3.4. Project Facets Specifying	

On the whole the dialog allows to select the "features" you want to use in your project. JBoss Developer Studio will then setup the appropriate tooling for your project. Since JBoss Seam integrates all popular Java EE frameworks, you can select any combination of technologies from the list. Here, for the default configuration, Dynamic Web Module, Java, JavaServer Faces (JSF), and Seam Facet are already selected for a typical database-driven web application. The default project facets should suffice.

In the Project Facets form you can also bring up server runtimes panel by clicking Runtimes tab on the right corner. This panel shows available server runtimes.

	Declark Frenchs			
89 · · · · · · · · · · · · · · · · · · ·	Project Facets			
Project Facets				
Select the facets that should be enabled for this proj	ect.			
<u>Configuration</u> : Dynamic Web Project with Seam 2.2				
Project Facet	Version			
🕨 📃 🗎 Axis2 Web Services				
CDI (Contexts and Dependency Injection)	1.0			
CXF 2.x Web Services	1.0			
🗹 🚋 Dynamic Web Module	2.5			
🗹 🛃 Java	1.5			
🗌 🗎 JavaScript	1.0			
🗹 📄 JavaServer Faces	1.2			
JAX-RS (REST Web Services)	1.1			
JBoss Portlets				
🗌 🏄 JBoss Web Services Core	3.0			
□ ↔ JPA	2.0			
🗹 🖹 Seam	2.2			
🗌 📄 WebDoclet (XDoclet)	1.2.3			
(?)				
Figure 3.5. Runtimes Selecting				

Click the **OK** and then the **Next** button to proceed to the next step.

A dynamic web application contains both web pages and Java code. The next wizard will ask you where you want to store Java files.

<b>9</b>	(	New Seam Pro	ject	
Java				
Configure project fo	or building a Ja	ava application.		
Source folders on b	uild path:			
SLC				
Default output fold	er:			
build/classes				
?	< <u>B</u> ack	<u>N</u> ext >	Canc	el

### Figure 3.6. Java Build Path

Following page provides you Web Module Settings .You can just leave the default values or choose another folder.

	Manu Garan Bashach
<b>(</b>	New Seam Project
Web Module	
Configure web mo	dule settings
configure web mo	duc sectings.
Context <u>r</u> oot:	workshop
Content directory:	WebContent
concerne <u>a</u> n eccory.	
✓ <u>G</u> enerate web.x	ml deployment descriptor
?	< <u>B</u> ack <u>N</u> ext > Cancel
$\odot$	

#### Figure 3.7. Web Module Settings

On the next form, you will be able to select where those library JARs come from. The easiest is just to select the JARs provided by the JBoss AS runtime associated with this project. That is why it is important to chose the right JBoss AS 4.2 runtime in the project setup window.

• Select *Library Provided by Target Runtime* as Type of JSF Implementation Library. We will use the JSF implementation that comes with JBoss server.

• Click the Next button

<b>9</b>	New Seam Project			
JSF Capabilities				
Add JSF capabilities to t	this Web Project			
JSF Implementation Lib	JSF Implementation Library			
Type: Library Provided	d by Target Runtime			
-	s able to provide the library required by this facet. re the project to use that library.			
JSF <u>Configuration File</u> :	/WEB-INF/faces-config.xml			
JSF <u>S</u> ervlet Name:	Faces Servlet			
JSF Servlet C <u>l</u> ass Name:	javax.faces.webapp.FacesServlet			
URL Mapping Patterns:	*.seam			
2	< Back Next > Cancel			
	< <u>B</u> ack <u>N</u> ext > Cancel			

# Figure 3.8. JSF Capabilities Adding

Next wizard step needs more settings that previous. Let's start with General section.

Leave the default Seam runtime and check a WAR deployment.

9	New Seam Project	
Seam Facet		
Configure Seam Fa	cet Settings	
General	General	
Seam Runtime:	Seam 2.2.EAP5	
Deploy as:	● WAR ○ EAR	
EJB project name:	workshop-ejb	
EAR project name:	workshop-ear	

#### Figure 3.9. Seam Facet Setting

The Database section is a little tricky. The Connection Profile needs to be edited so that the new project works properly with the external HSQLDB server. By default the project wizard tries to use the JBoss embedded HSQLDB, but the tutorial uses an external database to replicate a more real world development scenario. Click on the **Edit** button to modify the Connection Profile.

Database		
Database Type:	HSQL	
Connection profile:	DefaultDS 😂 Ed	i
Database Schema Name:		
Database Catalog Name:		
DB Tables already exists in database:		
Recreate database tables and data on deploy:		

### Figure 3.10. DataBase Setting

Select HSQLDB Profile Properties. Make sure the Database location is set to hsql://localhost:1701

Common   Default Schema Filter   Default Stored Procedure Filter   Default Table Filter   HSQLDB Profile Properties   Version   Database:   Default   Database location:   hsql:   User name:   sa   Password:   Save Password		
Common   Default Schema Filter   Default Stored Procedure Filter   Default Table Filter   HSQLDB Profile Properties   Version   Database:   Default Stored Procedure Filter   General Optional   Database location:   hsql:   Save Password:		Properties for
Default Schema Filter   Default Stored Procedure Filter   Default Table Filter   HSQLDB Profile Properties   Version   Database:   Default Stored Procedure Filter   General Optional   Database location:   hsql:   User name:   sa   Password:   Save Password	type filter text 🔏	HSQLDB Profile Properties
Default Stored Procedure Filter Default Table Filter HSQLDB Profile Properties Version Database location: hsql: User name: sa Password:	Common	
Default Table Filter   HSQLDB Profile Properties   Version   Database:   Default Table Filter   General Optional   Database:   Default Table Filter   General Optional   Database:   Default Table Filter   General Optional   Database:   Default Table Filter   Save Password:	Default Schema Filter	Drivers: Hypersonic DB
HSQLDB Profile Properties   Version   Database:   Defa   Database location:   hsql:   User name:   sa   Password:   Save Password		Properties
Version Database: Defa   Database location: hsql:   User name: sa   Password: Image: Ima		General Optional
User name: sa Password: Save Password		Database: Default
Password:		Database location: hsql://l
Save Password		User name: sa
		Password:
URL:     jdbc:		Save Password
		URL: jdbc:hs
?		
?		
?		
	3	

Figure 3.11. JDBC Connection Properties

Click the **Test Connection** button. At this point it probably won't work. This happens if the HSQL JDBC driver is not exactly the same. This can be solved by modifying the HSQLDB database driver settings. To modify the settings, click the **Edit Driver Definition Driver** button.

<b>9</b>	Properties for
type filter text 🔏	<b>HSQLDB</b> Profile Properties
Common	
Default Schema Filter	Drivers: Hypersonic DB
Default Stored Procedure Filter	Properties
Default Table Filter HSQLDB Profile Properties	General Optional
Version	Database: Default
	Database location: hsql://lo
	User name: sa
	Password:
	Save Password
	URL: jdbc:hsq

# Figure 3.12. Driver Details

The proper Driver JAR File should be listed under Jar List. Select the hsqldb.jar file found in the jbdevstudio/jboss-eap/jboss-as/common/lib/ directory and click the **OK** button.

ĺ	<b>9</b>			Edit Driver Definition
	Provide Dri	ver Detai	ls	
				o provide a unique name, a list of requir operty values.
	Name/Type	JAR List	Properties	
	Driver <u>fi</u> les:			
	/redhat/ap	os/jbdevst	udio4/jboss-	eap/jboss-as/common/lib/hsqldb.jar
	0			Canad
	$\bullet$			Cancel
11				

# Figure 3.13. Driver Details

Now, the **Test Connection** should succeed. After testing the connection, click the **OK** button.

		Success	
	Ping succeeded!		
8			

### Figure 3.14. Connection Testing

You can leave the Code Generation section as is. It refers to Java packages in which the generated code will be placed.

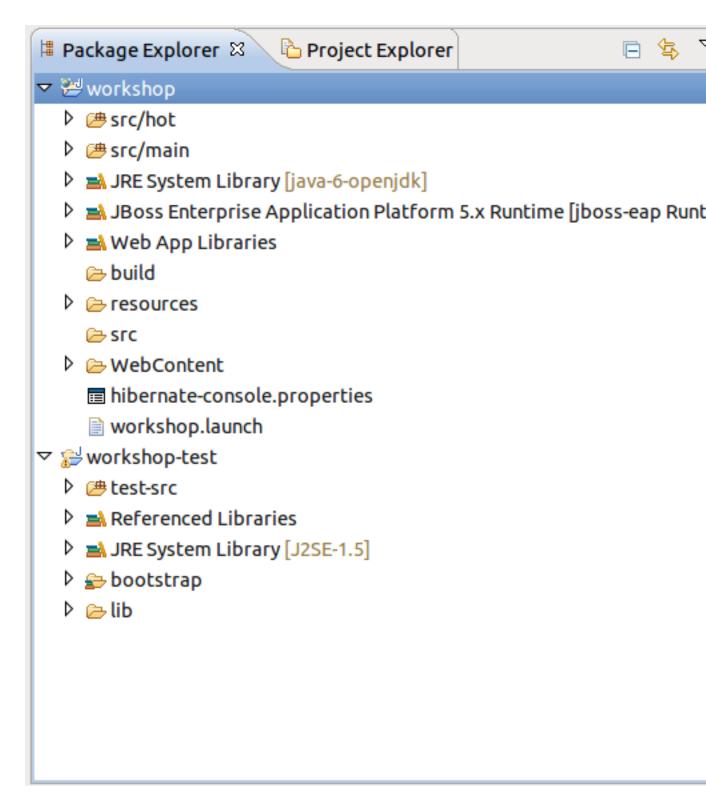
Code Generation	
Session Bean Package Name:	org.domain.workshop.session
Entity Bean Package Name:	org.domain.workshop.entity
Create Test Project:	
Test project name:	workshop-test
Test Package Name:	org.domain.workshop.test

### Figure 3.15. Code Generation Setting

Ш



Click on **Finish** button. Now, there should be a new Seam project called "workshop" listed in the Package Explorer view.





# 3.1.3. Start JBoss Application Server

The complete information on how to manage JBoss AS from JBoss Developer Studio can be found in the *corresponding chapter*.

Now you just need to start the server by clicking on the Start the server icon (

) in the Servers view.

Then run the project by selecting the project then selecting **Run As...**  $\rightarrow$  **Run on Server**.

🗏 Package Ex	kplorer 🛛 🕒 Project Explor	rer	
<ul> <li>✓ Worksho</li> <li>▷ # src/h</li> <li>▷ # src/n</li> <li>▷ A JRE S</li> <li>▷ A JBOS</li> <li>▷ A Web</li> <li>▷ A Web</li> <li>▷ Duild</li> <li>▷ A Web</li> <li>▷ Duild</li> <li>▷ A Web</li> <li>▷ Build</li> <li>▷ A Web</li> <li>▷ A</li></ul>	New         Go Into         Open in New Window         Open Type Hierarchy         Show In         Copy         Copy Qualified Name         Paste         Delete         Build Path         Source         Refactor         Import         Export	F4 Shift+Alt+W > Ctrl+C Ctrl+V Delete Shift+Alt+S > Shift+Alt+T > F5	oss-eap Runt
	<u>R</u> un As	$\rightarrow$	<u>1</u> Run on
	<u>D</u> ebug As	>	<u>2</u> Java Ap
	<u>P</u> rofile As	>	<u>3</u> Java Ap
<	Mark as Deployable		<u>4</u> TestNG
Properties			Ru <u>n</u> Conf
- · · ·	T <u>e</u> am	>	

Figure 3.17. "workshop" Run As

Select the server you want to run the project on, and click the **Finish** button.

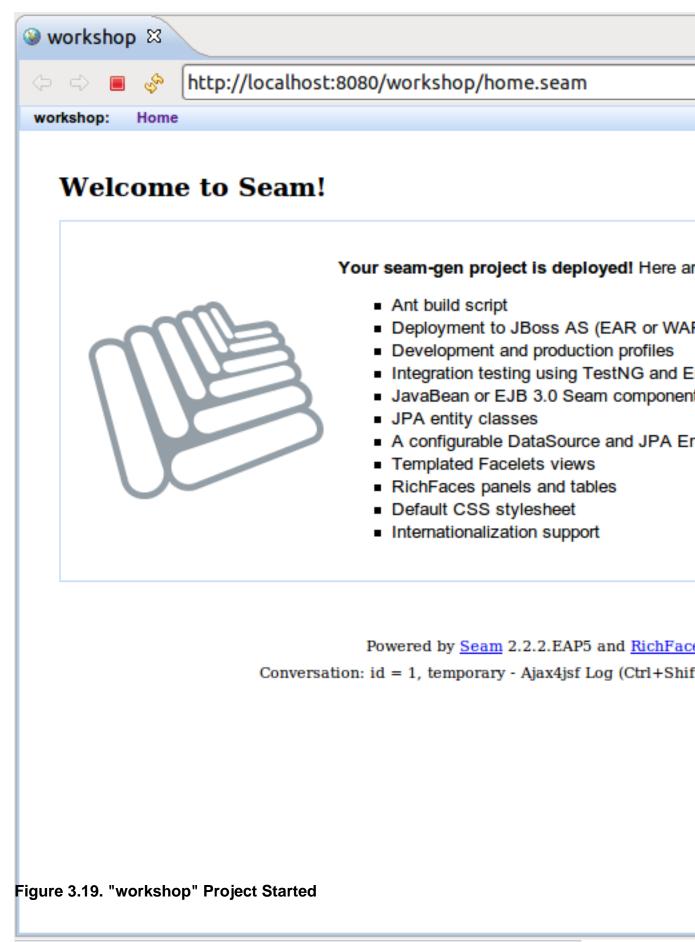
🛞 Run On Server	
Run On Server	
Select which server to use	
How do you want to select the server?	
Choose an existing server	
Manually define a new server	
Select the server that you want to use:	
type filter text	
🗢 🗁 localhost	
🕻 jboss-eap 🖡 Star	ted
JBoss Enterprise Application Platform (EAP) 5.x	
Always use this server when running this project	
?	Cancel
Figure 3.18. "workshop" Run On Server <u>N</u> ext >	



### Note:

If the project does not show up, then you can use a normal browser and use *http://localhost:8080/workshop/home.seam* as the URL.

Your project looks like this:

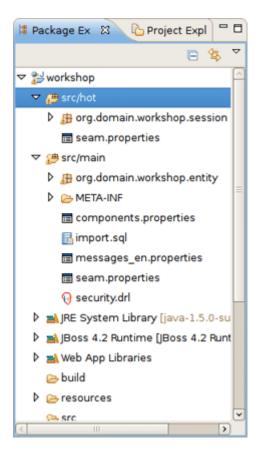


# 3.1.4. Workshop Project Code Overview

Now let's examine the project and its structure. Go back to the Package Explorer view in JBoss Developer Studio.

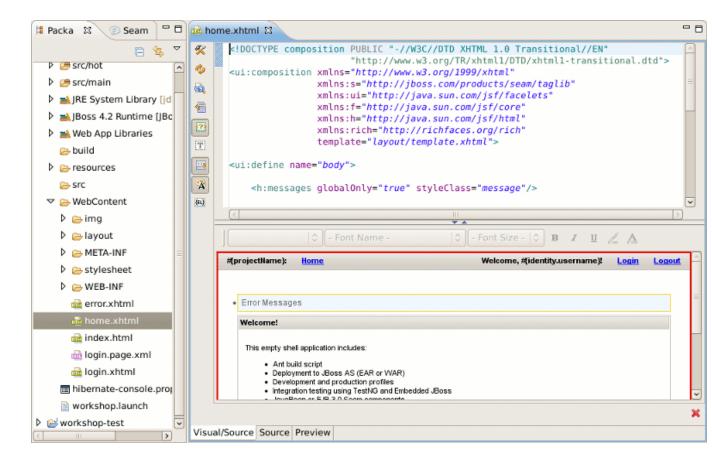
It seems like it's not much for a project but this shell application contains a login screen with default login logic, a menu template that can be further modified, and other layout templates.

It's important to note that the business logic will reside in the src/hot folder, by default. And, the package naming conventions that were used in New Seam project wizard could have been changed to something different from *org.domain.workshop.session*. Also, notice that there is a default Authenticator.java file. This is where custom security logic can be added. Seam has a nice declarative security model that we will explore in more detail later on. The src/main folder is a model directory. It stores the project's JPA entity beans.



### Figure 3.20. Project Structure

The view tier of the application is also important. Seam uses facelets and there is a built-in facelets GUI editor that includes nice WYSIWYG and component drag/drop functionality. Try this out by opening home.xhtml from WebContent folder.



### Figure 3.21. Facelets GUI Editor

Notice that the templates reside in the WebContent/layout folder. There is a stylesheet in the WebContent/stylesheet folder. There is also a login and default error page. The Facelet editor will be explored in more detail later in the lab.

The project already has a datasource that was created via the Seam project wizard database settings. All of the Seam specific configuration files and JAR dependencies are included and located in their proper locations. On last noteworthy line item is related to the build script. There isn't a build script because the Eclipse WTP (Web Tools Project) plugin is used to publish web application changes. As you can see, JBoss Developer Studio is removing a great deal of complexity from the enterprise Java project setup and deployment process. The end result is the developer is writing code, not spending time trying to figure out how to get a decent development environment and project build process.

# 3.2. Seam Action Development

Now it's time to write some code. The good news is that JBoss Developer Studio can also help out in this respect. In this section, we will create a new Seam Action POJO and facelet with some custom business logic and some GUI changes.

## 3.2.1. Create a New Seam Action

Go to main menu bar and click on File  $\rightarrow$  New  $\rightarrow$  New Seam Action to start the New Seam Action wizard.

Specify a Seam component name (e.g., "myAction"). The other properties will be auto-completed for you so there is no need to change them. Click on the **Finish** button.

New Seam Action
orkshop Browse
yAction
yAction
yActionBean
yAction
yAction
g.domain.workshop.session
Cancel

### Figure 3.22. New Seam Action Wizard

Now, open the MyAction. java file and replace the "myAction" method with this logic:

```
public void myAction() {
Calendar cal = Calendar.getInstance();
log.info("myAction.myAction() action called");
statusMessages.add("MyAction Executed on:" + cal.getTime());
}
```

You also need to import the java.util.Calendar class by clicking CTRL+Shift+O.

### 3.2.2. Test Seam Action

The new action can be tested by browsing the workshop-test project. JBoss Developer Studio has already created a TestNG test case for you.

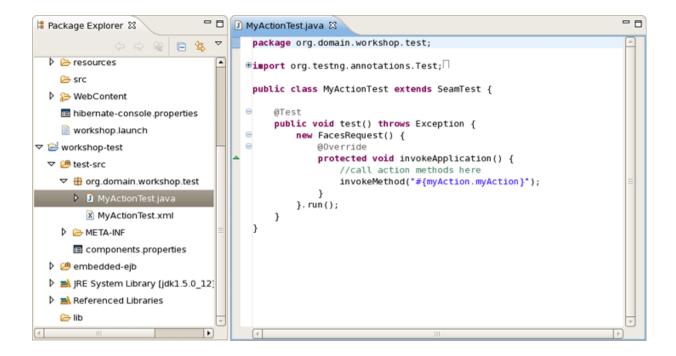


Figure 3.23. "workshop-test" Project



The test case simulates a Seam method execution for the MyAction.myAction() logic.

To run the test case, right click on MyActionTest.xml and click Run As  $\rightarrow$  TestNG Suite or use the Run As... toolbar shortcut as shown below.

□ •     □     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓     ↓	28 ❸ @▼ ] .4 ] ❷ ] 8 ] ? ▼   ▶ 兆 ● 令 兆 ] 21 ~ 31
Backage Explorer S Run As	I Run on Server Shift+Alt+X R
Copen Ru <u>n</u> Dialo     Organize Fa <u>v</u> ori	
<ul> <li>✓</li></ul>	<pre>public class MyActionTest extends SeamTest {     @Test     public void test() throws Exception {         new FacesRequest() {            @Override            @Override            @Override            @Override            @Override            ``````````````````````````</pre>
MyActionTest.xml     META-INF     Components.properties     Ø @ embedded-ejb	<pre>protected void invokeApplication() {     //call action methods here     invokeMethod("#{MyAction.myAction}");     } }.run();</pre>
<ul> <li>B JRE System Library [jdk1.5.0_12]</li> <li>B Referenced Libraries</li> <li>lib</li> </ul>	}

### Figure 3.24. TestNG Running

With any luck, the test case will pass. Look at the TestNG view.

Problems 🖉 Tasks 🖗 Seam Components 🍀 Servers	; 💷 Console 👫 R
	Tests: 1/1 Me
Search:	
📑 All Tests 📑 Failed Tests Summary	
✓ In Default suite ( 1/0/0/0 ) (0.205 s)	
🗢 🔚 Default test ( 0.205 s)	
🗢 🔚 org.domain.workshop.test.MyActionTest	
撞 test_myAction (0.205 s)	

# Figure 3.25. TestNG Results

Now, it's safe to test the new Seam Action in a web browser. The fastest way to do that is to right click on myAction.xhtml and use **Run As...**  $\rightarrow$  **Run On Server** which will show the appropriate URL in the browser. Alternatively you can manually enter *http://localhost:8080/workshop/myAction.seam* into a browser.

workshop	📠 home.xhtml	MyAction.java	Sector workshop Sector Sec
🗢 🗢 🔳 🖑	http://localhost:8	8080/workshop/myAc	tion.seam
workshop: Home	•		
MyAction 1	Executed on:Fri Jan	14 09:25:35 EST 2011	
myAction			
myAction!			
	Conversati		2.EAP5 and <u>RichFaces</u> . Gener ax4jsf Log (Ctrl+Shift+D) - <u>De</u>

#### Figure 3.26. Seam Action in a Web Browser

### 3.2.3. Modify Seam Action User Interface

Browse to *http://localhost:8080/workshop/myAction.seam* and click on the **myAction** button. This executes the "myAction" method. This looks pretty good, but we could make this page look a little better.

Open WebContent/myAction.xhtml in JBoss Developer Studio to use the nice facelets editor.

📾 m	yAction.xhtml 🖾	- 8
	<pre><!DOCTYPE composition PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//</td>    <td></td></pre>	
A		
(er)	<h:messages globalonly="true" styleclass="message"></h:messages>	
	0 - Font Name - 0 - Font Size - 0 B Z	33
	#projectName}: Home Welcome, #identity.username}! Login Logout	
	Error Messages      myAction      myAction!      Powered by Seam, Generated by seam-gen.	
Visu	al/Source Source Preview	~
Visu	myAction         myAction!         Powered by Seam, Generated by seam-gen.	×

# Figure 3.27. Open Seam Action with Editor

Right click on the "myAction!" button in the visual part of editor and select <h:commandButton> Attributes.

	ି - Font Na	ime -	🗘 - Font Size - 🗘 B Z 🖳 🦽
Jecmann	ex nome omesting	Customers List	CINEROVECS LIST PLOUDER LIST OF DET DELINE LIST OF DETS LIST
Error	Messages		
yAction	i		
9-0			
C myAd	ShicommandButton> Al	ttributes	<b></b>
	Parent Tag (h:form)		
¢	Functic rug (inform)		
	Insert around	2	commandButton
ual/Sour	Insert before	2	>
	Insert after	2	
	Insert into	2	>
	Replace with	;	>
	Strip Tag		
	Zoom	;	
	Cut	Ctrl+X	
	<u>C</u> opy	Ctrl+C	
	Paste	Ctrl+V	1

### Figure 3.28. Seam Action Editing

Change the value of the button to something different. If desired, you can change any other text on the page. Then, type **CTRL+S** to save the facelet.

3	Attributes	×
<h:commandb< th=""><th>utton&gt;</th><th></th></h:commandb<>	utton>	
name	value	<b>^</b>
rendered		
style		
styleClass		
tabindex		
title		
type		=
value	Click on My Action Button	🔻
		Close

### Figure 3.29. Attributes Dialog

Refresh http://localhost:8080/workshop/myAction.seam and now you should see your changes.

Notice that you did not have to publish the application. JBoss Developer Studio auto-published it for you.

🚾 myAction.xhtml 🕴		- 0
#(projectName): Home	Welcome, #{identity.username} Login	Logout
Error Messages myAction Click on My Action Button		
	Powered by <u>Seam</u> . Generated by seam-gen.	
Visual/Source Preview		

#### Figure 3.30. Seam Action Is Modified

# 3.3. Declarative Security

In this section you will see how easy it is to secure the facelets and facelet components in Seam. Let's go ahead and secure the action button, then we will secure the entire page.

### 3.3.1. Edit Login Authentication Logic

There is a class called Authenticator.java. The login page will execute the Authenticator.authenticate() method by default, so we'll start by viewing the authentication logic.

Open Authenticator.java in JBoss Developer Studio and you will see that it contains the authenticate() method with this code:

```
public boolean authenticate()
{
    log.info("authenticating {0}", credentials.getUsername());
    //write your authentication logic here,
    //return true if the authentication was
    //successful, false otherwise
    if ("admin".equals(credentials.getUsername()))
    {
        identity.addRole("admin");
        return true;
    }
    return false;
}
```

# 3.3.2. Secure Seam Page Component

Open myAction.xhtml and add a new secured command button:

```
<h:commandButton id="myActionSecured"
value="Secured Action Button"
action="#{myAction.myAction}"
rendered="#{s:hasRole('admin')}"/>
```

Refresh *http://localhost:8080/workshop/myAction.seam* If you are not logged in you will only see one button. If you are logged in, there will be two buttons.

🥹 workshop 🛿
🗇 🔿 🔳 🧇 http://localhost:8080/workshop/myAction.seam
workshop: Home
myAction
Click on My Action Button
Powered by <u>Seam</u> 2.2.2.EAP5 and <u>RichFac</u>
Conversation: id = 8, temporary - Ajax4jsf Log (Ctrl+Shi

### Figure 3.31. One Button on a Page

The secured button is not visible because the user isn't logged in as "admin".

Service Servic
🗢 🗢 🔳 🤣 http://localhost:8080/workshop/myAction.seam
workshop: Home
myAction
Click on My Action Button Secured Action Button
Powered by <u>Seam</u> 2.2.2.EAP5 and <u>RichFac</u> Conversation: id = 11, temporary - Ajax4jsf Log (Ctrl+Shi

### Figure 3.32. Secured Button is Visible

The user is logged in as "admin". Securing components is easy but securing pages is pretty simple as well.

Open *WebContent/WEB-INF/pages.xml*. Then add this markup directly underneath the <pages> element:

```
<page view-id="/myAction.xhtml" login-required="true"/>
```

Refresh *http://localhost:8080/workshop/myAction.seam* If you are not logged in you will get bounced back to the login page.

workshop <sup>™</sup>
🗢 🗢 🔳 🦑 http://localhost:8080/workshop/login.seam
workshop: Home
Login
Please login here
Username
Password
Remember me
Note - You may login with the username 'admin' and a blank password.
Login
Powered by Seam 2.2.2.EAP5 and RichFac
Conversation: id = 13, temporary - Ajax4jsf Log (Ctrl+Sh

### Figure 3.33. Login Page

Thus, if you enter login credentials for the "admin" user, you will be re-directed to the secured page and secured component. If you enter different login credentials, page access will be granted, but the secured component will not be displayed.

Congratulations! You have secured your new action both at the facelet component and page level. You also added custom authentication logic to the login action.

# 3.4. Browsing Workshop Database

In this section you get to know how to use the workshop database that was started at the beginning of the lab.

# 3.4.1. Database Connectivity Setup

The workshop data can be browsed inside of JBoss Developer Studio.

To open the Data Source Explorer, click on Window  $\rightarrow$  Open Perspective  $\rightarrow$  Other  $\rightarrow$  Database Development.

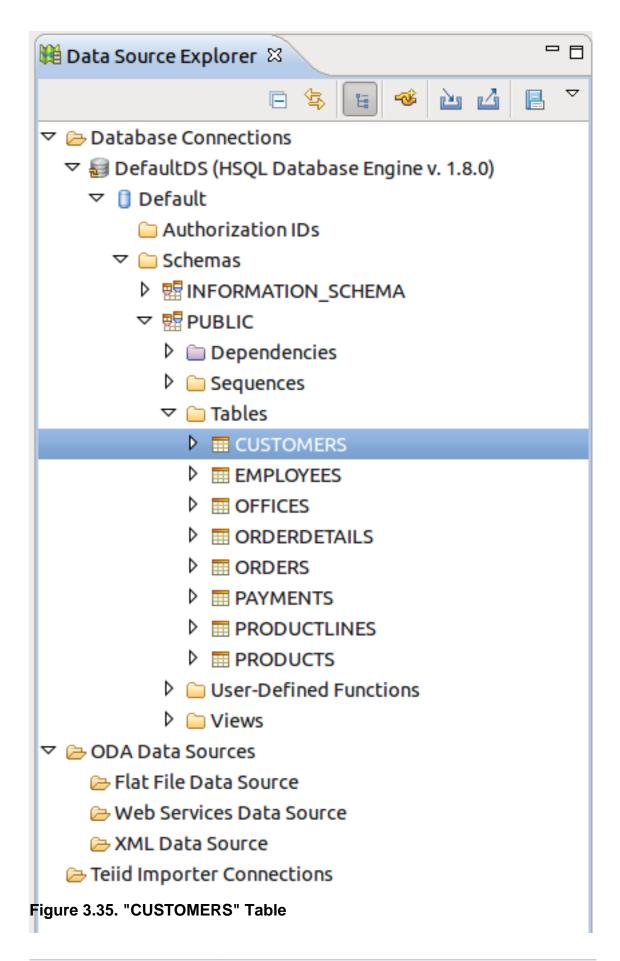
In the Data Source Explorer, expand the Databases node and select the Default database. Right click on it, select **Connect** from the context menu.

🛍 Data Source Explorer	X	- 8						
	🖻 🔄 🖬 📽 🔛 🗳	4 🖪 🍸						
🗢 🗁 Database Connections								
🗒 DefaultDS								
🗢 🗁 ODA Data Source:	<u>C</u> onnect							
🗁 Flat File Data S	<u>D</u> isconnect							
B Web Services D	<u>P</u> ing							
🗁 XML Data Sour	Work Offline							
🗁 Teiid Importer Co	<u>S</u> ave Offline							
	Re <u>n</u> ame	F2						
	D <u>e</u> lete	Delete						
	Duplicate							
	Ref <u>r</u> esh	F5						
	Open SQL Scrapboo <u>k</u>							
	<u>P</u> roperties	Alt+Enter						

### Figure 3.34. Data Source Explorer

### 3.4.2. Browse Workshop Database

Then in the current view, drill down to the CUSTOMERS table.



Right click on CUSTOMERS, select **Data**  $\rightarrow$  **Sample Contents** to view the data in the table.

There should be a SQL Results view on the workbench, but it could be hidden. Click on the "Result1" tab in the right side and you should see the data in the CUSTOMERS table.

🔲 SQL Resu	its 🛛 📮	Console	)				= 3	: 🖗 (		≱ ▽	- 6
Type query expression here		Status Result1									
Status	Operation	Date		CUSTOMER	CUSTOMERNAME	CONTACTLAST	CONTACT	PHONE	ADDRE	SSLINE1	A
✓ Succe		12/30/0	1	237	ANG Resellers	Camino	Alejandra	(91)74	Gran V	\u005cu	
			2	242	Alpha Cognac	Roulet	Annette	61.77.6	1 rue A	lsace-Lo	
			3	206	Asian Shopping N	Walker	Brydey	+612 9	Suntec	Tower T	8
			4	103	Atelier graphique	Schmitt	Carine	40.32.2	54, rue	Royale	
			5	144	Volvo Model Repli	Berglund	Christina	0921-1	Berguv	sv\u005¢	
			6	189	Clover Collections	Cassidy	Dean	+353 1	25 Maio	ien Lane	F
			7	141	Euro+ Shopping C	Freyre	Diego	(91) 55	C/ Mora	alzarzal,	
			8	216	Enaco Distributors	Saavedra	Eduardo	(93) 20	Rambla	a de Cata	-
			9	201	UK Collectables, L	Devon	Elizabeth	(171) 5	12, Ber	keley Ga	
			10	148	Dragon Souvenie	Natividad	Eric	+65 22	Bronz S	Sok.	в
			11	209	Mini Caravy	Citeaux	Fr\u005cu(	88.60.1	24, pla	ce Kl\u00	
			12	240	giftsbymail.co.uk	Bennett	Helen	(198) 5	Garden	House	c
			13	223	Nat\u005cu005cu	Kloss	Horst	0372-5	Tauche	rstra\u00	
			14	169	Porto Imports Co.	de Castro	Isabel	(1) 356	Estrada	a da sa\u	
			15	119	La Rochelle Gifts	Labrune	Janine	40.67.8	67, rue	des Cin	(
			16	112	Signal Gift Stores	King	Jean	702555	8489 S	trong St.	
Total 50 records shown											

Figure 3.36. SQL Results View

# i

### Note:

If you can't find the SQL Results view tab, click on Window  $\rightarrow$  Show View  $\rightarrow$  Other  $\rightarrow$  SQL Development  $\rightarrow$  SQL Results.

Congratulations! You just connected to the workshop database and queried the content using Database Explorer tools.

# 3.5. Database Programming

Now, it's time to reverse engineer the workshop database into a fully functioning Seam CRUD (Create Read Update Delete) application.

# 3.5.1. Reverse Engineer CRUD from a Running Database

In JBoss Developer Studio, switch to the Seam perspective, and then right-click the project and select New  $\rightarrow$  Seam Generate Entities.

The "workshop" project in the Seam Generate Entities wizard will be selected automatically. There is no need to change something more, click the **Next** button to proceed to the next step.

Gen	erate Seam Entities	
Generate Seam Entities		
Seam Project: Hibernate Console Configuration: Generation Mode O Use existing entities	workshop workshop	Browse
<ul> <li>Reverse engineer from data</li> <li>Use existing reveng</li> <li>reveng.xml:</li> </ul>	base	Br
? < <u>B</u> ack	<u>N</u> ext > Canc	el

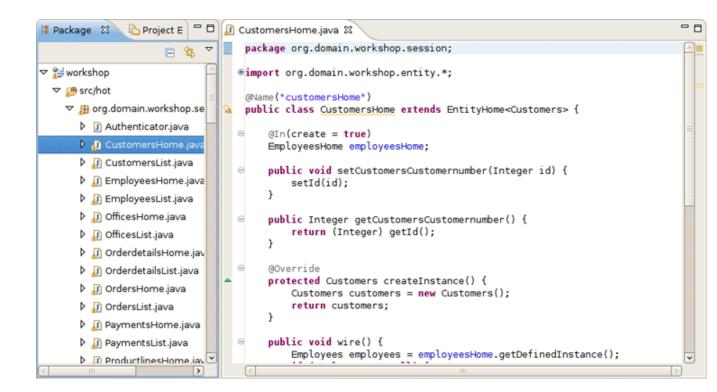
### Figure 3.37. Generate Seam Entities

On the next page use the **Refresh** button to display the database, then click the **Include** button to include all the tables from the database, and finally click the **Finish** button.

Sector 2010	Generate S	Seam Entities
Generate Seam Entities		
Select Tables		
Database schema: Customers Dublic Customers Employees Offices Orderdetails Orders Payments Productlines Products	Include Exclude Up Down Remove Remove All	Table filters:         !       Catalog         .*       .*         .*       .*         .*       .*         .*       .*         .*       .*         .*       .*         .*       .*         .*       .*         .*       .*         .*       .*         .*       .*         .*       .*
Refresh		
?	< <u>B</u> a	ack <u>N</u> ext >

# Figure 3.38. Selecting Tables

After running the Generate Entities action, you will see new *org.domain.workshop.entity* classes. These classes represent insert/update/delete/query logic.



#### Figure 3.39. org.domain.workshop.entity Classes

There is also the *org.domain.workshop.entity* package that contains the JPA classes. These are the entity beans that are mapped to database tables. Note that you can use Seam refactoring tools with Seam components. Read more about it in *Seam refactoring tools chapter* [http://download.jboss.org/jbosstools/nightly-docs/en/seam/html\_single/ index.html#seam\_refactoring] of Seam Dev Tools Reference Guide.

Last, but not least, there are facelets for all of the CRUD screens. The best way to get a feel for the generated code is to open a browser and play around with the application. Go to *http://localhost:8080/workshop* and insert/update/delete/query a few records. There is quite a bit of AJAX in this application, but we will explore that later on in the lab. For now, take note of the page tabs, required field logic and data table sorting in the list pages.



Tip

If you see the error java.lang.ClassNotFoundException: org.jboss.seam.servlet.SeamListener in the console output from the Application Server, you may need to copy the jboss-seam.jar file from the lib subdirectory in the Seam library (which can be downloaded from here [http://seamframework.org/Seam2/Seam2DistributionDownloads]) into the /server/default/deploy/workshop.war/WEB-INF/lib/ subdirectory in your Application Server (where "default" refers to the server profile that you are using).



# Tip

If you see the error Could not instantiate Seam component: org.jboss.seam.security.ruleBasedPermissionResolver, COPY the mvel2.jar file from the Seam library to the same destination directory mentioned in the tip above.

{projectName}:	Home	Offices List	Customers List	Employees List	Products List	Order	del
Error Messages	з						
Customers sear	ch param	eters				ĸ	
customernar	ne	#{customersLis	st.customer				
contactlastna	ime	#{customersLis	st.customer				
contactfirstn	ame	#{customersLis	st.customer				
phone		#{customersLis	st.customer				
addressline1		#{customersLis	st.customer				
addressline2		#{customersLis	st.customer				
city		#{customersLis	st.customer				
state		#{customersLis	st.customer				
postalcode		#{customersLis	st.customer				
country		#{customersLis	st.customer				
Search							
Customers sear	ch result	5					
No customers e	xists						

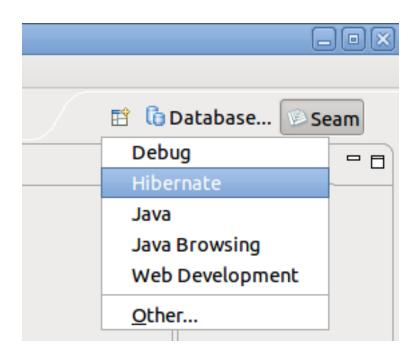
# Figure 3.40. CustomersList.xhtml in the Editor

Congratulations! You now have a fully functioning CRUD application that is AJAX enabled.

# 3.5.2. Use Hibernate Tools to Query Data via JPA

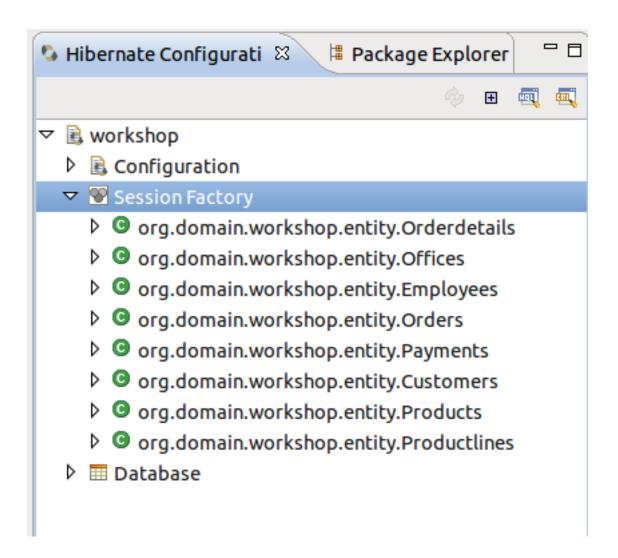
Now, it's time to write some JPA queries using the Hibernate perspective in JBoss Developer Studio.

In the upper right corner of the workbench there is a small icon (see the figure below), click on it and select **Hibernate**.



## Figure 3.41. Hibernate Perspective

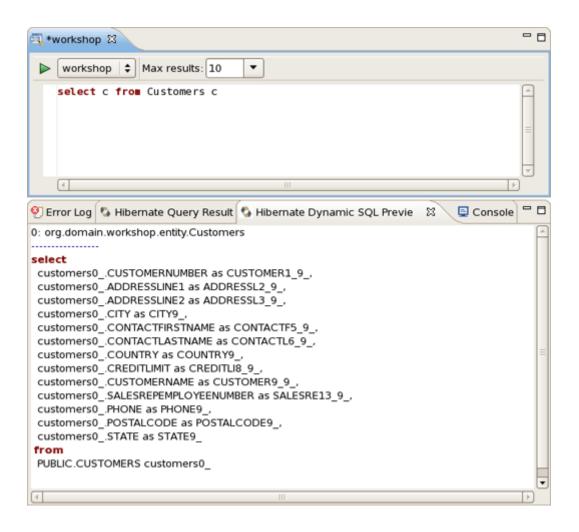
Look at the Hibernate Configurations view. In the "workshop" project, drill down on the Session Factory and notice that the JPA entities/attributes are listed in a nice tree view.



## Figure 3.42. Hibernate Configurations View

Right click on the Session Factory and select **HQL Editor**. This will open a JPA query scratch pad window.

Write your query and click on the "Hibernate Dynamic SQL Preview" tab. You should see the SQL that will be executed if this JPA query is run.



#### Figure 3.43. JPA Query Editor

Run the query by clicking on the green run icon.

The results are listed in the "Hibernate Query Result" view. There is a "Properties" tab in the workbench that can be used to see a specific JPA result. These results represent the JPA objects because our query did not specify column names.

	Properties 🛿	🖬 🐉 🗔 🔻 🗖	💐 *workshop 🕴	- 0
Pro	operty	Value	workshop 🗘 Max results: 10 💌	
▽	Identifier		select c from Customers c	<b>A</b>
	customernumbe	103		
▽	Properties			=
	addressline1	54, rue Royale		~
	addressline2			
	city	Nantes	🤨 Error Log 😘 Hibernate Query Result 🕱 💊 Hibernate Dynamic SQL Pr 📮 Console	
	contactfirstnam€	Carine		× %
	contactlastname	Schmitt		na 136
	country	France	0	Ê
	creditlimit	21000.0	org.domain.workshop.entity.Customers@c9fc25	
	customername	Atelier graphique	org.domain.workshop.entity.Customers@841423	
	employees		org.domain.workshop.entity.Customers@1463027	
	orderses		org.domain.workshop.entity.Customers@1038048	
	Þ paymentses		org.domain.workshop.entity.Customers@db0d07	
	phone	40.32.2555	org.domain.workshop.entity.Customers@134817a	
	postalcode	44000	org.domain.workshop.entity.Customers@207655	
	state		org.domain.workshop.entity.Customers@138d8fa	
			org.domain.workshop.entity.Customers@12d2db8	
			org.domain.workshop.entity.Customers@11499ff	•
4	111		select c from Custom select c from Custom select c from Custom 🕱 39	

# Figure 3.44. Hibernate Query Result View

The query can be refined, and take note that there is nice code completion in the JPA query editor.

्र *workshop छ	
<pre>workshop \$ Max results: 10 * select c.customername, c.employees. from Customers c</pre>	
Second Concernance, Creanployees <sup>(1)</sup> Customerses - Employees <sup>(1)</sup> Customerses <sup>(1)</sup> C	
<ul> <li>Contension - Employees</li> <li>firstname - Employees</li> <li>jobtitle - Employees</li> <li>jobtitle - Employees</li> <li>lastname - Employees</li> <li>officecode - Employees</li> <li>officecode - Employees</li> <li>reportsto - Employees</li> <li>reportsto - Employees</li> </ul>	
org.domain.workshop.entity.Customers@841423 org.domain.workshop.entity.Customers@1463027 org.domain.workshop.entity.Customers@1038048 org.domain.workshop.entity.Customers@db0d07 select c from Custom select c from Custom % Close	•
Writable Insert 1:36	

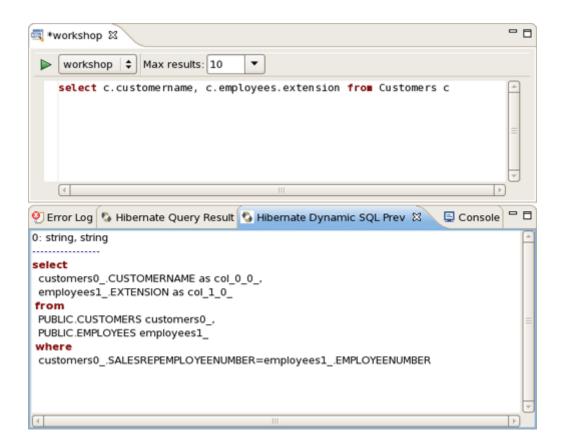
# Figure 3.45. Code Completion

A refined query will return results that are more ResultSet oriented. Notice the join logic that JPA supports.

*workshop &	
<pre>workshop \$ Max r select c.customern</pre>	ame, c.employees.extension from Customers c
4	
Error Log 😘 Hibernate G	Query Result 🛛 🔇 Hibernate Dynamic SQL Prev 🖳 Console 🖵
/	
0	1
Atelier graphique	x2028
Signal Gift Stores	x4065
Australian Collectors, Co.	×101
La Rochelle Gifts	x2028
Baane Mini Imports	x102
Mini Gifts Distributors Ltd.	x3291
Blauer See Auto, Co.	x102
Mini Wheels Co.	x3291
Land of Toys Inc.	x4102
Euro+ Shopping Channel	x2028

## Figure 3.46. The Hibernate Query Result

There was no need to specify an Employees table in the from part of the JPA query because JPA supports reference traversal via Java class attribute references. Not only are JPA and HQL queries fully supported, but Criteria based queries can also be written in the Criteria Editor. You should spend some time tinkering with different queries and possibly Criteria based queries, even though the instructions are not provided in this lab.

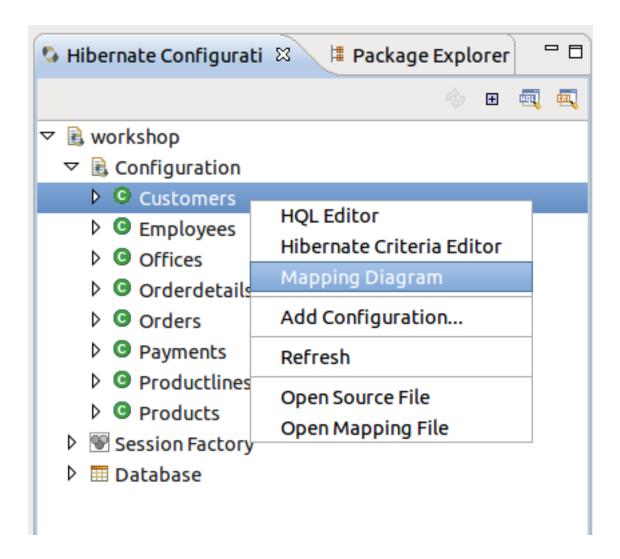


#### Figure 3.47. Criteria Editor

# 3.5.3. Use Hibernate Tools to visualize the Data Model

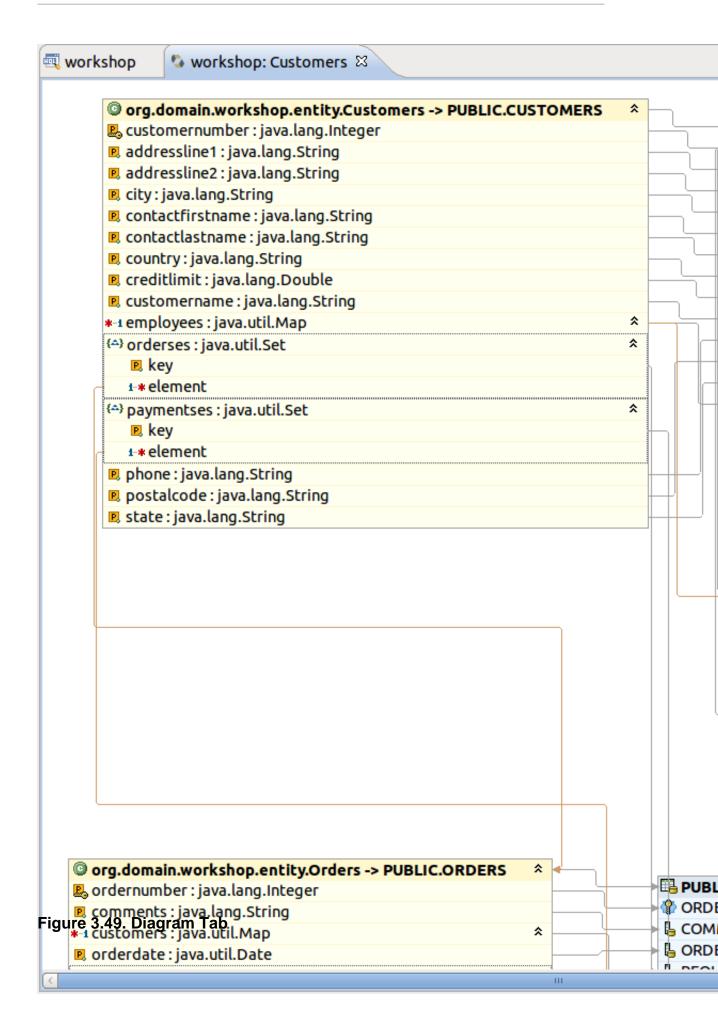
Now, it's time to view the data model for the workshop database.

In the Hibernate Configurations view, select "workshop" project and expand the Configuration node. Select the Customers entity, right click on it, choose **Mapping Diagram**.



#### Figure 3.48. Mapping Diagram Opening

You see a Diagram tab for the CUSTOMERS table and any tables that have FK references. This is a handy way to view the data model and JPA mappings. Now, you've got access to something that the Erwin Data Modeler can't do.



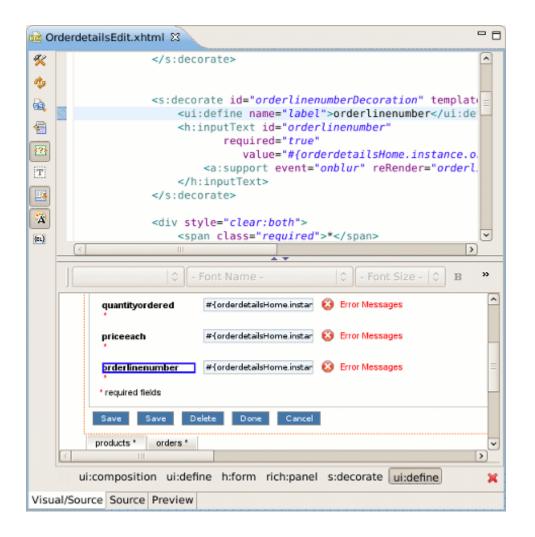
# **3.6. Rich Components**

This lab will conclude with one last AJAX twist. In this section we add a RichFaces inputNumberSlider to the Order Details edit screen.

### 3.6.1. Add a Richfaces component to the CRUD Application

Switch to Seam perspective, and open WebContent/OrderdetailsEdit.xhtml in JBoss Developer Studio.

Change the form field values using the visual editor. Seam has generated the form field names that match the database column names. This is not ideal for business users.



#### Figure 3.50. Form Fields Editing

Also, replace the QTY Ordered input field with a inputNumberSlider. You can use the JBoss Developer Studio palette or right click on the form and insert the RichFaces component.

][ ](	- Font Name -	C - Fon	t Size - 🛛 🗘	] 1	Β <u>Ι</u> <u>υ</u> ,	l.	<rich:dndparam> <rich:dragindicator></rich:dragindicator></rich:dndparam>	
· · · ·	orderdecanariorreanacan	Error Mes					<rich:draglistener> <rich:dragsupport> <rich:dropdownmenu></rich:dropdownmenu></rich:dragsupport></rich:draglistener>	
	orderdecanariorre.inatari	<ul> <li>Error Mes</li> <li>Error Mes</li> </ul>	-				<rich:droplistener> <rich:dropsupport> <rich:editor></rich:editor></rich:dropsupport></rich:droplistener>	
Orderlinenumber #{	Parent Tag (rich:panel)	B From Mess	sages				<rich:effect> <rich:extendeddatatable></rich:extendeddatatable></rich:effect>	
Save Save Delete	Insert around Insert before	>		_			<rich:fileupload> <rich:gmap></rich:gmap></rich:fileupload>	
Products * Orders * There is no products associa Productcode	Insert after Insert into Replace with	>	JBoss JSF name	>	Ajax4Jsf RichFaces Seam	> > >	<rich:graphvalidator> <rich:hotkey> <rich:inplaceinput></rich:inplaceinput></rich:hotkey></rich:graphvalidator>	lescription
#(_products.productcode) #{_	pr Zoom	>	oductname}	#(_	products produc	tscal	<rich:inplaceselect> <rich:inputnumberslider></rich:inputnumberslider></rich:inplaceselect>	oductolescription
✓ ui:composition ui:de sual/Source Source Preview	Cut fii ⊆opy	Ctrl+X Ctrl+C <b>Ctrl+V</b>	ui:define				<ri>crich:inputNumberSpinner&gt; <rich:insert> <rich:jquery> <rich:jsfunction></rich:jsfunction></rich:jquery></rich:insert></ri>	
							<rich:layout> <rich:layoutpanel> <rich:listshuttle></rich:listshuttle></rich:layoutpanel></rich:layout>	

# Figure 3.51. Insert RichFaces Component from Context Menu

One the last option is to use the source view and manually copy the inputNumberSlider markup listed below:

```
<rich:inputNumberSlider id="quantityOrdered" required="true"
value="#{orderdetailsHome.instance.quantityordered}"/>
```

nd *(	OrderdetailsEdit.xhtml 🖾 👘	8
** ** **	<pre><a:support event="onblur" quantityordered"<br="" rerender="orderlinenumbe&lt;br&gt;&lt;/h:inputText&gt;&lt;br&gt;&lt;/s:decorate&gt;&lt;br&gt;&lt;rich:inputNumberSlider id=">required="true" value="#{orderdetailsHome.instance.qua </a:support></pre>	
	<pre></pre>	

# Figure 3.52. Manually copying Source Code

The end result is an edit page that has better form labels and a new RichFaces control.

worksho	op 🛿					
\$ \$ <b>•</b>	🛚 🤣 🧄	p://localhost:	8080/worksh	op/Orderdeta	ailsEdit.seam	า
workshop:	Home E	Browse data				
Add Or	derdetails					
Quan	ntityordered *	0		) -	100	0
Price	each *	0.0		]		
Orde	rlinenumber*	0		]		
* requi	ired fields					
Save	Cancel					
Orders	s* Product	s *				
There	is no orders as:	sociated with this o	rderdetails.			
Sele	ect orders					
				Powered by	Seam 2.2.2.EA	P5 and <u>RichFace</u>
			Conversation			ljsf Log (Ctrl+Shi

# Figure 3.53. The Result Page

Congratulations! You have completed the JBoss Developer Studio lab.

# Developing a simple JSP web application



#### Note:

We highly recommend developing in Seam. This chapter is for users who for some reason cannot use Seam.

In this chapter you'll find out how to create a simple *JSP* [http://java.sun.com/products/jsp/] application using JBoss Developer Studio. The application will show a classic "Hello World!" on the page.

We'll assume that you have already launched JBoss Developer Studio and also that the Web Development perspective is the current perspective. If not, make it active by selecting **Window**  $\rightarrow$  **Open Perspective**  $\rightarrow$  **Web Development** from the menu bar or by selecting **Window**  $\rightarrow$  **Open Perspective**  $\rightarrow$  **Other...** from the menu bar and then selecting Web Development from the Select Perspective dialog box.

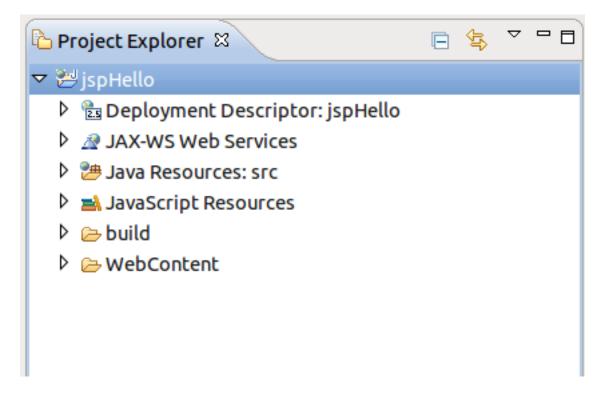
# 4.1. Setting Up the Project

We are going to start by creating a Dynamic Web Project with a minimal structure, i.e. with just required facets. Thus this section will perform you all necessary steps on how to do this.

- Go to the menu bar and select  $\textbf{File} \rightarrow \textbf{New} \rightarrow \textbf{Other...}$
- Select Web  $\rightarrow$  Dynamic Web Project in the New Project dialog box
- Click the Next button
- Enter "jspHello" as a project name
- Then select *Minimal Configuration* from the list of possible configurations and click the **Finish** button.

9	New D	ynamic Web	Project		
Dynamic Web Proje	ct				
Create a standalone [ Enterprise Applicatio		oject or add i	it to a new or	existin	g
Project na <u>m</u> e: jspHe	llo				
Project location					
✓ Use <u>d</u> efault locat	ion				
Location: /redhat/w	vorkspaces/wor	kspace-jbds4	/jspHello		
Target r <u>u</u> ntime					
jboss-eap Runtime				•	Nev
Dynamic web module	e <u>v</u> ersion				
2.5					
<u>C</u> onfiguration					
Minimal Configurat	ion			\$	
The most conservation Additional facets car					
EAR membership					
□ <u>A</u> dd project to an	EAR				
EAR pr <u>oj</u> ect name:	EAR			~	Ne
Working sets					
🔲 Add projec <u>t</u> to w	orking sets				
W <u>o</u> rking sets:				\$	
jure 4.1. Create New Web	Project				
2	< Back	Next >	Can	cel	

The *jspHello* node should appear in the upper-left Package Explorer view.



#### Figure 4.2. New Web Project

# 4.2. Creating JSP Page

This section covers all the points how to create, edit and then preview JSP page.

In our simple application we need to create only one JSP page which displays a "Hello World!" message.

- Right click the <code>WebContent</code> folder and select New  $\rightarrow$  JSP.
- Type hello.jsp for a file name and click the Next button.

In the next window you can choose a template for your JSP page and see its preview.

• Select New JSP File (xhtml) template and click the Finish button.

🕑 Ne	w JSP File
Select JSP Template	
Select a template as initial content in the JSP page.	
secce e compare as maint concent mane son pager	
✓ Use JSP Template	
Templates:	
Name	Description
New JavaServer Faces (JSF) Page (xhtml)	JSP with xhtn
New JavaServer Faces (JSF) Page (xhtml, xml syntax)	JSP with xhtm
New JSP File (html)	JSP with htm
New JSP File (xhtml)	JSP with xhtn
New JSP File (xhtml, xml syntax)	JSP with xhtm
	·
<u>P</u> review:	
xml version="1.0" encoding="\${encoding}" ?	
<pre>&lt;%@ page language="java" contentType="text/html</pre>	l; charset=\${en
<pre>pageEncoding="\${encoding}"%&gt; <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Tr</pre>    </pre>	ransitional//EN
<html xmlns="http://www.w3.org/1999/xhtml"></html>	
<head></head>	tml, charget-t

# Figure 4.3. Create JSP Page

Our hello.jsp page will now appear in the Project Explorer view.

# 4.2.1. Editing a JSP Page

Let's now make a little change so that a JSP page displays "Hello World!" message.

• Insert this line inside the <body> </body> tag:

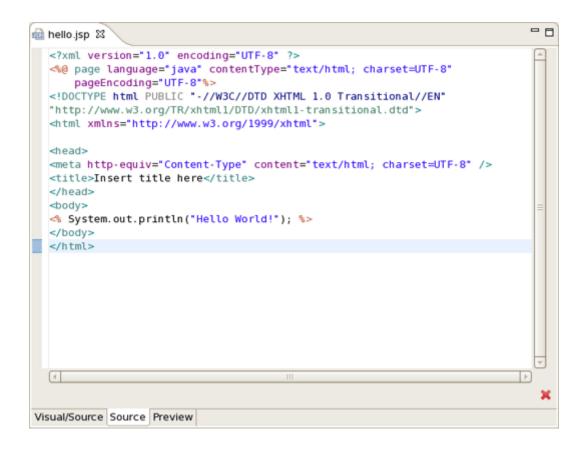
```
<% System.out.println("Hello World!"); %>
```

Notice that content assist functionality is always available when you are typing:

📾 *hello.jsp 🕱	- 8
<pre><?xml version="1.0" encoding="UTF-8" ?> &lt;%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%&gt; <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">     <html "-="" "http:="" 1.0="" dtd="" en"="" public="" tr="" transitional="" w3c="" www.w3.org="" xhtml="" xhtml1="" xhtml1-transitional.dtd"=""> <html "-="" "http:="" 1.0="" dtd="" en"="" public="" tr="" transitional="" w3c="" www.w3.org="" xhtml="" xhtml1="" xhtml1-transitional.dtd"=""> <html xmlns="http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"> <html xmlns="http://www.w3.org/1999/xhtml"> <head> <meta content="text/html; charset=utf-8" http-equiv="Content-Type"/> <title>Insert title here</title> </head> <body></body></html></html></html></html></pre>	
System.out. %>	
• autoFlush : boolean - PrintStream • charOut : OutputStreamWriter - PrintStream • closing : boolean - PrintStream • closing : boolean - PrintStream • formatter : Formatter - PrintStream • out : OutputStream - FilterOutputStream • textOut : BufferedWriter - PrintStream • trouble : boolean - PrintStream • trouble : boolean - PrintStream • checkError() : boolean - PrintStream • clearError() : void - PrintStream • clone() : Object - Object • outputStream • outoflust - Object • outoflust	11
html body jsp:scriptlet	×
Visual/Source Source Preview	

#### Figure 4.4. Content Assist in JSP Page

After changes made your hello.jsp page should look like this:



#### Figure 4.5. Hello.jsp Page Source

This line will actually output "Hello World!" message in the Console. To make the message displayed in the Browser, just replace this line with the simple Hello World!.

#### 4.2.2. web.xml file

When you are creating web project the wizard creates the web.xml file for you automatically. The web.xml file editor provided by JBoss Developer Studio is available in two modes: Tree and Source.

web		- Web Descrip	tor 2.5	A	
🗟 web.xml		Name:	web		
😋 Context Pa	arams	Display-Name:	DynamProject		
Can Listeners	📾 hello.jsp	🗴 web.xml 🛙			
<ul> <li>Servlets</li> <li>session-c</li> <li>Mime Maj</li> <li>Mime Come</li> <li>Error Page</li> <li>JSP Confit</li> <li>Security (</li> <li>Iogin-con</li> <li>Security (</li> <li>Env Entriticity (</li> <li>ElB</li> <li>Services</li> <li>Resource</li> </ul>	<pre><web-ap; xmlns="formation" xmlns:we xsi:sche <displ <welco <wel <wel <wel <wel <wel <wel <wel <wel< th=""><th><pre>lay-name&gt;jspHello lcome-file&gt;index.htm lcome-file&gt;index.jsp lcome-file&gt;default.h lcome-file&gt;default.h lcome-file&gt;default.j ome-file-list&gt;</pre></th><th><pre>www.w3.org/2001/XMU 'xml/ns/javaee" com/xml/ns/javaee/w 'java.sun.com/xml/ns iisplay-name&gt; nl i i itml itml </pre></th><th></th><th>sun.com/xml/</th></wel<></wel </wel </wel </wel </wel </wel </wel </welco </displ </web-ap; </pre>	<pre>lay-name&gt;jspHello lcome-file&gt;index.htm lcome-file&gt;index.jsp lcome-file&gt;default.h lcome-file&gt;default.h lcome-file&gt;default.j ome-file-list&gt;</pre>	<pre>www.w3.org/2001/XMU 'xml/ns/javaee" com/xml/ns/javaee/w 'java.sun.com/xml/ns iisplay-name&gt; nl i i itml itml </pre>		sun.com/xml/
- nesource					

#### Figure 4.6. Web.xml in Design and Source Mode

Both modes are fully synchronized. Let's add a mapping to our hello.jsp page in the web.xml file.

- Switch to the Source tab.
- Add the next code into <welcome-file-list>:

<welcome-file>hello.jsp</welcome-file>

If you go back to Tree tab you will see that the changes made in the Source tab are automatically reflected.

Actually you do not really need to do any configurations right now.

# 4.2.3. Deploying the project

Writing ant scripts and managing the packaging process can be quite a complicated and time consuming task for even the most trivial web applications. However, JBoss Developer Studio relieves you of this burden. All you need is to start JBoss Server and launch your application in your favorite browser.

You can also create a JAR archive with JBoss Developer Studio's Archive Tools and export it to any web server.

#### 4.2.3.1. JAR Config

Project archives managing is available through Project Archives view.

- \* Select  $\textbf{Window} \rightarrow \textbf{Show view} \rightarrow \textbf{Other} \rightarrow \textbf{JBoss Tools} \rightarrow \textbf{Project archives}$  from the menu bar
- Select a project in Package Explorer you want to be archived

In the Project Archives view you will see the that the project is now listed:

ි Project archives 🛿	E \$	ā	~ - 8
🔁 jspHello			

#### Figure 4.7. Project Archives

Right click on the project and select the JAR type of archive.

🛱 Packag	e Explore 🔂 Project archi	ives ස ්	- 8
		🖻 🕏 🖟	~
🕨 😑 jsfHe	New Archive	JAR	
	Build Project (Full)	*	

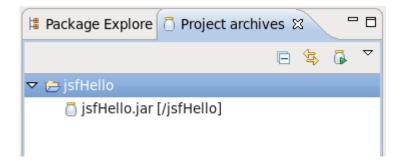
## Figure 4.8. Project Archives

In the New JAR dialog you can see automatically selected default values.

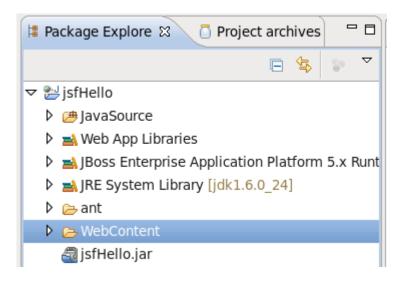
	New JAR					
	Create a new archive					
L						
	Archive information					
	Archive name: 🗍 jsfHello.jar					
	Destination: 🗁 /jsfHello					
	Relative to: O Filesystem Relative  Workspace Relative Variables Filesystem Work					
	Archive type					
	<ul> <li>Standard archive using zip compression</li> </ul>					
	<ul> <li>Exploded archive resulting in a folder (no compression)</li> </ul>					
_						
	Cancel     F					

#### Figure 4.9. New JAR Archive

• Click the **Finish** button. The *.JAR* file will appear in Package Explorer and also in Project Archives view as structure tree:

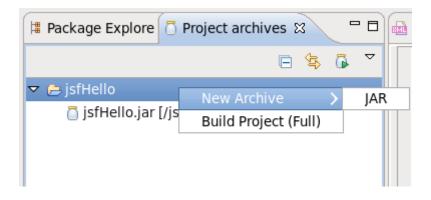


#### Figure 4.10. Archive is Created



#### Figure 4.11. Archive in Package Explorer

Using the Project Archives view you can rebuild the archive:



#### Figure 4.12. Configure Archive

#### 4.2.3.2. Auto redeploy

When you are creating a web application and register it on JBoss Server it is automatically deployed into the /deploy directory of the server. JBoss Developer Studio comes with the feature

of auto-redeploy. It means that you don't need to restart JBoss Server. Any changes made in the application in exploded format will trigger a redeployment on the server.

You can also use the "Finger touch" button for a quick restart of the project without restarting the server:



#### Figure 4.13. Finger Touch button

The "Finger" touches descriptors dependent on project (i.e. web.xml for WAR, application.xml for EAR, jboss-esb.xml in ESB projects).

#### 4.2.4. JSP Page Preview

JBoss Developer Studio comes with JSP design-time preview features. When designing JSP pages you can easily preview how they will look during runtime. You can even attach your stylesheet to the Preview.

• Make a little change to hello.jsp page, e.g. put this code snippet:

```
<%= new java.util.Date() %>
```

- Click the **Save** button.
- Switch to Preview page by clicking the Preview tab at the bottom of the page. You will see how the page will look at runtime.

#### 4.2.5. Launch JSP Project

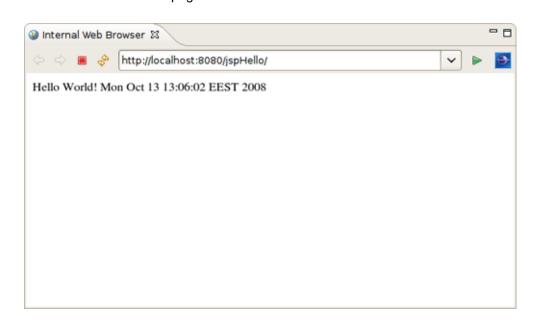
Let's now launch our project on server. We'll use JBoss Server that is shipped with JBoss Developer Studio. You can do it by performing one of the following actions:

- Start JBoss Server from Servers view by clicking the Start the server icon (
  - ).
- Click the **Run** icon or right click your project folder and select **Run As**  $\rightarrow$  **Run on Server**. If you haven't made any changes in the web.xml file or cleared it out you can launch

the application by right clicking the hello.jsp page and selecting Run on the Server(



You should see the next page in a Browser :



#### Figure 4.14. Running Project

Thus with the help of this chapter you've learnt how to organize a Dynamic Web Project with a minimal configuration, add new elements to it (in our case it's just one JSP page) and deploy and run it on the JBoss Server shipped with JBoss Developer Studio.

# RAD development of a simple JSF application



#### Note:

We highly recommend developing in Seam. This chapter is for users who for some reason cannot use Seam.

In this chapter you will learn how to create a simple JSF application being based on the "RAD" philosophy. We will create the familiar Guess Number application. The game is played according to the following rules. You are asked to guess a number between 0 and 100. If the guess is correct, a success page is displayed with a link to play again. If the guess is incorrect, a message is printed notifying that a smaller or a larger number should be entered and the game continues.

We'll show you how to create such an application from scratch, along the way demonstrating the powerful features included in JBoss Developer Studio such as project templating, Visual Page Editor, code completion and others. You will design the JSF application and then run the application from inside JBoss Developer Studio using the bundled JBoss server.

# 5.1. Setting up the project

First, you should create a JSF 1.2 project using an integrated JBoss Developer Studio's new project wizard and predefined templates. Follow the next steps:

 In the Web Projects view (if it is not open select Window → Show View → Others → JBoss Tools Web → Web Projects) click Create New JSF Project button.

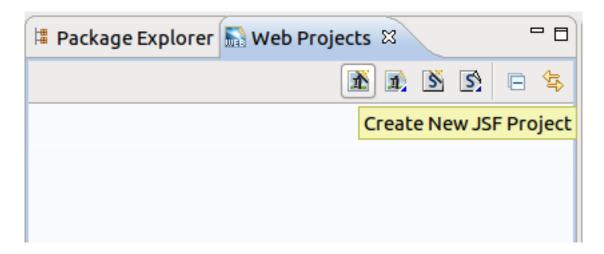
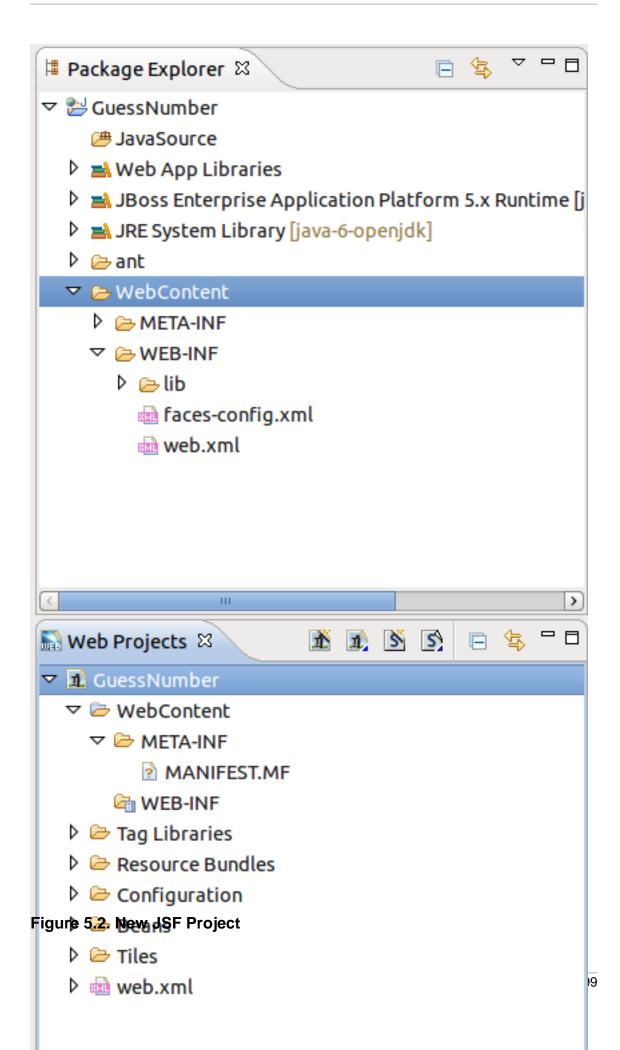


Figure 5.1. Create New JSF Project

- Enter GuessNumber as a project name, in JSF Environment drop down list choose JSF 1.2
- Leave everything else as it is and click the Finish button

Our project will appear in the Project Explorer and Web Projects views. As you can see JBoss Developer Studio has created the entire skeleton for the project with all required libraries, faces-config.xml file and web.xml file.



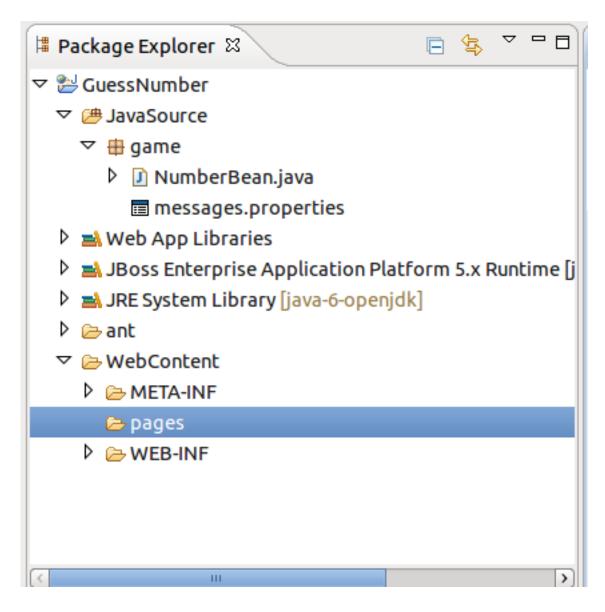
As the project has been set up, new JSP pages should now be created.

# 5.2. Creating JSP Pages

Here, we are going to add two pages to our application. The first page is called inputnumber.jsp. It prompts you to enter a number. If the guess is incorrect, the same page will be redisplayed with a message indicating whether a smaller or a larger number should be tried. The second page is called success.jsp. This page will be shown after you guess the number correctly. From this page you also have the option to play the game again.

Now, we will guide you through the steps on how to do this.

 First a folder called pages needs to be created under the WebContent folder. To do this right click on the WebContent folder in the Package Explorer view and select New → Folder. Set the Folder Name to pages and click the Finish button.



## Figure 5.3. Create pages folder

- Open the faces-config.xml file.
- Right click anywhere on the diagram mode
- From the context menu select New View

🔊 faces-config.xml 🛛								
٦,		New View						
		Auto Layout Select Element	:					
		Verify						
		Paste	Ctrl + V					
		Input Methods	>					

#### Figure 5.4. Create New View

- Type pages/inputnumber as the value for the From View ID field
- Leave everything else as is and click the **Finish** button
- In the same way create another JSF view. Type pages/success as the value for From View ID
- Select File  $\rightarrow$  Save

On the diagram you will see two created views.

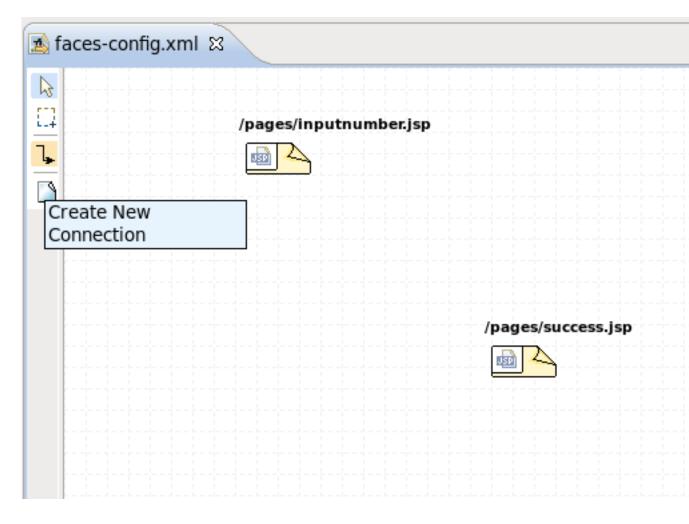
faces-config.xml	
	/pages/inputnumber.jsp
	/pages/success.jsp

#### Figure 5.5. New Views

# 5.3. Creating Transition between two views

Then, we should create connection between JSP pages.

• In the diagram, select the **Create New Connection** icon third from the top along the upper left side of the diagram to get an arrow cursor with a two-pronged plug at the arrow's bottom



## Figure 5.6. Create Connection

• Click on the pages/inputnumber page icon and then click on the pages/success page icon

A transition should appear between the two icons of views.

🔊 *faces-config.xm	83
<b>b</b>	
	/pages/inputnumber.jsp
Ъ	
	success
	/pages/success.jsp

#### Figure 5.7. Created Connection

\* Select  $\textbf{File} \rightarrow \textbf{Save}$  from the menu bar

## 5.4. Creating Resource File

A resource file is just a file with a *.properties* extension for collecting text messages in one central place. JBoss Developer Studio allows you to create quickly a resource file. The messages stored in resource file can be displayed to you on a Web page during application execution.

With resource file you don't hard code anything into the JSP pages. It also makes it easier to translate your application to other languages. All you have to do is to translate all your messages to the other language and save them in a new properties file with a name that ends with the appropriate ISO-639 language code.

It is a good idea to keep your resources inside the JavaSource folder, where you keep your .java files. Every time you build the project, all *.properties* files will then be copied to the classes folder by default.

\* Right click the <code>JavaSource</code> folder and select  $New \to Folder$ 

• Enter game as the Folder name and click the **Finish** button

Your resource file and java bean will be stored in this folder.

- Right click on the game folder and select  $\textbf{New} \rightarrow \textbf{Properties File}$
- Type messages as the value for "name" attribute and click the Finish button

JBoss Developer Studio will automatically open messages.properties file for editing.

🔊 faces-config.xml	📼 messages.properties 🛿	
<ul> <li>Filter</li> </ul>		
name	value	bb <u>A</u>
		Edit
		<u>D</u> elete
		<u>Ш</u> р
		Do <u>w</u> n
Properties Source		

#### Figure 5.8. Messages.properties File

- Click the Add button for adding new attribute to your resource file
- Enter *how\_to\_play* for the "name" and *Please pick a number between 0 and 100*. for the value

- Click the Finish button
- Add the following properties using the same process:

```
makeguess_button=Make Guess
trayagain_button=Play Again?
success_text=How cool.. You have guessed the number, {0} is correct!
tryagain_smaller=Oops..incorrect guess. Please try a smaller number.
tryagain_bigger=Oops..incorrect guess. Please try a bigger number.
```

\* Select  $\textbf{File} \rightarrow \textbf{Save}$  from the menu bar

Your .properties file should now look like follows:

▲ faces-config.xml 🔲 *messages.properties 🛛		
<ul> <li>Filter</li> </ul>		
name	value	
makeguess_button	Make Guess	
trayagain_button	Play Again?	
success_text	How cool You have guessed the number, {0}	
tryagain_smaller	Oopsincorrect guess. Please try a smaller n	
tryagain_bigger	Oopsincorrect guess. Please try a bigger nu	
how_to_play	Please pick a number between 0 and 100.	
Properties Source		

Figure 5.9. Properties are Added

The **Up** and **Down** buttons allow you to move the attributes in the list. To delete the attribute, select it and press the **Delete** button.

If you want to change a value or a name of your attribute, select it and then click the **Edit** button.

If the .properties file is rather big and there are a lot of entries in it, you can use filtering and regular expressions narrow down the list. The Filter and Regular Expressions Search is implemented by an expandable panel, closed by default:

When "Expression" is not selected (as by default), filter is case insensitive. When "Expression" is selected, filter uses regular expressions which are case sensitive

▲ faces-config.xml ■ *messages.properties ≅				
▼ Filter				
Name: *	Value:			
(* = any string, ? = any character, \ = escape for literals * ? \)				
name	value			
makeguess_button	Make Guess			
trayagain_button	Play Again?			
success_text	How cool You have guessed the number, {0} i			
tryagain_smaller	Oopsincorrect guess. Please try a smaller nu			
tryagain_bigger	Oopsincorrect guess. Please try a bigger nur			
how_to_play	Please pick a number between 0 and 100.			
Properties Source				

#### Figure 5.10. Filter and Regular Expressions Search Panel

Enter the characters that should be searched for in the entries to the 'name' or 'value' input fields accordingly. The filtered results will be displayed in the table below:

interset and a state of the st		
<ul> <li>Filter (matches 1 out of 6 items)</li> </ul>		
Name: play	Value:	
(* = any string, ? = any chara	cter, \ = escape for literals * ? \)	
name	value	
how_to_play	Please pick a number between 0 and 100.	
Properties Source		

#### Figure 5.11. Filter results

When using regular expressions please note, that regular expression syntax does not use "\*" for any characters and "?" for any one character. It's necessary to use "." for any one character and ".\*" for any characters. Symbols "\*" and "?" are used to show that the preceding token is not required, for example, "a.a" matches "aba" but not "aa", while "a.?a" or a.\*a" matches both; besides "a.\*a" matches "abcda".

To find the exact match, use sequences \A and \z in expression. For example, expression "\Adate \z" matches only string "date"; expression "\Adate" matches "date" and "dateline", expression "date\z" matches "date" and "Begin date", and expression "date" matches all of them.

## 5.5. Creating Java Bean

In this section you'll learn how to create a Java bean that will hold business logic of our application.

• Right click the game folder

- \* Select  $\textbf{New} \rightarrow \textbf{Class}$
- Type *NumberBean* for bean name
- A java bean is created.
- Declare the variable of your entered number:

Integer userNumber;

JBoss Developer Studio allows to quickly generate getters and setters for java bean.

- Right click the NumberBean. java file in the Package Explorer view
- \* Select Source  $\rightarrow$  Generate Getters and Setters...
- Check userNumber box and click the **OK** button

<b>9</b>	Gen	erate Getters a	and Setters
<u>S</u> elect getters ar	nd setters to creat	e:	
🔽 🗹 🔺 userNu	umber		
🗹 o gett	UserNumber()		
🗹 ° sett	JserNumber(Integ	ег)	
	for final fields (re	move 'final' mor	lifier from fields if ne
_	Tor machedos (re	move milat mot	
Insertion point:			
Last member			
S <u>o</u> rt by:			
Fields in getter	/setter pairs		
Access modifier			
o public	O protected	○ d <u>e</u> fault	○ pri <u>v</u> ate
🗆 <u>fi</u> nal	synchronized	I	
Generate mel	thod <u>c</u> omments		
The format of th	e getters/setters	may be configure	ed on the <u>Code Templa</u>
i 2 of 2 selecte	:d.		
			Cancel

Figure 5.12. Generate Getters and Setters

· Add the declaration of the second variable

int randomNumber;

• .. other bean methods:

```
public NumberBean ()
{
   randomNumber = (int)(Math.random()*100);
   System.out.println ( "Random number: "+randomNumber);
  }
 public String playagain ()
  {
   FacesContext context = FacesContext.getCurrentInstance();
   HttpSession session =
        (HttpSession) context.getExternalContext().getSession(false);
   session.invalidate();
   return "playagain";
 }
 public String checkGuess ()
  {
    // if guessed, return 'success' for navigation
   if ( userNumber.intValue() == randomNumber )
    {
     return "success";
    }
else
   {
     FacesContext context = FacesContext.getCurrentInstance();
     ResourceBundle bundle = ResourceBundle.getBundle("game.messages",
      context.getViewRoot().getLocale());
      String msg = "";
      // if number bigger, get appropriate message
      if ( userNumber.intValue() > randomNumber )
         msg = bundle.getString("tryagain_smaller");
      else // if number smaller, get appropriate message
         msg = bundle.getString("tryagain_bigger");
      // add message to be displayed on the page via <h:messages> tag
      context.addMessage (null, new FacesMessage(msg));
      // return 'tryagain' for navigation
     return "tryagain";
    }
  }
```

• And the import declarations:

```
import javax.faces.context.FacesContext;
import javax.servlet.http.HttpSession;
import javax.faces.application.FacesMessage;
import java.util.ResourceBundle;
```

The whole java bean contain the following code:

```
package game;
import javax.faces.context.FacesContext;
import javax.servlet.http.HttpSession;
import javax.faces.application.FacesMessage;
import java.util.ResourceBundle;
public class NumberBean
{
  Integer userNumber;
  int randomNumber; // random number generated by application
  public Integer getUserNumber ()
  {
   return userNumber;
  }
  public void setUserNumber (Integer value)
  {
     this.userNumber = value;
  }
  // constructor, generates random number
  public NumberBean ()
  {
   randomNumber = (int)(Math.random()*100);
   System.out.println (
   "Random number: " + randomNumber);
  }
  public String playagain ()
  {
   FacesContext context = FacesContext.getCurrentInstance();
   HttpSession session =
        (HttpSession) context.getExternalContext().getSession(false);
   session.invalidate();
   return "playagain";
  }
  // check if user guessed the number
  public String checkGuess ()
```

```
// if guessed, return 'success' for navigation
   if ( userNumber.intValue() == randomNumber )
   {
     return "success";
   }
   // incorrect guess
   else
  {
     // get a reference to properties file to retrieve messages
     FacesContext context = FacesContext.getCurrentInstance();
     ResourceBundle bundle =
         ResourceBundle.getBundle("game.messages",
         context.getViewRoot().getLocale());
     String msg = "";
     // if number is bigger, get appropriate message
     if ( userNumber.intValue() > randomNumber )
        msg = bundle.getString("tryagain_smaller");
     else // if number smaller, get appropriate message
        msg = bundle.getString("tryagain_bigger");
     // add message to be displayed on the page via <h:messages> tag
     context.addMessage (null, new FacesMessage(msg));
     // return 'tryagain' for navigation
     return "tryagain";
   }
 }
}
```

## 5.6. Editing faces-config.xml File

In this section you will learn about the faces-config.xml file.

This file holds two navigation rules and defines the backing bean used.

- Open the faces-config.xml file in a source mode
- Here we will add one more navigation rule and a managed bean declaration, so that the content of the file looks like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<faces-config
version="1.2"
xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:xi="http://www.w3.org/2001/XInclude"</pre>
```

```
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
   http://java.sun.com/xml/ns/javaee/web-facesconfig_1_2_.xsd">
 <navigation-rule>
   <from-view-id>*</from-view-id>
   <navigation-case>
     <from-outcome>playagain</from-outcome>
     <to-view-id>/pages/inputnumber.jsp</to-view-id>
    </navigation-case>
 </navigation-rule>
 <navigation-rule>
   <from-view-id>/pages/inputnumber.jsp</from-view-id>
   <navigation-case>
     <from-outcome>success</from-outcome>
     <to-view-id>/pages/success.jsp</to-view-id>
   </navigation-case>
 </navigation-rule>
 <managed-bean>
   <managed-bean-name>NumberBean</managed-bean-name>
   <managed-bean-class>game.NumberBean</managed-bean-class>
   <managed-bean-scope>session</managed-bean-scope>
 </managed-bean>
</faces-config>
```

The first navigation rule states that from any page (\* stands for any page) an outcome of playagain will take you to the /pages/inputnumber.jsp file. Outcome values are returned from backing bean methods in this example. The second navigation rule states that if you are at the page /pages/inputnumber.jsp, and the outcome is success, then navigate to the /pages/ success.jsp page.

## 5.7. Editing the JSP View Files

Now, we will continue editing the JSP files for our two "views" using the Visual Page Editor.

## 5.7.1. Editing inputnumber.jsp page

First, edit the inputnumber.jsp file.

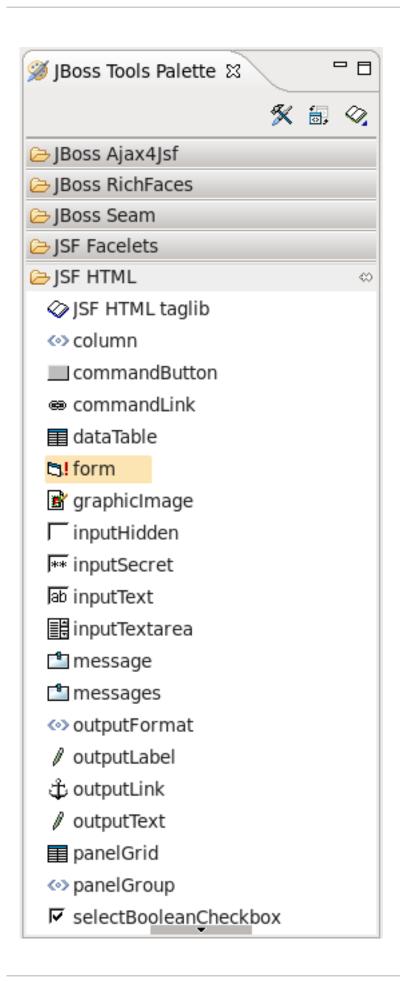
On this page we will have an output text component displaying a message, a text field for user's number entering and a button for input submission.

• Open the inputnumber.jsp file by double-clicking on the /pages/inputnumber.jsp icon

The Visual Page Editor will open in a screen split between source code along the top and a WYSIWIG view along the bottom. You can see that some JSF code will have already been generated since we chose a template when creating the page.

At the beginning it's necessary to create a <h:form> component that will hold the other components.

- Place the mouse cursor inside the <f:view></f:view> tag
- Go to JBoss Tools Palette and expand JSF HTML folder by selecting it
- Click on the <h:form> tag



• In the Insert Tag dialog select the *id* field and click on the second column. A blinking cursor will appear in a input text field inviting to enter a value of id

Insert Tag			
<h:form> attribut</h:form>	es		S
Attribute name	Value		
accept			
acceptcharset			
binding			
dir			
enctype			
id	inputNumbers		ſ
lang			
onclick			
ondblclick			
onkeydown			
onkeypress			
onkeyup			
?		Cancel	<u>F</u> inish

#### Figure 5.14. Define Id of Form

• Enter *inputNumbers* and click the **Finish** button

In source view you can see the declaration of a form.

```
🖹 inputnumber.jsp 🛛
    <%@ page language="java" contentType="text/html; charset=UT</pre>
        pageEncoding="UTF-8"%>
    <%@ taglib prefix="f" uri="http://java.sun.com/jsf/core"%>
    <%@ taglib prefix="h" uri="http://java.sun.com/jsf/html"%>
    <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//
  ⊖<html>
  ⊖<head>
    <meta http-equiv="Content-Type" content="text/html; charset
    <title>Insert title here</title>
    </head>
  ⊖ <body>
  ⊖<f:view>
        <h:form id="inputNumbers">
   \Theta
        </h:form>
    </f:view>
    </body>
    </html>
                                                             ш
Figure 5.15. Created Form
```

First let's declare the properties file in the inputnumber.jsp page using the loadBundle JSF tag.

• Add this declaration on the top of a page, right after the first two lines:

<f:loadBundle basename="game.messages" var="msg"/>

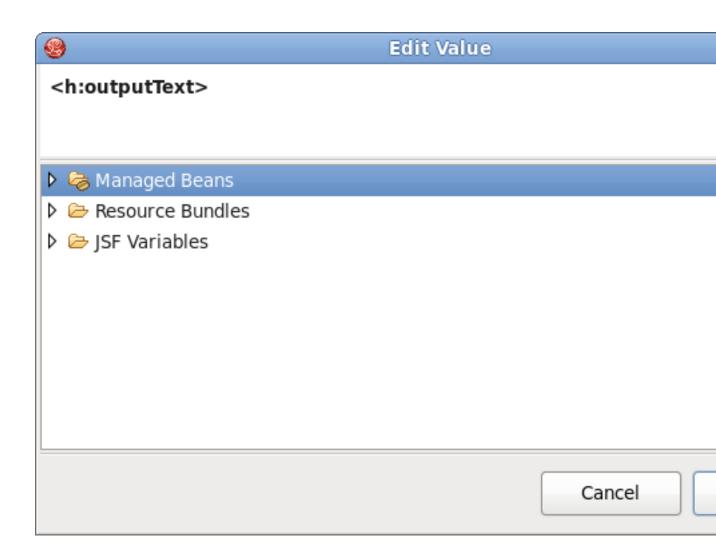
As always JBoss Developer Studio provides code assist:

🔊 faces-config.xml 🛛 😥 🖬 messages.pro	perties	- 0	🚿 JBoss Tools Palette 🕱 🛛 🗖
<pre><ul>     <li>de taglib uri="http://java.sun.com/jsf/html" prefix=</li>     <li>de taglib uri="http://java.sun.com/jsf/core" prefix=</li>     <li><f:1< li=""> </f:1<></li></ul></pre>		<b>A</b>	🛠 🗑 🔗
S C f:loadBundle	view, and exposi of the current red the "var" attribute a get() call is ma literal string ???Ki the key being loo MissingResource	undle localized for the Lo e it as a java.util.Map in th quest under the key speci e of this tag. The Map mus de for a key that does not EY??? is returned from the ked up in the Map, instear Exception being thrown. If JspException must be the	e request attributes fied by the value of it behave such that if exist in the Map, the e Map, where KEY is d of a f the ResourceBundle
f:I Visual/Source Preview		, , , , , , , , , , , , , , , , , , ,	<ul> <li>✓ outputLabel</li> <li>➡ outputLink</li> <li>✓ outputText</li> <li>➡ panelGrid</li> <li>◇ panelGroup</li> <li>✓ selectBooleanCheckbox</li> <li>➡ selectManyCheckbox</li> <li>➡ selectManyListbox</li> <li>➡ selectManyMenu</li> <li>➡ selectOneListbox</li> <li>➡ selectOneMenu</li> <li>&lt; selectOneRadio</li> </ul>

Figure 5.16. Code Assist

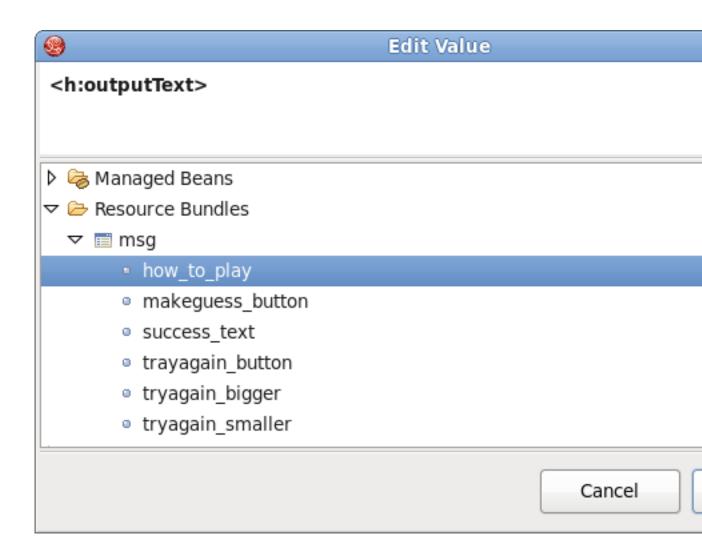
- Switch to Visual tab, where it is possible to work with the editor through a WYSIWYG interface
- Click the *outputText* item from the **JSF HTML** group in the **JBoss Tools Palette** view, drag the cursor over to the editor, and drop it inside the blue box in the editor
- Select the second column in the value row.
- Click the ... button next to the value field

JBoss Developer Studio will display a list of possible values:



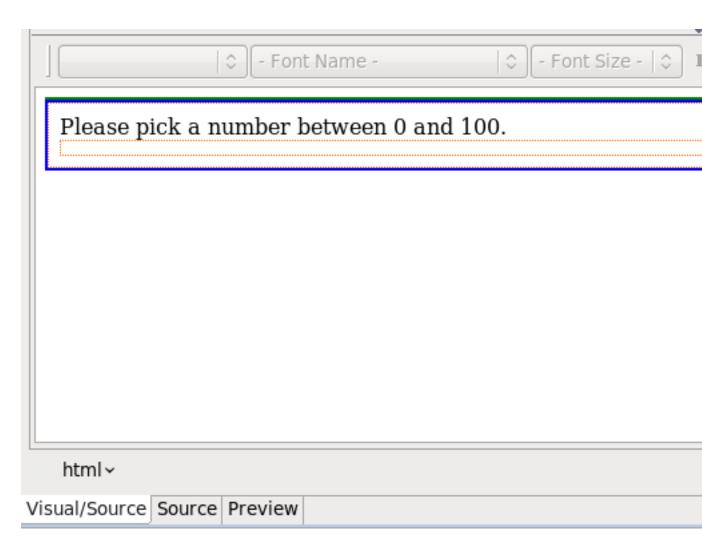
#### Figure 5.17. Choose Value

- \* Expand Resource Bundles  $\rightarrow$  msg
- Select the *how\_to\_play* value and click the **OK** button. Then click the **Finish** button.



#### Figure 5.18. Selecting Value

The text will appear on the page:



## Figure 5.19. Created OutputText Component

- Switch to Source mode and insert a <br/> tag after the <h:outputText> component to make a new line
- Click the Save button
- On the Palette click on *inputText,* drag the cursor over to the editor, and drop it inside the editor after the text
- Select the value row and click in the second column
- Click the ... button next to the value field
- \* Expand Managed Beans  $\rightarrow$  NumberBean
- Select userNumber value and click the **OK** button
- Select the Advanced tab

- Select the *id* row and click in the second column
- Type userNumber in the text field
- Select the *required* row and click in the second column
- Click ... button next to the value field
- Expand Enumeration and select true as a value

<b>ම</b>	Edit Required
<h:inputtext></h:inputtext>	
✓   Enumeration ⊨ false	
<ul> <li>− true</li> <li>Managed Beans</li> <li>Besource Bundles</li> <li>▷ ▷ JSF Variables</li> </ul>	
	Cancel

#### Figure 5.20. Add "required" Attribute

- Click the OK button, then click the Finish button
- · Go to Source mode
- Add the validation attribute to <f:validateLongRange> for user input validation

<h:inputText id="userNumber" value="#{NumberBean.userNumber}" required="true">

```
<f:validateLongRange minimum="0" maximum="100"/></h:inputText>
```

- Click the Save button
- Again select Visual mode
- On the Palette, click on *commandButton*, drag the cursor over to the editor, and drop it inside the editor after the inputText component.
- In the editing dialog select the value row and click on the second column
- · Click the ... button next to the value field
- \* Expand Resource Bundles  $\rightarrow$  msg and select <code>makeguess\_button</code> as a value
- Click the OK button
- Select the action row and click in the second column
- Type #{NumberBean.checkGuess} in the text field
- Click the Finish button
- In Source mode add <br/>
   tags between the <outputText>, <inputText> and <commandButton> components to place them on different lines

inputnumber.jsp page should look like this:

```
<%@
                language="java" contentType="text/html;
                                                              charset=UTF-8"
       page
pageEncoding="UTF-8"%>
<%@ taglib prefix="f" uri="http://java.sun.com/jsf/core"%>
<%@ taglib prefix="h" uri="http://java.sun.com/jsf/html"%>
<f:loadBundle basename="game.messages" var="msg"/>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://
www.w3.org/TR/html4/loose.dtd">
<html>
 <head>
 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
 <title>Insert title here</title>
 </head>
 <body>
  <f:view>
   <h:form id="inputNumbers">
   <h:outputText value="#{msg.how_to_play}"/>
    <br/>
   <h:messages style="color: blue" />
```

```
<br/>
                        <h:inputText
                                           id="userNumber"
                                                                required="true"
value="#{NumberBean.userNumber}">
    <f:validateLongRange minimum="0" maximum="100" />
    </h:inputText>
   <br/>
   <br/>
                        <h:commandButton
                                              action="#{NumberBean.checkGuess}"
value="#{msg.makeguess_button}"/>
   </h:form>
</f:view>
</body>
</html>
```

#### 5.7.2. Editing success.jsp page

We now edit the success.jsp page in the same way as we just edited the inputnumber.jsp file. The code for the success.jsp page should look like the following:

```
<%@ taglib uri="http://java.sun.com/jsf/html" prefix="h" %>
<%@ taglib uri="http://java.sun.com/jsf/core" prefix="f" %>
<f:loadBundle basename="game.messages" var="msg"/>
<html>
<head>
 <title></title>
</head>
<body>
 <f:view>
  <h:form id="result">
   <h:outputFormat value="#{msg.success_text}">
   <f:param value="#{NumberBean.userNumber}" />
   </h:outputFormat>
   <br />
   <br />
   <h:commandButton value="#{msg.trayagain_button}"
    action="#{NumberBean.playagain}" />
  </h:form>
 </f:view>
</body>
</html>
```

Again you can use code assist provided by JBoss Developer Studio when editing jsp page:

🗟 *s	success.jsp 🕴	- 8
≪ ⊚	<f:view></f:view>	^
64	<h:form id="result"></h:form>	
1	<h:outputformat value="#{msg.success_text}"></h:outputformat>	_
2	<f:param value=""></f:param>	
[T]	<th></th>	
	🖾 msg	
A	 • New JSF EL Expression - Create a new attribute value	vith #
(EL)	♦ JSP expression - JSP expression <%=%>	
<u>cs</u>	A New JSF EL Expression - Create a new attribute value	vith #
	] [ → ] - Font Name -	
	How cool You have guessed t	
	Play Again?	
		>
	html body f:view h:form h:outputFormat f:param	x
Visu	al/Source Source Preview	

#### Figure 5.21. Code Assist for <f:param>

The success.jsp page is shown if you correctly guessed the number. The <h:outputFormat> tag will get the value of success\_text from the properties file. The {0} in success\_text will be substituted for by the value of the value attribute within the <f:param> tag during runtime.

In the final result you have a button which allows you to replay the game. The action value references a backing bean method. In this case, the method only terminates the current session so that when you are shown the first page, the input text box is clear and a new random number is generated.

• Switch to Preview mode to see how this page will look in a browser:

🖬 inputnumber.jsp 😫
Please pick a number between 0 and 100. #{NumberBean.userNumbe
Make Guess
Visual/Source Source Preview

#### Figure 5.22. Success.jsp in Preview Mode

## 5.8. Creating index.jsp page

Now we need to create the index.jsp page.

The index.jsp page is the entry point of our application. It's just forwarding to the inputnumber.jsp page.

- \* Right click the <code>WebContent</code> folder and select  $New \to JSP$  File
- Enter index for name field and click the Next button.
- Untick the Use JSP Template check box and click the Finish button.
- Edit the source of the file so it looks like the following:

```
<!doctype html public "-//w3c//dtd html 4.0 transitional//en">
<html>
<body>
<jsp:forward page="/pages/inputnumber.jsf" />
</body>
</html>
```

Note the *.jsf* extension of a page. It means that we trigger the JSF controller servlet to handle the page according the servlet mapping in the faces-config.xml file.

## 5.9. Running the Application

Finally, we have all the pieces needed to run the application.

- Start up JBoss server by clicking on the **Start** icon in the Servers view. (If the JBoss Server is already running, stop it by clicking on the red icon and then start it again. After the messages in the Console tabbed view stop scrolling, JBoss is available)
- \* Right-click on the project and select  $\textbf{Run}~\textbf{As} \rightarrow \textbf{Run}~\textbf{on}~\textbf{Server}$
- · Play with the application by entering correct as well as incorrect values

2	Insert title here - Mozilla F
<u>F</u> ile <u>E</u> dit <u>V</u> iew Hi <u>s</u> tory <u>B</u> ookmarks <u>T</u>	<u>f</u> ools <u>H</u> elp
📀 📀 🖌 🥃 🔍 🏠 🕻 http://	ocalhost:8080/GuessNumber/
🕻 Insert title here 🗣	
Please pick a number between 0 a	nd 100.
Make Guess	
Done	

## Figure 5.23. You are Asked to Enter a Number Between 0 and 100

۷		Insert title here - Mozilla Firefox
<u>F</u> ile <u>E</u> dit <u>V</u> iew	/ Hi <u>s</u> tory <u>B</u> ookmark	s <u>T</u> ools <u>H</u> elp
	💈 🔵 🍙 🎑 htt	p://localhost:8080/GuessNumber/pages/inputnu
🕻 Insert title he	re	÷
Please pick a	number between	0 and 100.
<ul> <li>inputNu</li> <li>0 and 10</li> </ul>		er: Validation Error: Specified attribut
-6		
Make Guess		
Done		

# Figure 5.24. Your Input is Validated and an Error Message is Displayed if Invalid Input was Entered

Mozilla Firefox	
Elle Edit View Go Bookmarks Tools Help	
🖕 🗣 🚽 🤔 🛞 🐔 🗧 http://localhost:8080/GuessNumber/pages/inputnumber.jsf 💌 🛛 Go 🗌	
Please pick a number between a 0 and 100.	
• Oops., incorrect guess. Please try a smaller number.	
45	
Make Guess	
Done	

Figure 5.25. After You Enter a Guess, the Application Tells You Whether a Smaller or a Larger Number Should be Tried

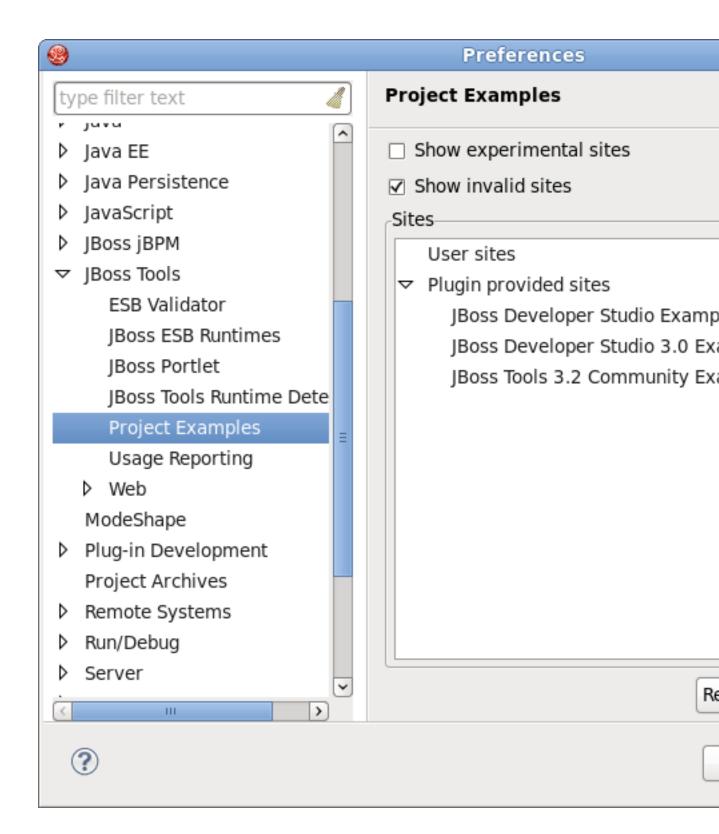
2					Mozilla Firefox	
<u>F</u> ile <u>E</u> dit	<u>V</u> iew	Hi <u>s</u> tory	<u>B</u> ookmarks	<u>T</u> ools	; <u>H</u> elp	
@~~~??	• C		lange 🕻 👔 http:	//localh	host:8080/GuessNumber/pages/inp	outnu
🕻 http://lo	ocalhost	/inputn	umber.jsf 🚽	•		
How co	ol You	u have g	guessed the	e numl	uber, 16 is correct!	
Play Aga	ain?					
Done						

Figure 5.26. Your Guess is Correct

## **Project Examples**

*JBoss Developer Studio* provides an option to download and import a ready-made project that you can explore learn from.

To adjust the settings of the Project Examples feature you need to navigate to **Windows**  $\rightarrow$  **Preferences**  $\rightarrow$  **JBoss Tools**  $\rightarrow$  **Project Examples**.



#### Figure 6.1. Project Examples Preferences

The Show experimental sites checkbox serves to enable or disable user sites in the Project Example dialog (Help  $\rightarrow$  Project Examples).

## 6.1. User Sites

As you can see from the **Project Examples Preferences** image you can add a custom project example that can be provided by anyone. This feature can, for example, facilitate project testing.

In oder to add a new project example you need to select the User sites option and press the **Add** button to the right.

When the **Add** button is pressed the Add Project Example Site dialog is displayed. The dialog contains two input fields: Name, where you need to specify the name of the new entry and URL, which has to point to the XML file that contains example project(s) properties. The structure of the XML file is discussed in more detail in a later chapter of this guide. Alternatively, if the XML is stored on your local machine, you can hit the **Browse** button to select the file in the file system.

Here is an example of the XML file that holds project example settings:

```
<projects>
   <project>
 <category>User Examples</category>
 <name>User Project Example</name>
  <shortDescription>
 Short project description.
 </shortDescription>
 <description>
    Full project description.
 </description>
 <size>10900</size>
 <url>
 http://projectexample.org/projectexample.zip
 </url>
</project>
</projects>
```

Once you define the location of the XML file with projects settings you will see a new user site entry added. Please note now if you select the entry you can edit and remove it with the corresponding buttons to the right. You can not perform such operations with the Plugin provides sites.

When the user sites location is set up you can download and install the project(s). Please see the next chapter of the guide for more details.

## 6.2. Downloading a Project Example

To download a project example and start working with it you need to perform a few steps:

```
• Go to the menu bar and select \textbf{File} \rightarrow \textbf{New} \rightarrow \textbf{Other...}
```

 Select Jboss Tools → Project Examples (You can also select Project Examples from menu bar: Help → Project Examples... or directly by selecting File → New → Example... menu )

New Project Example					
Project Example					
Import Project Example					
Show experimental sites					
Site: All					
Projects:					
type filter text					
Portlet					
▷ ESB for SOA-P 4.3					
ESB for SOA-P 5.0					
Teiid Designer					
Description:					
Draiast name:					
Project name:					
Project size:					
URL:					
Figure 6.2. Project Examples dialog					

#### **Chapter 6. Project Examples**

Alternatively, you should navigate to  $New \rightarrow Other...$ , scroll down to find the JBoss Tools option (or just type in the first letters of the word "JBoss" for quick search), expand the option and select **Project Examples**, and click the **Next** button.

• Now in the **New Project Example** dialog you can select a project you would like to explore and a site to download it from

The **Project Examples Wizard** provides a filter field to more easily locate the project examples you want, so you can type in the project you would like to explore in the field.

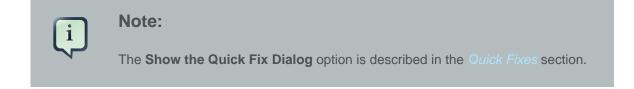
)	New Preisch Exemple	
	New Project Example	
Project Example		
Import Project Example		
Show experimental sites		
Site: All		
Projects:		
type filter text		
Seam Booking Example - EAR		
Seam Booking Example - EAR (including a test project)		
	oking Example - WAR Standalone	
Seam Bo	oking Example - WAR Standalone (including a test project)	
Description:		
Transaction and	emonstrates the use of Seam in a Java EE 5 environment. I persistence context management is handled by the EJB con equires JBoss EAP 4.3/JBoss AS 4.2.x and Seam 2.0.	
Project name:	jboss-seam-jpa	
Project size:	5.34M	
URL:	http://anonsvn.jboss.org/repos/jbosstools/workspace/snjez	
	· · · · · · · · · · · · · · · · · · ·	

If you have previously specified user sites (see the *User Sites* chapter) they also will be displayed in the list of project examples in the category that was defined in the XML file with user sites settings.

	New Project Example	×
Project Exam	ple	
Import Project	Example	
Show experies	imental sites	
Site: All		0
Projects:		
type filter text		
▷ Seam		A
Portlet		
▷ ESB		
RichFaces		
BPEL		
	oject description.	
ESB for SOA	A-P 5.0	
RESTEasy Description:		
Full project des	cription	
i un project des		
· · · · · · · · · · · · · · · · · · ·		
Project name:	User Project Example	
Project size:	10.64K	
URL:	http://projectexample.org/projectexample.zip	
Show the Q	uick Fix dialog	
?	< Back Next > Cancel	Finish
-		

### Figure 6.4. User Site

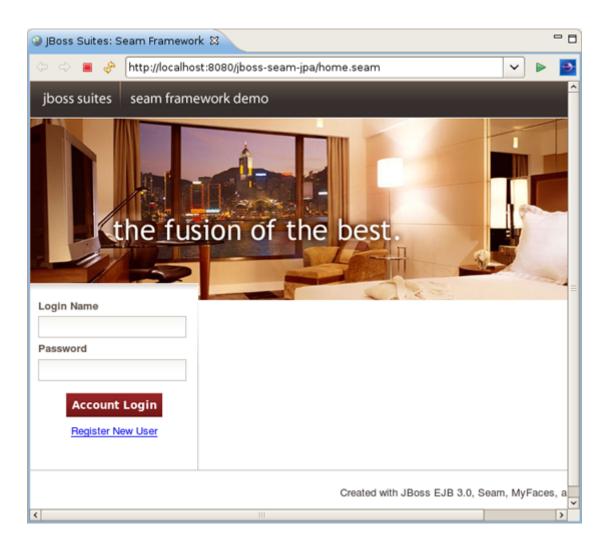
Pleas note that to view the user sites you need to have Show experimental sites checked.



• Press the Finish button to start downloading the project from the repository

When downloading is finished the project will be imported automatically and you will be able to see it in the **Package Explorer** view.

Now you can run the application on the server.



### Figure 6.5. Seam Demo Application run on the Server

For further operation add the following code to .project files of your Web project example.

```
<buildCommand>
```

It is needed for Code Assist and JSF EL Validation to function correctly.

### 6.3. Quick Fixes

The **Project Examples Wizard** has an option for making quick fixes for the imported project to easily fix possible issues like missing servers, Seam runtimes etc.

To enable the quick fix option you need to check the **Show the Quick Fix dialog** while choosing the *Project Example*.

When the project you selected is downloaded it will be checked for missing dependences and if any are detected you will see a dialog listing the problems.

Quick Fix		
Quick Fix		
Select a marker and click the	e Quick Fix button.	
Markers:		
Description	Resource	Туре
Seam Runtime "Seam 2.0.FP	" jboss-seam-jpa	Validation Message
0		<b>Finish</b>
		Finish

### Figure 6.6. Quick Fix Dialog box

To fix the problem you need to:

- Select the problem from the list
- Click the Quick Fix button

You will be offered a solution or a number of solutions to the problem.

Sector Quick Fix	
Quick Fix	
Select the fix for 'Seam Runtime "Seam 2.0.FP" specified for Sea	m project '
Select a fix:	
Set Seam properties	
Add 'Seam 2.0.FP' Seam Runtime	
Problems:	
Resource	Lo
✓ Ø jboss-seam-jpa	lin
Figure 6.7. Quick Fix Dialog box: Selecting a Fix	

In this case (see the image above), when the **Finish** button is pressed, the Seam Settings dialog box will be displayed where you need to provide a path to the Seam environment in order to fix the issue.

When the problem is fixed you will be returned to the Quick Fix dialog box with the remaining problems to be fixed.

You can also fix problems before downloading. When the project example is selected you will see warning message on the **New Project Example** dialog.

New Project Example         e         kample         hental sites         stars         king Example - EAR (including a test project)         king Example - WAR Standalone         king Example - WAR Standalone (including a test project)         Booking Example - EAR (including a tutorial)
kample hental sites king Example - EAR (including a test project) king Example - WAR Standalone king Example - WAR Standalone (including a test project)
hental sites hental sites king Example - EAR (including a test project) king Example - WAR Standalone king Example - WAR Standalone (including a test project)
king Example - EAR (including a test project) king Example - WAR Standalone king Example - WAR Standalone (including a test project)
king Example - EAR (including a test project) king Example - WAR Standalone king Example - WAR Standalone (including a test project)
king Example - EAR (including a test project) king Example - WAR Standalone king Example - WAR Standalone (including a test project)
king Example - EAR (including a test project) king Example - WAR Standalone king Example - WAR Standalone (including a test project)
king Example - EAR (including a test project) king Example - WAR Standalone king Example - WAR Standalone (including a test project)
king Example - EAR (including a test project) king Example - WAR Standalone king Example - WAR Standalone (including a test project)
king Example - WAR Standalone king Example - WAR Standalone (including a test project)
king Example - WAR Standalone (including a test project)
Booking Example - EAR (including a tutorial)
monstrates the use of Seam in a Java EE 5 environment. persistence context management is handled by the EJB con quires JBoss EAP 4.3/JBoss AS 4.2.x and Seam 2.0.
boss-seam-jpa
5.34M
http://anonsvn.jboss.org/repos/jbosstools/workspace/snjez
ple has some requirements that could not be ally configured. When importing the example you might errors which would need fixing manually or via Quick

To fix the problem immediately you need to:

- Click the Details... button in the New Project Example dialog
- Select the problem from the list in the Requirement details dialog box
- Click the Fix button

	Requirement details	
Requirements:		
Туре	Short description	
wtpruntime	Missing WTP Runtime	
seam	Missing Seam Runtime	
Description:		
This project example requires the JBoss EAP 4.3 or JBoss AS 4.2.x		
	Fix	

### Figure 6.9. Requirement Details Dialog box: Selecting a Fix

You will be offered a solution to the problem.

## FAQ

Refer to the following FAQ to get the answers on the most "popular" questions concerning JBoss Developer Studio.

# 7.1. What should I do if the Visual Page Editor does not start under Linux

Linux users may need to do the following to get the Visual Page Editor to work correctly on their machines.

1. On Red Hat based Linux distributions install the libXp.i386 package

### 2. Type

ln -s libstdc++.so.5.0.7 libstdc++.so.5

3. and/or use

yum install libXp

- 4. Open the JBoss Developer Studio perspective. If you see the Help view open, close it and restart JBoss Developer Studio
- Fedora with 5. If it doesn't help and you use Eclipse Version: 3.4.1, the issue can be produced because the libswt-xulrunner-gtk-3449.so file doesn't present in eclipse-swt-3.4.1-5.fc10.x86\_64.rpm/eclipse/plugins/ org.eclipse.swt.gtk.linux.x86\_64\_3.4.1.v3449c.jar. To add this file to eclipse you should:
  - Decompress eclipse/plugins/org.eclipse.swt.gtk.linux.x86\_3.4.1.v3449c.jar form eclipse-SDK-3.4.1-linux-gtk-x86\_64.tar.gz
  - Copy libswt-xulrunner-gtk-3449.so file to your Fedora Eclipse location.
  - Open the file jbdevstudio.ini, which can be found in your Fedora Eclipse location and add the following line:

-Dswt.library.path=/usr/lib/eclipse

,where /usr/lib/eclipse is the path to your eclipse folder.

- 6. If none of these work, do the following:
  - Clear the JBoss Developer Studio log file,<workspace>\.metadata\.log
  - Start JBoss Developer Studio with the -debug option:

jbdevstudio -debug		

• Post the JBoss Developer Studio log file(<workspace>\.metadata\.log) on the forums.

# 7.2. Visual Editor starts OK, but the Missing Natures dialog appears

6	Missing Nat	ures 🗙	
T	JBoss Tools Visual Editor might not fully work in project "WebProject" because it does not have JSF and code completion enabled completely.		
	Please use the Configure menu on the project to enable JSF if you want all features of the editor working.		
🗆 Do n	ot show this dialog again!	See JBoss Tools Visual Editor FAQ	
		ОК	

### Figure 7.1. Missing Nature

Some functionality of Visual Editor may not work if a project doesn't have org.jboss.tools.jsf.jsfnature or org.jboss.tools.jst.web.kb.kbnature in .project configuration. To fix this problem and turn off the message box execute next steps:

- 1. Right mouse button click on a project in Package Explorer.
- <sup>2.</sup> Select **Configure**  $\rightarrow$  **Add JSF Capabilities** from the context menu.
- 3. Configure your project using Add JSF Capabilities wizard and press Finish.

If you are sure that your project does not need JSF capabilities, just disable this message box by checking Do not show this dialog again! checkbox.

# 7.3. Do I need to have JBoss Server installed to run JBoss Developer Studio?

No. JBoss Developer Studio already comes bundled with JBoss Server. We bundle it together so that you don't need to download any additional software and can test your application in a Web browser right away.

If you want to use a different JBoss server installation, after JBoss Developer Studio is installed open Servers View (select **Window**  $\rightarrow$  **Show View**  $\rightarrow$  **Others**  $\rightarrow$  **Server**  $\rightarrow$  **Servers**), then right click and select **View**  $\rightarrow$  **New**  $\rightarrow$  **Server** and follow the wizards steps to point to another Jboss Server installation.

JBoss Developer Studio works with any servlet container, not just JBoss. For more information on deployment, please see the Deploying Your Application section.

## 7.4. I have an existing Seam 1.2.1 project. Can I migrate or import the project into a JBoss Developer Studio Seam project?

Use the following steps to manually transfer an existing Seam 1.2.1 project into a new JBoss Developer Studio Seam project:

· Create a Seam Web project to get the JBoss tools structure

Then from your Seam 1.2.1 seam-gen project start doing the following:

- Copy src to src
- Copy view to Web content
- Copy resources individual files to where they are in the seam web project etc.

## 7.5. I have an existing Struts or JSF project. Can I open the project in JBoss Developer Studio?

Yes. From main menu select File  $\rightarrow$  File  $\rightarrow$  Import  $\rightarrow$  Other  $\rightarrow$  JSF Project (or Struts Project) and follow wizards steps.

### 7.6. Can I import a WAR file?

Yes. Select File  $\rightarrow$  Import  $\rightarrow$  Web  $\rightarrow$  WAR file then follow importing steps.

# 7.7. Is it possible to increase the performance of Eclipse after installing your product?

JBoss Developer Studio configures eclipse via the jbdevstudio.ini file to allocate extra memory, but if you for some reason need more memory then by default, you can manually make adjustments in this file. For example:

-vmargs -Xms128m -Xmx512m -XX:MaxPermSize=128m

## 7.8. How can I add my own tag library to the JBoss Tools Palette?

See the section on Adding Tag Libraries in the Visual Web Tools Guide.

# 7.9. How to get Code Assist for Seam specific resources in an externally generated project?

To get Code Assist for Seam specific resources in an externally generated project, you should enable Seam features in Project Preferences. Right click an imported project and navigate **Properties**  $\rightarrow$  **Seam Settings**. Check *Seam support* box to enable all available Seam Settings.

# 7.10. How to import an example Seam project from jboss-eap directory?

To import an example Seam project from *jboss-eap* into your working directory, you should perform the following steps:

- \* Select  $\textbf{New} \rightarrow \textbf{Other} \rightarrow \textbf{Java Project from Existing Buildfile}$
- Point to the build.xml file of any chosen project by clicking the Browse button
- Click the Finish button to open the project

As these seam examples are non WTP projects, next you should enable Seam support for them. To do that, right click the project and select **Properties**  $\rightarrow$  **Seam Settings**.

## 7.11. Is a cross-platform project import possible for JBoss Developer Studio?

Yes. You can easily import created in Linux JSF, Struts or Seam project to Windows and vice versa.

To do the transferring JSF, Struts or Seam project, select Menu  $\rightarrow$  Import  $\rightarrow$  General  $\rightarrow$  Existing Projects into Workspace.

## **Further Reading**

### • Seam Dev Tools Reference Guide

This guide helps you to understand what Seam is and how to install Seam plug-in into Eclipse. It tells you the necessary steps to start working with Seam Framework and assists in a simple Seam Project creation. Also you will learn how to create and run the CRUD Database Application with Seam as well as find out what Seam Editors Features and Seam Components are.

### • Visual Web Tools Reference Guide

provides general orientation and an overview of visual web tools functionality. This guide discusses the following topics: editors, palette, web properties view, openOn, content assist, RichFaces support.

### • JBoss Server Manager Reference Guide

This guide covers the basics of working with the JBoss server manager. You will read how to install runtimes and servers and quickly learn how to configure, start, stop the server and know how deployment and archiving process. You will find out how to manage installed JBoss Servers via JBoss AS Perspective. You will also read how to deploy modules onto the server.

### • jBPM Tools Reference Guide

With jBPM Tools Reference Guide we'll help you to facilitate a cross-product learning and know how you can speed your development using special editors and visual designers. We'll also guide you through the steps on how to create a simple process and test it within jBPM jPDL perspective.

### Hibernate Tools Reference Guide

Throughout this guide you will learn how to install and use Hibernate Tools bath via Ant and through Eclipse. We'll supply you with the information on how to create mapping files, configuration file as well as a file for controlling reverse engineering by using specific wizards that Hibernate tooling provides. Also you will know about Code Generation and peculiarities of work within Hibernate Console Perspective.

### • ESB Editor Reference Guide

This guide provides you with the information on ESB Editor and all necessary wizards for ESB files development.

### JBoss Portal Tools Reference Guide

The guide gives a detail look at how you can easily build a Portlet Web Application with JBoss Tools and deploy it onto JBoss Portal.

### • JBoss WS User Guide

This guide gives you practical help on JBossWS usage. You will learn how to create a web service using JBossWS runtime, find out how to create a web service client from a WSDL document using JBoss WS and also see how to set your development environment.

### Smooks Tools Reference Guide

This guide is packed with useful and easy-to-understand information about graphical, configuration and source editor pages.

### • Drools Tools Reference Guide

The guide help you to discover how to create a new Drools project, use debugging rules and work with different editors.

### • JMX Tools Reference Guide

With the help of this guide you'll explore the best practices to follow when working with MBean Explorer, MBean Editor, Connections and etc.

### • Eclipse Guvnor Tools Reference Guide

The purpose of this guide is to describe briefly the functionality present in the Eclipse Guvnor Tools (EGT) for Drools 5.

### • JSF Tools Tutorial

This tutorial will describe how to deal with classic/old style of JSF development and how to create a simple JSF application.

### • JSF Tools Reference Guide

From this guide you'll discover all peculiarities of work at a JSF project. You'll learn all shades that cover the process of project creation and take a closer look at the JSF configuration file. Also you'll get to know managed beans and how to work with them and find out, how to create and register a custom converter, custom validator and referenced beans in a JSF project.

### Struts Tools Reference Guide

In Struts Tools Reference Guide you will learn how to create and work with a new struts project. This guide also provides information about graphical editor for struts configuration files, tiles files, and struts validation files.

### • Struts Tools Tutorial

This tutorial will describe the classical style of Struts development, and will step-by-step show you how to create a simple Struts application.