Getting Started with JBoss Developer Studio

ISBN: Publication date: April 2008

Getting Started with JBoss Developer Studio

PDF version

Getting Started with JBoss Developer Studio

Copyright © 2007, 2008 JBoss, a division of Red Hat Inc.

1. Getting Started with JBoss Developer Studio	. 1
1.1. What is JBDS?	. 1
1.2. Configuring Your Java Environment	. 1
1.2.1. Installing and Configuring 32-bit Sun JDK 5.0 on Linux	. 1
1.2.2. Installing and Configuring 32-bit Sun JDK 5.0 on Microsoft Windows	. 3
1.3. JBoss Developer Studio Installation	. 4
1.4. JBoss Developer Studio and JBoss Tools	
1.4.1. What is the difference?	. 8
1.4.2. JBoss Tools Installation	. 9
1.5. Welcome to JBoss Developer Studio	10
1.6. Upgrading	14
1.7. Uninstalling	14
1.8. Support	14
1.9. Other relevant resources on the topic	
2. Manage JBoss AS from JBoss Developer Studio	
2.1. How to Manage the JBoss AS Bundled in JBDS	
2.1.1. Starting JBoss Server	17
2.1.2. Stopping JBoss Server	
2.1.3. Server Container Preferences	-
2.2. How to Use Your Own JBoss AS Instance with JBDS	
2.2.1. JBoss AS Installation	
2.2.2. Adding and configuring JBoss Server	
3. Write Your First Project with JBoss Developer Studio	
3.1. Create a Seam Application	
3.1.1. Start Development Database	
3.1.2. Create and deploy Seam Web Project	
3.1.3. Start JBoss Application Server	
3.1.4. Workshop Project Code Overview	
3.2. Seam Action Development	
3.2.1. Create a New Seam Action	
3.2.2. Test Seam Action	
3.2.3. Modify Seam Action User Interface	
3.3. Declarative Security	
3.3.1. Edit Login Authentication Logic	
3.3.2. Secure Seam Page Component	
3.4. Browsing Workshop Database	
3.4.1. Database Connectivity Setup	
3.4.2. Browse Workshop Database	
3.5. Database Programming 3.5.1. Reverse Engineer CRUD from a Running Database	
3.5.2. Use Hibernate Tools to Query Data via JPA	
3.5.3. Use Hibernate Tools to visualize the Data Model	
3.6. Rich Components	
3.6.1. Add a Richfaces component to the CRUD Application	
o.o Add a Montaces component to the ONOD Application	07

4. Developing a simple JSP web application	71
4.1. Setting Up the Project	71
4.2. Creating JSP Page	73
4.2.1. Editing a JSP Page	74
4.2.2. web.xml file	
4.2.3. Deploying the project	77
4.2.4. JSP Page Preview	80
4.2.5. Launch JSP Project	81
5. RAD development of a simple JSF application	83
5.1. Setting up the project	
5.2. Creating JSP Pages	84
5.3. Creating Transition between two views	86
5.4. Creating Resource File	
5.5. Creating Java Bean	
5.6. Editing faces-config.xml File	
5.7. Editing the JSP View Files	94
5.7.1. Editing inputnumber.jsp page	
5.7.2. Editing success.jsp page 1	101
5.8. Creating index.jsp page 1	
5.9. Running the Application 1	
6. FAQ 1	
6.1. What should I do if Visual Page Editor does not start under Linux 1	
6.2. Do I need to have JBoss Server installed to run JBoss Developer Studio? 1	09
6.3. I have an existing Seam 1.2.1 project. Can I migrate/import the project to a JBDS	
Seam project? 1	
6.4. I have an existing Struts or JSF project. Can I open the project in JBDS? 1	
6.5. Can I import a .war file? 1	
6.6. Is it possible to increase the performance of Eclipse after installing your product? 1	
6.7. How can I add my own tag library to the JBoss Tools Palette? 1	11
6.8. How to get Code Assist for Seam specific resources in an externally generated	
project? 1	
6.9. How to import an example Seam project from jboss-eap directory? 1	
7. Further Reading 1	13

Getting Started with JBoss Developer Studio

1.1. What is JBDS?

JBoss Developer Studio is a set of eclipse-based development tools that are pre-configured for JBoss Enterprise Middleware Platforms and Red Hat Enterprise Linux. Developers are not required to use JBoss Developer Studio to develop on JBoss Enterprise Middleware and/or Red Hat Linux. But, many find these pre-configured tools offer significant time-savings and value, making them more productive and speeding time to deployment.

This guide covers the first steps to get started working with JBoss Developer Studio. You will learn how to install and configure necessary software for your OS (currently Linux, Microsoft Windows or Mac OSX).

Thus this guide will provide you with detailed info on how to start JDK, JBDS and JBoss Tools.

1.2. Configuring Your Java Environment

You must have a working installation of JDK 5 before you install JBoss Developer Studio. Currently it will only fully work with a 32-bit JVM, not a 64-bit JVM. On a 64-bit JVM the visual editor fails to launch because of feature of OS architecture. Thus in this guide we will show you how to install a 32-bit Sun JDK 5.0 on a Linux Platform and Microsoft Windows Platform.

1.2.1. Installing and Configuring 32-bit Sun JDK 5.0 on Linux

To install 32-bit Sun JDK 5.0 on Linux and configure it, you should follow the next steps:

- Download the <u>Sun JDK 5.0 (Java 2 Development Kit</u>) from Sun's website. Choose "JDK 5.0 Update <x>" (where "x" is the latest update number) for download and then select "Red Hat Package Manager in self-extracting" file for Linux. Read the instructions on Sun's website for installing the JDK.
- If you don't want to use SysV service scripts you can install the "self-extracting file" for Linux instead of choosing the "RPM in self-extracting" file. In that case you can skip the next step mentioned here. But it is recommended to use the SysV service scripts for production servers.
- Download and install the appropriate -compat RPM from JPackage <u>here</u>. Please ensure you choose a matching version of the -compat package to the JDK you've installed.
- Create an environment variable that points to the JDK installation directory and call it JAVA_HOME. Add *\$JAVA_HOME/bin* to the system path to be able to run java from the command line. You can do this by adding the following lines to the .bashrc file in your home directory.

#In this example /usr/java/jdk1.5.0_11 is the JDK installation directory. export JAVA_HOME=/usr/java/jdk1.5.0_11 export PATH=\$PATH:\$JAVA_HOME/bin



Note:

#If you have JDK already installed and added in your system path, you should add \$JAVA_HOME/bin before the old \$PATH (not after it) so that the new version of JDK can be found first, i. e.

export PATH=\$JAVA_HOME/bin:\$PATH This way, the machine will pick up the new JVM first. You only need to run "alternative" as a safe guard for the right JVM.

Set this variable for your account doing the installation and also for the user account that will run the server.

 If you have more than one version of JVM installed on your machine, make sure you are using the JDK 1.5 installation as the default java and javac. You can do this using the alternatives system. The alternatives system allows different versions of Java from different sources to coexist on your system.

1.2.1.1. Select alternatives for java, javac and java_sdk_1.5.0

• As a root user, type the following command at the shell prompt and you should see something like this:

[root@vsr ~]\$ /usr/sbin/alternatives --config java There are 2 programs that provide 'java'. Selection Command

1 /usr/lib/jvm/jre-1.4.2-gcj/bin/java

*+ 2 /usr/lib/jvm/jre-1.5.0-sun/bin/java

Enter to keep the current selection[+], or type selection number:

Make sure the Sun version [jre-1.5.0-sun in this case] is selected (marked with a '+' in the output), or select it by entering its number as prompted.

• Repeat the same for javac and java_sdk_1.5.0.

You should verify that java, javac and java_sdk_1.5.0 all point to the same manufacturer and version.



Note:

You can always override this step by setting the JAVA_HOME environment variable as explained in the previous step.

• Make sure that the java executable is in your path and that you are using an appropriate version. To verify your Java environment, type "java -version" at the shell prompt and you should see something like this:

[root@vsr ~]\$ java -version java version "1.5.0_11" Java(TM) 2 Runtime Environment, Standard Edition (build 1.5.0_11-b03) Java HotSpot(TM) Client VM (build 1.5.0_11-b03, mixed mode, sharing)

1.2.2. Installing and Configuring 32-bit Sun JDK 5.0 on Microsoft Windows

To install and configure 32-bit Sun JDK 5.0 on Microsoft Windows, follow these steps:

Download the <u>Sun JDK 5.0 (Java 2 Development Kit</u>) from Sun's website. Choose "JDK 5.0 Update <x>" (where "x" is the latest update number) for download and then select your Windows Platform options to perform the installation.

• Create an environment variable called JAVA_HOME that points to the JDK installation directory, for example:



In order to run java from the command line, add the *jre\bin* directory to your path, for example:

C:\Program Files\Java\jdk1.5.0_11\jre\bin

To do this, open the *Control Panel* from the *Start* menu, switch to Classic View if necessary, open the System Control Panel applet (*System*), select the *Advanced* Tab, and click on the *Environment Variables* button.

Now, when 32-bit Sun JDK 5.0 has been successfully installed, we can pass on to the next step.

1.3. JBoss Developer Studio Installation

This chapter will provide you with detailed information on how to install JBoss Developer Studio.

JBDS comes with a simple installer, bundled with tested/pre-configured versions of Eclipse, WTP, JBossEAP, Seam, and SpringIDE. Thus, to start perform the next steps:

- First of all you need the appropriate installation file for your platform from <u>Red Hat website</u>.
- Then run in console:

java -jar jbdevstudio-linux-gtk-1.1.0.GA.jar

· Follow the instructions presented by the installation wizard

	JBoss Developer Studio GA	
Introduction Step 1 of 10		Jises DEVELOPER STUDIO
A Please read the following inform	nation:	
This installer will guide you t	nrough the installation of JBoss Deve	loper Studio GA.
It is strongly recommended th	nat you quit all programs before conti	inuing with this installation.
Click the "Next" button to pro previous screen, click the "Pre	ceeded to the next screen. If you want wious" button.	t to change something on a
You may stop this installation	at any time by clicking the "Quit" bu	tton.
JBoss, a division of Red Hat		
		🜍 Next 😣 Quit

Figure 1.1. JBoss Developer Studio Installation Wizard

- Provide the installation path
- Select Java VM

JBoss Developer Studi	o GA
Select Java VM Step 4 of 10	JBoss DEVELOPER STUDIO
JBoss Developer Studio needs Java 5 and does not work with gij java	1
🔿 Default Java YM	
Specific Java VM	
/usr/java/jdk1.5.0_12/jre	📔 Browse
Dare a decision of Dad Man	
JBoss, a division of Red Hat	🚱 Previous 🛛 😧 Next 😣 Quit

Figure 1.2. Select Java VM

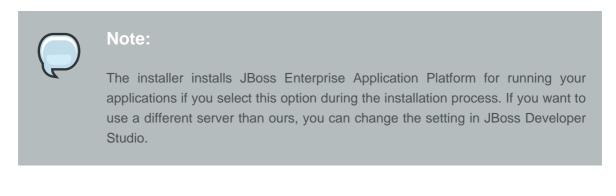




 Installation process includes <u>JBoss Enterprise Application Platform</u>. Select Yes to use it in JBoss Developer Studio.

JBoss Developer Studio GA	_×_
JBoss Enterprise Application Platform Step 5 of 10	JBess DEVELOPER STUDIO
Do you want to install the JBoss Enterprise Application Platform that comes with JBos Developer Studio ?	:5
JBoss, a division of Red Hat	lext 😣 Quit

Figure 1.3. JBoss Enterprise Application Platform Installing



• Check your installation paths and see the components to install. If you'd like to change something, press *Previous* button. Click *Next* to start installation.

JBoss Developer	Studio GA
Summary Information Step 7 of 10	JBoss DEVELOPER STUDIO
Ready to install. Important data are listed below. Press "Next	t" to start installation,
Installation path /opt/work/jbds_ga Java VM path: /usr/java/jdk1.5.0_12/jre Following components will be installed: JBoss Developer Studio GA Eclipse JBoss Enterprise Application Platform Disk Space: Available Disk Space: 1.28 GB Required Disk Space: 635.9 MB	
JBoss, a division of Red Hat	G Previous Sext Quit

Figure 1.4. Summary Information

1.4. JBoss Developer Studio and JBoss Tools

This section uncovers the points on the differences between JBoss Developer Studio and JBoss Tools and provides the steps on JBoss Tools installation as well.

1.4.1. What is the difference?

JBoss Tools is what went into our JBoss Developer Studio which comes as an easy-to-install Eclipse based IDE fully configured and ready to run with the bundled JBoss Enterprise Application Platform.

In short JBoss Tools are just a set of Eclipse plugins and JBoss Developer Studio adds:

- An installer
- Eclipse and Web Tools preconfigured
- JBoss EAP with JBoss AS and Seam preconfigured
- 3rd party plugins bundled and configured
- Access to RHEL and Red Hat Network
- · Access to the JBoss/Red Hat supported software

For additional information see <u>JBoss.com</u>

1.4.2. JBoss Tools Installation

Here, let's consider the installation of the JBoss Tools modules.

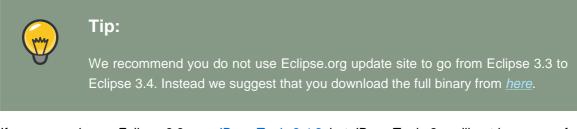
JBoss Tools is an umbrella project for the JBoss developed plugins that will make it into JBoss Developer Studio. The JBoss Tools modules are:

- JBoss AS Tools
- Seam Tools
- Hibernate Tools
- Visual Page Editor
- JST Tools
- JBPM Tools

To install the JBoss Tools plugins for Eclipse, you need the following:

· Get Eclipse and Web Tools

The currant version of JBoss Tools (3.0.0.beta1) targets at Eclipse 3.4/Ganymede and WTP 3.0.



If you can only use Eclipse 3.3 use <u>JBoss Tools 2.1.2</u>, but JBoss Tools 2.x will not have any of the new features.



Note:

Remember to choose the download that matches your OS and use Java 5 when you run it.

• Get the *latest JBoss Tools build*

Some of our newer plugins, like TPTP and BIRT, need additional drivers. On the left side of the download page you can find all of the required drivers for chosen build and their versions.

Instead of downloading the nightly build version manually, it's also possible to get the latest release of JBoss Tools from one of our update sites:

- Stable Updates: http://download.jboss.org/jbosstools/updates/stable
- Development Updates: <u>http://download.jboss.org/jbosstools/updates/development</u>
- Finally, install the build

Unzip the file(s) directly into your Eclipse *plugins/features* directory and it will be readily available. It might be necessary to start Eclipse with eclipse -clean to make sure it starts clean and rereads the new list of plugins.

If you need to install any standalone plug-in from JBoss Tools visit a <u>JBoss Tools Wiki</u> page to read about dependencies between standalone plug-ins.

1.5. Welcome to JBoss Developer Studio

In this section we'll show you how to work with the welcome page of the JBoss Developer Studio.

The welcome page is the first page you see when you first launch JBoss Developer Studio.

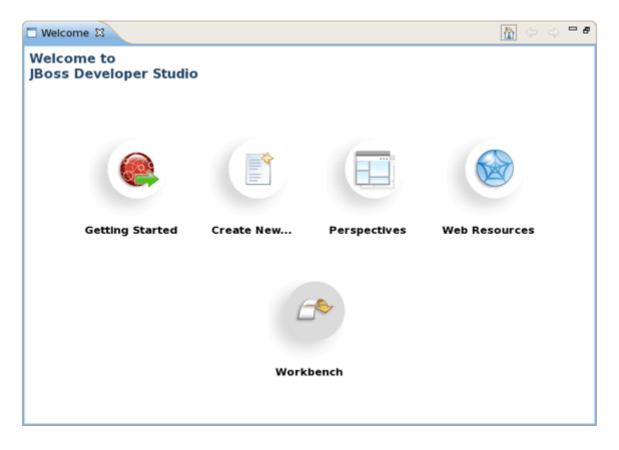


Figure 1.5. Welcome to JBoss Developer Studio

With the help of its page you will be able:

• to get quick access to Getting Started Documentation (guides, tutorials and viewlets)

□ Welcome 🛙	🏠 🔶 🗢 🖱
Welcome to JBoss Developer Studio	
Getting Started	
JBoss Developer Studio is a set of eclipse-based development tools that are pre-config JBoss Enterprise Middleware Platforms and Red Hat Enterprise Linux. Developers may pre-configured tools offer significant time-savings and value, making them more prod speeding time to deployment.	find these
Getting Started with JBoss Developer Studio Getting Started with JBoss Developer Studio.	
Viewlets and Tutorials	
Hibernate Console Viewlets Hibernate Tools Viewlets illustrating various parts of Hibernate Tools.	
JBoss Seam Tutorial JBoss Seam Reference Guide and Tutorial	
WTP J2EE Tutorials Web Tools Platform J2EE Tutorials	
Getting Started Create New Perspectives Web Resources Workbench	

Figure 1.6. Getting Started Documentation

• to create new Seam projects, jBPM Process, JSF or Struts projects using JBDS wizards

C Welcome &	🖞 🗢 🗢 🗗
Welcome to JBoss Developer Studio	
Create New	
Start using JBoss Developer Studio	
Create Seam Project JBoss Developer Studio includes a New Seam Project Wizard that allows you to setup pr featured web application in Seam. You can use any combination of Java EE frameworks Seam.	
Create JBPM Process Project JBoss Developer Studio includes a New jBPM Process Project that allows you to cre using jBPM Framework	eate applications
Create JSF Project JBoss Developer Studio allows you to create brand new JSF projects. A new JSF project v libraries, tag libraries and a JSF configuration file. JBoss Developer Studio comes with a predefined project templates. These templates are flexible and easily customizable.	
Create Struts Project JBoss Developer Studio includes a New Struts Project Wizard that radically simplifies the started with a new Struts project.	e process for getting
Getting Started Create New Perspectives Web Resources Workbench	

Figure 1.7. Create New...

• to get short description of perspectives that JBDS offers for more productive development

□ Welcome 🕴	🚡 🗢 🗢 🗗
Welcome to JBoss Developer Studio	
Available	
JBoss Developer Studio offers a number of perspectives making the development mo	re productive
The Java EE Perspective The Java EE perspective includes workbench views that you can use when developing reso enterprise applications, EJB modules, Web modules, application client modules, and connect modules.	
Seam Perspective The Seam perspective helps you to use all advantages of Seam Framework in your applicat	ions.
Database Development perspective The Database Development perspective allows you to create, modify, and execute databas SQL files.	e objects and
Hibernate Console perspective The Hibernate Console perspective allows you to configure database connections, provides of classes and their relationships and allows you to execute HQL queries interactively agains database and browse the query results.	
Getting Started Create New Perspectives Web Resources Workbench	

Figure 1.8. Perspectives

• to visit JBoss Developer Studio web resources.

□ Welcome 🛛	奋	¢	⇔	- 8
Welcome to JBoss Developer Studio				
Related Links				
JBoss Developer Studio Web Resources				
JBoss Developer Studio Users Forum				
News News on JBoss Developer Studio				
in the second se				
Getting Started Create New Perspectives Web Resources Workbench				

Figure 1.9. Web Resources

Start work with JBoss Developer Studio by clicking on *Workbench* button or simply close the Welcome page.

1.6. Upgrading

To upgrade, just uninstall your current version and install the new one.

1.7. Uninstalling

- Make sure JBoss Developer Studio is not running
- Uninstall your current version of JBoss Developer Studio by running Uninstaller

1.8. Support

If you have comments or questions, you can discuss them at our <u>JBoss Developer Studio Forum</u>.

When writing to the forum for questions, please include the following information:

- 1. JBoss Developer Studio version
- 2. Exact error message
- 3. Steps to reproduce the issue

JBDS subscribers can get necessary support on our Support Portal.

1.9. Other relevant resources on the topic

JBDS on JBoss: JBoss Developer Studio

Forum: JBoss Forum

Subscription: JBDS Subscription

The latest documentation builds are available *here*.

Manage JBoss AS from JBoss Developer Studio

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

JBoss Developer Studio ships with JBoss EAP v.4.2 bundled. When you followed the default installation of JBoss Developer Studio, you should already have a JBoss 4.2 Server installed and defined. To run JBoss AS 4.2 you need JDK 1.5, JDK 6 is not formally supported yet, although you may be able to start the server with it.

2.1. How to Manage the JBoss AS Bundled in JBDS

This section covers the basics of working with the JBoss Server supported directly by JBDS via bundled AS plug-in. To read more about AS plug-in, refer to the <u>Server Manager guide</u>.

2.1.1. Starting JBoss Server

Starting JBoss Server is quite simple. JBoss Developer Studio allows you to control its behaviour 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 on the JBoss Server View or right click server name in this view and select Start. If this view is not open, select Window > Show View > Other > Server > JBoss Server View



Figure 2.1. Starting from Icon

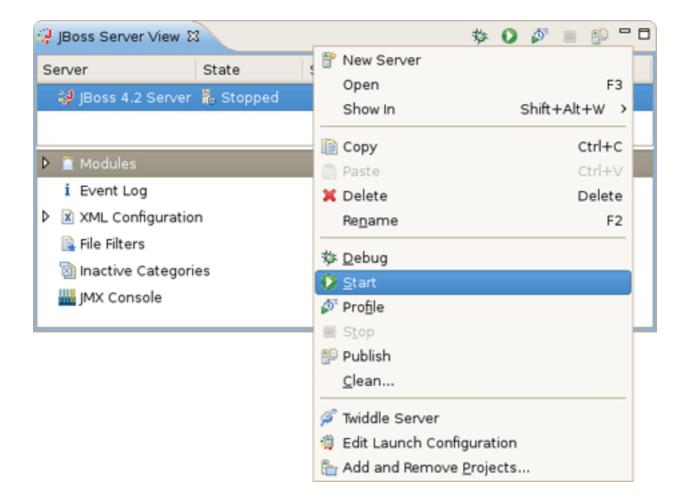


Figure 2.2. Starting from JBoss Server View

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

🖹 Problems 🧔 Tasks	🤪 JBoss Server View 🕒 Console 🕴	- 8
JBoss-server 4.2 [JBoss	Application Server] /usr/java/jdk1.5.0_12/bin/java (Jul 11, 2007 4:31:14 PM)	
🔳 🗶 🔆 🕞 🚮		
16:31:17,324 INF0	[Server] Starting JBoss (MX MicroKernel)	
16:31:17,370 INF0	[Server] Release ID: JBoss [EAP] 4.2.0.GA (build: SVNTag=JBPAPP_4_2_0_G	A d
16:31:17,373 INF0	[Server] Home Dir: /opt/work/rhds_beta/jboss-eap/jboss-as	
16:31:17,373 INF0	[Server] Home URL: file:/opt/work/rhds_beta/jboss-eap/jboss-as/	
16:31:17,377 INF0	[Server] Patch URL: null	
16:31:17,378 INF0	[Server] Server Name: default	
16:31:17,379 INF0	[Server] Server Home Dir: /opt/work/rhds_beta/jboss-eap/jboss-as/server	/de
16:31:17,381 INF0	[Server] Server Home URL: file:/opt/work/rhds_beta/jboss-eap/jboss-as/s	erv
16:31:17,382 INF0	[Server] Server Log Dir: /opt/work/rhds_beta/jboss-eap/jboss-as/server/	def
16:31:17,384 INF0	[Server] Server Temp Dir: /opt/work/rhds_beta/jboss-eap/jboss-as/server	/de
16:31:17,396 INFO	[Server] Root Deployment Filename: jboss-service.xml	
16:31:19,103 INFO	[ServerInfo] Java version: 1.5.0_12, Sun Microsystems Inc.	
16:31:19,107 INFO	[ServerInfo] Java VM: Java HotSpot(TM) Client VM 1.5.0_12-b04, Sun Micro	sys
16:31:19,110 INFO	[ServerInfo] OS-System: Linux 2.6.18-8.el5xen,i386	
16:31:20,559 INFO	[Server] Core system initialized	
16:31:29,515 INF0	[WebService] Using RMI server codebase: http://127.0.0.1:8083/	
16:31:29,534 INF0	[Log4jService\$URLWatchTimerTask] Configuring from URL: resource:jboss-l	og4
16:31:32,045 INF0	[TransactionManagerService] JBossTS Transaction Service (JTA version) -	JB
16:31:32,049 INF0	[TransactionManagerService] Setting up property manager MBean and JMX L	aye
16:31:32,829 INF0	[TransactionManagerService] Starting recovery manager	
16:31:33,323 INFO	[TransactionManagerService] Recovery manager started	
16:31:33,325 INF0	[TransactionManagerService] Binding TransactionManager JNDI Reference	
CI.		
<u>U</u>	III III III III III III III III III II	

Figure 2.3. Console Output

When the server is started you should see *Started* right to its name in JBoss Server View (column "Status").

🥥 JBoss Server View Σ	3		蓉	0	ø	ŧP	- 0
Server	State	Status					
🤯 JBoss 4.2 Server	🖡 Started	Synchronized					

Figure 2.4. Server is Started

To see event log after the server is started, expand Event Log branch beneath JBoss Server View:

🤪 JBoss Server View Σ	3	蓉	0	ø	P	- 0
Server	State	Status				
🤯 JBoss 4.2 Server	🖡 Started	Synchroniz	ed			
👂 🛅 Modules						
▽ i Event Log						
🕨 🖡 Starting the S	erver					
👂 👈 Publishing [3 r	nodules, 51 res	ources chan	ged]			
👂 🖹 XML Configuratio	n					
🗟 File Filters						
🖄 Inactive Categori	es					
▷ 🏭 JMX Console						

Figure 2.5. Event Log

2.1.2. Stopping JBoss Server

To stop the server, click the *Stop* icon in JBoss Server view or right click the server name and press *Stop*.

😲 JBoss Server View 🛿	参	0	× • •

Figure 2.6. Stopping Server

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

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.

How to Use Your Own JBoss AS Instance with

JBDS

😲 JBoss 4.2 Server 🛿		
🗄 Overview		
General Information		- Dublishing
General Internation	nd other common settings.	 Publishing Modify settings for publishing.
Server name:	IBoss 4.2 Server	 Never publish automatically
Host name:	localhost	· · · · · · · · · · · · · · · · · · ·
		 Automatically publish when resources change
<u>Runtime Environment:</u>	JBoss 4.2 Runtime	Publishing interval (in seconds):
Open launch configurati	ion	Timeouts
- Deployment		Specify the time limit to complete server operations.
The temporary deployme	ent location is where files will g renamed into the deploy directory. the same file system	Start (in seconds): Stop (in seconds):
Deploy Directory	/home/ochikvina/Desktop/jboss Browse	- Server Polling
Temporary Deploy Direct	ory /home/ochikvina/Desktop/jboss Browse	Startup Poller JMX Poller 😂
- Login Credentials		Shutdown Poller Process Terminated Poller 💲
Set the login and passwo This will ensure it starts a		- Server Ports
User Name Password		The ports entered here are which ports the tools will poll the server on. Changing these fields will not change the ports the server itself listens on. JNDI Port: 1099 ☑ Automatically detect Po Web Port: 8080 ☑ Automatically detect Po
Overview		

Figure 2.7. Server Overview

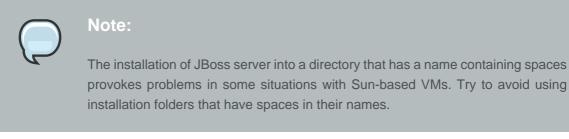
Here you can specify some common settings: host name, server name, runtime as well as settings related to the publishing, timeouts and server ports.

2.2. How to Use Your Own JBoss AS Instance with JBDS

Although JBoss Developer Studio works closely with JBoss EAP 4.2 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 4.2.1 server. First of all you need to install it.

2.2.1. JBoss AS Installation

• Download the binary package of JBoss 4.2.1 and save it on your computer: <u>http://</u> labs.jboss.com/jbossas/downloads It does not matter where on your system you install JBoss server.



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-4.2.1.zip archive contents into a location of your choice. The jboss-4.2.1.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-4.2.1 directory.

2.2.2. Adding and configuring JBoss Server

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

- Open the JBoss Server View by selecting Window > Show View > Other > Server > JBoss Server View. You will see JBoss Server view.
- Right click anywhere in this view and select New Server.
- Select JBoss, a division of Red Hat > JBoss v4.2 and click the Installed Runtimes button to select a new installed runtime.

9	New Ser	ver		×
Define a New Server				
Choose the type of server to create				
Server's <u>h</u> ost name: localhost				•
		Downle	oad additional s	erver adapters
Select the server type:				
type filter text				
JBoss AS 3.2				
JBoss AS 4.0				
🤯 JBoss AS 4.2				
🤪 JBoss System Copy Server				=
👂 🗁 Macromedia				
👂 🗁 Mort Bay				
👂 🗁 ObjectWeb				
JBoss Application Server 4.2				
Server runtime: JBoss 4.2 Runtime			Install	ed Runtimes
(<u> </u>				
0	< <u>B</u> ack	<u>N</u> ext >	Enish	Cancel

Figure 2.8. Selecting Server Type

- Click Add button to add a new JBoss runtime.
- Select JBoss, a division of Red Hat > JBoss v4.2 and press Next.

9	Pre	ferences (Filte	red)		×
type filter text	Installed Ser	ver Runtime En	vironments		⇔* ⇔*
▽ Server	Add, remove, o	r edit installed se	rver runtime defi	nitions.	
Installed Runtimes	Installed server	runtimes:			
	Name		Туре		<u>A</u> dd
	🤯 JBoss 4.2 Se	erver Adapter Rur	JBoss 4.2 Runtim	ne	Edit
					<u>R</u> emove
					Search
•	/opt/work/rhds1	1/jboss-eap/jboss	-as		
Ø				ОК	Cancel

Figure 2.9. Installed Runtimes

• In the next step make JBoss Developer Studio to know where you have installed the server and define JRE.

9	New	Server Runt	ime	×
Create a new JBo	oss Server F	Runtime	1	JBoss 4.2
A JBoss Server run It can be used to so as well as by a "se JBoss.	et up classpat	hs for project	s which depend o	on this runtime,
Name (unique iden	tifier)			
JBoss-runtime				
Home Directory				
/opt/work/jboss-a	s 4.2.1			Browse
JRE				
jdk1.5.0_12			\$	Installed JREs
Configuration				
🧃 default				
i all				•
0	< <u>B</u> ack	<u>N</u> ext >	Einish	Cancel

Figure 2.10. Defining JBoss Runtime



• In the following window leave all settings default or give your name to a new jboss server and press *Finish*.

New Server Runtime
Create a new JBoss Server
Add a JBoss Server and have great fun
A JBoss Server manages starting and stopping instances of JBoss. It manages command line arguments and keeps track of which modules have been deployed
If the information below is incorrect, please press back, Installed Runtimes, and then Add to create a new runtime from a different location.
Name (unique identifier)
JBoss-server 4.2.1
Runtime Information
Home Directory
/opt/work/jboss-as 4.2.1
JRE jdk1.5.0_12
Configuration
default
(?) < <u>Back</u> <u>Next</u> > <u>Finish</u> Cancel

Figure 2.11. Adding New Runtime

A new runtime will now appear in the *Preferences > Server > Installed Runtimes* dialog.

9	Preferences (Fil	tered)	×
type filter text	Installed Server Runtime I	Invironments	() = () =
	Add, remove, or edit installed	server runtime definitions.	
Installed Runtimes	Installed server runtimes:		
	Name	Type	<u>A</u> dd
	JBoss 4.2 Server Adapter R	-	Edit
	🤯 JBoss-runtime	JBoss 4.2 Runtime	Bemove
			Bemete
			<u>S</u> earch
	/opt/work/jboss-as 4.2.1		
Ø		ОК	Cancel

Figure 2.12. Runtime is Added

• Click *OK*. Then select a new added runtime in Server runtime drop down list and click *Next* button twice.

9	New Server	×
Define a New S	Server	
Choose the type	of server to create	
Server's <u>h</u> ost na	me: localhost	•
	Downle	oad additional server adapters
Select the server	r type:	
type filter text		
Boss		•
	ivision of Red Hat	
🤪 JBoss A	IS 3.2	
😲 JBoss A	S 4.0	
🤯 JBoss A	S 4.2	-
🖓 JBoss S	ystem Copy Server	.
JBoss Application	Server 4.2	
Server <u>r</u> untime:	JBoss 4.2 Server Adapter Runtime	Installed Runtimes
-	JBoss-runtime	·
0	< Back Next >	Einish Cancel

Figure 2.13. Choosing Runtime

• In the next dialog verify a JBoss runtime information and if something is unfair go back and correct it.

New Server	
reate a new JBoss Server	5S 4.2
A JBoss Server manages starting and stopping instances of JBoss. t manages command line arguments and keeps track of which modules have been deploy Name	/ed.
JBoss 4.2 Server	٦
Runtime Information	
If the runtime information below is incorrect, please press back, Installed Runtimes, and then Add to create a new runtime from a different location. Home Directory /opt/work/jboss-4.2.0.GA JRE /usr/java/jdk1.5.0_12 (jdk1.5.0_12)	
Configuration default	
Login Credentials JMX Console Access User Name Password	
Deployment Deploy Directory /opt/work/jboss-4.2.0.GA/server/default/deploy Browse	
⑦	

Figure 2.14. Configuring Projects

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

9	New Server		×
Add and Remove Projects Modify the projects that are config	gured on the server		
Move projects to the right to confi	gure them on the server		
<u>Available projects:</u>		Configured projects:	
🐻 JSFKickStart			
	A <u>d</u> d >		
	< <u>R</u> emove		
	Add All >>		
	<< Remove All		
	<< Remove All		
0	De Back	- Enlah	Canaal
Ø	< Back Next >	Einish	Cancel

Figure 2.15. Configuring Projects

A new JBoss server should now appear in JBoss Server View.

🙆 Tasks 觽 Data Source E	xplorer 📮 Cons	ole 🤪 JBoss Server View 🛿	莽	0	ø	8	<u>ا</u> م
Server	State	Status					
🚆 JBOSS 4.2.1	🚡 Stopped	Synchronized					
JBoss-server 4.2.0	指 Stopped	Synchronized					

Figure 2.16. 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 the JBoss Developer Studio can be used during the development process.

3.1. Create a Seam Application

In this section you get to know how to create a Seam project in JBDS, how to start the server and what a structure your project has after creating.

3.1.1. Start Development Database

Before opening the JBoss Developer studio you need to <u>download</u> and start a Workshop Database.

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	
<u>File Edit View T</u> e	erminal Ta <u>b</u> s <u>H</u> elp	
<pre>[Server@la758cb]: [Server@la758cb]: [Server@la758cb]: perties] [Server@la758cb]: [Server@la758cb]: [Server@la758cb]: ssfully in 930 ms [Server@la758cb]: [Server@la758cb]: [Server@la758cb]:</pre>	<pre>[Thread[main,5,main]]: checkRunning(false) entered [Thread[main,5,main]]: checkRunning(false) exited Startup sequence initiated from main() method Loaded properties from [/opt/work/database/database/server.p Initiating startup sequence Server socket opened successfully in 19 ms. Database [index=0, id=0, db=file:employee, alias=] opened successfully in 19 ms.</pre>	

Figure 3.1. Starting DataBase

3.1.2. Create and deploy Seam Web Project

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

First of all you get the Workspace Launcher. Change the default workspace location if it's needed. Click on Ok.

9	Workspace Launcher			
Select a workspace				
JBoss Develo workspace.	oper Studio stores your projects in a folder called a			
<u>W</u> orkspace:	/root/workspace	▼ <u>B</u> rowse		
Use this a	as 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 directory 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 needed datasource and driver libraries need to go.

New Seam Project	×
Seam Web Project	and a
Create standalone Seam Web Project	E
Project name: workshop	
Project contents:	
☑ Use <u>d</u> efault	
Directory: //home/ochikvina/java/eclipse-tools/eclipse/workspace3/work	s Browse
Target Runtime	
JBoss 4.2 Runtime	≎ N <u>e</u> w
Dynamic Web Module version	
2.5	0
Target Server	
JBoss 4.2 Server	0 New
Configuration	
Dynamic Web Project with Seam 1.2	C Modify
Configures a Dynamic Web application to use Seam v1.2	v Produly
conliguies a Dynamic web application to use Seam V1.2	
(?) < Back Next > Finish	Cancel
C Dack Mext > Built	Cancer

Figure 3.3. New Seam Project Wizard

All settings are already specified here, you can just modify the Configuration. Chose needed one from the list of already predefined configurations or click on the *Modify...* to configure your custom facets pattern.

On the hole appeared dialog allows to select the "features" you want to use in your project. Doing this JBoss Developer Studio setups 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.

Select Facets Select the facets that should be enal		
onfigurations: Dynamic Web Projec	with Coord 1.2	
	t with Seam 1.2	\$ Save As Delete
Project Facet	Version	Details Runtimes
🖸 📄 Axis2 Web Services		Axis2 Web Services
🗹 👼 Dynamic Web Module	2.5	Enables Web services generation
🗹 🔬 Java	5.0	through the Axis2 Web services engine.
🗌 🚸 Java Persistence	1.0	
🗌 📄 JavaScript Toolkit	1.0	
🗹 📄 JavaServer Faces	1.2	
🗌 📄 JBoss ESB	4.3	
🗌 📄 JBoss Portlets		
🗌 📄 JBoss Web Services		
🗹 📄 Seam	1.2	
🗌 📄 WebDoclet (XDoclet)	1.2.3	
		>
0		OK Cancel

Figure 3.4. Project Facets Specifying

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

9	Project Face	ets X
Project Facets Select the facets that should be ena	abled for this project.	
Configurations: Dynamic Web Proje	ct with Seam 1.2	Save As Delete
Project Facet	Version	Details Runtimes
🕨 🔲 🖥 Axis2 Web Services		JBoss 4.2 Runtime
🗹 💑 Dynamic Web Module	2.5	
🗹 🛃 Java	5.0	
🗌 ↔ Java Persistence	1.0	
🗌 📄 JavaScript Toolkit	1.0	
🗹 📄 JavaServer Faces	1.2	
🔲 📄 JBoss ESB	4.3	
IBoss Portlets		
IBoss Web Services		
🗹 📄 Seam	1.2	
🗌 📄 WebDoclet (XDoclet)	1.2.3	
		Show all runtimes
		Make Primary New
		Runtime Composition:
		<no runtime="" selected=""></no>
0		OK Cancel

Figure 3.5. Runtimes Selecting

Click on Ok and then Next to proceed further.

A dynamic web application contains both web pages and Java code. The wizard will ask you where you want to put those files. You can just leave the default values or choose another folder.

)	New Seam Project	
Veb Module		
Configure web module setting	s.	
Context Root:		
workshop		
Content Directory:		
WebContent		
ava Source Directory:		
src		
🗹 Generate Deployment Des	criptor	
Generate Deployment Desc	criptor	

Figure 3.6. 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.

- Check Server Supplied JSF Implementation. We will use <u>JSF implementation</u> that comes with JBoss server
- Click Next

😣 New Seam Project 🗙			
JSF Capabilities Add JSF capabilities to this Web Project			
JSF Libraries:	Server Supplied JSF Implementation		
	Component Libraries Deploy Library Name >>		
JSF Configuration File:	/WEB-INF/faces-config.xml		
JSF <u>S</u> ervlet Name:	Faces Servlet		
JSF Servlet Classname:	javax.faces.webapp.FacesServlet		
<u>U</u> RL Mapping Patterns:	*.seam		
0	< Back Next > Finish Cancel		

Figure 3.7. 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	X
Seam Facet		and a
Configure Seam l	Facet Settings	
l		
General		
Seam Runtime:	Seam 1.2.AP	\$ Add
Deploy as:		

Figure 3.8. Seam Facet Setting

Next 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 Edit to modify the Connection Profile.

Database	
Database Type:	HSQL \$
Connection profile:	Default 🗘 Edit New
Database Schema Name:	
Database Catalog Name:	
DB Tables already exists in database:	
Recreate database tables and data on deploy:	

Figure 3.9. DataBase Setting

Select JDBC Connection Properties. Make sure the URL is set to *jdbc:hsqldb:hsql:// localhost:1701*

Properties for			
type filter text	JDBC Conne	ection Properties	⇔* ⇔*
Common Default Schema Filter	Select a driv Hypersonic	er from the drop-down:	¢
Default Stored Procedure Filter Default Table Filter	Database:	SAMPLE	
JDBC Connection Properties	URL:	jdbc:hsqldb:hsql://localhost:1701	
Version	User name:	sa	
	Password:		
	Save Pas	sword	

Figure 3.10. JDBC Connection Properties

Try click on Test Connection button. It probably won't work. This happens if the hsql jdbc driver is not exactly the same. This can be worked around by modifying the HSQLDB database driver settings. To modify the settings, click the "..." next to the drop-down box.

The proper Driver JAR File should be listed under Driver File(s). Select the hsqldb.jar file found in the database/lib directory and click on Ok.

😣 Edit Driver Definit	ion	×	
Provide Driver Details			
Modify details in the fields below to provide a unique nan set any available and applicable property values.	ne, a list of required	jars, and	
Driver Name			
Hypersonic DB			
Driver Type:			
Hypersonic DB			
Driver File(s):			
/opt/work/jbds/jboss-eap/jboss-as/server/default/lib/hsq	idb.jar	Add Jar/Zip	
		Edit Jar/Zip	
		Remove Jar/Zip	
		Clear All	
Properties:		1	
Property	Value		
▼ General			
Connection URL	jdbc:hsqldb:MyD	В	
Database Name	SAMPLE		
User ID	sa		
(III		Þ	
0	ОК	Cancel	

Figure 3.11. Driver Details

Select Hypersonic DB and click on Ok. Again, this only happens if the selected hsqldb.jar is different from the running database.

Oriver Definitions		×
Available Driver Definitions		
Add, edit, or remove driver definitions to manage available driver definitions.		
🗢 🖼 Derby	•	Add
G 10.0		Edit
G 10.1		
G 10.2		Remove
👒 Generic JDBC Driver		Сору
▽ 👒 Hypersonic DB		
▽ 🕞 1.8		
Hypersonic DB		
🗢 🖼 Informix		
G 10.0	=	
G 9.2		
G 9.3		
G 9.4		
🗢 🖼 MySQL		
G 4.0		
G 4.1		
G 5.0		
G 5.1		
🗢 🖼 Oracle		
G 10	•	
0 ок		Cancel
U OK		Cancer

Figure 3.12. Hypersonic DB Selecting

Now, the Test Connection should succeed. After testing the connection, click on Ok.

@	Propert	des for X
type filter text	JDBC Conne	ection Properties $(r + c) +$
Common Default Schema Filter		er from the drop-down:
Default Stored Procedure Filter	Hypersonic	DB 🔷 🗘
Default Table Filter	Database:	SAMPLE
JDBC Connection Properties	URL:	jdbc:hsqldb:MyDB
Version	User name:	sa
	Password:	
	Save Pas	sword
3	S	uccess
Ping succee	ded!	dd Jp
		OK pwn
		Clear All
		Test Connection
0		OK Cancel

Figure 3.13. 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						
Test Package Name:	rg.domain.workshop.test						
0	< Back Next > Finish Cancel						

Figure 3.14. Code Generation Setting

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

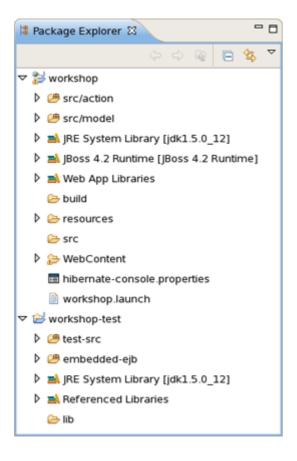


Figure 3.15. "worskhop" Project in the Package Explorer

3.1.3. Start JBoss Application Server

The complete information on how to manage JBoss AS from JBoss Developer Studio you can read in a *corresponding chapter*.

Now you just need to start the server by clicking on green circle/triangle icon in the JBoss Server View.

State		
state	Status	Start the server
Stopped	Synchronized	
	= Stopped	Stopped Synchronized

Figure 3.16. Starting the Server

Then run the project by selecting the project and use *Run As... > Run on Server*.

\bigcirc	Note:
	If the project does not show up, then you can use a normal browser and use <i>http://localhost:8080/workshop/home.seam</i> as the url.

Your project looks like this:

Se workshop ∞	- 0
💠 👄 😹 🚸 http://localhost:8080/workshop/home.seam	- 🕨 🕨
workshop: Home	Login
Welcome!	
This empty shell application includes: Ant build script Deployment to JBoss AS Integration testing using TestNG and JBoss Embeddable EJB3 EJB 3.0 Seam components Templated Facelets views HSQL (or MySQL) Datasource Default CSS stylesheet Internationalization support	
Powered by <u>Seam</u> . Generated by seam-gen.	

Figure 3.17. "workshop" Project Started

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 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.

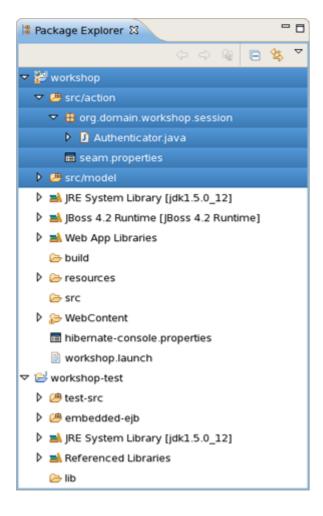


Figure 3.18. Project Structure

It's important to note that the business logic will reside in the *src/action* folder, by default. And, the package naming conventions that were used in Seam project setup 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 a bit more detail later on. The *src/model* folder will store the project's JPA entity beans. The view tier of the application is also important. Seam uses facelets and there is a built-in facelets GUI editor that has some nice WYSIWYG and component drag/drop functionality. Try this out by opening home.xhtml from WebContent folder.

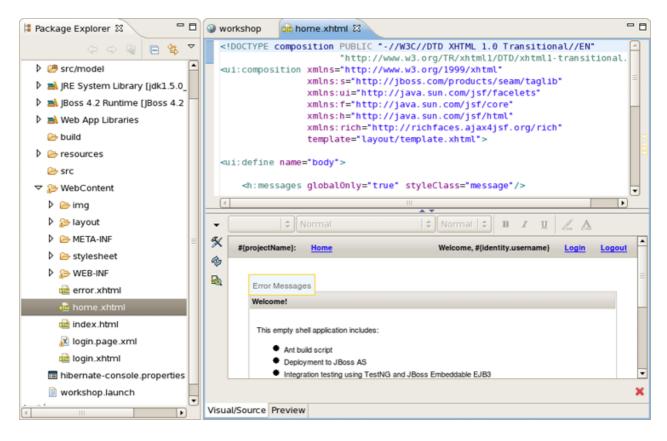


Figure 3.19. Facelets GUI Editor

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 per the Seam project wizard database settings. And, obviously all of the Seam specific configuration files and JAR dependencies are included and placed in the 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/deployment process. The end result is a developer that is writing code, not spending days/weeks 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 > New > 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 Finish.

9	New Seam Action
Seam Action Create a new Seam action	
Seam Project:	workshop Browse Settings
Seam component name:	myAction
POJO class name:	MyAction
Bean name:	MyActionBean
Method name:	myAction
Page name:	myAction
Package name:	org.domain.workshop.session Browse
0	Einish Cancel

Figure 3.20. 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");
facesMessages.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.

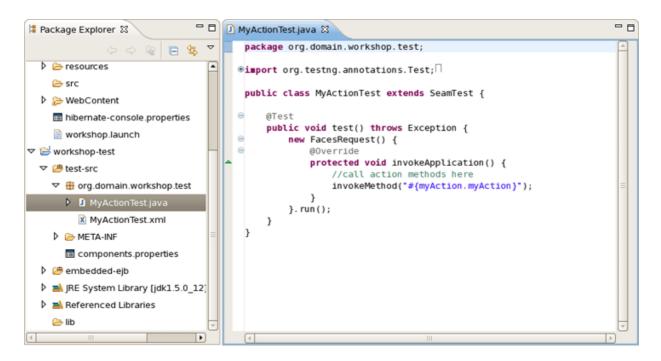


Figure 3.21. "workshop-test" Project

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

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

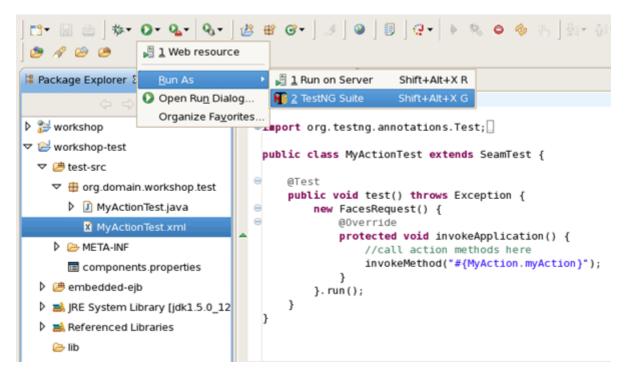


Figure 3.22. TestNG Running

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

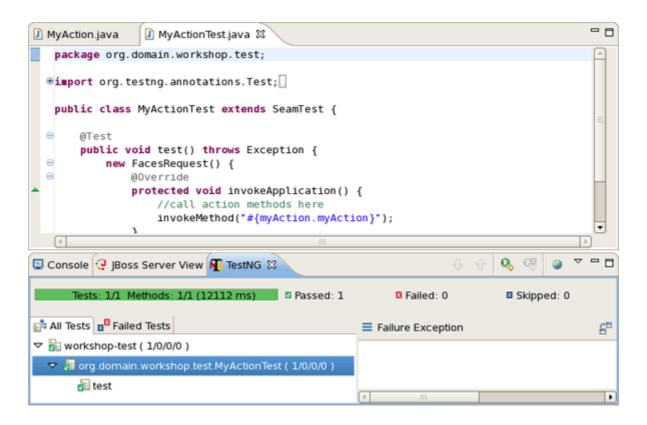


Figure 3.23. TestNG Results

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... > 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.

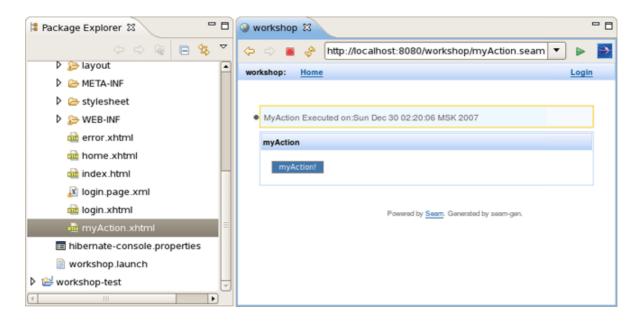


Figure 3.24. Seam Action in a Web Browser

3.2.3. Modify Seam Action User Interface

Browse to *http://localhost:8080/workshop/myAction.seam* and click on myAction. 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.

📾 myAction.xhtml 😫	•
<pre><ui:composition ht<br="" xmlns="http
xmlns:s=">xmlns:ui="h xmlns:f="ht xmlns:h="ht xmlns:rich= xmlns:a="ht</ui:composition></pre>	<pre>CC *-//W3C//DTD XHTML 1.0 Transitional//EN* "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.c ://www.w3.org/1999/xhtml* :ttp://java.sun.com/jsf/facelets* :ttp://java.sun.com/jsf/facelets* :ttp://java.sun.com/jsf/html* ="http://richfaces.ajax4jsf.org/rich" :ttps://ajax4jsf.dev.java.net/ajax" .ayout/template.xhtml"> </pre>
▼ \$ Normal	⇒ Normal ⇒ B I U / A
	Welcome, #{identity.username} Logout
Error Messages myAction	
myAction!	
	ا د

Figure 3.25. Open Seam Action with Editor

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

-	¢ Nor	mal	🗘 Nor	mal 🕴	в	I	<u>u</u>	Ζ Δ			
%	#{projectName}:	lome		W	Velcome,	#{ider	ntity.u	username}	Login	Logout	ŕ
à	Error Messages										=
	myAction										
	myAction	<pre>P </pre> A:commandButton> A									_
		Parent Tag	,								-
	mposition ui:defin	Insert Around Insert Before	•	ndBut	ton						×
visua	Il/Source Preview	Insert After	•								
		Cut Copy Paste									
		Re <u>m</u> ove									

Figure 3.26. 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.

•	Attributes	×
h:commandB	utton>	
name	value	•
rendered		
style		
styleClass		
tabindex		
title		
type		=
value	Click on My Action Button	
		Close

Figure 3.27. 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 🕴		•	٥
#{projectName}: <u>Home</u>	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 3.28. Seam Action Is Modified

3.3. Declarative Security

In this section you will see how it's easy 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 adding some custom login logic.

Open Authenticator.java in JBoss Developer Studio and replace the authenticate() method with this code:

```
public boolean authenticate() {
  if (identity.getUsername().equals("admin")
  && identity.getPassword().equals("password")) {
    identity.addRole("admin");
    return true;
  }
  else
  return true;
}
```

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.

] A	uthe	ntica	ator.ja	iva	📾 myAct	on.xhtml	🎯 worksho	p 83		1	- 0
¢			Ŷ	http://	localhost:8	080/worksh	op/myAction.	seam	•	►	÷
wor	kshop	:	Home						!	Login	
	myA	ction	1								
	(Click	on My	Action B	utton						
						Powered by Sear	. Generated by sea	m-gen.			

Figure 3.29. One Button on a Page

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

Authenticator.java	amyAction.xhtml	🥹 workshop 🕄		- 8
🗢 🔿 🔳 🦑 http:/	/localhost:8080/worksh	op/myAction.seam	•	۵ 🔺
workshop: <u>Home</u>			Welcome, admin	Logout
myAction				
Click on My Action 8	Button Secured Action Bu	utton		
	Powered by Sean	n. Generated by seam-gen.		

Figure 3.30. 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.

Authenticator.java	amyAction.xhtml	🛞 workshop 🛿 👜 pages.xn	nt 🗖
🗢 🔿 🔳 🖑 http://	localhost:8080/worksh	op/login.seam?cid=62&clr=true	🔹 🕨 💽
workshop: Home			Login
Please log in first			
Login			
Please login using any us	sername and password		
Username			
Password]	
Remember me	<u> </u>		
Login			
	Powered by Seam	. Generated by seam-gen.	

Figure 3.31. 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 > Open Perspective > Other > Database Development.*

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

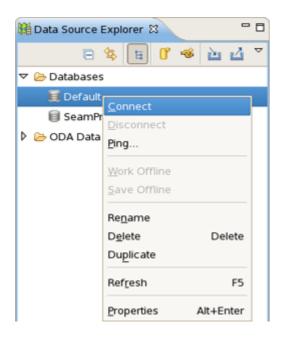


Figure 3.32. Data Source Explorer

3.4.2. Browse Workshop Database

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

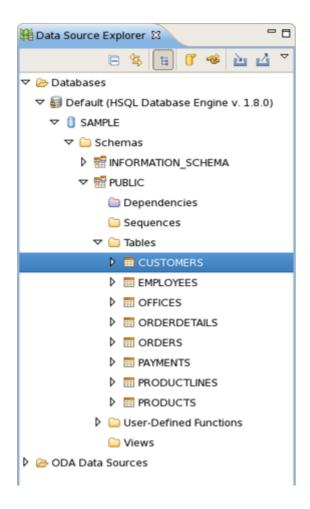


Figure 3.33. "CUSTOMERS" Table

Right click on CUSTOMERS, select *Data > 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.

Type quer	y expression h	nere	Statu	us Result1						
Status	Operation	Date		CUSTOMER	CUSTOMERNAME	CONTACTLAST	CONTACT	PHONE	ADDRESSLINE1	Α
🖌 Suce	ce	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 Alsace-Lo	
			з	206	Asian Shopping N	Walker	Brydey	+612 %	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	Berguvsv\u005	i.
			6	189	Clover Collections	Cassidy	Dean	+353 1;	25 Maiden Lane	F
			7	141	Euro+ Shopping C	Freyre	Diego	(91) 55	C/ Moralzarzal,	
			8	216	Enaco Distributors	Saavedra	Eduardo	(93) 20	Rambla de Cata	ł
			9	201	UK Collectables, L	Devon	Elizabeth	(171) 5?	12, Berkeley Ga	
			10	148	Dragon Souvenie	Natividad	Eric	+65 22	Bronz Sok.	В
			11	209	Mini Caravy	Citeaux	Fr\u005cul	88.60.1	24, place Kl\u00	1
			12	240	giftsbymail.co.uk	Bennett	Helen	(198) 53	Garden House	c
			13	223	Nat\u005cu005cu	Kloss	Horst	0372-5	Taucherstra\u00	;
			14	169	Porto Imports Co.	de Castro	Isabel	(1) 356	Estrada da sa\u	
			15	119	La Rochelle Gifts	Labrune	Janine	40.67.8	67, rue des Cin	L
			16	112	Signal Gift Stores	King	Jean	702555	8489 Strong St.	
										Þ

Figure 3.34. SQL Results View



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 Seam perspective and click on *File > New > Seam Generate Entities.*

In the Seam Generate Entities wizard, select "workshop" project using the Browse button. There is no need to change something more, click on Finish.

Seneral General	te Seam Entities 🛛 🗙
Generate Seam Entities	
Seam Project: Hibernate Console Configuration Generation Mode	
0	<u>F</u> inish Cancel

Figure 3.35. Generate Seam Entities

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

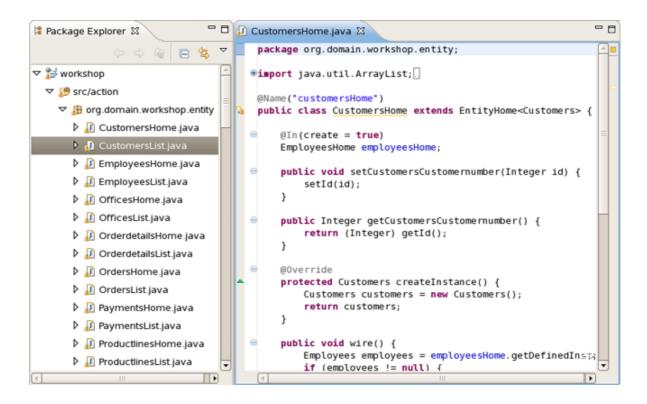


Figure 3.36. org.domain.workshop.entity Classes

There are also *org.domain.workshop.entity* package that contains the JPA classes. These are the entity beans that are mapped to database tables.

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 which we will explore further later on in the lab. For now, take note of the page tabs, required field logic and data table sorting in the list pages.

CustomersList.xht	mi 83				-
{projectName}: H	ome Offices List	Customers List	Employees List	Products List	Orderde
Error Messages	parameters				æ
	#{customersl	ist customer			
customername					
contactlastnam	#{customersl	ist.customer			
contactfirstnam	#{customersl	ist.customer			
phone	#{customersl	.ist.customer			
addressline1	#{customersl	ist.customer			
addressline2	#{customersl	ist.customer			
city	#{customersl	ist.customer			
state	#{customersl	ist.customer			
postalcode	#{customersl	ist.customer			
country	#{customersl	ist.customer			
Search					
Customers search	results				
No customers exis	sts				
III)

Figure 3.37. CustomersList.xhtml in the Editor

Congratulations! You now have a fully functioning CRUD application that is already 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 choose Hibernate.

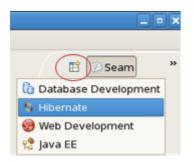


Figure 3.38. Hibernate Perspective

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.

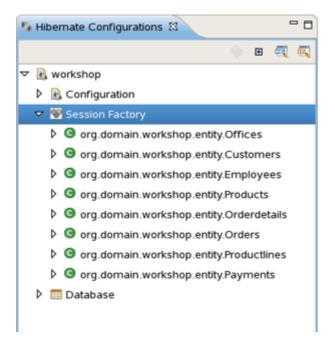


Figure 3.39. Hibernate Configurations View

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.

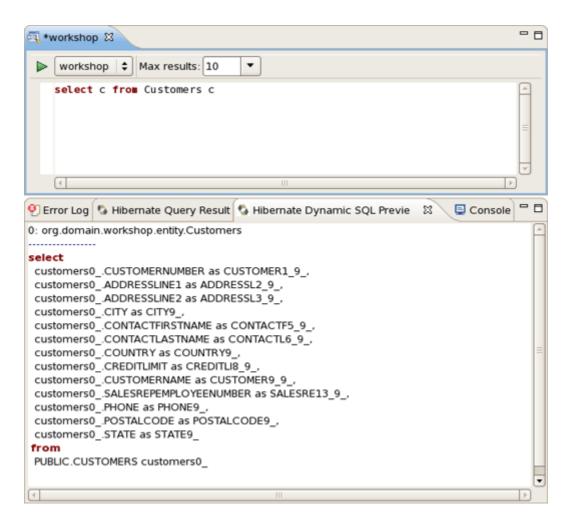


Figure 3.40. JPA Query Editor

Run the query by clicking on the green run icon.

The results are listed in the "Hibernate Query Result" tab. 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.

🔲 Pr	roperties 🛿		*workshop 🛙	- 8
Prop	erty	Value	▶ workshop 🗘 Max results: 10 💌	
⊽ lo	dentifier		select c from Customers c	A
	customernumbe	103		
▽ P	roperties			=
	addresslinel	54, rue Royale		*
	addressline2			Þ
	city	Nantes	Error Log 🥵 Hibernate Query Result 🕱 💊 Hibernate Dyn	namic SQL Pr 📮 Console 🗖 🗖
	contactfirstnam€	Carine		Xž
	contactlastname	Schmitt		
	country	France		-
	creditlimit	21000.0	rg.domain.workshop.entity.Customers@c9fc25	=
	customername	Atelier graphique	rg.domain.workshop.entity.Customers@841423	
Þ	employees		rg.domain.workshop.entity.Customers@1463027	
⊳	orderses		rg.domain.workshop.entity.Customers@1038048	
⊳	paymentses		rg.domain.workshop.entity.Customers@db0d07	
	phone	40.32.2555	rg.domain.workshop.entity.Customers@134817a	
	postalcode	44000	rg.domain.workshop.entity.Customers@207655	
	state		rg.domain.workshop.entity.Customers@138d8fa	
			rg.domain.workshop.entity.Customers@12d2db8	
			rg.domain.workshop.entity.Customers@11499ff	•
1	III		lect c from Custom select c from Custom select c from Cus	stom 🕱 "9

Figure 3.41. Hibernate Query Result View

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

्य *workshop ध	- 8
<pre>workshop \$ Max results: 10 \$ select c.customername, c.employees. from Customers c</pre>	
 Setect C. Customer Hame, C. emptoyees. Customerses - Employees email - Employees employeenumber - Employees extension - Employees firstname - Employees jobtitle - Employees jobtitle - Employees lastname - Employees officecode - Employees officecode - Employees reportsto - Employees reportsto - Employees 	
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	V

Figure 3.42. Code Completion

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

🕨 workshop 😫 Max r	esults: 10 💌	
select c.customern	aame, c.employees.extension from Customers	c
•		>
Error Log 😘 Hibernate (Query Result 🖾 💊 Hibernate Dynamic SQL Prev	🖳 Console 🗖
		x 8
		•••
0	1	
Atelier graphique	x2028	
Signal Gift Stores	x4065	
Australian Collectors, Co.	x101	
La Rochelle Gifts	x2028	
La Rochelle Gifts		
Baane Mini Imports	x102	
Baane Mini Imports		
Baane Mini Imports Mini Gifts Distributors Ltd.	x3291	
Baane Mini Imports Mini Gifts Distributors Ltd. Blauer See Auto, Co.	x3291 x102	

Figure 3.43. The Results

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.

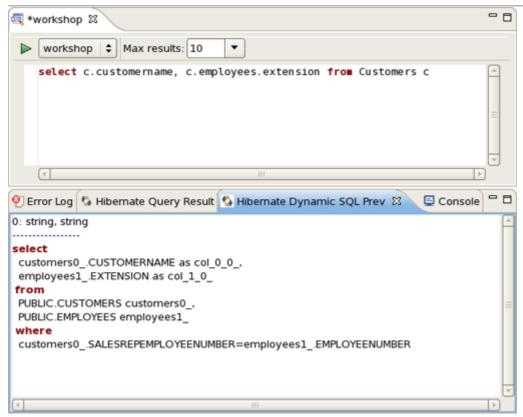


Figure 3.44. Criteria Editor

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 Open Mapping Diagram.

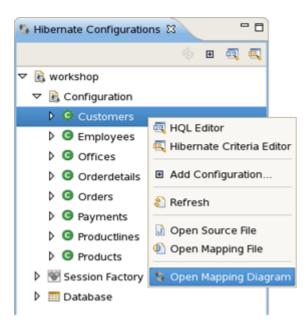


Figure 3.45. Mapping Diagram Opening

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.

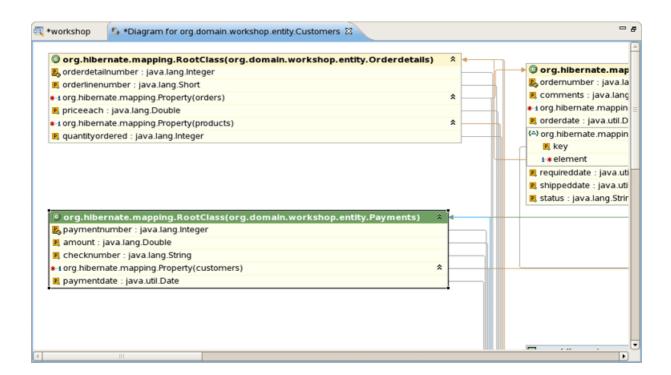


Figure 3.46. Diagram Tab

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, 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.

or 📾	derdetailsEdit.xhtml 🛛			- 0			
		'ext>		-			
	4 51 4 6 6 7 4 6 6 7						
			ecoration" template="layout/edit.xhtml">	= =			
		AVW0V0	rlinenumber	H			
<pre><h:inputtext <="" id="orderlinenumber" pre="" required="true"></h:inputtext></pre>							
	Tec		ilsHome.instance.orderlinenumber}">				
	<a: sup<="" th=""><th></th><th><pre>" reRender="orderlinenumberDecoration"/></pre></th><th></th></a:>		<pre>" reRender="orderlinenumberDecoration"/></pre>				
	<td>Text></td> <td></td> <td></td>	Text>					

Figure 3.47. 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.

	1	<s:decorate <ui:de< th=""><th><pre>required="true" disabled="#{orderdet value="#{orderdet support event="onblu putText> te> e id="quantityordered fine name="label">gua utText id="quantityor required="true" value="#{orderdet</pre></th><th>ailsHome.inst r" reRender=" Decoration" t ntityordered< dered"</th><th>ance.ord orderdet emplate: /ui:def:</th><th>ailnumber "layout/e ne></th><th>Decorati dit.xhtm</th><th>on*/></th><th><rich: datascroller=""> <rich: dndparam=""> <rich: dragindicator=""> <rich: draglistener=""> <rich: dragsupport=""> <rich: dropdownmenu=""> <rich: droplistener=""> <rich: dropsupport=""> <rich: effect=""></rich:></rich:></rich:></rich:></rich:></rich:></rich:></rich:></rich:></th></ui:de<></s:decorate 	<pre>required="true" disabled="#{orderdet value="#{orderdet support event="onblu putText> te> e id="quantityordered fine name="label">gua utText id="quantityor required="true" value="#{orderdet</pre>	ailsHome.inst r" reRender=" Decoration" t ntityordered< dered"	ance.ord orderdet emplate: /ui:def:	ailnumber "layout/e ne>	Decorati dit.xhtm	on*/>	<rich: datascroller=""> <rich: dndparam=""> <rich: dragindicator=""> <rich: draglistener=""> <rich: dragsupport=""> <rich: dropdownmenu=""> <rich: droplistener=""> <rich: dropsupport=""> <rich: effect=""></rich:></rich:></rich:></rich:></rich:></rich:></rich:></rich:></rich:>
-		🗘 Normal	\$ No	mai 🗘 🖪 🕽	ζ <u>υ</u>	<u>.</u> A			<rich:gmap> <rich:inputnumberslider></rich:inputnumberslider></rich:gmap>
X		Enor wessages	~						<rich:inputnumberspinner></rich:inputnumberspinner>
a,		Edit Orderdetails							<rich:insert></rich:insert>
			(# forderdetaile i on a locked		1				<rich:jquery></rich:jquery>
		orderdetailnumber*	#{orderdetailsHome.instar,	Error Messages					<rich:listshuttle></rich:listshuttle>
		quantityordered *	#{orderdetailsHome.instar	Error Messages					<rich:menugroup></rich:menugroup>
			<s:decorate> Attributes</s:decorate>						<rich:menultem></rich:menultem>
		priceeach*	Parent Tag 🔹 🕨	Error Messages					<rich:menuseparator></rich:menuseparator>
		orderlinenumber*	Insert Around	Error Messages					<rich:message></rich:message>
		* required fields	Insert Before						<rich:messages></rich:messages>
			Insert After	From Palette	• IBoss		Ajax4jsf		<rich:modalpanel></rich:modalpanel>
		Save Save [Insert Alter	FormFalette	JSF		RichFace	· ·	<rich:nodeselectlistener></rich:nodeselectlistener>
		products * orders *	Cut		Jar		Seam		<rich:orderinglist></rich:orderinglist>
	1		Сору				Seam		<rich:paint2d></rich:paint2d>
		No products	Paste						<rich:panel></rich:panel>
		productcode	Remove	product	name	products	cale	produ	<rich:panelbar></rich:panelbar>
	4		Strip Tag						<rich:panelbarltem></rich:panelbarltem>
Visua	al/Sou	urce Preview	sulp lag						<rich:panelmenu></rich:panelmenu>

Figure 3.48. RichFaces Component Inserting

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

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

Add a Richfaces component to the CRUD



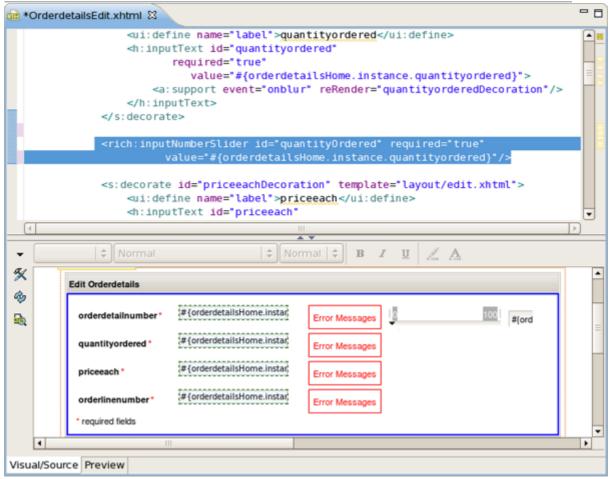


Figure 3.49. Adding the Source

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

rkshop: <u>Hon</u> Edit Orderdeta	offices	List Custom	ers List E	mployees List	Products List Orderdetail	s List Orders Li	at Decidere	tlines List			
Edit Orderdets						Unders Li	st produc	tiines Lisi	Pa	yments	L
Edit Orderdets											
Earl Graenaeta	ills										
orderdetailn	umber 0	1	0		100 0						
quantityord	ered •)									
priceeach*		.0									
orderlinenu	mber 0										
 required field 											
	ancel										
products *	orders *										
productcode	productlines productline	productname	productscale	productvendor	productdescriptic	on qu	antityinstock	buyprice	msrp	action	
S18_1749	Vintage Cars	1917 Grand Touring Sedan	1:18	Welly Diecast Productions	This 1:18 scale replica of the 19 Touring car has all the features expect from museum quality reg four doors and bi-fold hood oper engine and instrument panel, ch and tufted upholstery, all topped factory baked-ename! finish.	you would productions: all ning, detailed 27 prome-look trim,	24	86.7	170.0	View	

Figure 3.50. 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 <u>JSP</u> application using the 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* > *Open Perspective* > *Web Development* from the menu bar or by selecting *Window* > *Open Perspective* > *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 with the 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 File > New > Other
- Select Web > Dynamic Web Project in the New Project dialog box
- Click Next
- Enter "jspHello" as a project name
- Then select Minimal Configuration from the list of possible configurations and click Finish

	New Dynamic Web Project	
ynamic Web Project		2
Create a standalone Dynamic W Application.	/eb project or add it to a new or existing Enterprise	
Project name: jspHello		
Project name. Jspriello		
Project contents:		
✓ Use <u>d</u> efault		
Directory: //home/ochikvina/jav	va/eclipse-tools/eclipse/workspace2/jspHello	Browse
Target Runtime		
JBoss 4.2 Runtime		\$ N <u>e</u> w
Dynamic Web Module version		
2.5		0
Configuration		
Minimal Configuration		Modify
The most conservative starting can later be installed to add ne	point. Only the required facets are installed. Additi w functionality to the project.	onal facets
EAR Membership		
Add project to an EAR		
EAR Project Name: EAR		✓ <u>N</u> ew
0	< Back Next > Finish	Cancel

Figure 4.1. Create New Web Project

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

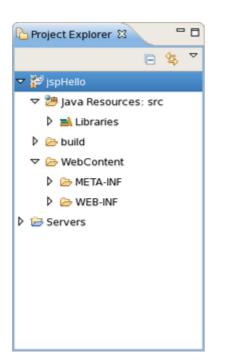


Figure 4.2. New Web Project

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 *WebContent* > *New* > *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 Finish button.

	w JavaServer Page
elect JSP Template elect a template as initial content in the JSP pa	age.
Use JSP Template	
emplates are 'New JSP' templates found in the	J <u>SP Templates</u> preference page.
Name	Description
New JSP File (html)	JSP with html markup
New JSP File (xhtml)	JSP with xhtml markup
New JSP File (xhtml, xml syntax)	JSP with xhtml markup and xml style syntax
review	
review	(in m) # . 2n
<pre>?xml version="1.0" encoding="\${encod %@ page language="java" contentType=</pre>	
<pre>?xml version="1.0" encoding="\${encod %@ page language="java" contentType= pageEncoding="\${encoding}"%></pre>	"text/html; charset=\${encoding}"
<pre>?xml version="1.0" encoding="\${encod %@ page language="java" contentType= pageEncoding="\${encoding}"%> !DOCTYPE html PUBLIC "-//W3C//DTD XH html xmlns="http://www.w3.org/1999/x</pre>	<pre>"text/html; charset=\${encoding}" ITML 1.0 Transitional//EN" "http://www.w3.org/TR/x</pre>
<pre>?xml version="1.0" encoding="\${encod %@ page language="java" contentType= pageEncoding="\${encoding}"%> !!DOCTYPE html PUBLIC "-//W3C//DTD XH html xmlns="http://www.w3.org/1999/x head></pre>	<pre>"text/html; charset=\${encoding}" ITML 1.0 Transitional//EN* "http://www.w3.org/TR/x chtml*></pre>
<pre>?xml version="1.0" encoding="\${encod %@ page language="java" contentType= pageEncoding="\${encoding}"%> !!DOCTYPE html PUBLIC "-//W3C//DTD XH html xmlns="http://www.w3.org/1999/x head></pre>	<pre>"text/html; charset=\${encoding}" ITML 1.0 Transitional//EN* "http://www.w3.org/TR/x</pre>
<pre>?xml version="1.0" encoding="\${encod %@ page language="java" contentType= pageEncoding="\${encoding}"%> !!DOCTYPE html PUBLIC "-//W3C//DTD XH html xmlns="http://www.w3.org/1999/x head> meta http-equiv="Content-Type" conte</pre>	<pre>"text/html; charset=\${encoding}" ITML 1.0 Transitional//EN* "http://www.w3.org/TR/x chtml*></pre>
<pre>?xml version="1.0" encoding="\${encod %@ page language="java" contentType= pageEncoding="\${encoding}"%> PUDCTYPE html PUBLIC "-//W3C//DTD XH html xmlns="http://www.w3.org/1999/x head> meta http-equiv="Content-Type" content title>Insert title here</pre>	<pre>"text/html; charset=\${encoding}" ITML 1.0 Transitional//EN" "http://www.w3.org/TR/x html"> ent="text/html; charset=\${encoding}" /></pre>
<pre>?xml version="1.0" encoding="\${encod %@ page language="java" contentType= pageEncoding="\${encoding}"%> PUDCTYPE html PUBLIC "-//W3C//DTD XH html xmlns="http://www.w3.org/1999/x head> meta http-equiv="Content-Type" content title>Insert title here</pre>	<pre>"text/html; charset=\${encoding}" ITML 1.0 Transitional//EN" "http://www.w3.org/TR/x html"> ent="text/html; charset=\${encoding}" /></pre>
<pre>?xml version="1.0" encoding="\${encod %@ page language="java" contentType= pageEncoding="\${encoding}"%> PUDCTYPE html PUBLIC "-//W3C//DTD XH html xmlns="http://www.w3.org/1999/x head> meta http-equiv="Content-Type" content title>Insert title here</pre>	<pre>"text/html; charset=\${encoding}" ITML 1.0 Transitional//EN" "http://www.w3.org/TR/x html"> ent="text/html; charset=\${encoding}" /></pre>

Figure 4.3. Create JSP Page

Our *hello.jsp* page will now appear in Project Explorer.

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:

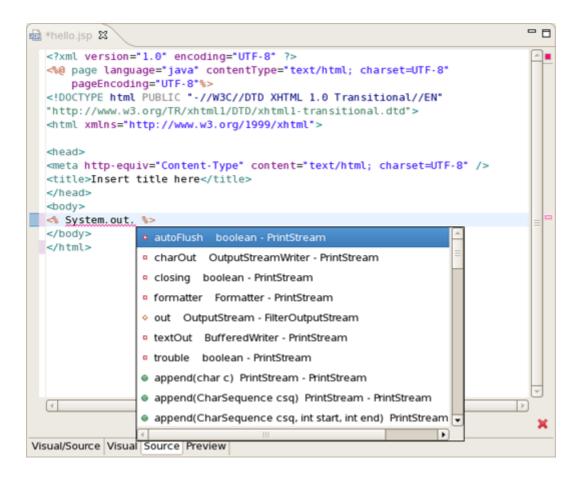


Figure 4.4. Content Assist in JSP Page

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

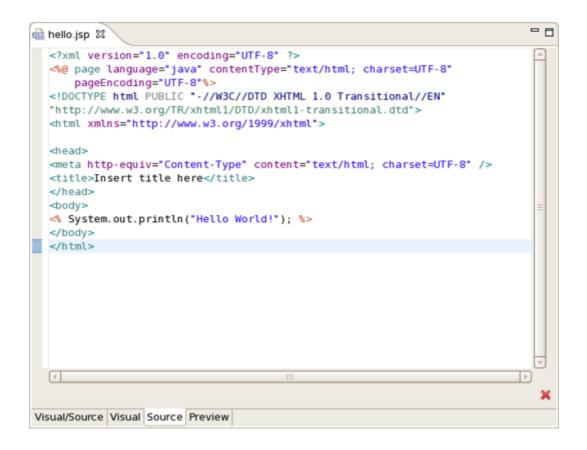


Figure 4.5. Hello.jsp Page

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* 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		<u> </u>	
z 📾 web.xml	A	Name:	web		ווו ר	
襑 Context Para	ms	Display-Name:	DynamProject		i	
🔄 Filters		Display-Marrie.	Dynamiroject			
😂 Listeners 📾	hello.jsp	🗴 web.xml 🛿				-
a Servlets		sion="1.0" encodin	·			
🐨 session-c		cmlns:xsi="http:// tp://java.sun.com/	/www.w3.org/2001/XM	LSchema-instanc	e"	
🗢 Mime Mai	xmlns:web=	"http://java.sun.	com/xml/ns/javaee/	web-app 2 5.xsd		
A Mime Maj	xsi:schema	aLocation="http://	com/xml/ns/javaee/ 'java.sun.com/xml/n			om/xml/
Melcome	xsi:schema <display< td=""><td>aLocation="http:// /-name>jspHello<td>java.sun.com/xml/n</td><td></td><td></td><td>om/xml/</td></td></display<>	aLocation="http:// /-name>jspHello <td>java.sun.com/xml/n</td> <td></td> <td></td> <td>om/xml/</td>	java.sun.com/xml/n			om/xml/
Welcome Error Pagi	xsi:schema <display <welcome< td=""><td>aLocation="http://</td><td>′java.sun.com/xml/n display-name></td><td></td><td></td><td>om/xml/</td></welcome<></display 	aLocation="http://	′java.sun.com/xml/n display-name>			om/xml/
Melcome Error Pagi JSP Configure	xsi:schema <display <welcome <welco <welco< td=""><td>alocation="http:// /-name>jspHelloe-file-list> ome-file>index.htm ome-file>index.htm</td><td>′java.sun.com/xml/n display-name> n n</td><td></td><td></td><td>om/xml/</td></welco<></welco </welcome </display 	alocation="http:// /-name>jspHelloe-file-list> ome-file>index.htm ome-file>index.htm	′ java.sun.com/xml/n display-name> n n			om/xml/
Welcome Error Pagi	xsi:schema <display <welcome <welco <welco <welco< td=""><td>alocation="http:// /-name>jspHelloe-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp</td><td><pre>/java.sun.com/xml/n display-name> nl n u</pre></td><td>s/javaee http:/</td><td></td><td>om/xml/</td></welco<></welco </welco </welcome </display 	alocation="http:// /-name>jspHelloe-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp	<pre>/java.sun.com/xml/n display-name> nl n u</pre>	s/javaee http:/		om/xml/
Melcome Error Pagi JSP Configure	xsi:schema <display <welcome <welco <welco <welco <welco< td=""><td>alocation="http:// /-name>jspHelloe-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h</td><td>′java.sun.com/xml/n display-name> n n</td><td>s/javaee http:/</td><td></td><td>om/xml/</td></welco<></welco </welco </welco </welcome </display 	alocation="http:// /-name>jspHelloe-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h	′ java.sun.com/xml/n display-name> n n	s/javaee http:/		om/xml/
 Melcome Error Pagi JSP Confie Security (xsi:schema <display <welcome <welco <welco <welco <welco <welco< td=""><td>alocation="http:// /-name>jspHelloe-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h ome-file>default.h</td><td><pre>/java.sun.com/xml/n display-name> nl n u u utml</pre></td><td>s/javaee http:/</td><td></td><td>om/xml/</td></welco<></welco </welco </welco </welco </welcome </display 	alocation="http:// /-name>jspHelloe-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h ome-file>default.h	<pre>/java.sun.com/xml/n display-name> nl n u u utml</pre>	s/javaee http:/		om/xml/
 Melcome Error Page JSP Config Security Iogin-con 	xsi:schema <display <welcome <welco <welco <welco <welco <welco <td>alocation="http:// /-name>jspHelloa-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h ome-file>default.h ome-file>default.j a-file-list></td><td><pre>/java.sun.com/xml/n display-name> nl n v/welcome-file> utml utml </pre></td><td>s/javaee http:/</td><td></td><td>om/xml/</td></welco </welco </welco </welco </welco </welcome </display 	alocation="http:// /-name>jspHelloa-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h ome-file>default.h ome-file>default.j a-file-list>	<pre>/java.sun.com/xml/n display-name> nl n v/welcome-file> utml utml </pre>	s/javaee http:/		om/xml/
 Melcome Error Page JSP Config Security Iogin-con Security I 	xsi:schema <display <welcome <welco <welco <welco <welco <welco <welco< td=""><td>alocation="http:// /-name>jspHelloa-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h ome-file>default.h ome-file>default.j a-file-list></td><td><pre>/java.sun.com/xml/n display-name> nl n v/welcome-file> utml utml </pre></td><td>s/javaee http:/</td><td></td><td>om/xml/</td></welco<></welco </welco </welco </welco </welco </welcome </display 	alocation="http:// /-name>jspHelloa-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h ome-file>default.h ome-file>default.j a-file-list>	<pre>/java.sun.com/xml/n display-name> nl n v/welcome-file> utml utml </pre>	s/javaee http:/		om/xml/
 Melcome Error Pagi JSP Config Security i Iogin-con Security i Env Entriti EJB 	xsi:schema <display <welcome <welco <welco <welco <welco <welco <td>alocation="http:// /-name>jspHelloa-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h ome-file>default.h ome-file>default.j a-file-list></td><td><pre>/java.sun.com/xml/n display-name> nl n v/welcome-file> utml utml </pre></td><td>s/javaee http:/</td><td></td><td>om/xml/</td></welco </welco </welco </welco </welco </welcome </display 	alocation="http:// /-name>jspHelloa-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h ome-file>default.h ome-file>default.j a-file-list>	<pre>/java.sun.com/xml/n display-name> nl n v/welcome-file> utml utml </pre>	s/javaee http:/		om/xml/
 Security (Security (xsi:schema <display <welcome <welco <welco <welco <welco <welco <td>alocation="http:// /-name>jspHelloa-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h ome-file>default.h ome-file>default.j a-file-list></td><td><pre>/java.sun.com/xml/n display-name> nl n v/welcome-file> utml utml </pre></td><td>s/javaee http:/</td><td></td><td>om/xml/</td></welco </welco </welco </welco </welco </welcome </display 	alocation="http:// /-name>jspHelloa-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h ome-file>default.h ome-file>default.j a-file-list>	<pre>/java.sun.com/xml/n display-name> nl n v/welcome-file> utml utml </pre>	s/javaee http:/		om/xml/
 Melcome Error Pagi JSP Confie Security i Iogin-con Security i Env Entrii EJB Services Resource 	xsi:schema <display <welcome <welco <welco <welco <welco <welco <td>alocation="http:// /-name>jspHelloa-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h ome-file>default.h ome-file>default.j a-file-list></td><td><pre>/java.sun.com/xml/n display-name> nl n v/welcome-file> utml utml </pre></td><td>s/javaee http:/</td><td></td><td>om/xml/</td></welco </welco </welco </welco </welco </welcome </display 	alocation="http:// /-name>jspHelloa-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h ome-file>default.h ome-file>default.j a-file-list>	<pre>/java.sun.com/xml/n display-name> nl n v/welcome-file> utml utml </pre>	s/javaee http:/		om/xml/
 Melcome Error Pagi JSP Confie Security i Iogin-con Security i Env Entriti EJB Services 	xsi:schema <display <welcome <welco <welco <welco <welco <welco <td>alocation="http:// /-name>jspHelloa-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h ome-file>default.h ome-file>default.j a-file-list></td><td><pre>/java.sun.com/xml/n display-name> nl n v/welcome-file> utml utml </pre></td><td>s/javaee http:/</td><td></td><td>om/xml/</td></welco </welco </welco </welco </welco </welcome </display 	alocation="http:// /-name>jspHelloa-file-list> ome-file>index.htm ome-file>index.htm ome-file>index.jsp ome-file>default.h ome-file>default.h ome-file>default.j a-file-list>	<pre>/java.sun.com/xml/n display-name> nl n v/welcome-file> utml utml </pre>	s/javaee http:/		om/xml/

Figure 4.6. Web.xml in Design and Source Mode

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

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

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

If you come back to Tree mode you will see that the changes made are automatically reflected in that mode.

Actually you don't really need to do any configurations right now.

4.2.3. Deploying the project

While creating any web project you could experience a pain writing ant scripts and managing the packaging even when writing the most trivial web applications. With JBoss Developer Studio you are saved from such a pain. All you need is to start JBoss Server and launch your application in your favorite browser.

You can also create a war archive with <u>JBDS's Archive Tools</u> and export it to any web server.

4.2.3.1. WAR Config

Project archives managing is available through *Project Archives view*.

- Select Window > Show view > Other > JBoss Tools > Project archives from menu bar
- Select a project in Package Explorer you want to be archived

In Project Archives you will see available archive types for the project:

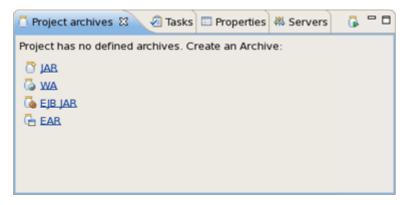


Figure 4.7. Project Archives

• Click, for example, WAR option to create war archive

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

9	New WAR	X
Create a new archive		
Create a new archive consisti	ng of filesets in the workspace.	
Archive information		
Archive name: 📋 jspHello.v	var	
Destination: 😂 jspHello	Workspace Filesystem	
Archive type		
Standard archive using zi	p compression	
 Exploded archive resulting 	g in a folder (no compression)	
		-
2	Back Next > Einish Cancel]

Figure 4.8. New WAR Archive

• Click *Next* to see a stub archive configuration for your project:

Sew WAR	×
WAR Archive Stub	
Below is a stub archive configuration for your project. You can customize this structure further after pressing finish.	
Preview	
🗢 🦲 jspHello.war	A
VEB-INF	
▽ 🗁 lib	
🗟 **/*.jar: jspHello	=
🗟 **/*: classes	
🗟 **/*: WebContent	v
() ()	Cancel

Figure 4.9. Stub Archive Configuration

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

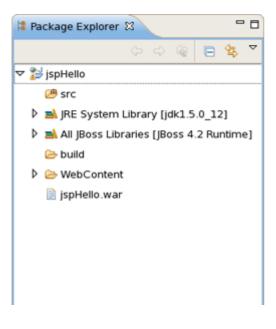


Figure 4.10. Archive is Created

🗇 Project archives 🛿 🧔 Tasks 🤪 jBoss Server View 🖳 Console	G - D
🗢 🚊 jspHello.war	
VEB-INF	
👂 🗁 lib	
🗟 **/*: classes	
🗟 **/*: WebContent	

Figure 4.11. Archive in Project Archives View

Via Project Archives view you could now edit your archive, add new folders, publish to server, and so on:

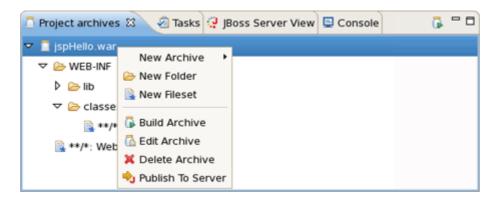


Figure 4.12. Configure Archive

4.2.3.2. Auto redeploy

When you are creating a web application and register it on JBoss Server it is automatically deployed into */deploy* directory of the server. JBDS 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.

4.2.4. JSP Page Preview

JBDS comes with JSP design-time preview features. When designing JSP pages you can easily preview how they will look during runtime. You can even <u>attach your stylesheet to the Preview</u>.

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

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

• Click Save button.

• Switch to Preview page by clicking 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 <u>JBoss Server view</u> by clicking the Start icon.

🤪 JBoss Server View 🕱	*0)	• 1	• 🗖

Figure 4.13. Starting Server

Click the Run icon or right click your project folder and select Run As > Run on Server. If you
haven't made any changes in 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.



Figure 4.14. Run Project

You should see the next page in a Browser :

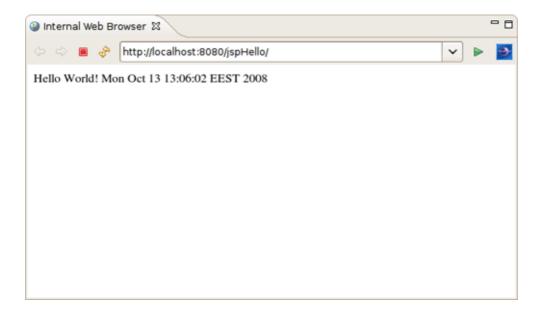


Figure 4.15. Running Project

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

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 see how to create a simple JSF application being based on "RAD" philosophy. We will create the familiar Guess Number application. The scenario is the following. 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 powerful features of 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 JBDS's new project wizard and predefined templates. Follow the next steps:

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

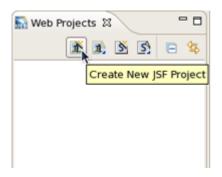


Figure 5.1. Create New JSF Project

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

Our project will appear in Project Explorer and Web Projects Views. As you can see JBoss Developer Studio has created for us the whole skeleton for the project with all needed libraries, faces-config.xml and web.xml files.

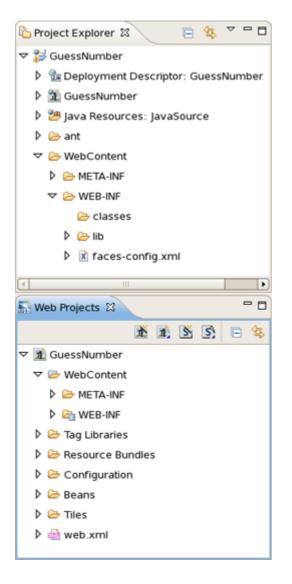


Figure 5.2. New JSF Project

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

5.2. Creating JSP Pages

Here, we are going to add two pages to our application. The first page is 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 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.

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

Figure 5.3. Create New View

- Type pages/inputnumber as the value for From-view-id
- Leave everything else as is and click *Finish*
- In the same way create another jsf view. Type pages/success as the value for From-view-id
- Select File > Save

On the diagram you will see two created views.

🖄 faces-c	config.xml 🛿			- 6
		/pages/inputnumber.jsp		
		/pa	ges/success.jsp	
Diagram	Tree Source			

Figure 5.4. 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

	/pages/inputnumber.jsp
2	
Create New	
Create New Connection	

Figure 5.5. 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.

/pages/inputnumber.jsp success /pages/success.jsp	🏂 faces-con	ʻig.xml ⊠	
/pages/success.jsp	7	👜 🐴	

Figure 5.6. Created Connection

• Select File > 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 first, you don't hard code anything into the JSP pages. And second, it 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 JavaSource folder and select New > Folder
- Type game for Folder name and click Finish

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

- Right click on game folder and select New > Properties File
- Type messages as the value for "name" attribute and click Finish

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

🔲 Messag	es.prope	rties 🛙					- 0
Messages.	oropertie	s					
name			value	 	 	 A	dd
						E	dit
						Del	lete
						<u> </u>	ļp
						Do	<u>w</u> n
Properties	Source						

Figure 5.7. Messages.properties File

- Click Add button for adding new attribute to your resource file
- Type how_to_play for "name" and Please pick a number between 0 and 100. for value
- Click Finish
- In such a way add the next properties:

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.

• Click *File > Save* from the menu bar

Your .properties file should now look like follows:

🖄 faces-config.xml 🛛	inputnumber.jsp 🔄 messages.properties 🔀	-
nessages.properties		
name	value	Add
how_to_play	Please pick a number between a 0 and 100.	
makeguess_button	Make Guess	Edit
tryagain_button	Play Again?	Delete
success_text	How cool You have guessed the number, {0} is correct!	
tryagain_smaller	Oops incorrect guess. Please try a smaller number.	Цр
tryagain_bigger	Oops incorrect guess. Please try a bigger number.	Do <u>w</u> n

Figure 5.8. Properties are Added

Up and *Down* buttons allow you to move you attribute on the list. For delete the attribute, choose it and press *Delete* button.

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

5.5. Creating Java Bean

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

- Right click game folder
- Select New > Class
- Type NumberBean for bean name

A java bean is created.

• Declare the variable of your entered number:

Integer userNumber;

JBDS allows to quickly generate getters and setters for java bean.

• Right click NumberBean.java in Package Explorer

- Select Source > Generate Getters and Setters...
- Check userNumber box and click OK

9	Generate Getters and Setters	×
<u>S</u> elect getters a	and setters to create:	
🔻 🗹 🔺 user	Number	Select <u>A</u> ll
⊡ o ge	etUserNumber()	Deselect All
⊘ ∘ se	tUserNumber(Integer)	
		Select Getters
		Select Setters
Allow setter	s for final fields (remove 'final' modifier from fields if necessar	ry)
Insertion point:		
Last method		•
Sort by:		
Fields in gette	r/setter pairs	•
Access modif	er	
<u> public</u>	○ protected ○ default ○ private	
🗌 <u>f</u> inal	synchronized	
Generate m	nethod <u>c</u> omments	
The format of I	he getters/setters may be configured on the <u>Code Templates</u> p	reference page.
		, , , , , , , , , , , , , , , , , , , ,
i 2 of 2 selec	cted.	
0	ОК	Cancel

Figure 5.9. 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.Locale; import java.util.ResourceBundle;

The whole java bean should look as follows:

```
import javax.faces.context.FacesContext;
import javax.servlet.http.HttpSession;
import javax.faces.application.FacesMessage;
import java.util.Locale;
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 know about faces-config.xml file.

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

- · Open faces-config.xml file in a source mode
- Add here one more navigation rule and a managed bean declarations 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"
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 */pages/inputnumber.jsp*. 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 Visual Page Editor.

5.7.1. Editing inputnumber.jsp page

First, let's dwell on how to edit inputnumber.jsp.

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 inputnumber.jsp 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 be already generated as we choose a template when creating the page.

At the beginning it's necessary to create a <h:form> component where all others components are put.

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

🚿 JBoss Tools Palette 🛛 🦳 🗖
🛠 🗟 🛇
🗁 JBoss Ajax4jsf
➢ JBoss RichFaces
🕞 JBoss Seam
🕞 JSF HTML 🔹
🐼 JSF HTML taglib
column
commandButton
ee commandLink
🖽 dataTable
S! form
🖻 graphicImage
🗍 inputHidden
👀 inputSecret
ab inputText
🔢 inputTextarea
🖆 message
🖆 messages
 outputFormat
🖉 outputLabel
🖞 outputLink
🖉 outputText
📰 panelGrid
See A se
selectBooleanCheckbox
selectManyCheckbox
selectManyListbox
selectManvMenu

Figure 5.10. Insert h:form

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

<h:form> attributes Attribute name Value accept acceptcharset binding</h:form>	
accept acceptcharset	
acceptcharset	
binding	
dir	
enctype	
id	
lang	
onclick	
ondblclick	
onkeydown	
onkeypress	

Figure 5.11. Define Id of Form

• Type inputNumbers and click Finish

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



Figure 5.12. Created Form

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

selectManyMenu

selectOneMenu

×

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

<f:loadbundle <b="">ba</f:loadbundle>	asename="game.messages" var	="msg"/>		
As always JBDS	provides code assist:			
sces-config.xml	😥 •inputnumber.jsp 🕴 🔳 messages.prop	erties	•) 🚿 JBoss Tools Palette 🕱 🛛 🗖
	http://java.sun.com/jsf/html* prefix=*		A	• × • •
<pre><ful> </ful></pre>	http://java.sun.com/jsf/core* prefix=*	t. <i>1</i> >		Boss Ajax4jsf
S <> f:loadBundle	v 0 11 a 11 11 11 11 11 11 11 11 11 11 11 1	view, and expose of the current requine the "var" attribute a get() call is mad iteral string ???KE the key being look MissingResourceE	le for a key that does n Y??? is returned from ed up in the Map, inste	the request attributes cified by the value of ust behave such that if iot exist in the Map, the the Map, where KEY is ad of a . If the ResourceBundle hrown.
				 <i>i</i> outputLabel

Figure 5.13. Code Assist

Visual/Source Visual Source Preview

f:I

- Switch to Visual tab, so it could be possible to work with the editor completely in its WYSIWYG mode
- Click on *outputText*, drag the cursor over to the editor, and drop it inside the blue box in the editor
- Select *value* and click on this line below "value" header
- Click ... button next to the value field

JBDS will nicely propose you to choose within available values:

9	Edit Value	×
<h:outputtext></h:outputtext>		
🍓 Managed Beans		
Resource Bundles		
SF Variables		
		Ok Cancel

Figure 5.14. Choose Value

- Expand Resource Bundles > msg
- Select *how_to_play* value and click *Ok*. Then click *Finish*

9	Edit Value	×
<h:outputtext></h:outputtext>		
😂 Managed Beans		<u>^</u>
🗢 偏 Resource Bundles		
🗢 🛅 msg		
how_to_play		=
makeguess_button		
success_text		
tryagain_bigger		
tryagain_button		
• tryagain smaller		
	Ok	Cancel

Figure 5.15. Selecting Value

The text will appear on the page:

🙇 fac	ces-config.xml 🔞 🔹	inputnumber.jsp 🛛 🔚 messages.properties	- 0
• (\$]	÷ ÷ B I <u>U</u> <u>A</u>	
×	Please pick a nun	nber between a 0 and 100.	
🤣 R			
~~			
html	body f:view h:form		×
Visua	I/Source Visual Source	Preview	

Figure 5.16. Created OutputText Component

- Switch to Source mode and insert
 tag after <h:outputText> component to make a new line.
- Click Save button.
- On the Palette click on *inputText,* drag the cursor over to the editor, and drop it inside the editor after the text.
- Switch to a Source mode and insert
 tag after <h:outputText> component to make a new line
- Click 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 *value* and click on this line below "value" header
- Click ... button next to the value field
- Expand Managed Beans > NumberBean
- Select userNumber value and click Ok
- Switch Advanced tab
- Select id and click on this line below "value" header
- Type userNumber in text field

- Select required and click on this line below "value" header
- Click ... button next to the value field
- Expand Enumeration and select true as a value

9	Edit Required	×
<h:inputtext></h:inputtext>		0
😑 false		
– true		
👂 🍓 Managed Beans		
Resource Bundles		
ISF Variables		
	Ok	Cancel

Figure 5.17. Add "required" Attribute

- Click Ok, then click Finish
- · 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 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 value and click on this line below "value" header
- Click ... button next to the value field
- Expand Resource Bundles > msg and select makeguess_button as a value

- Click Ok
- Select action and click on this line below "value" header
- Type NumberBean.checkGuess in text field
- Click Finish
- In Source mode add
tags between <outputText> , <inputText> and
 commandButton> components to place them on different lines

inputnumber.jsp page should look like this:

```
<%@ 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>
<f:view>
  <h:form id="inputNumbers">
   <h:outputText value="#{msg.how_to_play}"/>
   <br/>br/>
   <h:messages style="color: blue"/>
   <br/>br/>
   <h:inputText id="userNumber" value="#{NumberBean.userNumber}" required="true">
      <f:validateLongRange minimum="0" maximum="100"/>
   </h:inputText>
   <br/><br/>
   <h:commandButton value=
       "#{msg.makeguess_button}" action="#{NumberBean.checkGuess}"/>
 </h:form>
</f:view>
</html>
```

5.7.2. Editing success.jsp page

In the same way like inputnumber.jsp, edit success.jsp page. Its whole source should be the next:

```
<%@ 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>
<f:view>
```

<h:form id="result"></h:form>
<h:outputformat value="#{msg.success_text}"></h:outputformat>
<f:param value="#{NumberBean.userNumber}"></f:param>
<h:commandbutton value="</td"></h:commandbutton>
"#{msg.trayagain_button}"

Again you can use code assist provided by JBDS when editing jsp page:

👜 faces-config.xml 🛛 📾 inputnumber.jsp	🖬 messages.properties 🛛 🙆 *success.jsp 🖾 🔭 1	- 0	🚿 JBoss Tools Palette 🛿 🛛 🗖
<pre></pre>		<u>A</u>	🛠 🗐 🤕
<pre><%@ taglib uri="http://java.sun.com/jsf/core" prefix="f" %> <f:loadbundle basename="game.messages" var="msg"></f:loadbundle></pre>		Boss Ajax4jsf	
<pre></pre>		Boss RichFaces	
<head></head>		_	🕞 Boss Seam
<title></title>			≥ JSF HTML *
			✓ JSF HTML taglib
<pre><body> <f:view></f:view></body></pre>		6	column
<h:form id="result"></h:form>			
	e="#{msg.success_text}">		@ commandLink
f:param_value=*	#1		ataTable
	(a) msg	E C	L! form
4	Ø NumberBean		graphicImage
			inputHidden
▼ ↓ ↓	applicationScope		** input/idden
X Hannah Van hannah da	(a) cookie		ab inputText
How cool You have quessed the	facesContext		🗄 inputTextarea
	() header		imessage
	③ headerValues		messages
	(a) initParam	L	🐼 outputFormat
	(i) param		/ outputLabel
	(a) paramValues		t outputLink
		-	/ outputText
	(requestScope		🖬 panelGrid
			opanelGroup
blad hade follow bifered biotech formant formant			SelectBooleanCheckbox
html body f:view h:form h:outputFormat f:param		E selectManyCheckbox	
Visual/Source Visual Source Preview		selectManyListbox	

Figure 5.18. Code Assist for <f:param>

This page, success.jsp, 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.

At the end, 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:

🔊 faces-config.xml	🔞 inputnumber.jsp 🛛 🔳 messag	es.properties	📾 success.jsp	"1	- 0
Please pick a nun #{NumberBean.user	nber between a 0 and 100. Numbé				
Visual/Source Visual S	purce Preview				

Figure 5.19. Success.jsp in Preview Mode

5.8. Creating index.jsp page

Now you know how to create index.jsp page.

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

- Right click WebContent > New > JSP File
- Type *index* for name field and choose *JSPRedirect* as a template
- Click Finish
- The source for this page should be 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 JBoss Server view. (If JBoss 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 project Run AS > Run on Server
- Play with the application by entering correct as well as incorrect values

🕹 Mozilia Firefox	
<u>File Edit View Go Bookmarks Tools H</u> elp	\sim
🖕 • 🛶 - 🤔 🙁 🏠 🛴 http://localhost:8080/GuessNumber/ 💌 🛇 Go 💽	
Please pick a number between a 0 and 100.	
Make Guess	
Done	

Figure 5.20. You are Asked to Enter a Number Between 0 and 100

Nozilia Firefox
Ele Edit View Go Bookmarks Tools Help
🔷 🔹 🛶 - 🥰 💿 🏠 🤇 http://localhost:8080/GuessNumber/pages/inputnumber.jsf 💌 💿 Go 💽
Please pick a number between a 0 and 100.
• inputNumbers:userNumber: Validation Error: Specified attribute is not between the expected values of 0 and 100.
-6
Make Guess
Done

Figure 5.21. Your Input is Validated and an Error Message is Displayed if Invalid Input was Entered

😢 Mozilia Firefox 📃	
Ele Edit View Go Bookmarks Tools Help	$\langle \rangle$
🔷 • 🍦 - 🤔 💿 😭 🧲 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 5.22. After You Enter a Guess, the Application Tells You Whether a Smaller or a Larger Number Should be Tried

🕹 Mozilia Firefox		
Eile Edit View Go Bookmarks Tools Help		ं
🖕 🔹 🧼 - 🥰 💿 🏠 🛴 http://localhost:8080/GuessNumber/pages/inputnumber.jsf	🔽 🔘 Go	G,
How cool You have guessed the number, 32 is correct!		
Play Again?		
riay Againt		
Done		

Figure 5.23. Your Guess is Correct

FAQ

For more information on JBoss Developer Studio features, refer to the following FAQ to get the answers on the most "popular" questions.

6.1. What should I do if 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 xpLib.i386 package

2. Type

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

3. and/or use

yum install libXp

- 4. Open the JBDS perspective. If you see the Help view open, close it and restart JBDS
- 5. If none of these work, do the following:
 - Clear the Eclipse log file, <workspace>\.metadata\.log
 - Start Eclipse with the -debug option:

eclipse -debug

• Post the Eclipse log file (<workspace>\.metadata\.log) on the forums.

6.2. 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 > Show View > Others > Server > Servers*), then right click on this *view > New > 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.

6.3. I have an existing Seam 1.2.1 project. Can I migrate/ import the project to a JBDS Seam project?

We highly recommend you to create Seam 1.2.1 project using the JBDS. In other case try to do manually:

· 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.

6.4. I have an existing Struts or JSF project. Can I open the project in JBDS?

Yes. From main menu select *File > Import > Other > JSF Project (or Struts Project)* and follow wizards steps.

6.5. Can I import a .war file?

Yes. Select *File > Import > Web > WAR file*, then follow importing steps.

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

JBoss Developer Studio preconfigures eclipse via the eclipse.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

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

See Adding Tag Libraries in Visual Web Tools Guide.

6.8. 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 > Seam Settings*. Check *Seam support* box to enable all available <u>Seam Settings</u>.

6.9. 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:

- Go to New > Other > Java Project from Existing Buildfile
- Point to the *build.xml* of any chosen project by pressing *Browse* button
- Hit *Finish* 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 go to *Properties > Seam Settings*.

Further Reading

JSF Tools Reference Guide (html)

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.

• JSF Tools Tutorial (html)

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

• Struts Tools Reference Guide (html)

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 (html)

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

• Seam Dev Tools Reference Guide (html)

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 (html)

JBoss Server Manager Reference Guide (html)

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 (html)

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 (html)

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 (html)

This guide provides you with the information on ESB Editor which allows you to develop ESB file.

• JBoss Portal Tools Reference Guide (html)

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 (html)

This guide gives you practical help on JBossWS usage.

• Exadel Studio Migration Guide (html)

This document is intended to help you to migrate an existing Exadel JSF or Struts projects from Exadel Studio into JBoss Developer Studio.