## Visual Web Tools Reference Guide

Version: 2.0.0.GA

1. Visual Web Tools	1
1.1. Key Features of Visual Web Tools	. 1
2. Spring Tools	. 5
2.1. Spring IDE guide	5
2.1.1. Add Spring Project Nature	. 5
2.1.2. Create New Spring Project	5
2.1.3. Add References To Other Spring Projects	5
2.1.4. Add Spring Beans Config Files	5
2.1.5. Create Spring Beans Config Sets	. 5
2.1.6. Open Spring Explorer	. 5
2.1.7. Validate Spring Beans Config	5
2.1.8. Open Spring Beans Graph	5
2.1.9. Search Spring Beans	5
3. Editors	7
3.1. Editors Features	7
3.1.1. OpenOn	7
3.1.2. Content Assist	12
3.1.3. Synchronized Source and Visual Editing	27
3.2. Visual Page Editor	29
3.2.1. Advanced Settings	35
3.2.2. Setup notes for Linux	40
3.2.3. JSP syntax validation	41
3.2.4. JSP Page Preview	41
3.3. More Editors	41
3.3.1. Graphical Properties Editor	41
3.3.2. Graphical TLD Editor	43
3.3.3. Graphical Web Application File (web.xml) Editor	48
3.3.4. CSS Editor	53
3.3.5. JavaScript Editor	55
3.3.6. XSD Editor	57
3.3.7. Support for XML Schema	
4. JBoss Tools Palette	65
4.1. Palette Options	66
4.1.1. Palette Editor	66
4.1.2. Show/Hide	73
4.1.3. Import	75
4.2. Using the Palette	
4.2.1. Inserting Tags into a JSP File	75
4.2.2. Adding Custom JSF Tags to the JBoss Tools Palette	
4.3. Rich Faces Support	
5. Web Projects View	
5.1. Project Organization	
5.2. Drag and Drop	
5.2.1. For a Property	

5.2.2. For Managed Bean Attributes	88
5.2.3. Navigation Rules	88
5.2.4. For a Tag Library File Declaration	90
5.2.5. For JSP Pages	91
5.3. Developing the Application	92
5.4. Expanding Tag Library Files	93
5.5. Drag and Drop Tag Libraries on to JBoss Tools Palette	94
5.6. Create and Import JSF and Struts Projects	94
6. JBoss Tools Preferences	97
6.1. CodeAssist	98
6.2. Editors	99
6.3. JBoss Servers	. 100
6.4. JSF	101
6.5. JSF Flow Diagram	102
6.6. JSF Page	104
6.7. JSF Propject	105
6.8. Packaging Archives	107
6.9. Plug-in Insets	108
6.10. Resource Insets	110
6.11. Seam	111
6.12. Seam Validator	112
6.13. Struts	113
6.14. Struts Automatic	114
6.15. Struts Customization	. 115
6.16. Struts Flow Diagram	116
6.17. Struts Pages	118
6.18. Struts Project	119
6.19. Struts Support	121
6.20. Title Diagram	122
6.21. Verification	123
6.22. View	125
6.23. Visual Page Editor	126
6.24. XDoclet	128
6.25. XDoclet Templates	
6.26. XDoclets Variables	134
6.27. Changing Default Environment During Project Creation	135
6.28. Changing Default Project Template During Project Creation	136

## **Visual Web Tools**

This guide covers the usage of Visual Web Tools in JBoss Developer Studio and JBoss Tools. The difference between these products is that 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, please visit the JBoss Developer Studio home page at: *http://www.jboss.com/products/devstudio*.

In JBoss Tools there is an extensive collection of specialized wizards, editors and views that can be used in various scenarios while developing Web applications. The following chapters walk through these features.

## **1.1. Key Features of Visual Web Tools**

Here is the table of main features of Visual Web Tools:

#### Table 1.1. Key Functionality for Visual Web Tools

Feature	Benefit	Chapter
Visual Page Editor	Powerful and customizable visual page editor. Possibility to develop an application using any web technology: jsf, seam, struts, jsp, html and others. Developing using four tabs: visual/source, visual, source and preview. Fast and easy switching between these tabs. Split screen design of visual and source views. Full and instant synchronization between source and visual views. Integration with properties and outline views. Graphical toolbar to add inline styling to any tag.	visual page editor
JBoss Tools Palette	Organizing various tags by groups, inserting tags into a jsp or xhtml page with one click, adding custom or 3rd party tag libraries into	jboss tools palette

Feature	Benefit	Chapter
	the palette, easy controling the number of tag groups shown on the palette.	
Web Projects View	Visualizing and displaying projects by function. Easy selecting of different kinds of items and dropping them into jsp pages. Using context menus to develop the application. Using icon shortcuts to create and import JSF and Struts projects. Expanding and inspecting tag library files. Selecting custom and third-party tag libraries to drag and drop onto the JBoss Tools Palette.	web projects view
JSF and Facelets support	Step-by-step wizards for creating new JSF and Facelets projects with a number of predefined templates, importing existing ones and adding JSF capabilities to non-jsf web projects.	jsf support
Flexible and customizable project template management		working with projects
Support for Managed Beans	Adding new managed beans, generating code for attributes, properties and getter/setter methods.	managed beans
OpenOn	Easy navigation between views and other parts of your projects.	facelets support
Content Assist	Code completion proposals while working in java, xml, jsp, xhtml, xhtml, seam project and jsf configuration files. Content assist based on project data (dynamic code assist); with graphical editor. Code completion for values from property files, beans attributes and methods, navigation rule outcomes and jsf variables.	content assist
Support for Custom Converters and Validators		converters and validators
Verification and Validation	All occuring errors will be immediately reported by verification feature, no matter in what view you are working. Constant validation and errors checking allows to catch many of the errors during development process that significantly reduces development time.	verification and validation

Feature	Benefit	Chapter
Drag-and-Drop	Possibility of inserting any tag onto the page you are editing by just drag-and-droping it from the palette to this page. Adding any properties, managed bean attributes, navigation rules, tag library file declarations, jsp files from web projects view by clicking them and draging to source code.	visual page editor drag-and-drop
Struts Support	Step-by-step wizards for creating a new struts project with a number of predefined templates, importing existing ones and adding struts capabilities to non-struts web projects.	struts support
Support for JSF and Struts Configuration Files	Working on files using three modes: diagram, tree and source. Synchronization between the modes and full control over the code. Easy moving around the diagram using the Diagram Navigator. Working with struts projects that have multiple modules. Possibility to use Struts configuration file debugger allowing to set break points on struts diagram and then launch the server in debug mode.	graphical editor for jsf graphical editor for struts
Rich Faces Support	Tight integration between JBDS and RichFaces frameworks. Easy managing RichFaces components in any web application. Support for RichFaces and Ajax4jsf libraries in JBoss Tools Palette. Rendering RichFaces components in Visual Page Editor.	rich faces support

# **Spring Tools**

JBoss Developer Studio is bundled with *Spring IDE* [http://springide.org/project] for Eclipse. Visit Spring IDE site for the latest versions and documentation.

## 2.1. Spring IDE guide [http://springide.org/project/wiki/ SpringideGuide]

*Spring IDE* [http://springide.org/project] is a graphical user interface for the configuration files used by the *Spring Framework* [http://www.springframework.org/]. It's built as a set of plugins for the Eclipse platform.

2.1.1. Add Spring Project Nature [http://springide.org/project/ wiki/SpringideGuide#AddProjectNature]

2.1.2. *Create New Spring Project* [http://springide.org/project/ wiki/SpringideGuide#CreateNewProject]

2.1.3. Add References To Other Spring Projects [http://springide.org/project/wiki/ SpringideGuide#AddProjectReferences]

2.1.4. Add Spring Beans Config Files [http://springide.org/ project/wiki/SpringideGuide#AddBeansConfigs]

2.1.5. Create Spring Beans Config Sets [http://springide.org/ project/wiki/SpringideGuide#CreateBeansConfigSets]

2.1.6. *Open Spring Explorer* [http://springide.org/project/wiki/ SpringideGuide#OpenSpringExplorer]

2.1.7. Validate Spring Beans Config [http://springide.org/ project/wiki/SpringideGuide#ValidateBeansConfig]

2.1.8. *Open Spring Beans Graph* [http://springide.org/project/ wiki/SpringideGuide#OpenBeansGraph]

2.1.9. Search Spring Beans [http://springide.org/project/wiki/ SpringideGuide#SearchBeans]

## **Editors**

In previous chapters you had possibility to read about Graphical Editor for *JSF* and *Struts* configuration files, *Graphical Editor for Tiles Files*, *Graphical Editor for Struts Validation Files*. All these editors have *OpenOn* and *Content Assist* features, they are described more detail in this chapter. In addition you get to know a *Visual Page Editor* for combined visual and source editing of Web pages and many *other editors* for different types of files.

### 3.1. Editors Features

JBoss Developer Studio has powerful editor features that help you easily navigate within your application and make use of content and code assist no matter what project file (jsp, xhtml, xml, css, etc...) you are working on.

#### 3.1.1. OpenOn

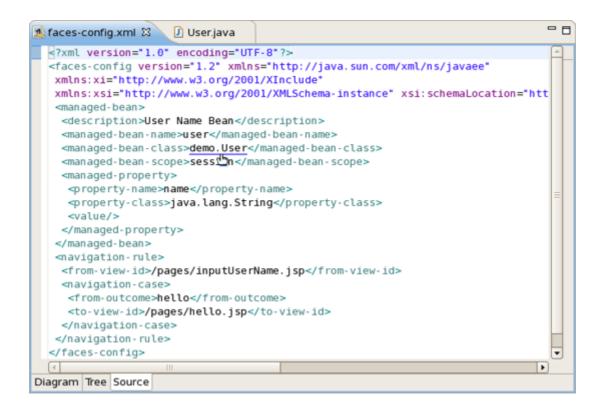
OpenOn let's you easily navigate through your project without using the Package Explorer view (project tree). With OpenOn, you can simply click on a reference to another file and that file will be opened.

OpenOn is available for the following files:

- XML files
- JSP/XHTML Pages
- Java files

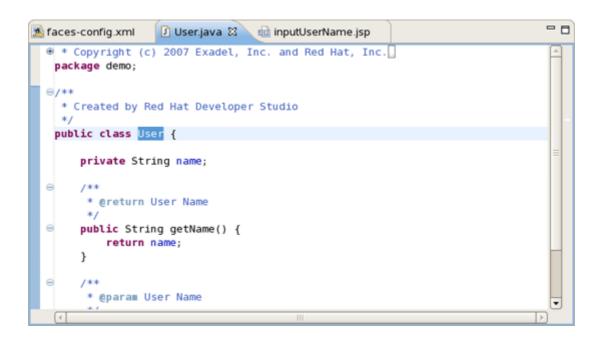
#### 3.1.1.1. XML Files

Press and hold down the Ctrl key. As you move the mouse cursor over different file references in the file, they display an underline. When you have the mouse cursor over the name of the file you want to open, click and the file will open in its own editor. In this example the managed bean NameBean will open.



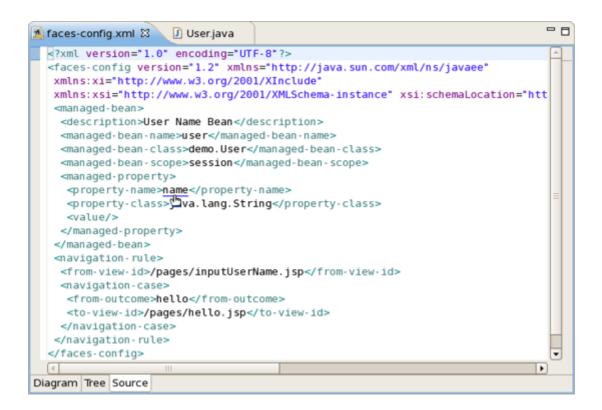
#### Figure 3.1. NameBean Managed Bean

This is the result of using OpenOn.



#### Figure 3.2. NameBean Java Class

You can also try OpenOn with defined attributes.



#### Figure 3.3. OpenOn With Defined Attributes

You can also open any JSP pages.

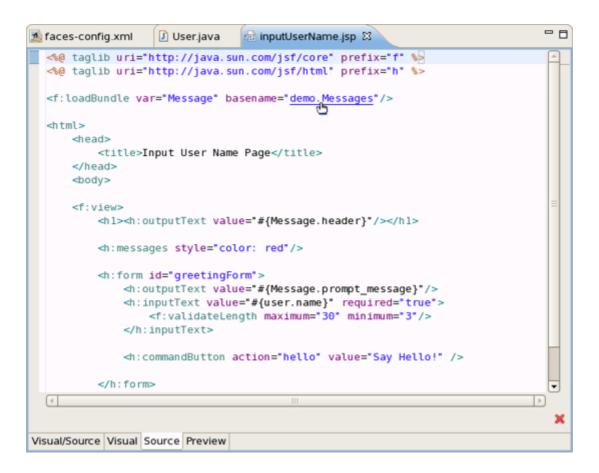
🔊 faces-config.xml 🕴 🕖 User.java 🖓	٥
<pre>?xml version="1.0" encoding="UTF-8"?&gt;</pre>	-
<pre><faces-config <="" pre="" version="1.2" xmlns="http://java.sun.com/xml/ns/javaee"></faces-config></pre>	F
<pre>xmlns:xi="http://www.w3.org/2001/XInclude"</pre>	
<pre>xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="htt</pre>	
<man -="" aged="" bean=""></man>	
<pre><description>User Name Bean</description></pre>	
<managed-bean-name>user</managed-bean-name>	
<pre><managed-bean-class>demo.User</managed-bean-class></pre>	
<managed-bean-scope>session</managed-bean-scope>	
<managed-property></managed-property>	
<property-name>name</property-name> ====================================	
<property-class>java.lang.String</property-class>	
<value></value>	
<navigation-rule></navigation-rule>	
<from-view-id>/pages/inputUserName.jsp</from-view-id>	
<navigation-case></navigation-case>	
<from-outcome>hello</from-outcome>	
<to-view-id>/pages/hello.jsp</to-view-id>	
	•
Diagram Tree Source	

Figure 3.4. JSP Page OpenOn

#### 3.1.1.2. JSP Pages

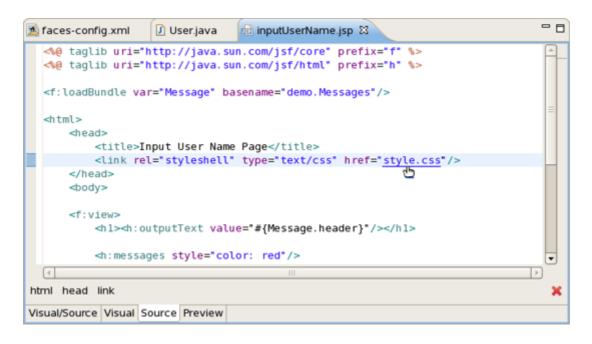
OpenOn is also very useful in JSP pages. It will allow you to quickly jump to the reference instead of having to hunt around in the project structure.

You can easily open the imported property files.



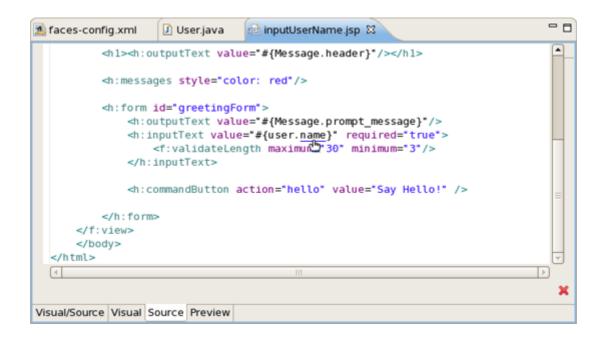
#### Figure 3.5. OpenOn With Imported Property Files

Use OpenOn to open a CSS file used with a JSP page:



#### Figure 3.6. OpenOn With CSS File

Open managed beans:



#### Figure 3.7. OpenOn With Managed Beans

For JSP files in a JSF project, you can also easily open the navigation rules by applying OpenOn to the JSF tag for the navigation outcome:

staces-config.xml	🚺 User.java	inputUserName.jsp 🛛		- 8
<h 1=""><h :="" ou<="" th=""><th>itputText valu</th><th>e="#{Message.header}*/</th><th>&gt;</th><th><b>A</b>_</th></h></h>	itputText valu	e="#{Message.header}*/	>	<b>A</b> _
<h:messaç< td=""><th>ges style="col</th><th>or: red"/&gt;</th><td></td><td></td></h:messaç<>	ges style="col	or: red"/>		
<h:form 1<="" th=""><th>id="greetingFo</th><th>rm" &gt;</th><th></th><th></th></h:form>	id="greetingFo	rm" >		
<h :="" ou<="" td=""><th>tputText valu</th><th>e="#{Message.prompt_me</th><td>ssage}"/&gt;</td><td></td></h>	tputText valu	e="#{Message.prompt_me	ssage}"/>	
<h:ir< td=""><th>putText value</th><th>="#{user.name}" requir</th><td>ed="true"&gt;</td><td></td></h:ir<>	putText value	="#{user.name}" requir	ed="true">	
· ·	<f:validatelen< th=""><th>gth maximum="30" minim</th><th>um="3"/&gt;</th><th></th></f:validatelen<>	gth maximum="30" minim	um="3"/>	
<th>inputText&gt;</th> <th></th> <td></td> <td></td>	inputText>			
<h:co< td=""><th>ommandButton a</th><th>ction="<u>hello</u>" value="S</th><td>ay Hello!" /&gt;</td><td>=</td></h:co<>	ommandButton a	ction=" <u>hello</u> " value="S	ay Hello!" />	=
<th></th> <th></th> <td></td> <td></td>				
				-
4		111	>	)
				×
Visual/Source Visual S	ource Preview			

#### Figure 3.8. OpenOn With JSF Tag

#### 3.1.2. Content Assist

Content assist is available when working with

- Seam project files [../../../seam/en/html\_single/index.html#ContentAssist]
- JSF project files
- Struts project files
- JSP files
- RichFaces components

#### 3.1.2.1. JSF Project Files

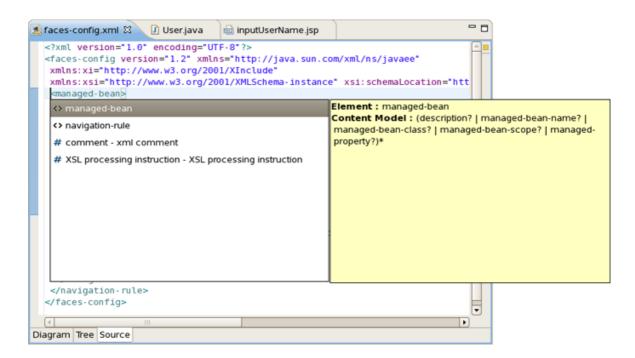
When working with JSF project in JBoss Developer Studio, you can use various Content Assist features while developing:

- Content Assist for XML, JSP and JSF configuration files
- · Content Assist based on project data
- Content Assist with graphical JSF editor

#### 3.1.2.1.1. Content Assist for XML, JSP and JSF configuration files

At any point when working with any XML, JSP and JSF configuration files Content Assist is available to help you. Simply type *Ctrl-Space* to see what is available.

Content Assist for JSF configuration file:



#### Figure 3.9. Content Assist in JSF Configuration File

Content Assist for JSF JSP file:

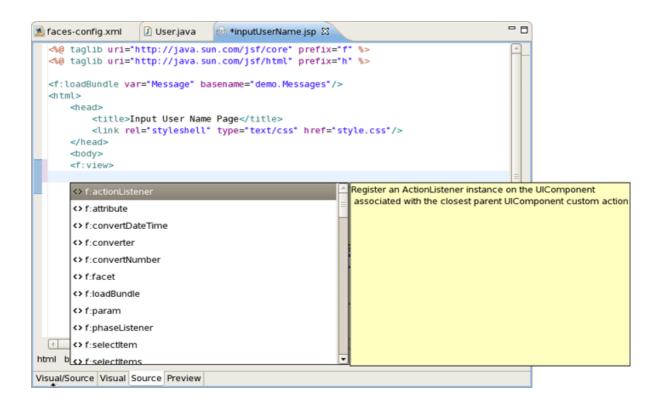
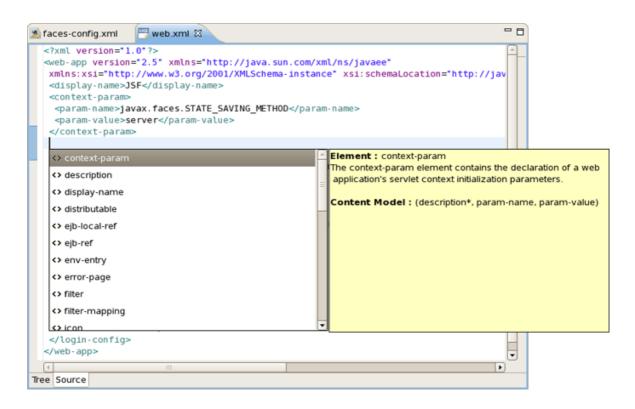


Figure 3.10. Content Assist in JSP File

Content Assist for other JSF XML project files (web.xml shown):



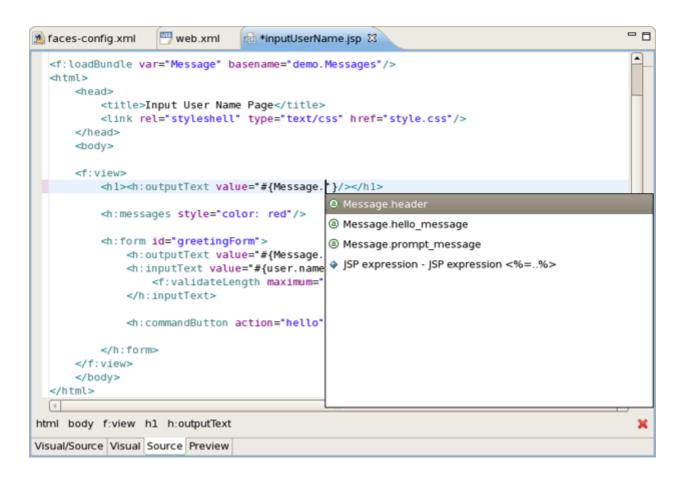
#### Figure 3.11. Content Assist in web.xml File

#### 3.1.2.1.2. Content Assist Based on Project Data

JBoss Developer Studio takes Content Assist to the next level. Studio will constantly scan your project and you will be able to insert code into the JSP page from your project that includes:

- Values from Property files
- "Managed beans" attributes and methods
- Navigation Rule Outcomes
- JSF variables (context, request etc...)

The first screenshot shows how to insert message from a Properties files. You simply put the cursor inside the *"value"* attribute and press *Ctrl-Space*. JBoss Developer Studio will scan your project and show a list of possible values to insert.



#### Figure 3.12. Inserting Message

In the following screenshot we are inserting a "Managed bean" attribute value. Again, by simply clicking *Ctrl-Space*, JBoss Developer Studio will show a list of all possible values that you can insert.

Once you select a Managed bean, it will show you a list of all available attributes for the selected Managed bean (userBean).

🔊 faces-config.xml 🛛 🖑 web.xml 🔯 *inputUserName.jsp 😂	- 0
<pre><f:view></f:view></pre>	=
Introl body f:view h:form h:inputText   Visual/Source Visual Source Preview	×

#### Figure 3.13. Attributes List

Code Assist based on project data will also prompt you for navigation rules that exist in your JSF configuration file.

		_
<h:commandbutton action="&lt;/td&gt;&lt;td&gt;" value="Say Hello!"></h:commandbutton>	Ξ	
	© #{user}	
	In the second	
	♦ JSP expression - JSP expression <%=%>	
html body f:view h:form h:commandButton		x
Visual/Source Visual Source Preview		

#### Figure 3.14. Code Assist

#### 3.1.2.1.3. Content Assist within Tree JSF Editor

JBoss Developer Studio also provides Content Assist when working within the Tree JSF configuration editor. Just click *Ctrl-Space.* 

faces-config.xml 🛿 🔛 we	b.xml 👜 inputUserName.jsp	• •
ces Config Editor		
faces-config	▼ Factories	
🔊 faces-config.xml*	Application-Factory:	m.su <u>B</u> rowse
Application Components	Faces-Context-Factory:	⊕ com.sun
a Components	Lifecycle-Factory:	com.sun.accessibility     Gom.sun.accessibility.internal
🕨 😽 Managed Beans	Render-Kit-Factory:	Com.sun.accessibility.internal.resources
Navigation Rules	▼ Lifecycle	🖶 com.sun.activation
le Referenced Beans and the Referenced Beans and the Render Kits and the Render Kits and the Reference Ref	phase-listener	com.sun.activation.registries
Validators		com.sun.activation.viewers
🧭 Extensions		com.sun.codemodel
		🖶 com.sun.codemodel.fmt
		Com sun codemodel util
		Down
ram Tree Source		

#### Figure 3.15. Content Assist in Tree JSF Configuration Editor

### 3.1.2.2. Struts Project Files

Content Assist features are available when you work with Struts projects.

#### 3.1.2.2.1. Content Assist for Struts Configuration File

Content Assist helps you in Struts Configuration file.

🐟 *struts-config.xml 🛛	- E
<pre><?xml version="1.0" encoding="UTF-8"?> <!DOCTYPE struts-config PUBLIC "-//Apache Softwa     "http://struts.ap"</pre>    </pre>	<pre>re Foundation//DTD Struts Configuration 1.2//EN ache.org/dtds/struts-config_1_2.dtd"&gt;</pre>
<pre><struts-config> <data-sources></data-sources> <form-beans> <form-bean <="" getname"="" getnameform"="" name="GetNameForm" path="/greeting" pre="" type="demo.GetNa &lt;/form-beans&gt; &lt;global-exceptions/&gt; &lt;global-forwards&gt; &lt;forward name="><td><pre>scope="request" type="demo.GreetingAction"&gt;</pre></td></form-bean></form-beans></struts-config></pre>	<pre>scope="request" type="demo.GreetingAction"&gt;</pre>
Attribute : attribute Data Type : CDATA	attribute     className     forward     id     include     input
	parameter     prefix     roles     suffix     unknown="true"

#### Figure 3.16. Struts Content Assist

#### 3.1.2.2.2. Content Assist for Struts JSP File

Using Code Assist in Struts JSP file is shown below.

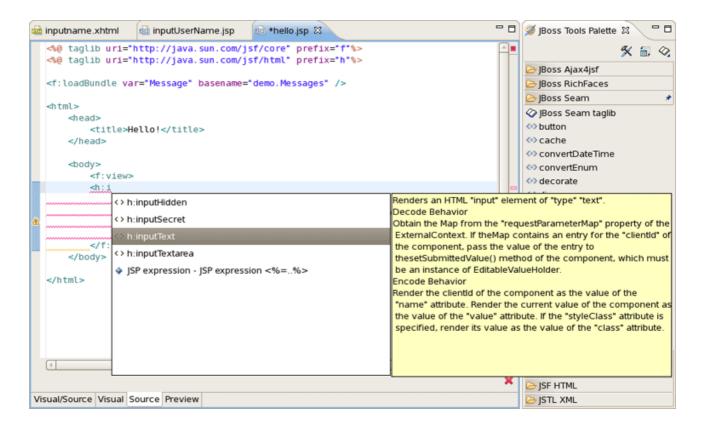
<bean :<="" th=""><th>/ 51</th><th></th><th></th></bean>	/ 51		
<table <t< td=""><td>&lt;&gt;&gt; bean:cookie</td><td>ŕ</td><td>Element : bean:cookie</td></t<></table 	<>> bean:cookie	ŕ	Element : bean:cookie
	<> bean:define		
</td <td>&lt;&gt; bean:header</td> <td></td> <td></td>	<> bean:header		
	<> bean:include		
	<> bean:message		
	<> bean:page		
<br <td>&lt;&gt; bean:parameter</td> <td></td> <td></td>	<> bean:parameter		
<td>&lt;&gt; bean:resource</td> <td></td> <td>-</td>	<> bean:resource		-
<% %>	<> bean:size		
	<> bean:struts		
G	11		
Micual/Course Micu		-	
Visual/Source Visu	al Source		

Figure 3.17. Struts JSP Content Assist

#### 3.1.2.3. JSP Pages

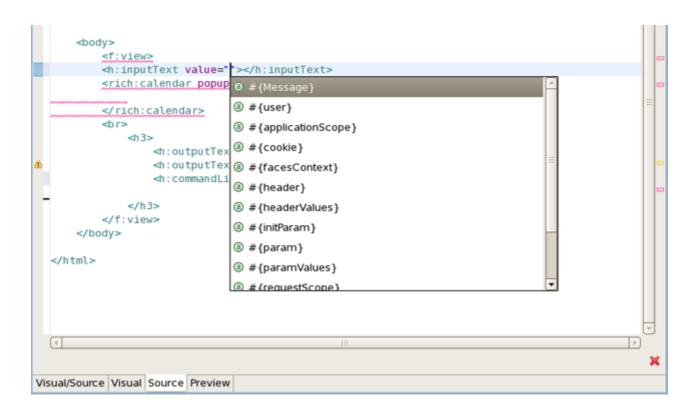
#### 3.1.2.3.1. Content Assist for JSF Tags

JBDS provides full code completion for JSF tags:



#### Figure 3.18. JSF Tags Content Assist

When the tag is selected the required attributes, if there any, are already inserted and the cursor is located to the first attribute. As this point you can ask for attribute proposals.



#### Figure 3.19. Attributes Content Assist

#### 3.1.2.3.2. Content Assist for JSTL Tags

-1+	able>			
<c:< th=""><th></th><th></th><th></th><th></th></c:<>				
	⇔ c:catch	-	Catches any Throwable that occurs in its body	
	<> c:choose		and optionally exposes it.	
<th>&lt;&gt; c:forEach</th> <th></th> <th></th> <th></th>	<> c:forEach			
	<> c:forTokens	Ξ		
	<> c:if			
	<> c:import			
	<> c:otherwise	-		
4	<> c:out			
	<> c:param			
	<> c:redirect	•		

#### Figure 3.20. JSTL Tags Content Assist

#### 3.1.2.3.3. Content Assist for HTML Tags

Content assist for HTML tags has the same mechanism as for JSF tags:

 din	dy>		
<u>s10</u>		Defines a form control for user input	
	<>> input	Defines a form control for user input	
	<> ins		
	JSP expression - JSP expression <%=%>		
۵.			
-			
<th></th> <th></th> <th></th>			
		v	
4	III	Þ	
		×	
Visual/Sour	ce Visual Source Preview		

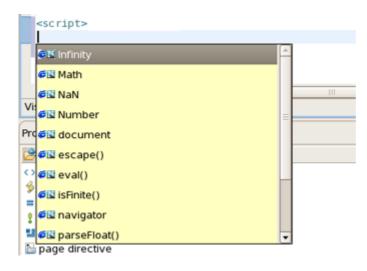
### Figure 3.21. HTML Tags Content Assist

You can use as well attributes proposals for HTML tags:

<body> <input ""<="" th="" type=""/><th>n ame=**&gt;</th><th></th></body>	n ame=**>	
<f:view> <rich:cal </rich:cal  </f:view>	"button"     "checkbox"     "file"     "hidden"     "image"     "password"     "radio"     "recent"	Attribute : type Data Type : ENUM Enumerated Values :         - text         - password         - checkbox         - radio         - submit         - reset         - file         - hidden         - image         - button
Visual/Source Visual So		

Figure 3.22. HTML Tags Content Assist

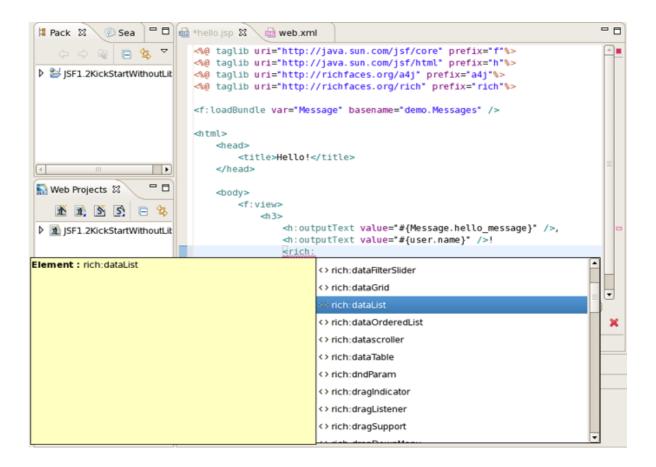
#### 3.1.2.3.4. Content Assist for JavaScript Tags



#### Figure 3.23. JavaScript Tags Content Assist

#### 3.1.2.4. Rich Faces components

JBDS indeed provides code completion for Rich Faces framework components. All you have to do is to install Rich Faces libraries into your project. See *here* [http://labs.jboss.com/file-access/ default/members/jbossrichfaces/freezone/docs/devguide/en/html/GettingStarted.html] how to install.



#### Figure 3.24. Content Assist for Rich Faces Components

- To insert a Rich Faces component on a page expand JBoss Rich Faces group on the palette
- Click on some component
- Put the needed attributes in the Insert Tag dialog and click Finish button

)	Insert Tag	9
rich:calendar> at	tributes	
Attribute name	Value	
jointPoint		
localValueSet		
ocale		
monthLabels		
monthLabelsShort		
optionalFooter		
optionalHeader		
popup	false	
preloadDateRangeBe	2g	
preloadDateRangeEr	nd	
rendered		
0		Einish Cancel

#### Figure 3.25. Insert Tag

The Rich Faces tag will be inserted on your page displayed in source and visual modes:

🗟 he	llo.js	spΣ	3	æ	web	.xm	I			- 8
• (	\$								¢ ♥ B I ∐ / A	
X	<<	<		July.	2008		>	>>	U	
4					Wed					
	27			1	2	3	4	5		
~~	28	6	7	8	9	10	11	12		
	29	13	14	15	16	17	18	19		
	30	20	21	22	23	24	25	26		
	31	27	28	29	30	31				
	07/				Т	xlay				
	He	llo,	#{ı		.na	me]	}!			
html	bo	dy	f:vi	ew						×
Visua	I/So	urce	Vis	sual	Sou	rce	Pre	viev	w	

#### Figure 3.26. Rich Faces Component

# 3.1.2.5. Adding dynamic code assist to custom components that were added to JBoss Tools Palette

Here is what you need to do to add project based code assist to a custom component added in JBoss Developer Studio:

 Create a new xml file in <JBDS\_home>studio/eclipse/plugins/org.jboss.tools.common.kb\_\*\*\*/ schemas/tld/. For example call it JeniaFaces.xml. The file should be written according to <JBDS\_home>/studio/eclipse/plugins/org.jboss.tools.common.kb/kb.jar/org/jboss/tools/ common/kb/kb-schema\_1.0.dtd

Here is how you set what is available for code assist:

• Adds code assist for JSF pre-defined objects, such as value= "#{param}":

```
<AttributeType ...>
<proposal type="jsfVariables"/>
</AttributeType>
```

• Adds bundle resource (property file) code assist:

```
<AttributeType ...>
<proposal type="bundleProperty"/>
</AttributeType>
```

• Adds managed bean property code assist:

```
<AttributeType ...>
<proposal type="beanProperty"/>
</AttributeType>
```

• Adds managed bean property but with a specified type:

```
<AttributeType ...>
<proposal type="beanProperty">
<param name="type" value="java.lang.Boolean"/>
</proposal>
</AttributeType>
```

• Adds managed bean method with a signature:

```
<AttributeType ...>
<proposal type="beanMethodBySignature">
<param name="paramType" value="javax.faces.context.FacesContext"/>
<param name="paramType" value="javax.faces.component.UIComponent"/>
<param name="paramType" value="java.lang.Object"/>
<param name="returnType" value="java.lang.Object"/>
</proposal>
</AttributeType>
```

2. Add information on your xml file in <JBDS\_home>/studio/eclipse/plugins/ org.jboss.common.kb\_\*\*\*/plugin.xml

```
<tld

jsf="true"

name="Jenia Faces"

schema-location="schemas/tld/myJSF.xml"
```

uri="http://www.jenia.org/jsf/dataTools"/>

3. Restart Eclipse. You should now have code assist for the component.

#### 3.1.3. Synchronized Source and Visual Editing

JBoss Developer Studio offers the flexibility to edit any files in either source or extra visual modes at the same time.

The project is yours and so is the source. JBoss Developer Studio provides you many different graphical editors to speed your application development. At the same time, you always have full control over all project source files. Any changes you make in the source view, will immediately appear in the graphical view.

The JSF configuration file editor has three views: Diagram, Tree and Source. All views are synchronized, you can edit the file in any view.

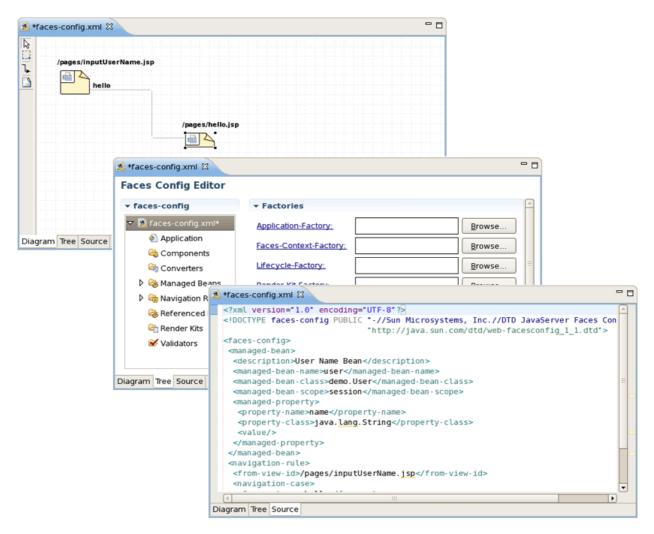


Figure 3.27. Three Views are Synchronized

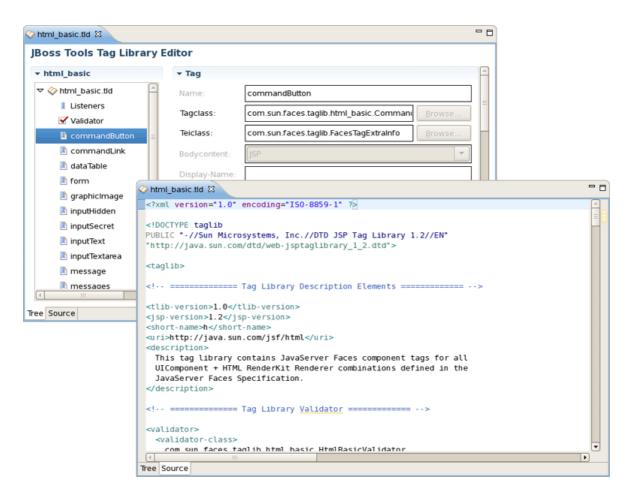
The same applies to all other JBoss Developer Studio editors.

Web XML editor is shown. Web XML editor has a graphical view (Tree) and source (Source).

📟 web.xml 🛙				- 8		
Web XML Editor						
▼ web		- Web Descript	or 2.4	Ĥ		
	<pre></pre> <pre>&lt;</pre>	Name: Display-Name: Description: xml & ht version="1.0" -app version="2. mis:xsi="http://w. splay-name>JSFki ontext-param> baram-name>Javax. varam-value>serve: context-param> istener> .istener .istene	<pre>web JSFkickstart  JFkickstart  ////interfactors ////interfactors ////interfactors ////interfactors ////////////////////////////////////</pre>		="http://java.sun.co	
	<u <lo <a< td=""><td><pre>servlet-name&gt;Face irl-pattern&gt;*.jsf servlet-mapping&gt; ogin-config&gt; nuth-method&gt;BASIC ogin-config&gt;</pre></td><td></td><td></td><td></td><td>•</td></a<></lo </u 	<pre>servlet-name&gt;Face irl-pattern&gt;*.jsf servlet-mapping&gt; ogin-config&gt; nuth-method&gt;BASIC ogin-config&gt;</pre>				•
	Tree So	ource			Þ	

#### Figure 3.28. Two Views are Synchronized

JBoss Developer Studio TLD file editor shown in Tree view. At any point you can edit the source by going switching to Source view.



#### Figure 3.29. Two Views are Synchronized

## 3.2. Visual Page Editor

JBoss Developer Studio comes with a powerful and customizable Visual Page Editor (VPE). You can use the Visual Page Editor to develop an application using any technology: JSF, Struts, JSP, HTML and others.

Current VPE version has four tabs: Visual/Source, Visual, Source and Preview.

Split screen design with instant synchronization between source and visual views:

🗟 cu	stomerinfo.jsp 🛛		- 8
	<h:outputtext td="" va<=""><td></td><td>-</td></h:outputtext>		-
	<pre><h:inputtext id<br=""><f:valuechan </f:valuechan </h:inputtext></pre>	alue="#{bundle.firstLabel}" /> d="firstName" value="#{customer.firstName}" required="true"> ngeListener type="carstore.FirstNameChanged" /> eClass="validationMessage" for="tirstName"/>	=
	<h:outputtext v<br=""><h:inputtext id<="" td=""><td>alue="#{bundle.middleLabel}" /&gt; ="middleInitial" size="1" maxlength="1" #{customer.middleInitial}" &gt;</td><td>•</td></h:inputtext></h:outputtext>	alue="#{bundle.middleLabel}" /> ="middleInitial" size="1" maxlength="1" #{customer.middleInitial}" >	•
4			•
•	\$		
≶X ⊚ ₽	CAR DE	мо	
	Please fill in your na	me and address.	
	Title	#{customer.titleOptions} 👤	
	First Name	# {customer.firstName} 🖌 Error Message	
	Middle Initial	#{c Error Message	
	Last Name	#{customer.lastName}	•
	-	n h:panelGrid h:panelGrid h:inputText	×
	First Name Middle Initial Last Name	# {customer.firstName} Error Message # {c Error Message # {customer.lastName} m h:panelGrid h:panelGrid h:inputText	

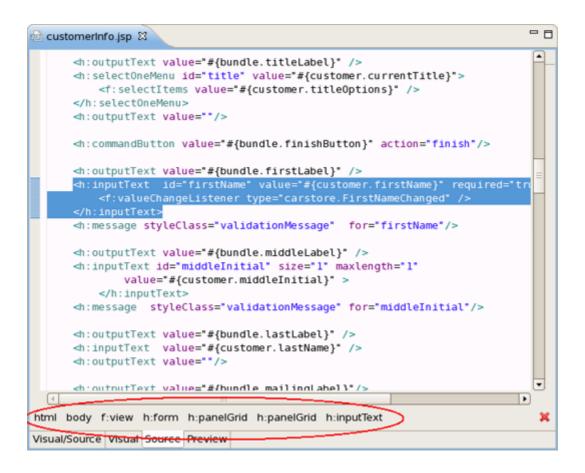
### Figure 3.30. Visual/Source View

You can also switch to pure Visual design:

🗟 cus	tomerinfo.jsp 🛙				- 0
• [ % ]	\$	‡ ][   ‡	BIU	L. A.	
49 R	CAR DE	мо	1020	000	6
	Please fill in your na	me and address.			
	Title	#{customer.titleOptions} 💌			
	First Name	#{customer.firstName}	Error Message		
	Middle Initial	#{c	Error Message		
	Last Name	#{customer.lastName}			
	Mailing Address	#{customer.mailingAddres			=
	City	#{customer.city}			
	State	AL 🗾			
	Zip Code	#{customer.zi;	Error Message		
	Credit Card Number		Error Message		
	Expiry Date	01 🔽 2002 💌			
	Finish				
	2				
	Thanks for stopping	by!			•
Ē					•
html	body f:view h:form	n h:panelGrid h:panelGrid h	inputText		×
Visual	/Source Visual Source	e Preview			

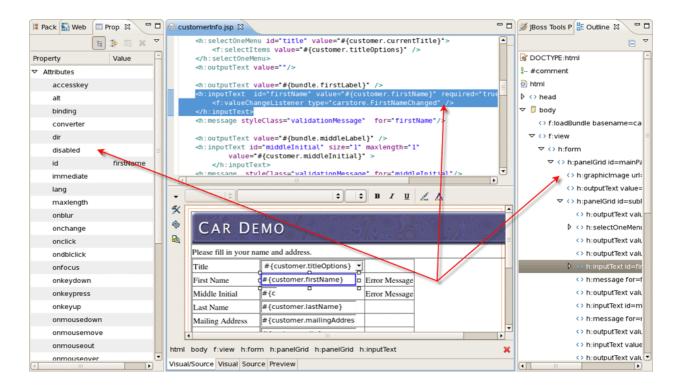
#### Figure 3.31. Visual View

Or work just in Source view. Note that selection bar is now available not only in Visual mode but also in Source one:



#### Figure 3.32. Source View

No matter what view you select, you get full integration with Properties and Outline views:



## Figure 3.33. Properties And Outline Views

Preview mode is read-only, it only shows how the page will look like in a browser.

customerinfo.jsp	B		- c
CAR DE		620	050
Please fill in your na	me and address.		
Title	#{customer.titleOptions} 💌		
First Name	#{customer.firstName}	Error Message	
Middle Initial	#{c	Error Message	
Last Name	#{customer.lastName}		
Mailing Address	#{customer.mailingAddres		
City	#{customer.city}		:
State	AL 🔽		
Zip Code	#{customer.zip	Error Message	
Credit Card Number		Error Message	
Expiry Date	01 - 2002 -		
Finish			
<u></u>			
Thanks for stopping	by!		
•			
<ul> <li>Visual/Source Visua</li> </ul>			•
visual/source visua	Source Treview		

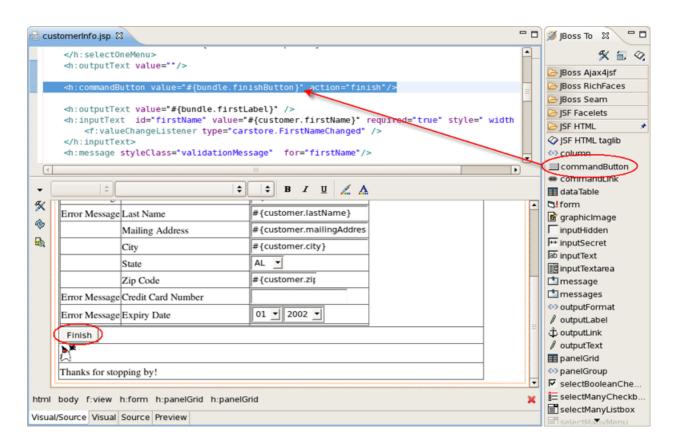
# Figure 3.34. Preview Mode

Use the graphical toolbar to add inline styling to any tag.

4		<pre>leClass="validationMessage" for="firstName"/&gt; value="#{bundle.middleLabel}" /&gt;</pre>	•
$\leq$	\$	⇒ ⇒ B <i>I</i> U <i>L</i> A	
≪ ⊚ ₽	CAR D	ЕМО	
	Please fill in your r		
	Title	#{customer.titleOptions}	
	First Name	#{customer.firstName}	
	Middle Initial	#{c Error Message	-
	1		Þ
html	body f:view h:fo	rm h:panelGrid h:panelGrid h:inputText	×
Visua	al/Source Visual Sou	Irce Preview	

Figure 3.35. Graphical Toolbar

With just a click or drag-and-drop insert any tags from the palette on to the page you are editing.



#### Figure 3.36. Inserting Tags From Palette

## 3.2.1. Advanced Settings

There are three buttons on the Visual Page Editor left side:

\$	<b>\$</b> ]	🗘 B I 🗓 🔏 🛕
CAR I	DEMO our name and address.	20200
Title	#{customer.titleOptions}	
Finish	First Name	#{customer.firstName}
Error Message	Middle Initial	#{c
Error Message	Last Name	#{customer.lastName}
	Mailing Address	#{customer.mailingAddres
	City	#{customer.city}
	State	AL -

# Figure 3.37. Visual Page Editor Buttons

• Preferences

Provides quick access to Visual Page Editor preferences.

9	Preference	s (Filtered) X
type filter text	Visual Page Editor	\$- \$-
	General Templates	
✓ Web ✓ Editors		Show Border for Unknown Tags
Visual Page Editor		Show Resource Bundles Usage as EL Expressions
		☑ Always Prompt for Tag Attributes During Tag Insert
		☑ Show Selection Tag Bar
		Always Hide Selection Bar Without Prompt
	Default Editor Tab:	Visual/Source
<		Restore Defaults Apply
0		OK Cancel

# Figure 3.38. Visual Page Editor Preferences Window

Refresh

Refresh displaying information with this button.

Page Design Options

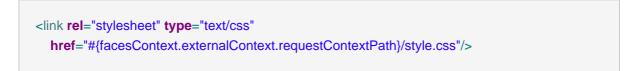
This button leads to page design options.

References to Resources	
sign Options	0
-Time Absolute Folder	
	<u>C</u> hange
Page	•
-Time Relative Folder	
	<u>C</u> hange
Page	•
CSS File Path	
CSS File Path	Add
CSS File Path	Edit
CSS File Path	
CSS File Path	Edit
	Edit Remove
	-Time Absolute Folder Page -Time Relative Folder

#### Figure 3.39. Page Design Options

This dialog let's you set resources which are usually only resolved in runtime. To set a stylesheet, click *Add* (for CSS File Path section) and add your stylesheet. It works when CSS is defined on your page in the following way:

#### Code:



This will work fine in runtime, but the Visual Page Editor doesn't know what requestContextPath in design time is.

The next section (URI), let's you add URI taglibs if you are using includes so that the editor knows where to find the tag libraries.

The first two sections let you define actual runtime folders. Here is an example.

Let's say you have the following project structure:

WebContent/ pages/ img/ a.gif header.jsp main.jsp

header.jsp content:

My Header <img src="img/a.gif"/>

main.jsp:

<jsp:include page="pages/header.jsp" />

When you open *main.jsp* in Visual Page Editor, it will not be able to resolve the image from the header, however, it will work fine in runtime. To fix this in design time, click the *Page Design Options* button and set *Actual Run-Time Relative* Folder to *<project>WebContent > pages* and you will see the image appear.

• Hide Selection bar

By clicking on the component in Visual view or selecting a code snippet in Source mode you can see the tags tree. If you want to hide the selection bar, use the "Hide Selection Bar" button on the lower right side.

•	\$	(\$)	
<ul><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li><li></li></ul>	CAR	Dемо	
	Please fill in y	our name and address.	
	Title #{customer.titleOptions}		
	Finish	First Name	#{customer.firstName
	Error Message	Middle Initial	#{c
	Error Message	Last Name	# {customer.lastName
		Mailing Address	# {customer.mailingAd
Ŀ	•	111	
html	body f:view	h:form h:panelGrid h:panelG	rid 🗙
Visua	l/Source Visual	Source Preview	$\sim$

## Figure 3.40. Selection Bar

# 3.2.2. Setup notes for Linux

## 3.2.2.1. How to Start the Visual Page Editor under Linux

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

The Visual Page Editor requires the library libstdc++.so.5. This library is contained in the compatlibstdc++-33.i386 package.

• To install this package on Fedora Core or Red Hat Enterprise Linux run the following command:

yum install compat-libstdc++-33.i386

• On any other rpm based distributions download libstdc++.so.5 and run the following command:

rpm -Uvh compat-libstdc++-33.i386

On Debian based distributives run the following command:

apt-get install compat-libstdc++-33.i386

In case you have the library installed and you still have issue with starting the visual page editor then close all browser views/editors and leave one visual page editor open and restart eclipse. This should force a load of the right XULRunner viewer.

# 3.2.3. JSP syntax validation

When working in JBoss Tools JSP editor you are constantly provided with feedback and contextual error checking as you type.

# 3.2.4. JSP Page Preview

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

JSP preview is available for:

- Struts Pages
- JSF Pages

The preview features are available with Visual Page Editor.

# **3.3. More Editors**

Besides Visual Page Editor JBDS is supplied with a huge range of various editors for different file types: properties, TLD, web.xml, tiles, and so on.

# 3.3.1. Graphical Properties Editor

The Properties editor allows you to work in two different modes and also supports unicode characters.

To create a new properties file, in the Package Explorer view, select *New > Properties File* from the right-click context menu on the folder where you want to create the file.

Ne <u>w</u>	Þ	🖹 JSF Project
Go Into		Struts Project
Open in <u>N</u> ew Window		🎦 Project
Ope <u>n</u> Type Hierarchy	F4	🍄 Package
Sho <u>w</u> In	Shift+Alt+W	🞯 Class
<u>[]</u> еору	Ctrl+C	Interface Source Folder
Copy Qualified Name	Ctrl+V	😂 Folder
X Delete	Delete	📑 File 🐳 Struts Config
<u>B</u> uild Path	•	1 Faces Config
<u>S</u> ource	Shift+Alt+S ▶	🖹 Tiles File
Refactor	Shift+Alt+T 🕨	🕙 Validation File
import		💸 TLD File
Export		📸 JSP File
Export		🚵 XHTML File
🔘 Run XDoclet	Shift+Ctrl+F1	🚵 HTML File
🔗 Refresh	F5	🚳 CSS File
Clo <u>s</u> e Project		🖄 JS File
Close Unrelated Projects		📸 Properties File
Assign Working Sets		📑 E <u>x</u> ample
<u>R</u> un As	•	Cther
<u>D</u> ebug As	•	
Profile As	•	

# Figure 3.41. Selecting Properties File

You can edit the file using a table-oriented "Properties" viewer:

🔲 message.properties	s 🖾	- 6
message.properties		
name	value	Add
header	Hello Demo Application	
prompt_message	Name:	Edit
hello_message	Hello	Delete
		Up
		Do <u>w</u> n
Properties Source		

## Figure 3.42. "Properties" Viewer

You can also use a Source viewer for editing the file:

🖬 message.properties 🕱	- 0
message.properties	
<pre>header=Hello Demo Application prompt_message=Name: hello_message=Hello</pre>	<u>^</u>
	=
	v
	•
Properties Source	

#### Figure 3.43. Source Viewer

# **3.3.2. Graphical TLD Editor**

The TLD editor comes with same features you will find in all other JBoss Developer Studio editors:

- Graphical and source edit modes
- Validation and error checking

#### 3.3.2.1. Tree view

⊘html_basic.tld 怒		- C
JBoss Tools Tag Library Edit	or	
w html_basic	▼ Tag Library	A
🗢 🔷 html_basic.tld	Tlibversion:	1.2
Listeners	Shortname:	h
Validator	URI:	http://java.sun.com/jsf/html
CommandButton      CommandLink	Display-Name:	
dataTable		
👂 🖹 form	Small-Icon:	
👂 🖹 graphicImage	Large-lcon:	
InputHidden	Description:	his tag library contains JavaServer Faces
<ul> <li>InputSecret</li> <li>InputText</li> </ul>		
Inputiext	- Defined	
Imposed in the state of the	name	tagclass Add
👂 🖹 messages	commandButte	c com.sun.faces.taglib.html_b
I outputFormat	commandLink	com.sun.faces.taglib.html_b
Tree Source		

## Figure 3.44. Tree View

#### 3.3.2.2. Source view

You can easily switch from Tree to Source by selecting the Source tab at the bottom of the editor.



Figure 3.45. Source View

You can easily add a new tag:

⊘MyTLD.tld ⊠			- 8
JBoss Tools Tag Lil	orary Editor		
▼ MyTLD	+ Tag I	Library	A
V VILD.tld	Rename		
Validator	New 🔸	Validator	
I choose I out	Properties	Listener //java.sun.com/jsp/jstl/xml1	
▷ 🚺 ir	Displa Small-it	Tag File	$\exists$
<ul> <li>forEach</li> <li>param</li> </ul>	Large-I	icon:	
Iset	Descrip	ption: My Tag Library	
Functions	▼ Defir	(	
	name	tagclass	id
	choos	<u>R</u> em	iove
	out	org.apache.taglibs.standard.tag.rt.xml.ExprTag org.apache.taglibs.standard.tag.common.xml.lfTag	sit
4 111	+ forEad		lp 🗸
Tree Source			

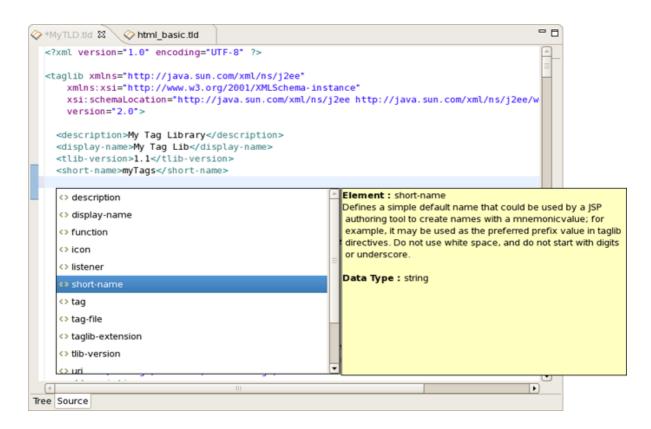
# Figure 3.46. Adding a New Tag

You can also easily add a new attribute to an existing tag:

JBoss Too	ls Tag Library Ed	litor		
▼ MyTLD		▼ Tag		<b>^</b>
▼ 🔷 MyTLD I Liste		Name: <u>Tagclass:</u>	if org.apache.taglibs.standard.tag.common.xml.lfTa Browse	
🗹 Vali	ose	Teiclass:	Browse	5
I out		Bodycontent:	JSP	
<ul> <li>If</li> &lt;</ul>	Add Attribute	ragment		
	<b>Copy</b>	Ctrl + C		
	X Delete	Delet	e	
	Properties			-
4 111	Þ	name	required Add	

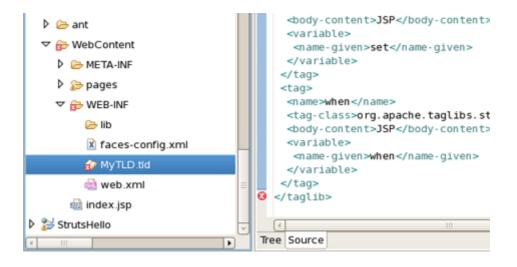
# Figure 3.47. Adding a New Attribute

Content assist is available when editing the file using the Source viewer:



#### Figure 3.48. Content Assist

In the Source viewer, if at any point a tag is incorrect or incomplete, an error will be indicated next to the line and also in the Problems view below.



#### Figure 3.49. Error Reporting

# 3.3.3. Graphical Web Application File (web.xml) Editor

The Web Application File editor comes with the same features you will find in all other JBoss Developer Studio editors:

- Graphical and source edit modes
- Validation and error checking

## 3.3.3.1. Tree View

> MyTLD.tld 🔤 web.xml 🛙				- t
Web XML Editor				
- web	✓ Serviet			-
マ 🗟 web.xml	Servlet-Name:	Faces Servlet		
Context Params Filters	Servlet-Class:	javax.faces.webapp.f	aces Browse	
A Listeners	Load-on-Startup:	1		
	▼ Init Params			
Faces Servlet:javax.faces.	param-name	param-value	Add	
Faces Servlet:*.jsf	paramename	pararri-value		
😭 session-config			<u>R</u> emove	
襑 Mime Mappings			<u>E</u> dit	
a welcome-file-list			Up	
襑 Error Pages			<u>Ob</u>	
🖉 JSP Config			Down	
a Security Constraints	- Security Roles			
🧈 login-config	role-name	role-link	Add	
😂 Security Roles	Tore Harrie	TOTE IN R		
😂 Env Entries			<u>R</u> emove	
😂 EJB			<u>E</u> dit	
😂 Services			Up	
😂 Resources				
😂 Message Destinations			Down	
🖋 locale-encoding-mapping-list	- Advanced			
	Small-Icon:		<u>C</u> hange	.
ee Source				

#### Figure 3.50. Tree View

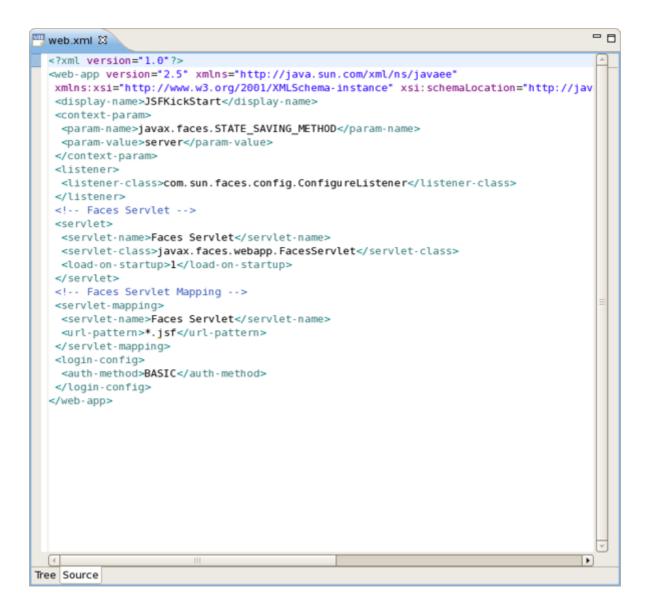
You can add any new elements right in the Tree viewer:

web		- Web Des	criptor 2	.5	
💩 web.xml 🚬		Managa	wah		
👂 🔄 Context F	New	•	× Conte	ext Param	
Rilters	Rename		脊 Filter.		
👂 🔄 Listeners	Heridi He		👸 Filter	Mapping	(
	📱 Change Timestam	np	🚯 Lister		
<ul> <li>Faces</li> </ul>	Copy	Ctrl + C	Servi		
🖏 Faces 🕯		Ctrl + X		et Mapping	
Session-c	Paste	Ctrl + V		Mapping	Add
w session-c			Error	2	
🗟 Mime Maj	Properties			rity Constraint	<u>R</u> emove
a) welcome-f	ile-list			rity Role	Edit
襑 Error Pages	;		🔉 Env E		
🥖 JSP Config			🕈 Ejb Ri		<u>Ш</u> р
🗟 Security Co	onstraints			ocal Ref	Down
🧈 login-config	9		🍇 Servi	ce Ref urce Env Ref	
🗟 Security Ro	les			urce Env Ref	
ຊ Env Entries		Small-Icon:			<u>C</u> hange
🔄 EJB		Large-lcon:		age Destination Ref == age Destination	Change
a Services		Bischie a sta			
Resources		Distributable	-	no	
a Message D	estinations	Metadata-Co	mplete:		
+ •	oding-mapping-list	-			
p locale enco	sourd-mapping-use	Encoding:			

# Figure 3.51. Adding New Elements

## 3.3.3.2. Source View

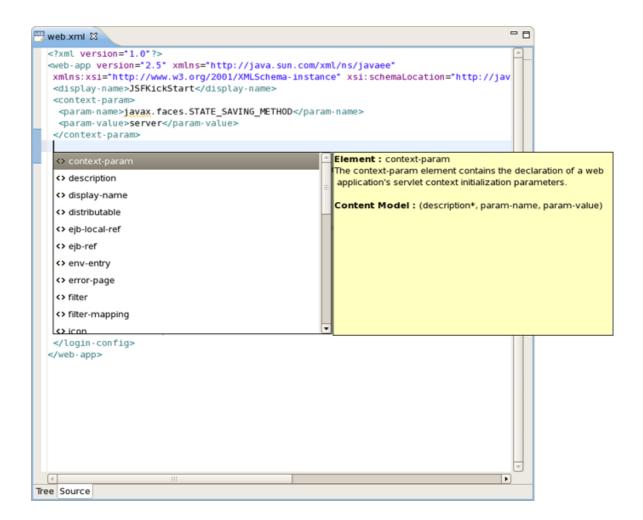
Switch to the Source viewer to edit the web.xml file by hand at any time:



#### Figure 3.52. Source View

#### 3.3.3.3. Content Assist

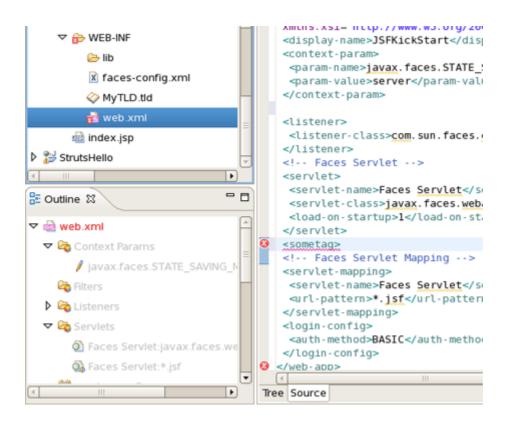
Content assist is available in the Source viewer. Simply click CTRL-Space anywhere in the file.



## Figure 3.53. Content Assist

# 3.3.3.4. Errors Checking and Validation

If errors occur anywhere in the file, small red dots will appear next to the lines where the errors occurred. Also, note that the file is marked by a small x in the Package Explorer view.



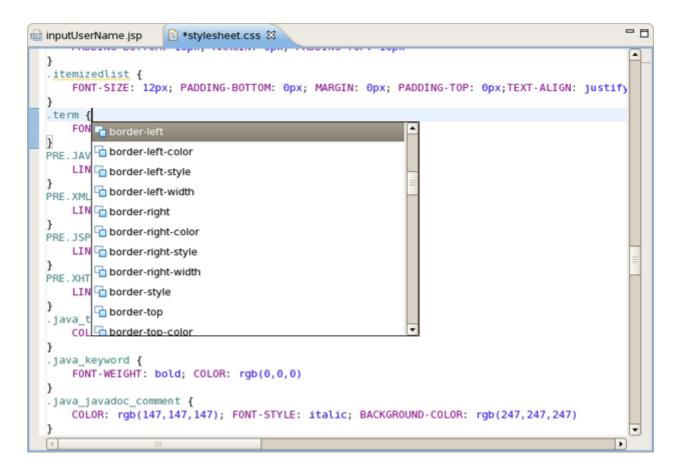
#### Figure 3.54. Errors Reporting

# 3.3.4. CSS Editor

The CSS editor comes with the same features you will find in all other JBoss Developer Studio editors.

- Content assist
- Validation and error checking

With the CSS (Cascading Style Sheet) editor, you can take advantage of code prompting:



#### Figure 3.55. CSS Editor

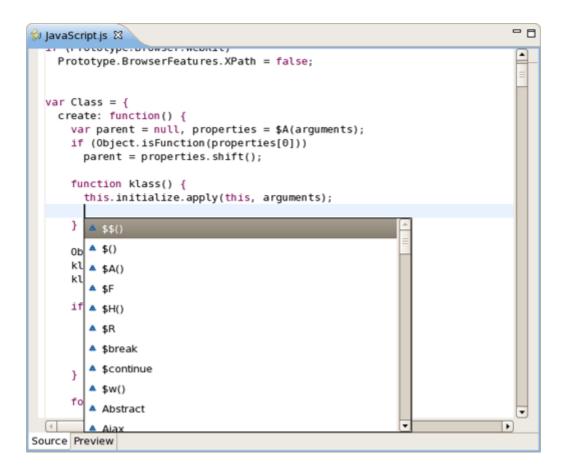
And you can also use the Properties view next to the editor to edit existing stylesheet declaration properties:

inputUserName.jsp	- 8	Properties 23
text-align: center;	<b>A</b> _	🗉 🕸 🕷 🗶
)		Property Value
.mediaobject{		Aural
<pre>padding : 5px 10px 5px 35px; }</pre>		Box model
H1 {	=	Colors and Background:
MARGIN: 0px; FONT-SIZE: 22px; COLOR: #ff6600; PADDING: 45px 0px 1 }	10px	
H2 {		font
<pre>MARGIN: 0px; FONT-SIZE: 18px; COLOR: #2a7bd4; PADDING: 25px 0px 1 }</pre>	10px	font-family
НЗ {		font-size 15px
MARGIN: 0px; FONT-SIZE: 15px; COLOR: #000000; PADDING: 20px 0px 1 }	10px	font-size-adjust
H4 {		font-stretch
MARGIN: 0px; FONT-SIZE: 12px; COLOR: #000000; PADDING: 15px 0px 1 }	10px	font-style
H5 {		font-variant
MARGIN: 0px; FONT-SIZE: 12px; COLOR: #000000; PADDING: 15px 0px 1 }	10px	font-weight
л Нб {		Generated content/List
MARGIN: 0px; FONT-SIZE: 11px; COLOR: #000000; PADDING: 5px 0px 0p	ox Op	Paged media
3		▶ Tables
div.book div.section div.mediaobject{		Þ Text
<pre>text-align:left; }</pre>	-	User interface
	•	Visual

## Figure 3.56. Properties View

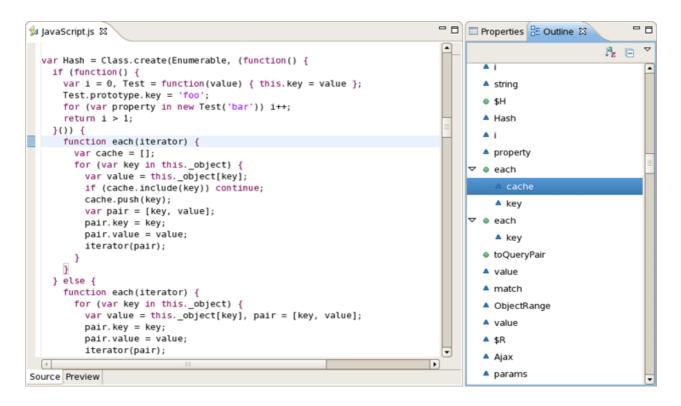
# 3.3.5. JavaScript Editor

The JavaScript editor includes a Preview viewer and a Source viewer. In the Source viewer, you can use code assist:



#### Figure 3.57. JavaScript Editor

You can also use the Source viewer with the Outline view to navigate around the file:



#### Figure 3.58. Source Viewer

## 3.3.6. XSD Editor

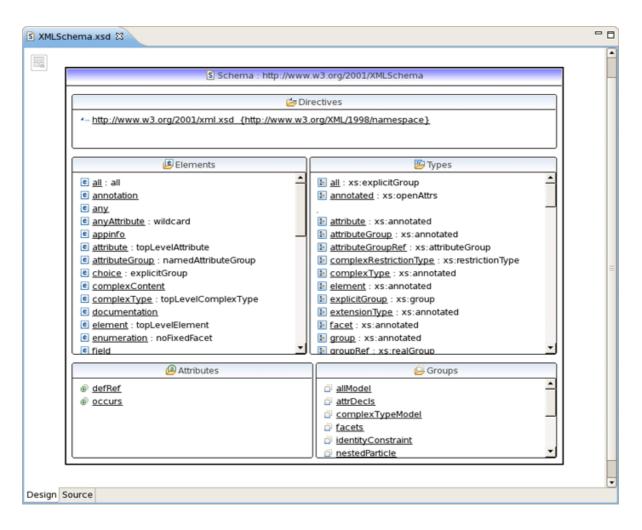
JBoss Developer Studio comes with an XSD Editor for XML Schema files. This editor comes from the Web Tools Project (WTP) (see *WTP Getting Started* [http://www.eclipse.org/webtools/ testtutorials/gettingstarted/GettingStarted.html]).

To create a new XSD file, right-click a folder in the Package Explorer view, select *New > Other...* from the context menu and then select *XML > XML Schema* in the dialog box.

<b>@</b>	New	×
Select a wizard		
Create a new XML schema file		
<u>W</u> izards:		
type filter text		
· · · · · · · · · · · · · · · · · · ·		
User Assistance		
👂 🗁 Web		
Web Services		
🗢 🗁 XML		
📑 DTD		
XML		
S XML Schema		=
👂 🗁 Examples		-
-		
O < Back	Next > Einish	Cancel
⑦ < <u>B</u> ack	<u>N</u> ext > ⊟nish	Cancer

# Figure 3.59. Creating New XSD file

The XSD Editor includes two viewers for working on the file, a Design viewer and a Source viewer:



#### Figure 3.60. Source Viewer

In the Design viewer, you can drill down on an element by double-clicking on it:

(MLSchema.xsd 🛛		
R		
choice	explicitGroup	
	Iname NCName	
	(a) ref QName	
	🔞 anyAttribute	
	e group groupRef	
	element localElement	
	Choice explicitGroup	
	🚛 sequence explicitGroup	
	🖉 any (anyType)	

# Figure 3.61. Design Viewer

Various edit options are available when you right-click an element in the diagram:

# Figure 3.62. Edit Options

You can also use the Properties view to edit a selected element:

S XMLSchema.xsd	x							- 8
e choice			🛃 explicitGroup					1
				③ name		NCName		
				In the second		QName		
				🔞 anyAttribute				
				🚽 annotation	[01]	(annotationType)	7	
			Г	📧 group		groupRef		
				element 🕘		localElement		=
			<sup></sup>	🚽 choice		explicitGroup		
				🖉 sequence		explicitGroup		
				🐙 any		(anyType)		
								*
Design Source								<u> </u>
Properties	_							~ - 0
③ attribute								
General	Name:	ref						<b></b>
Constraints	Type:	xs:Ql	Name					•
Documentation	Usage:							•
Extensions								

# Figure 3.63. Properties View

You can also use a Source viewer for the file. In this viewer, along with direct editing of the source code, you can also edit the file by using the Properties view on the right:

MLSchema.xsd 🛛		F	roperties 🕄			~ (	-
<pre><xs:attribute name="maxOccurs" td="" type="xs:allNNI" use="optional&lt;/td&gt;&lt;td&gt;1" 🖻<=""><td>0</td><td>attribute</td><td></td><td></td><td></td><td></td></xs:attribute></pre>	0	attribute					
waystributeCrown, name-IdefDeft		Ge	neral	Name:	ref		J
<pre><xs:attributegroup name="defRef">     <xs:annotation></xs:annotation></xs:attributegroup></pre>	-	Co	nstraints	Type:	xs:QName	-	ĺ.
for element, group and attributeGroup,	=			.ype.			4
which both define and reference <td>ta</td> <td>Do</td> <td>cumentation</td> <td>Usage:</td> <td></td> <td>•</td> <td>L</td>	ta	Do	cumentation	Usage:		•	L
<pre><xs:attribute name="name" type="xs:NCName"></xs:attribute></pre>		Ext	ensions	-			6
<pre><xs:attribute name="ref" type="xs:QName"></xs:attribute></pre>				1			
<pre><xs:group name="typeDefParticle"></xs:group></pre>							
<xs:annotation></xs:annotation>							
<xs:documentation></xs:documentation>	1						
'complexType' uses this							
<xs:choice></xs:choice>							
<xs:element name="group" type="xs:groupRef"></xs:element>							
<xs:element ref="xs:all"></xs:element>							
<xs:element ref="xs:choice"></xs:element>							
<xs:element ref="xs:sequence"></xs:element>	1						
<xs:group name="nestedParticle"></xs:group>							
<xs:choice></xs:choice>							
<pre><xs:element name="group" type="xs:groupRef"></xs:element></pre>	-						
	Þ						
ign Source							

Figure 3.64. Source Viewer

# 3.3.7. Support for XML Schema

JBoss Developer Studio fully supports XML files based on schemas as well as DTDs:

	web.xml 🕅
	xml version="1.0"?
	<pre><web-app <="" pre="" version="2.4" xmlns="http://java.sun.com/xml/ns/j2ee"></web-app></pre>
	xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="ht
	<description>Facelets StarterKit</description>
	<pre><display-name>myFaces</display-name></pre>
	<context-param></context-param>
	<param-name>javax.faces.DEFAULT_SUFFIX</param-name>
	<pre><param-value>.xhtml</param-value></pre>
	<context-param></context-param>
	<param-name>facelets.REFRESH_PERIOD</param-name>
	<param-value>2</param-value>
	<context-param></context-param>
	<param-name>facelets.DEVELOPMENT</param-name>
	<param-value>true</param-value>
	<context-param></context-param>
	<param-name>javax.faces.STATE_SAVING_METHOD</param-name>
	<param-value>client</param-value>
	<context-param></context-param>
	<param-name>com.sun.faces.validateXml</param-name>
	<param-value>true</param-value>
	<context-param></context-param>
	<param-name>com.sun.faces.verifyObjects</param-name>
	<pre><param-value>true</param-value></pre>
Tr	ee Source

Figure 3.65. XML File

# **JBoss Tools Palette**

This chapter will introduce you to the functionality provided by JBoss Tools Palette. The Palette allows you to quickly and easily create your JSP or JSF pages. Now you can do it more faster without additional knowledge.

The JBoss Tools Palette allows you to:

- Insert tags into a JSP or JSF page with one click
- Add custom and 3rd party tags

The JBoss Tools Palette contains a developer's project tag libraries and provides possibility to add any tag libraries to it. Also you can choose a necessary one from the list of already existed tag libraries:

- HTML
- JBoss
- JSF
- JSTL
- MyFaces
- Oracle ADF Faces
- Struts
- XHTML



#### Figure 4.1. Default View of The JBoss Tools Palette

By default the Palette is represented in Web Development Perspective with four groups. If you can't see it, select *Window > Show View Other... > JBoss Tools Web > JBoss Tools Palette* from the menu bar.

# 4.1. Palette Options

To facilitate your work, you can configure the Palette in your own way, by selecting the corresponding icon on the Palette toolbar.

There is a possibility to configure the JBoss Tools Palette:

- · to edit the palette content by adding, removing or changing the palette elements
- to show/hide groups, subgroups
- to import groups, subgroups

🧭 JBoss Tools Palette 😫 🦳 🗖
🗶 🖩 🥎
🕞 JBoss Ajax4jsf
🕞 JBoss RichFaces
🕞 JBoss Seam
▷ JSF Core
▷ JSF Facelets
JSF HTML
🕞 JSF html 🔹 🖈
column
commandButton
commandLink
<> dataTable
form
graphicImage
inputHidden
inputSecret
inputText
inputTextarea
message
🐼 messages 🔻

#### Figure 4.2. Palette Buttons

#### 4.1.1. Palette Editor

JBoss Tools Palette contains existing libraries of tags, thus the Palette editor is intended to work with them or create your new one, as well.

To open the editor, click on the Palette Editor icon:



Figure 4.3. Palette Editor Icon

The window has two parts. There is a reflected grouped list of components on the left side of the palette editor. Each group is divided into multiple groups, every of which is a tag library. The right side of the palette editor is an editing window where it's possible to change values of group or tag library attributes that you've chosen on the left part of the window.

It can also be done by right click and using *Edit...* option.

For example, JSF group consists of Core, Facelets, HTML tag libraries and the attributes as name, description and hidden which are available for editing:

<b>9</b>	Palette Editor							
▽ 🗁 XStudio	Ĥ	name	value					
👂 🌇 Icons		element type	group					
▽ 🚿 Palette		name	JSF					
D 😑 HTML		description						
🕨 🔒 JBoss	=	hidden	no					
🔻 🚔 JSF								
👂 🐸 Core								
👂 🐸 Facelets								
D 😂 HTML								
Þ 😑 jstl								
🕨 🔒 MyFaces								
				OK Cancel				
<ul> <li>▷ ➢ Facelets</li> <li>▷ ➢ HTML</li> <li>▷  JSTL</li> <li>▷  MyFaces</li> </ul>	•			OK Cancel				

#### Figure 4.4. Tag Libraries of the JSF Group

The Palette Editor provides the following possibilities when working with existing tags or icons:

• to work with a set of icons

*Icons* is the root folder for the icon sets. The first step is creating the icon set. Right click on the *Icons* folder and select *Create* > *Create* Set...

Set the value of the name in the *Add Icons* window and click *Finish* button. A new element will appear in the list.

<b>9</b>	Palette	Editor	•
▼ 🗁 XStudio	name	value	
🗢 🍇 Icons	element type	partition	
له <mark>Create</mark>	•	🌋 Create Set	
Paste	Ctrl + V		
👂 🍇 Frames 📄		1	
👂 🏪 JSF Core			
Lists			
👂 鞼 Struts Bean			
👂 鞼 Struts Common 🔄			
Struts HTML			
👂 鞼 Struts Logic 📃			
			OK Cancel

#### Figure 4.5. Creating a Set of Icons

Also you can delete the set. Right click on the set of icons that you wish to remove and chose the *Delete Set* option from the pop-up menu or click the *Delete* keyboard button.

• to edit icons in the chosen set

When the set of icons is created, new icons can be imported to it. Choose the required set and select the option *Create > Import Icon...* from the pop-up menu that appears after you right-click on a folder.

9		Palette I	Editor		×
マ 🗁 XStudio マ 職 Icons ト 積 Commo	n	name element type		lue t of icons	
<ul> <li>Fonts</li> <li>Frames</li> <li>Frames</li> <li>JSF Cor</li> <li>Lists</li> <li>Lists</li> <li>Tuts E</li> <li>Struts C</li> <li>Struts H</li> <li>Struts List</li> </ul>	Cut Set	c	ttrl + C ttrl + X ttrl + V Delete	Create Set	
				ОК	Cancel

#### Figure 4.6. Creating Icons

Set the name of the icon and the path and click *Finish* button.

• to work with a group of tag libraries

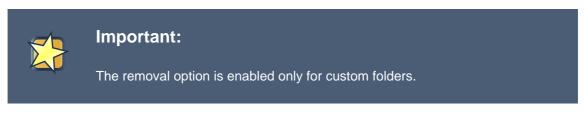
The first step in work with the editor is creating a group of libraries. It's very easy to do, right mouse button click on the *Palette* folder and select *Create > Create Group...* 

Set a name of a group in the Create Group window and click *OK* button. A new element will appear at the end of the list.

	Palette E	ditor	×
▼ 🗁 XStudio	name	value	
🕨 🖏 Icons	element type	partition	
▼ 9 Palette		Delete-	
D 🔒 HTML Create		🔸 🗳 Create Group	
🕨 🔒 JBoss 💼 Paste	Ctrl	+ V	
D 🔒 JSF	=		
Þ 🔒 JSTL			
MyFaces			
👂 🔒 Oracle ADF Fac			
👂 🚊 Some Group			
D 🔒 Struts			
	•		
	L		
		ОК	Cancel

#### Figure 4.7. Creating a Group of Tag Libraries

You are allowed to edit or delete a group, as well. If you'd like to change attributes of a group, use the right editing window of the palette editor or the *Edit...* option, like it was mentioned before. In order to remove the group, right click on the group that you wish to remove and chose the *Delete* option or click the *Delete* keyboard button.



• to work with a tag library

The group maintains a list of tag libraries. If you'd like to create your own library, click right mouse button on the group and choose *Create Group...* option.

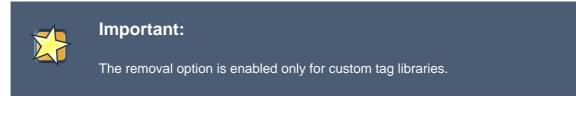
9		Palette	Editor		×
▽ 🗁 XStudio	Á	name	value		
👂 鞼 Icons		element type	group		
		name	HTML		
🕨 🚔 HTML	🛎 Create	description			
🕨 🗎 JBoss	G Create	Group	res		
👂 🗎 JSF	Edit				
D 🗎 JSTL	📄 Сору	Ctr	1 + C		
D 🚊 MyFace	of Cut	Ctr	1 + X		_
👂 🗎 Oracle	💼 Paste	Ctr	1 + V		
D 📄 Some (	X Delete	D	elete		
D 🔷 Struts	. Delete		elete		
( III					
				ок	Cancel

#### Figure 4.8. Creating a tag library

After setting the attribute name and the path of the icon, click *Ok* button.



You are allowed to edit or delete the tag library, as well. If you'd like to change attributes of the library or choose another icon, use the right editing window of the palette editor or the *Edit...* option. In order to remove the tag library, right click on the library that you wish to remove and chose the *Delete* option or click the *Delete* keyboard button.



· to work with a tag element

When the library folder is created, new tags can be added to it. Choose the required library and select the option *Create > Create Macro...* from the pop-up menu that appears after you right-click on a folder.

•		Palette	Editor		>
▽ 🗁 XStudio	Á	name	value	2	
👂 🖏 Icons		element type	sub-	group	
▽ 🚿 Palette		name	Block	ĸ	
🗢 😑 HTML	=	icon			
🕨 🐸 Block	Edit.	Ob a man a cond	to Adver	//www.w3.org/TR/REC-html40	
🕨 🐸 Core	Crea		•	🕈 Create Macro	
🕨 🐸 Form					
👂 🐸 Frame:		/	Ctrl + C		
👂 🐸 Scripts			Ctrl + X		
🕨 🐸 Table	💼 Paste	2	Ctrl + V		
👂 🐸 Text	X Dele	te	Delete		
				OK Cancel	

#### Figure 4.9. Creating a tag element

In the Add Palette Macro window, you can configure the tag element. Attribute *Name* is mandatory to fill and it will be the name of the tag element. Other settings are optional. You can choose the icon and set the *Start Text* and the *End Text* for your tag element. If your tag text is too long, use the *Change...* button to see it all. For *start text* and *end text* there is a possibility to control the cursor position by using "|" symbol.

		Palette Ed	lite	or 🛛
▽ 🗁 XStudio	A	name		value
👂 🖏 Icons		element type		macro
	=	name		a
🗢 😑 HTML		icon		%lcons%/Struts HTML/link
🕨 🐸 Block		large icon		
🗢 📂 Core		description		<pre><html>   &lt;</html></pre>
⊛ a		start text	(	<a href=" "></a>
address		end text		
🐼 area		automatically ref	on	yes
body				
h1	•			
	Ŭ			
				OK Cancel

#### Figure 4.10. Parameters of the Palette element

After all the attributes are set, click *Finish* button.



You are also allowed to edit or delete the tag. If you'd like to change the attributes of the tag or choose another icon for it, use the right editing window of the palette editor or the *Edit...* option from the pop-up menu. In order to remove the tag, right click on the tag that you wish to remove and chose the *Delete* option or click the *Delete* keyboard button.



#### Important:

The removal option is enabled only for custom tags. JBoss Palette tags can not be removed but can be modified.

#### 4.1.2. Show/Hide

*Show/Hide* is a very useful feature that allows you to control the number of tag groups that are shown on the palette.

• Click Show/Hide button, at the top right side of the JBoss Tools Palette.

#### Figure 4.11. Show/Hide Button

• In the dialog Show/Hide Drawers check the groups the libraries of which you want to be shown on the palette:

Show/Hide Drawers	×
Customize Palette	
<ul> <li>HTML</li> <li>JBoss</li> <li>JSF</li> <li>JSTL</li> <li>MyFaces</li> <li>Oracle ADF Faces</li> <li>Struts</li> <li>XHTML</li> </ul>	Show All Hide All
Ok	Cancel

#### Figure 4.12. Show/Hide Drawers

If libraries are not displayed in the palette, check whether they are selected. Click the plus sign to expand the libraries of the group and make sure that a tick is put next to the wanted libraries.

• Click OK. The new groups will now be shown on the palette:

🚿 jBoss Tools Palette 🕱 🖵 🗖
🛠 🗟 🛇
Block
C HTML Core
C> HTML Form
🗁 HTML Frames
C> HTML Scripts
🗁 HTML Table
🗁 HTML Text
🗁 JBoss Ajax4jsf
JBoss RichFaces
궏 JBoss Seam
🗁 JSF Core
JSF Facelets
JSF HTML
MyFaces Extensions
MyFaces Sandbox
MyFaces Tomahawk
> XHTML List
> XHTML Object
> XHTML Structural
> XHTML Table
C XHTML Text
C XHTML XForms Basic

#### Figure 4.13. New Added Groups

The names of the elements are compound. The first part is the group name and the second is the library name.

#### 4.1.3. Import

The Import button lets you add a custom or 3rd party tag library to JBoss Tools Palette. Find out more information on how to add particular tags see the *Adding Custom JSF Tags* section.

# 4.2. Using the Palette

#### 4.2.1. Inserting Tags into a JSP File

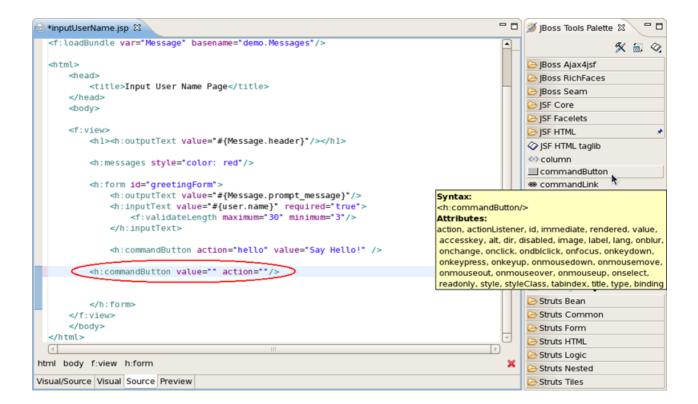
A new tag can be added into any text file including jsp, htm, html and xhtml.

Let's do it. Open your JSP file and place the cursor in a place where you'd like to add a tag and then click that tag in the palette. In the Insert Tag window, that appears, you can set the value of *general* and *advanced* attributes of the tag that you chose.

•	Insert Tag	×
<h:commandbutto< th=""><th>on&gt; attributes</th><th></th></h:commandbutto<>	on> attributes	
General Advanced		
Attribute name	Value	A
action		=
value		
accesskey		
actionListener		
alt		
binding		
dir		
disabled		
id		
image		
0		<u>F</u> inish Cancel

# Figure 4.14. Inserting Tag

In the example below, the *commandButton* tag has been inserted.



#### Figure 4.15. Inserting Tag



The cursor position after adding a tag into a file is specified by "|" symbol in the tag template on the right in the Palette Editor window.

<b>9</b>		Palette Edito	r	
Di 😂 Core		name	value	
🗢 😂 Form		element type	macro	
button	Γ	name	input	
checkbox	=	icon		
choosing		large icon		
fieldset		description	<html></html>	
<ol> <li>file</li> </ol>		start text	<input< td=""><td>type=" " name=""&gt;</td></input<>	type=" " name="">
<ul> <li>form</li> </ul>		end text	_	
hidden		automatically re	efc no	
input				
input button				
				OK Cancel

#### Figure 4.16. Palette Editor

Above you can see where the cursor position for HTML > Form > input is set. Thus, after adding this tag into your file the cursor will be in the attribute "type". Then, you can straight use the combination of buttons Ctrl + Space to inquire about a prompting.

à home.xhtml	im ∗inputname.xhtml ⊠		-
<ui:< td=""><td>composition template="/template</td><td>es/common.xhtml"&gt;</td><td>•</td></ui:<>	composition template="/template	es/common.xhtml">	•
	<ui:define name="pageTitle">JS</ui:define>	F 1.2 and Facelets under Tomcat. KickStart	Application
	<ui:define name="pageHeader">J</ui:define>	SF 1.2 Hello Application	
	<ui:define name="body"></ui:define>		
		ue" showDetail="false" style="color: red; f	ont-weight: bold;
	<form id="he&lt;/td&gt;&lt;td&gt;lloForm" jsfc="h:form"></form>		
	ut type="" name="">		
tribute : type		O "button"	
nta Type : ENUM		③ "checkbox"	
- text		() "file"	
- password		() "hidden"	=
- checkbox - radio		() "image"	
- submit		(a) "password"	_
- reset			
- file - hidden		(8) "radio"	
- image		③ "reset"	
- button		I submit	
- button		@ "text"	
- Datton		@ text	

#### Figure 4.17. Cursor position

# 4.2.2. Adding Custom JSF Tags to the JBoss Tools Palette

There are two ways to add any custom or 3rd party tag library to the JBoss Tools Palette:

- Drag-and-drop from the Web Projects view
- The Import button on the JBoss Tools Palette

Before you can add your custom component library, you need to make sure it is included in your project. Either place the ".*tld*" file or the ".*jar*" that includes your tag library under the lib folder in your project.

#### 4.2.2.1. Drag-and-Drop

Switch to the Web Projects view and expand the Tag Libraries folder. If the view is not active, select *Window > Show View > Web Projects* from the menu bar.

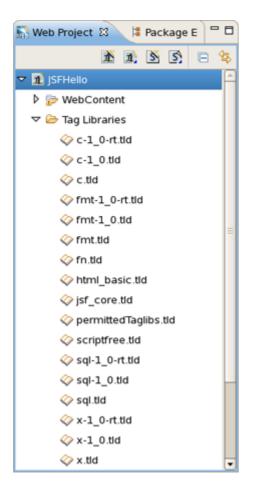


Figure 4.18. Web Projects View

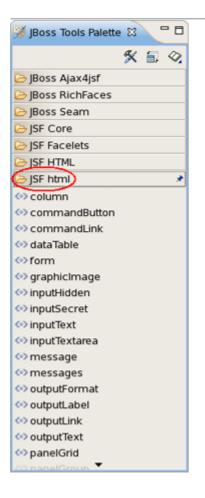
Also make sure that the JBoss Tools Palette is open. Select the tag library that you want to add and simply drag-and-drop it on to the JBoss Tools Palette.

You will see the following dialog window. As you can see JBoss Developer Studio takes care of all the details. Chosen *TLD file*, *name* and *prefix* of the library and *Library URL* are detected, thus just need to set the *Group* name to which you wish to place this tag library. You can either add this tag library to an existing Group or just create a new one.

<b>9</b>	Import Tags from TLD File	×	
Create Macro from Tags in TLD file Parent group must be specified.			
TLD File*	/JSFHello//META-INF/html_basic.tld		
Name:*	html		
Default Prefix	h	]	
Library URI	http://java.sun.com/jsf/html	]	
<ul> <li>Add to Exist</li> <li>Create New</li> </ul>		]	
	OK Cancel	]	

# Figure 4.19. Import Tags From TLD File Form

Once you are finished, you will see the new tag library added to the JBoss Tools Palette.



#### Figure 4.20. JBoss Tools Palette with New Tag Library

#### 4.2.2.2. Import Button

The same you can do with *Import* button. You can see this button at the top right side of the JBoss Tools Palette.

<ul> <li>JBoss Ajax4jsf</li> <li>JBoss RichFaces</li> <li>JBoss Seam</li> <li>JSF Core</li> <li>JSF Facelets</li> <li>JSF JSTL core</li> <li>Struts Bean</li> </ul>
<ul> <li>JBoss RichFaces</li> <li>JBoss Seam</li> <li>JSF Core</li> <li>JSF Facelets</li> <li>JSF HTML</li> <li>JSF JSTL core</li> </ul>
<ul> <li>JBoss Seam</li> <li>JSF Core</li> <li>JSF Facelets</li> <li>JSF HTML</li> <li>JSF JSTL core</li> </ul>
JSF Core JSF Facelets JSF HTML JSF JSTL core
JSF Facelets     JSF HTML     JSF JSTL core
SF HTML
JSF JSTL core
🗁 Struts Bean
Struts Common
Struts Form
C Struts HTML
🕞 Struts Logic
Struts Nested
Struts Tiles

#### Figure 4.21. Import Button

By clicking on the *Import button* you will see the Import Tag window a similar like in the *Drag-and-Drop* method. Set the name and prefix of the library and Library URL. Also you need to set the Group name to which you'd like to add your tag library. And like in the previous method you can add it to an existing Group or create a new one. On this Import Tag form you can use *Browse...* button to locate the tag library that you want to add:

۲	Import Tags from TLD File	×				
Create Macro from Tags in TLD file						
🙆 Attrit	Edit TLD X					
TLD File*						
TLD File	🗢 🗁 JSFHello					
Name:		h				
	🐼 c.tid	K.				
Default	🗇 permittedTaglibs.tld	μ				
Library	♦ sql-1_0-rt.tld	h				
	🐼 sql.tid	E				
Ade		h.				
O Cre	♦ scriptfree.tld	F.				
O Cre	<pre>   fmt-1_0-rt.tld </pre>	μ.				
	<pre>   fmt.tld </pre>	h				
		H.				
	<pre>   fmt-1_0.tld</pre>					
	html_basic.tld					
	⊘ jsf_core.tld					
	OK Cancel					

#### Figure 4.22. Select TLD File

# 4.3. Rich Faces Support

JBoss Developer Studio comes with a tight integration with *Rich Faces* component framework. Rich Faces and Ajax4jsf in JBoss Tools Palette always exist, this palette is not dynamic and is not changed from file to file. JBoss Tools Palette supports only the last version of the Rich Faces components.



Rich Faces components, as well, as *Ajax4jsf* ones are already on the JBoss Tools Palette:

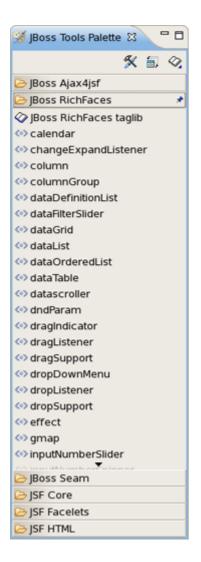


Figure 4.23. Rich Faces Components

# **Web Projects View**

Web Projects is a special view that comes with JBoss Developer Studio.

If the Web Projects view's tab is not visible next to the Package Explorer tab, select *Window* > *Show View* > *Other* > *JBoss Tools Web* > *Web Projects* from the menu bar.

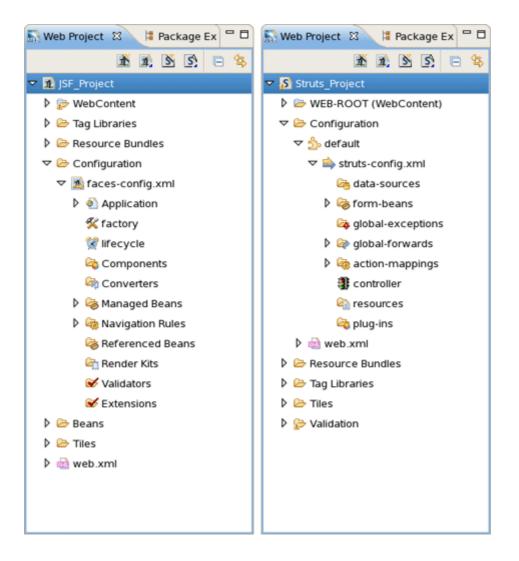
With the Web Projects view, you can:

- Visualize the project better because the project artifacts for JSF and Struts projects are organized and displayed by function.
- Select these kinds of items to drag and drop into JSP pages:
  - JSF managed bean attributes
  - JSF navigation rules outcomes
  - Property file values
  - Tag library files
  - Tags from tag libraries
  - JSP page links
- Use context menus to develop the application (all create and edit functions are available)
- · Use icon shortcuts to create and import JSF and Struts projects
- Expand and inspect tag library files
- Select custom and third-party tag libraries to drag and drop onto the JBoss Tools Palette

# 5.1. Project Organization

The Web Projects view organizes your project in a different way. The physical structure of course stays the same. The new organization combines common project artifacts together which makes it simpler to locate what you are looking for and develop.

The screen shot below shows a JSF project and a Struts project in Web Projects view.



#### Figure 5.1. Web Projects View

# 5.2. Drag and Drop

Web Projects View has a drag and drop option that can be used for property, managed bean attributes, navigation rules, tag library file declaration and JSP Pages.

# 5.2.1. For a Property

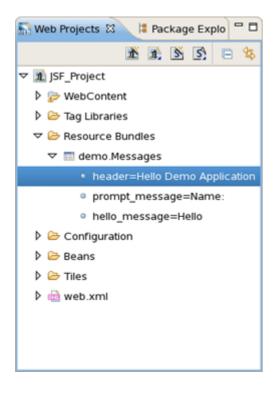
Expand the Resources Bundles folder that holds all the Property files in your project. Select the file from which you want to add the property and then select the property.

We will be dragging and dropping a property file value inside the outputText tag for the "value" attribute.

```
<html>
<head>
<title>Input User Name Page</title>
</head>
<body>
<f:view>
<hl><h:outputText value=""/></hl>
```

#### Figure 5.2. OutputText Tag

Select the property:



#### Figure 5.3. Selecting Property

Drag the property and drop it between the quotes for the value attribute in the JSP file. Notice that JBoss Developer Studio added the correctly formatted expression for referring to the property value #{*Message.header*} automatically.

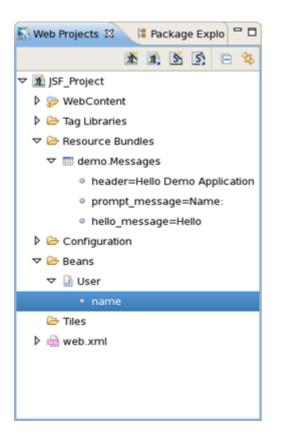
```
<html>
<head>
<title>Input User Name Page</title>
</head>
<body>
<f:view>
<hl><h:outputText value="#{Message.header}"/></hl>
<h:messages style="color: red"/>
```

#### Figure 5.4. Inserted Property

You can actually place the tag anywhere in the page, not just inside an existing tag. In this case, JBoss Developer Studio will place the complete tag <h:outputText value="#{Message.header}"/> in the page.

#### 5.2.2. For Managed Bean Attributes

Select a *"managed bean"* attribute and then drag and drop it onto the JSP page. We are going to place it inside the *"value"* attribute of the inputText tag.



#### Figure 5.5. Selecting Managed Bean Attribute

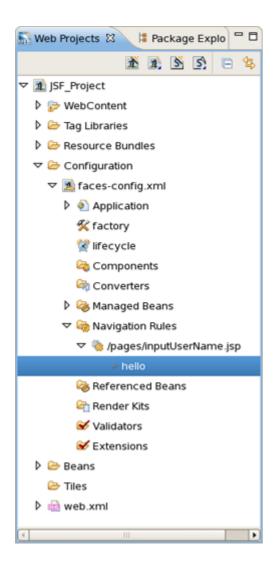
Once again, JBoss Developer Studio adds the correct expression, #{user.name}.

```
<h:form id="greetingForm">
    <h:outputText value="#{Message.prompt_message}"/>
    <h:inputText value="#{user.name}" required="true">
        <f:validateLength maximum="30" minimum="3"/>
    </h:inputText>
```

#### Figure 5.6. Added Expression

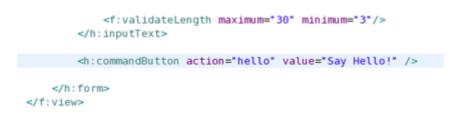
#### 5.2.3. Navigation Rules

Select the navigation rule under Configuration > faces-config.xml > Navigation Rules:



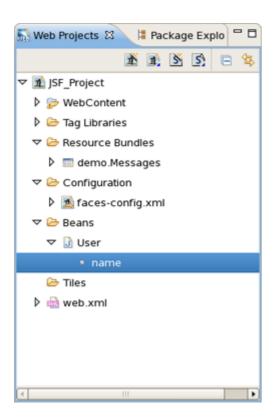
#### Figure 5.7. Selecting Navigation Rule

Drag and drop it inside the commandButton tag:



#### Figure 5.8. Naviagation Rule in CommandButton Tag

You could do the same if the navigation rule was defined inside an action method:



#### Figure 5.9. Navigation Rule in Action Method

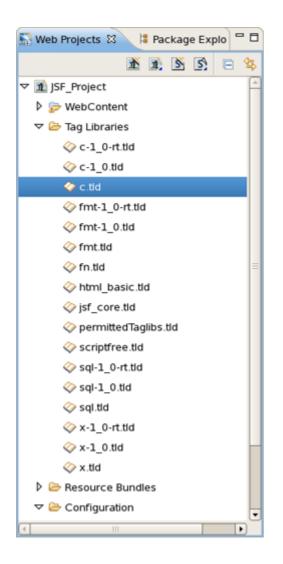
Here is how it would look after drag and drop:

```
<f:validateLength maximum="30" minimum="3"/>
</h:inputText>
<h:commandButton action="#{user.name}" value="Say Hello!" />
</h:form>
```

#### Figure 5.10. Inserted Navigation Rule

#### 5.2.4. For a Tag Library File Declaration

Select a TLD file:



#### Figure 5.11. Selecting TLD File

Then drag and drop it onto the JSP page to add a declaration at the top of the page:

#### Figure 5.12. Inserted TLD File

#### 5.2.5. For JSP Pages

You can also drag and drop a JSP page path to a JSP page to create a forward as shown:

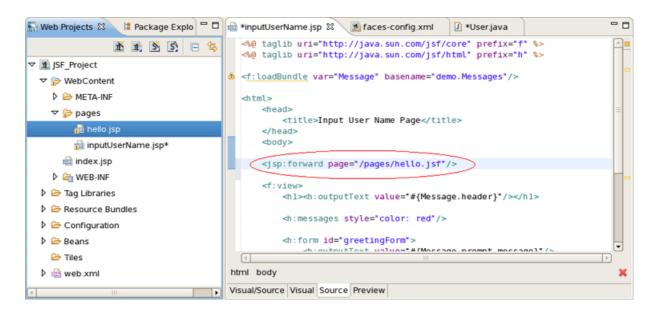


Figure 5.13. Creating JSP Forward

# 5.3. Developing the Application

It is also possible to develop your application right from the Web Projects view. Simply right-click any node in the tree and select an appropriate action from the context menu. For instance, this screen capture shows creating a new navigation rule.

🔝 *Web Proje 🕴 🛤 Package B	=x - 🗆	🖄 *faces-config.xml 🛙	• 8
<ul> <li>▶ Decontent</li> <li>&gt; Tag Libraries</li> <li>&gt; Resource Bundles</li> <li>&gt; Configuration</li> <li>&gt; Configuration</li> <li>&gt; Application</li> <li>Application     faces-config.xml*     <li>&gt; Application     <li>factory     </li> <li>Components     </li> <li>Converters     </li> <li>Managed Beans </li> </li></li></li></li></li></li></li></li></li></li></li></li></ul>		/pages/inputUserl	Name.jsp /pages/hello.jsp
Navigation Rules	New	•	🔞 Rule
leans			
🖳 Render Kits	😽 Cut	Ctrl + X	
🧭 Validators	📔 Copy	/ Ctrl + C	
🧭 Extensions	👘 Paste	Ctrl + V	
👂 🗁 Beans	💢 Delet	te Delete	
🕨 🗁 Tiles			
👂 📾 web.xml	Prope	erties	
StrutsHello	🤥 Verify	*	
		Diagram Iree Source	

#### Figure 5.14. Creating New Navigation Rule

# 5.4. Expanding Tag Library Files

You can easily expand any TLD file in the project. Browse to the Tag Libraries folder. Right-click a TLD file and select *Expand*. The TLD file will now be expanded.

You can then select any tag and drag it onto a JSP page.

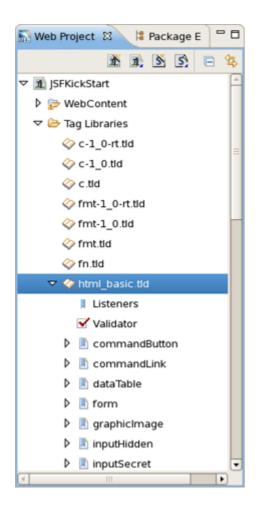


Figure 5.15. Expanding Tag Library File

# 5.5. Drag and Drop Tag Libraries on to JBoss Tools Palette

Read Adding Tag Libraries to learn about this.

# 5.6. Create and Import JSF and Struts Projects

You can also create and import JSF and Struts project from Web Projects view by selecting the buttons below.

From left to right:

- 1. Create New JSF Project
- 2. Import JSF Project
- 3. Create New Struts Project
- 4. Import Struts Project

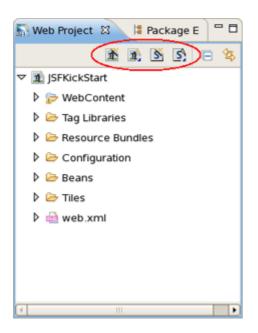


Figure 5.16. Web Projects View Buttons

# **JBoss Tools Preferences**

Configuring the various JBoss Developer Studio features is done via the Preferences screen by selecting *Window > Preferences > JBoss Tools* from the menu bar.

۲			Preferences	×
ty	pe filter text		JBoss Tools	(⊃≖ ⊂)≖
⊳	General	*	JBossTools preferences	
⊳	Ant			
⊳	Connectivity			
	FreeMarker Editor			
⊳	Help			
	HQL editor			
⊳	Install/Update			
₽	Internet			
₽	Java			
⊳	JBoss jBPM			
~	JBoss Tools			
	JBoss Servers			
	Packaging Archives			
	Veb			
	XDoclet			
	JPA			
₽	Plug-in Development			
₽	Run/Debug			
⊳	Server			
⊳	Spring			
₽	SQL Development	H		
⊳	Team			
	TestNG			Restore Defaults Apply
	Validation	•		Restore Defaults Apply
C	D			OK Cancel

#### Figure 6.1. Preferences are included in this dialog.

From this screen, you can select these more specific sets of JBoss Tools preferences:

- Code Assist
- Editors
- JBoss Servers

- JSF
- JSF Flow Diagram
- JSF Page
- JSF Propject
- Packaging Archives
- Plug-in Insets
- Resource Insets
- Seam
- Seam Validator
- Struts
- Struts Automatic
- Struts Customization
- Struts Flow Diagram
- Struts Pages
- Struts Project
- Struts Support
- Title Diagram
- Verification
- View
- Visual Page Editor
- XDoclet
- XDoclet Templets
- XDoclets Variables

# 6.1. CodeAssist

Select JBoss Tools > XDoclet > Code Assist to see Code Assist preference page.

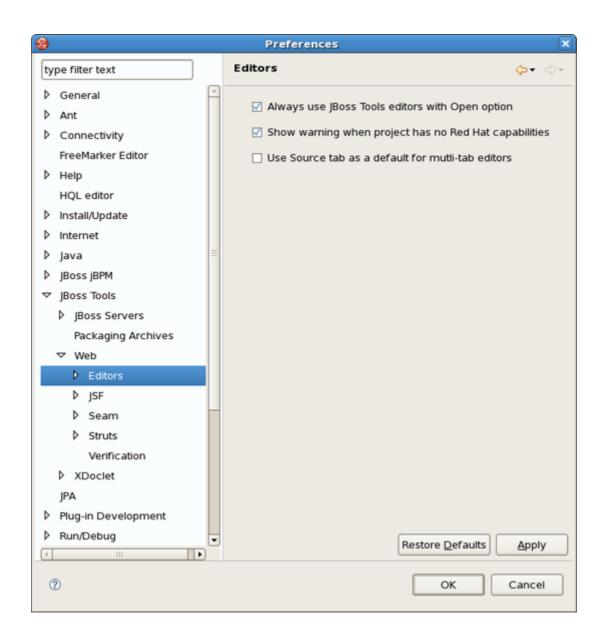
Here is what the Code Assist preference page looks like:

۲			Preferences	×
ty	pe filter text		Code Assist	<b>⇔</b> • ⇔•
⊳	General	Ĥ	Allows to refresh the XTags from the XDoclet modules	
⊳	Ant		Refresh XDoclet Data	
⊳	Connectivity		nerresit Aboolet bata	
	FreeMarker Editor			
⊳	Help			
	HQL editor			
⊳	Install/Update			
⊳	Internet			
⊳	Java			
⊳	JBoss jBPM	=		
~	JBoss Tools			
	JBoss Servers			
	Packaging Archives			
	◊ Web			
	Code Assist			
	JPA			
₽	Plug-in Development			
₽	Run/Debug	Н		
₽	Server			
₽	Spring			
₽	SQL Development			
₽	Team	•	Restore Defaults	Apply
4				дрру
Ċ	D		ОК	Cancel

#### Figure 6.2. CodeAssist

# 6.2. Editors

You can set various preferences for the editors that JBoss Developer Studio adds to the Eclipse environment by selectind *JBoss Tools* > *Web* > *Editors*.



#### Figure 6.3. Editors

In the initial Editors screen, you can decide on some global settings for JBoss Tools editors. You can select whether an available JBoss Tools editor should always be the default editor for a type of file, whether the user should be warned that making a project an Red Hat project will make an JBoss Tools editor fully available for a particular type of file, and whether, for JBoss Tools editors, the Source mode should be the default instead of a visual mode.

# 6.3. JBoss Servers

The following preferences can be changed on the *JBoss Tools* > *JBoss Servers* preference page.

Here is what the JBoss Servers preference page looks like:

9	Preferences
type filter text	JBoss Servers 🔅 🗘 🗸
<ul> <li>▷ General</li> <li>▷ Ant</li> <li>▷ Connectivity FreeMarker Editor</li> <li>▷ Help HQL editor</li> <li>▷ Install/Update</li> <li>▷ Install/Update</li> <li>▷ Internet</li> <li>▷ Java</li> <li>▷ JBoss JBPM</li> <li>♥ JBoss Tools</li> <li>▷ JBoss Servers</li> <li>Packaging Archives</li> <li>♥ Web</li> <li>▷ Editors</li> <li>▷ JSF</li> <li>▷ Seam</li> <li>▷ Struts</li> <li>∨erification</li> <li>▷ XDoclet</li> <li>JPA</li> <li>▷ Plug-in Development</li> <li>▷ Run/Debug</li> </ul>	Servers Boss Application Server 4.2 Server Timeouts Start Timeout 450 Stop Timeout 450 Upon Timeout: Abort Server Transition Set Server State to Target State Startup Poller JMX Poller $\Rightarrow$ Shutdown Poller Process Terminated Poller $$$
Kun/Debug	Restore Defaults     Apply
٢	OK Cancel

# Figure 6.4. JBoss Servers

# 6.4. JSF

Select JBoss Tools > Web > JSF to get to the JSF Project specific preferences.

9	Preferences	×
type filter text	JSF	<b>⇔</b> ∙ ⇔∗
▶ General	JSF Projects specific preferences.	
▶ Ant		
▶ Connectivity		
FreeMarker Editor		
▶ Help		
HQL editor		
▶ Install/Update		
▶ Internet		
▶ Java =		
▷ JBoss jBPM		
JBoss Servers		
Packaging Archives		
∀ Web		
▶ Editors		
▶ JSF		
▷ Seam		
Struts		
Verification		
XDoclet		
JPA		
Plug-in Development		
Run/Debug		
۲	ОК	Cancel

Figure 6.5. JSF

# 6.5. JSF Flow Diagram

Selecting *JBoss Tools* > *Web* > *Editors* > *JSF Flow Diagram* allows you to specify some aspects of the Diagram mode of the JSF configuration file editor.

9	Preferences	×
type filter text	JSF Flow Diagram	<b>⇔</b> • ⇔-
<ul> <li>General</li> <li>Ant</li> <li>Connectivity FreeMarker Editor</li> <li>Help</li> <li>HQL editor</li> <li>Install/Update</li> <li>Internet</li> <li>Java</li> <li>JBoss JBPM</li> <li>JBoss Tools</li> <li>JBoss Servers Packaging Archives</li> </ul>	JSF Flow Diagram       Add View         ISF Flow Diagram       Add View         Image: Show Grid       Image: Show Grid         Grid Step:       16         Link Path Font:       default.size=8,style=1         View Path Font:       default.size=8,style=1         Image: Switch to standard control mode for cursor after transition       Show shortcut icon         Image: Show shortcut path       Image: Show shortcut path	← <u>C</u> hange <u>C</u> hange vigation case
<ul> <li>✓ Web</li> <li>✓ Editors</li> <li>JSF Flow Diagram</li> <li>Struts Flow Diagram</li> <li>Tiles Diagram</li> <li>Visual Page Editor</li> <li>JSF</li> <li>Seam</li> <li>Struts</li> <li>Verification</li> </ul>	Restore Defaults	Apply

#### Figure 6.6. JSF Flow Diagram

The first two items control the background grid for the diagram. The next two items allow you to control the appearance of the labels for views (pages) and the transitions between views. For these two items, clicking the *Change...* button allows you to assign a font with a dialog box.

The first check box determines whether a view in the diagram that doesn't have a transition connecting it to another view yet should be written to the source code as a partial navigation rule. The next check box determines whether the diagram cursor reverts immediately to the standard selection mode after it's used in the transition-drawing mode to draw a transition. Finally, the last two check boxes concern shortcuts. A shortcut is a transition that is there but isn't actually displayed in the diagram as going all the way to the target view it's connected to, in order to make the diagram clearer. With the check boxes, you can decide whether to display a small shortcut icon as part of the shortcut and also whether to display the target view as a label or not.

•	Preferences	×
type filter text	JSF Flow Diagram	<b>⊳</b> • ⇔•
General	JSF Flow Diagram Add View	
Ant		
Connectivity	Page Template: JSFBasePage	-
FreeMarker Editor	Extension: jsp	•
Help	heb.	
HQL editor		
Install/Update		
Internet	Ξ	
Þ Java		
JBoss jBPM		
JBoss Servers		
Packaging Archives		
⊽ Web		
JSF Flow Diagram		
Struts Flow Diagram		
Tiles Diagram		
Visual Page Editor		
▷ JSF		
Seam		
Struts		
Verification	Restore Defaults	Apply
0	ОК С	ancel

#### Figure 6.7. Add View

Selecting the Add Page tab in the JSF Flow Diagram screen allows you to determine the default template and file extension for views (pages) you add directly into the diagram using a context menu or the view-adding mode of the diagram cursor.

# 6.6. JSF Page

By selecting *JBoss Tools* > *Web* > *JSF* > > *JSF Pages* you can add jsf pages or remove existing ones.

<b>9</b>	Preferences	×
type filter text	JSF Pages	⇔- ⇔-
<ul> <li>General</li> <li>Ant</li> </ul>	Blank [default]	Add
Connectivity FreeMarker Editor	JSFBasePage JSPRedirect	Remove
Help     HQL editor     Install/Update		
▷ Internet ▷ Java		
<ul> <li>▷ JBoss jBPM</li> <li>▽ JBoss Tools</li> <li>▷ JBoss Servers</li> </ul>		
Packaging Archives		
▷ Editors ♥ JSF		
JSF Pages Project		
<ul> <li>Seam</li> <li>Struts</li> <li>Verification</li> </ul>		
▷ XDoclet JPA		
<ul> <li>Plug-in Development</li> <li>Run/Debug</li> </ul>		
Server     Spring		Set Default
0	ОК	Cancel

#### Figure 6.8. JSF Page

# 6.7. JSF Propject

Select JBoss Tools > Web > JSF > Project to see JSF Project preference page.

On *Project* panel you define a template for a new created project: servlet version, page template and so on.

S Preferences						
type filter text	Project					
General	New Project Import	Project				
▶ Ant	,					
Connectivity	Version:	+				
FreeMarker Editor	Project Template:	<b>(</b>				
▷ Help						
HQL editor		☑ Use Default Path				
Install/Update	Projects Root:	/opt/work/workspace_beta				
Internet	Servlet Version:	2.4				
⊅ Java	Serviet version:	2.4				
▷ JBoss jBPM		Register Web Context in server.xml				
	Page Template:	<b>(</b>				
JBoss Servers						
Packaging Archives						
⊽ Web						
Editors						
⊽ JSF						
JSF Pages						
Project						
▷ Seam						
▷ Struts						
Verification						
XDoclet						
JPA						
Plug-in Development						
Run/Debug						
Server						
▶ Spring		Restore Defaults Apply				
0		OK Cancel				

### Figure 6.9. JSF Propject

Selecting the Import Project tab in the JSF Project screen allows you to determine the default servlet version and whether to register Web Context in server.xml.

9	Preferences	×
type filter text	Project	<b>⇔</b> • ⇔∗
<ul> <li>General</li> <li>Ant</li> <li>Connectivity</li> <li>FreeMarker Editor</li> <li>Help</li> <li>HQL editor</li> <li>Install/Update</li> <li>Internet</li> </ul>	New Project       Servlet Version:       2.4       Register Web Con	· ·
<ul> <li>▷ Java</li> <li>▷ JBoss jBPM</li> <li>▽ JBoss Tools</li> <li>▷ JBoss Servers</li> <li>Packaging Archives</li> <li>▽ Web</li> <li>▷ Editors</li> <li>▽ JSF</li> <li>JSF Pages</li> </ul>		
Project       > Seam       > Struts       Verification       > XDoclet       JPA       > Plug-in Development	Restor	re <u>D</u> efaults <u>A</u> pply
0		OK Cancel

### Figure 6.10. Import JSF Propject

# 6.8. Packaging Archives

The following preferences can be changed on the *JBoss Tools > Packaging Archives* page.

On *Packaging Archives* panel you determine settings for Project Packages view and core preferences.

<b>9</b>	Preferences	×
type filter text	Packaging Archives	⇔- ⇒-
<ul> <li>▷ General</li> <li>▷ Ant</li> <li>▷ Connectivity</li> <li>FreeMarker Editor</li> <li>▷ Help</li> <li>HQL editor</li> <li>▷ Install/Update</li> <li>▷ Internet</li> <li>▷ Java</li> <li>▷ JBoss jBPM</li> <li>▽ JBoss Tools</li> </ul>	Core Preferences Enable incremental builder Project Packages View Show full output path next to packages. Show the full root directory of filesets. Show project at the root Show all projects that contain packages	
<ul> <li>JBoss Servers</li> <li>Packaging Archives</li> <li>Web</li> <li>XDoclet</li> <li>JPA</li> </ul>		
<ul> <li>Plug-in Development</li> <li>Run/Debug</li> <li>Server</li> <li>Spring</li> <li>SQL Development</li> <li>Team TestNG</li> <li>Validation</li> </ul>		
<ul> <li>Validation</li> <li>Veb and XML</li> <li>Veb Services</li> <li>XDoclet</li> </ul>	Restore <u>D</u> efaults OK	<u>A</u> pply Cancel

#### Figure 6.11. Packaging Archives

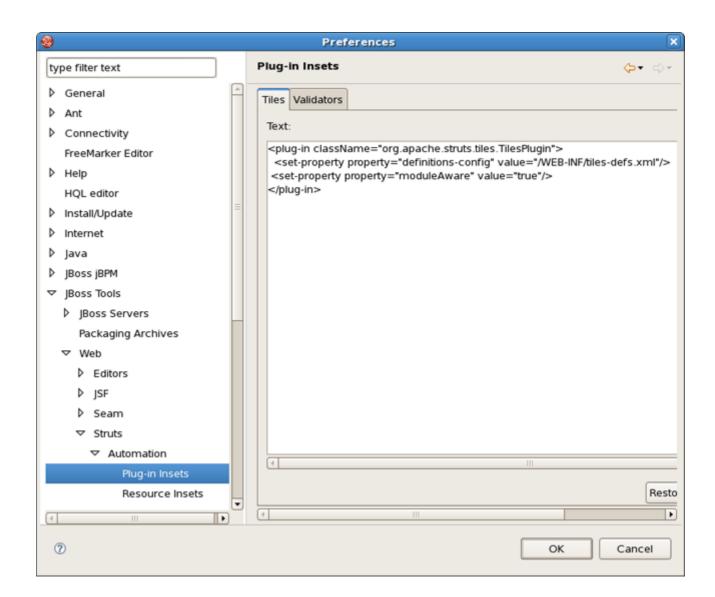
# 6.9. Plug-in Insets

By selecting *Web* > *Struts* > *Automation* > *Plug-in Insets* on tab Tiles you can define a default text for tiles plugin.

9	Preferences	×
type filter text	📋 Packaging Archives	⇔• ⇒∗
<ul> <li>General</li> <li>Ant</li> <li>Connectivity FreeMarker Editor</li> <li>Help HQL editor</li> <li>Install/Update</li> <li>Internet</li> <li>Java</li> <li>JBoss jBPM</li> </ul>	Core Preferences Enable incremental builder Project Packages View Show full output path next to packages. Show the full root directory of filesets. Show project at the root Show all projects that contain packages	
<ul> <li>✓ JBoss Tools</li> <li>▷ JBoss Servers</li> <li>Packaging Archives</li> <li>▷ Web</li> <li>▷ XDoclet</li> </ul>		
JPA Plug-in Development Run/Debug Server Spring SQL Development Team TestNG		
Validation  Validation  Web and XML  Veb Services  XDoclet	Restore <u>D</u> efaults	<u>A</u> pply Cancel

## Figure 6.12. Plug-in Insets

The same is done but for validator plugin on the tab Validators.



#### Figure 6.13. Plug-in Insets of Validators

## **6.10. Resource Insets**

To see Resource Insets preference page select *JBoss Tools* > *Web* > *Strats* > *Automation* > *Resource Insets*.

On Resource Insets panel you determine default error messages for error resource files.

9	Preferences	
type filter text	Resource Insets	<b>⇔</b> • ⇔•
General	Default Error Messages	
Ant		
Connectivity	Text:	
FreeMarker Editor	errors.required={0} is required.	A
> Help	errors.minlength={0} can not be less than {1} c errors.maxlength={0} can not be greater than {	
HQL editor	errors.invalid={0} is invalid.	-,
> Install/Update	errors.byte={0} must be a byte.	
> Internet	errors.short={0} must be a short.	
> Java	errors.integer={0} must be an integer.	
> JBoss jBPM	errors.long={0} must be a long. errors.float={0} must be a float.	
7 JBoss Tools	errors.double={0} must be a double.	
JBoss Servers	errors.date={0} is not a date.	
,	errors.range={0} is not in the range {1} through	{2}.
Packaging Archives Veb	errors.creditcard={0} is an invalid credit card nu errors.email={0} is an invalid e-mail address.	mber.
<ul> <li>Web</li> <li>Editors</li> </ul>	errors.errain= {0} is air invalid e-rriair address.	=
▷ JSF ▷ Seam		
✓ Struts		
Plug-in Insets		
Resource Insets		
Customization		
Project		
Struts Pages		
Verification		
XDoclet		
JPA	Restore Defau	ults Apply
0	ОК	Cancel

#### Figure 6.14. Resource Insets

## 6.11. Seam

The following preferences can be changed on the *JBoss Tools* > *Web* > *Seam* page.

On Seam screen you can add and remove Seam runtimes.

Here is what Seam preference page looks like:

۲				Preferen	ices		×
ty	pe filter text	5	Sea	m			<b>⇔</b> • ⇔-
Þ	General	-	_				
⊳	Ant			Name	Versio	Path	Add
⊳	Connectivity		V	Seam 1.2.AP	1.2	/opt/work/rhds_rc	Edit
	FreeMarker Editor		Ŀ				Remove
⊳	Help		Ŀ				Remove
	HQL editor						
⊳	Install/Update		Ŀ				
⊳	Internet		Ŀ				
⊳	Java	=	Ŀ				
⊳	JBoss jBPM		-				
~	JBoss Tools						
	JBoss Servers						
	Packaging Archive						
	⊽ Web						
	Editors						
_	Þ js⊧						
	Seam						
	Struts						
	Verification						
	XDoclet						
	JPA						
Þ	Plug-in Development						
•	Run/Debug	•				Restore Default	s <u>A</u> pply
•							
Ċ	D					ОК	Cancel

Figure 6.15. Seam

# 6.12. Seam Validator

The following preferences can be changed on the *JBoss Tools* > *Web* > *Seam* > *Validator* page.

In Validator panel you configure seam problems that will be processed by validator.

<b>8</b>	Preferences	×
type filter text	Validator	<b>⇔</b> ∙ ⇔∽
General     Ant     Connectivity	Configure Project Sp Select the severity level for the following optional Seam Validator problems:	ecific_
FreeMarker Editor ▷ Help HQL editor ▷ Install/Update ▷ Internet ▷ Java ▷ JBoss jBPM ▽ JBoss Tools	<ul> <li>Components</li> <li>Duplicate component name:</li> <li>Stateful component does not contain @Remove method:</li> <li>Stateful component does not contain @Destroy method:</li> <li>Stateful component has wrong scope:</li> <li>Component class name cannot be resolved to a type:</li> <li>Component class does not contain setter for property:</li> </ul>	Error Er
<ul> <li>▶ JBoss Servers Packaging Archives</li> <li>♥ Web</li> <li>▶ Editors</li> <li>▶ JSF</li> <li>♥ Seam</li> <li>Validator</li> <li>▶ Struts Verification</li> <li>▶ XDoclet</li> </ul>	<ul> <li>Entities</li> <li>Component life-cycle methods:</li> <li>Factories</li> <li>Bijections</li> <li>Context variables</li> <li>Expression language</li> </ul>	
JPA Plug-in Development        Image: Construction of the second se	Restore Defaults	<u>Apply</u> Cancel

## Figure 6.16. Seam Validator

## 6.13. Struts

By selecting *JBoss Tools > Web > Struts* you can configure Struts projects specific preferences.

8		Preferences	
type filter text	]	Struts	<b>⇔</b> • ⇔•
Ø General	<u></u>	Struts Projects specific preferences.	
▶ Ant			
Connectivity			
FreeMarker Editor			
▶ Help			
HQL editor			
Install/Update			
Internet			
👂 Java	=		
JBoss jBPM			
✓ JBoss Tools			
JBoss Servers			
Packaging Archives			
▽ Web			
Editors			
▷ JSF			
Seam	H		
Struts			
Verification			
XDoclet			
JPA			
Plug-in Development			
Run/Debug	-		
۹ III	D		
0			OK Cancel
U		L	Cancel

## Figure 6.17. Struts

## 6.14. Struts Automatic

On *Automation* panel you can modify default text for the Tilel Struts plug-in element, the Validator Struts plug-in element, and error message resource files.

type filter text     Image: system in the system	<b>9</b>	Preferences	X
<ul> <li>Ant</li> <li>Connectivity</li> <li>FreeMarker Editor</li> <li>Help</li> <li>HQL editor</li> <li>Install/Update</li> <li>Internet</li> <li>Java</li> <li>JBoss JBPM</li> <li>JBoss Servers</li> <li>Packaging Archives</li> <li>✓ Web</li> <li>Editors</li> <li>JSF</li> <li>Seam</li> <li>✓ Struts</li> </ul>	type filter text	Automation 🗇 🗸	÷۳
<ul> <li>▷ Editors</li> <li>▷ JSF</li> <li>▷ Seam</li> <li>▽ Struts</li> <li>▷ Automation</li> </ul>	<ul> <li>General</li> <li>Ant</li> <li>Connectivity</li> <li>FreeMarker Editor</li> <li>Help</li> <li>HQL editor</li> <li>Install/Update</li> <li>Install/Update</li> <li>Internet</li> <li>Java</li> <li>JBoss jBPM</li> <li>▼ JBoss Tools</li> <li>▷ JBoss Servers</li> </ul>	Here you can modify default text for the Tiles Struts plug-in element, the Validator Struts plug-in element, and error message resource files.	\$~* 
Project     Struts Pages     Verification     OK Cancel	<ul> <li>▷ Editors</li> <li>▷ JSF</li> <li>▷ Seam</li> <li>▽ Struts</li> <li>▷ Automation</li> <li>Customization</li> <li>▷ Project</li> <li>Struts Pages</li> <li>Verification</li> </ul>		

#### Figure 6.18. Struts Automatic

# 6.15. Struts Customization

The following preferences can be changed on the *JBoss Tools* > *Web* > *Struts* > *Customization* page.

In the Customization screen you configure Link Recognizer for Struts tags.

		Pr	eferences			
ype filter text		Customizatio	n			<b>⇔</b> ∙ ⇔
General	*	Link Recognize	er			
> Ant		Tag	Attribute	Refer to	Link Type	Add
> Connectivity		html:link	action	action	Struts	
FreeMarker Editor		html:link	page	page	Struts	Edit
> Help		html:link	forward	forward	Struts	Delete
HQL editor		html:frame	action	action	Struts	
Install/Update		html:frame	page	page	Struts	
Internet	=	html:frame	forward	forward	Struts	
Java		html:form	action	action	Struts	
JBoss jBPM		logic:forward	name	forward	Struts	
7 JBoss Tools		logic:redirect		forward	Struts	
JBoss Servers		,,				
Packaging Archives						
▽ Web						
Editors	H					
▷ JSF						
Seam						
Automation						
Customization						
Project						
Struts Pages						
Verification	•				Restore Defaults	Apply
0					ок	Cancel

#### Figure 6.19. Struts Customization

# 6.16. Struts Flow Diagram

Similarly to the JSF Flow Diagram screen, selecting *JBoss Tools > Web > Editor > Struts Flow Diagram* page allows you to specify aspects of the Diagram mode of the Struts configuration file editor. The Struts Flow Diagram screen adds an option to hide the Diagram tab and labeling settings for additional artifacts.

<b>9</b>	Prefe	rences
type filter text	Struts Flow Diag	ram 🔶 🕶 🖒 👻
<ul> <li>General</li> <li>Ant</li> <li>Connectivity</li> <li>FreeMarker Editor</li> <li>Help</li> <li>HQL editor</li> <li>Install/Update</li> <li>Internet</li> <li>Java</li> <li>JBoss jBPM</li> </ul>	Struts Flow Diagra Struts Flow Diagra Grid Step: Action Font: Forward Font: Path Font:	
<ul> <li>✓ JBoss Tools</li> <li>▷ JBoss Servers</li> <li>Packaging Archives</li> <li>✓ Web</li> <li>✓ Editors</li> <li>JSF Flow Diagram</li> <li>Struts Flow Diagram</li> <li>Visual Page Editor</li> <li>▷ JSF</li> <li>▷ Seam</li> <li>▷ Struts</li> <li>Verification</li> </ul>	Comment Font:	default,size=10,style=0       ⊆hange         Switch to control mode after transition is made         Show shortcut icon         Show shortcut path             Restore Defaults
$\widehat{O}$		OK Cancel

#### Figure 6.20. Struts Flow Diagram

Selecting the Add Page tab in the Struts Flow Diagram screen allows you to determine the default template and file extension for views (pages) you add directly into the diagram using a context menu or the view-adding mode of the diagram cursor.

<b>9</b>	Preferences	×
type filter text	Struts Flow Diagram	<b>⇔</b> • ⇔-
General	Struts Flow Diagram Add Page	
▷ Ant		
Connectivity	Page Template: StrutsForm	<b></b>
FreeMarker Editor	Extension: jsp	-
▶ Help		
HQL editor		
Install/Update		
Internet	=	
▷ Java		
JBoss jBPM		
JBoss Servers		
Packaging Archives		
∀ Web		
JSF Flow Diagram		
Struts Flow Diagram		
Tiles Diagram		
Visual Page Editor		
▷ JSF		
Seam		
Struts		
Verification	Restore Defaults	Apply
Doclet XDoclet	•	
٢	ОК	Cancel

## Figure 6.21. Adding Page

# 6.17. Struts Pages

You can change the following preferences on the JBoss Tools > Web > Struts > Struts Pages preference page.

On Struts Pages panel you can add or remove Struts pages.

<b>8</b>	Preferences	×
type filter text	Struts Pages	<b>⇔</b> • ⇔•
General		
▶ Ant	Blank [default]	Add
Connectivity	StrutsForm	Remove
FreeMarker Editor		
▶ Help		
HQL editor		
Install/Update		
Internet	=	
⊅ Java		
JBoss jBPM		
JBoss Servers		
Packaging Archives		
⊽ Web		
Editors		
▷ JSF		
Seam		
Automation		
Customization		
Project		
Struts Pages		
		Set Default
0	ОК	Cancel

#### Figure 6.22. Struts Pages

# 6.18. Struts Project

You can change the following preferences on the *JBoss Tools* > *Web* > *Struts* > *Project* preference page:

On *Project* panel you define a template for a new Struts created project: servlet version, page template and so on.

😌 Preferences 🗙			
type filter text		Project	<b>⇔</b> • ⇔∗
General	<u>^</u>	New Project Import	tProject
▶ Ant			
Connectivity		Struts Version:	Struts 1.1 🗘
FreeMarker Editor		Project Template:	Blank
Help		inoject lemplate.	
HQL editor			☑ Use Default Path
Install/Update		Projects Root:	/opt/work/workspace_beta
Internet	=	Servlet Version:	2.3
▷ Java		Serviet version:	2.3
JBoss jBPM			Register Web Context in server.xml
		Page Template:	<b>(</b>
JBoss Servers			
Packaging Archives			
⊽ Web			
Editors	Н		
▷ JSF			
Seam			
Automation			
Customization			
Project			
Struts Pages			
Verification	J		Restore Defaults Apply
0			OK Cancel

#### Figure 6.23. Struts Project

Selecting the Import Project tab in the Struts Project screen allows you to determine the default servlet version and whether to register Web Context in server.xml.

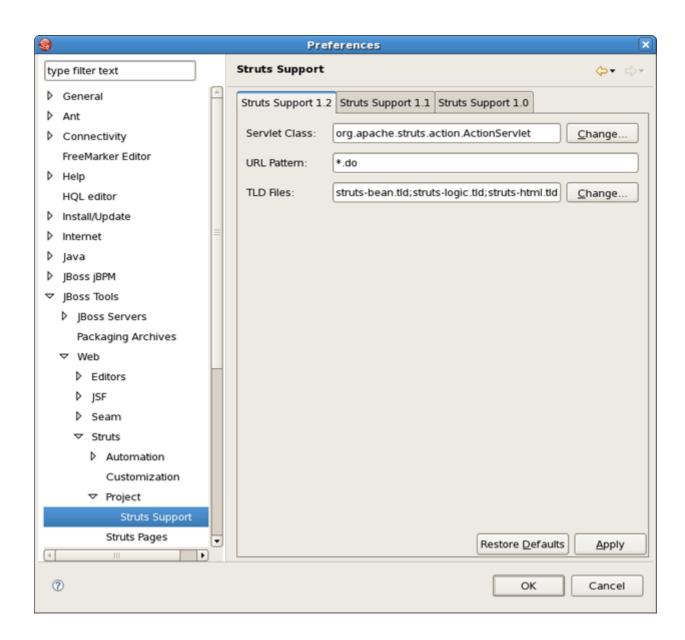
<b>9</b>	Preferences	×
type filter text	Project	<b>⇔</b> ∙ ⇔∗
General	New Project Import Project	
▷ Ant		
Connectivity	Servlet Version: 2.3	<b></b>
FreeMarker Editor	Register Web Context in server.xml	
▶ Help		
HQL editor		
Install/Update		
Internet	=	
▷ Java		
JBoss jBPM		
JBoss Servers		
Packaging Archives		
∀ Web		
Editors		
▷ JSF		
Seam		
Automation		
Customization		
Project		
Struts Pages		
Verification	Restore Defaults	Apply
	Restore Derauits	дрруу
0	ОК	Cancel

#### Figure 6.24. Import Struts Pages

# 6.19. Struts Support

The following preferences can be changed on the *JBoss Tools* > *Web* > *Struts* > *Project* > *Struts Support* page.

Select Struts Support screen if you want to configure Struts versions support settings.



#### Figure 6.25. Struts Support

## 6.20. Title Diagram

JBoss Tools > Web > Editors > Title Diagram screen allows you control some settings for the placement of Tiles definitions in the Diagram mode of the JBoss Tools Tiles editor.

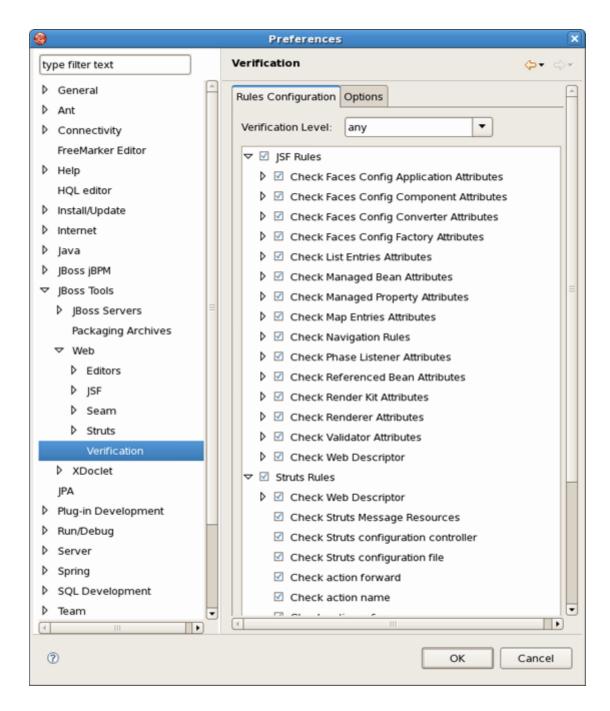
<b>8</b>	Preferences	×
type filter text	Tiles Diagram	<b>⇔</b> • ⇔⇒
<ul> <li>▷ General</li> <li>▷ Ant</li> <li>▷ Connectivity</li> <li>FreeMarker Editor</li> <li>▷ Help</li> <li>HQL editor</li> <li>▷ Install/Update</li> <li>▷ Internet</li> <li>▷ Java</li> <li>▷ JBoss JBPM</li> <li>▽ JBoss Tools</li> </ul>	Tiles Diagram         Vertical Spacing:         Vertical Spacing:         default(20)         Horizontal Spacing:         default(185)         Definition Name Font:         default,size=7,style=1         Layout Animation	
<ul> <li>JBoss Servers Packaging Archives</li> <li>Web</li> <li>Editors JSF Flow Diagram Struts Flow Diagram</li> <li>Tiles Diagram</li> <li>Visual Page Editor</li> <li>JSF</li> <li>Seam</li> <li>Struts Verification</li> </ul>	▼ Restore Default	s <u>A</u> pply
0	ОК	Cancel

## Figure 6.26. Title Diagram

## 6.21. Verification

The following preferences can be changed on the *JBoss Tools* > *Web* > *Verification* page.

On Rules Configuration tab of Verifcation panel you can determine JSF and Struts rules.



#### Figure 6.27. Verification

On Options tab you can define a limit for the reported errors number.

<b>9</b>	Preferences	×
type filter text	Verification	<b>⇔</b> • ⇔•
<ul> <li>▷ General</li> <li>▷ Ant</li> <li>▷ Connectivity</li> <li>FreeMarker Editor</li> <li>▷ Help</li> <li>HQL editor</li> <li>▷ Install/Update</li> </ul>	Rules Configuration Options Reported Errors Number Limit: unlimited	
<ul> <li>▷ Internet</li> <li>▷ Java</li> <li>▷ JBoss jBPM</li> <li>▽ JBoss Tools</li> <li>▷ JBoss Servers</li> <li>Packaging Archive</li> <li>▽ Web</li> <li>▷ Editors</li> <li>▷ JSF</li> </ul>		E
<ul> <li>▷ Seam</li> <li>▷ Struts</li> <li>Verification</li> <li>▷ XDoclet JPA</li> <li>▷ Plug-in Development</li> <li>▷ Run/Debug</li> <li>▷ Server</li> <li>♥</li> </ul>		•
٢	ОК	Cancel

Figure 6.28. Options of Verification

## 6.22. View

The following preferences can be changed on the *JBoss Tools* > *JBoss Servers* > *View* page.

The View shows you preferences for JBoss Servers view.

•				Prefe	rences	
type filter text	]	View				⇔- ⇔
General	<b>^</b>					
Þ Ant			[			
Connectivity			Name	Enabled	Description	
FreeMarker Editor			Modules		Allow modules to be included in the tree.	
> Help		Move Up	i Event Log		A place to view server-related events	
HQL editor		Move Down	XML Configuration		An extension that lets you quickly change often-used fields or ports	
Install/Update			🕞 File Filters		A front end to open, edit, or view files in your server's directories.	
Internet			lnactive Categories		Lists inactive categories and allows you to turn them back on.	
Java						
JBoss jBPM						
JBoss Tools	=	Below, you m	av set or clear any pref	erences sp	ecific to each view extension above.	
✓ Boss Servers			ons may have no prefe			
View						
Packaging Archives						
♦ Web						
XDoclet						
JPA						
Plug-in Development						
Run/Debug						
Server						
Spring						
SQL Development						
Team						
TestNG	-				Restore Defaults	Apply
0					ок	Can

#### Figure 6.29. View

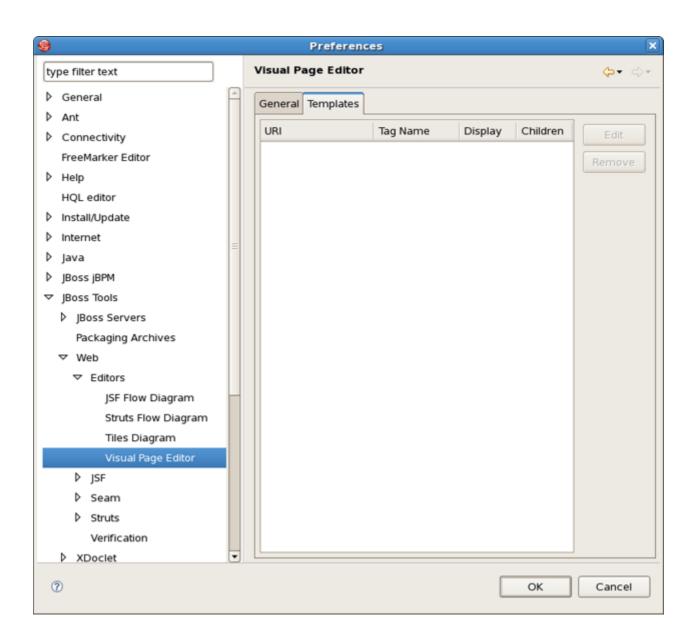
# 6.23. Visual Page Editor

*JBoss Tools > Web > Editors > Visual Page Editor* screen allows you to control some aspects of the behavior of the Visual Page Editor (VPE) for JSP files. Also you can define a default editor tab.

<b>8</b>	Prefere	nces X
type filter text	Visual Page Editor	<b>⇔</b> • ⇔∗
General	General Templates	
▶ Ant		
Connectivity		Show Border for Unknown Tags
FreeMarker Editor		Show Resource Bundles Usage as EL Expressions
▷ Help		Always Prompt for Tag Attributes During Tag Insert
HQL editor		
Install/Update		✓ Show Selection Tag Bar
▶ Internet =		Always Hide Selection Bar Without Prompt
⊅ Java	Default Editor Tab:	Visual/Source
▷ JBoss jBPM		
JBoss Servers		
Packaging Archives		
⊽ Web		
✓ Editors     ✓		
JSF Flow Diagram		
Struts Flow Diagram		
Tiles Diagram		
Visual Page Editor		
Þ js⊧		
Seam		
Struts		
Verification		Restore Defaults Apply
		Restore Defaults Apply
0		OK Cancel

## Figure 6.30. Visual Page Editor

On the Templates tab you can edit or remove VPE templates.



#### Figure 6.31. Visual Page Editor Templates

## 6.24. XDoclet

The following preferences can be changed on the *JBoss Tools* > *XDoclet* page.

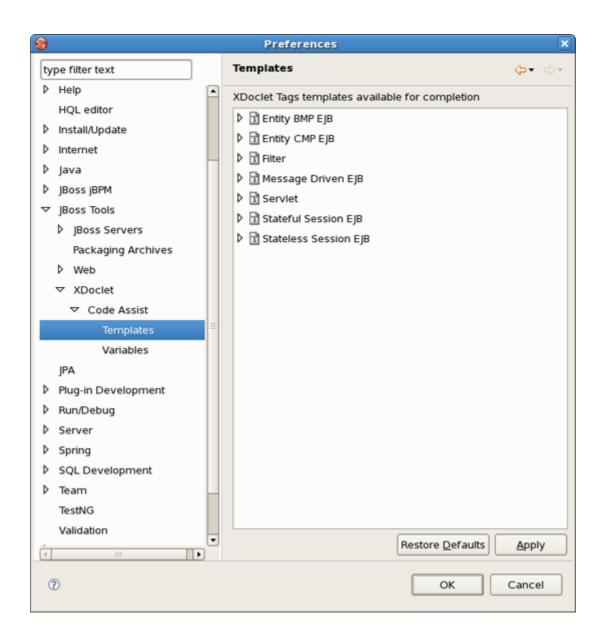
On XDoclet screen you determine XDoclet module versions.

<b>9</b>	Preferences	×
type filter text	XDoclet	⇔• ⇔+
<ul> <li>▶ General</li> <li>▶ Ant</li> <li>▶ Connectivity FreeMarker Editor</li> <li>▶ Help HQL editor</li> <li>▶ Install/Update</li> <li>▶ Internet</li> <li>▶ Java</li> <li>▶ JBoss JBPM</li> <li>▼ JBoss Tools</li> <li>▶ JBoss Servers</li> </ul>	Allows to refresh the modules list from the disk JBoss.net xdoclet module version <ul> <li>4.0.x</li> <li>3.2.x</li> </ul> Refresh XDoclet Modules	
Packaging Archives  Veb  XDoclet JPA  Plug-in Development Run/Debug Server Spring		
<ul> <li>SQL Development</li> <li>Team</li> <li>TestNG</li> <li>Validation</li> <li>7</li> </ul>	▼ Restore <u>D</u> efau	Its <u>A</u> pply Cancel

Figure 6.32. XDoclet

# 6.25. XDoclet Templates

To see what XDoclet tags templates available for completion select *JBoss Tools > XDoclet > Code Assist > Templates*.



#### Figure 6.33. XDoclet Templates

By right clicking on any tree element you can add a new template, remove or rename it and also add a new tree.

Select, for example, Add and you'll be prompted by a list of available elements to add.

<b>9</b>	Preferences	×
type filter text	Templates	<b>⇔</b> + ⇔+
type filter text <ul> <li>General</li> <li>Ant</li> <li>Connectivity</li> <li>FreeMarker Editor</li> <li>Help</li> <li>HQL editor</li> <li>Install/Update</li> <li>Internet</li> <li>Java</li> <li>JBoss JBPM</li> <li>JBoss Tools</li> <li>JBoss Servers</li> <li>Packaging Archives</li> <li>Web</li> <li>XDoclet</li> <li>Code Assist</li> </ul> <li>Templates</li> <li>JPA</li> <li>Plug-in Development</li> <li>Run/Debug</li>	Templates         XDoclet Tags templates available for completion         Image: Entity BMP EJB         Image: Entity CMP EJB         Image: Entity Edge:	
▷ Server		
▷ Spring	Restore Def	aults Apply
SQL Development	•	
0	ок	Cancel

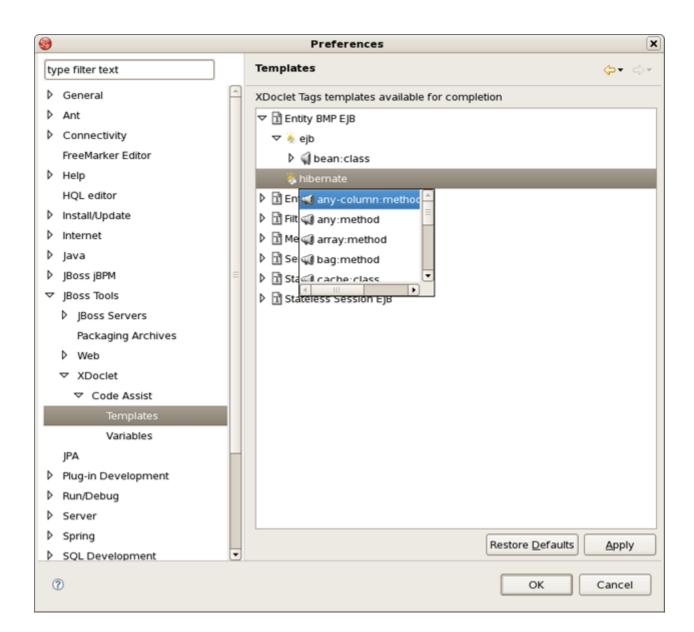
## Figure 6.34. Add New XDoclet Template

By double-clicking some element a new list will be shown: now for available methods.

<b>(</b>	Preferences	×
type filter text	Templates	<b>⇔</b> • ⇔∗
<ul> <li>General</li> <li>Ant</li> <li>Connectivity</li> <li>FreeMarker Editor</li> <li>Help</li> <li>HQL editor</li> <li>Install/Update</li> </ul>	XDoclet Tags templates available for completion         Image: Completion in the second seco	
<ul> <li>▷ Internet</li> <li>▷ Java</li> <li>▷ JBoss jBPM</li> <li>▽ JBoss Tools</li> <li>▷ JBoss Servers</li> <li>Packaging Archives</li> <li>▷ Web</li> <li>▽ XDoclet</li> <li>▽ Code Assist</li> </ul> Templates Variables	Stateless Session EJB	
JPA <ul> <li>Plug-in Development</li> <li>Run/Debug</li> <li>Server</li> <li>Spring</li> </ul>		
SQL Development      ⑦	•	Defaults Apply

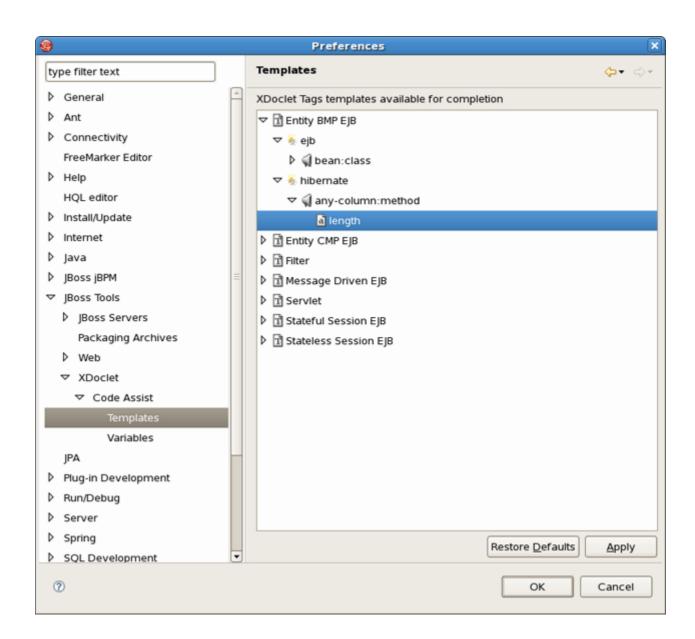
## Figure 6.35. Select New Element

Choose any element from the list, then select an attribute.



#### Figure 6.36. Select New Element

It will be added to the available templates.



#### Figure 6.37. Select New Element

## 6.26. XDoclets Variables

By selecting JBoss Tools > XDoclet > Variables you define variables used in templates.

Changing Default Environment During Project

Creation

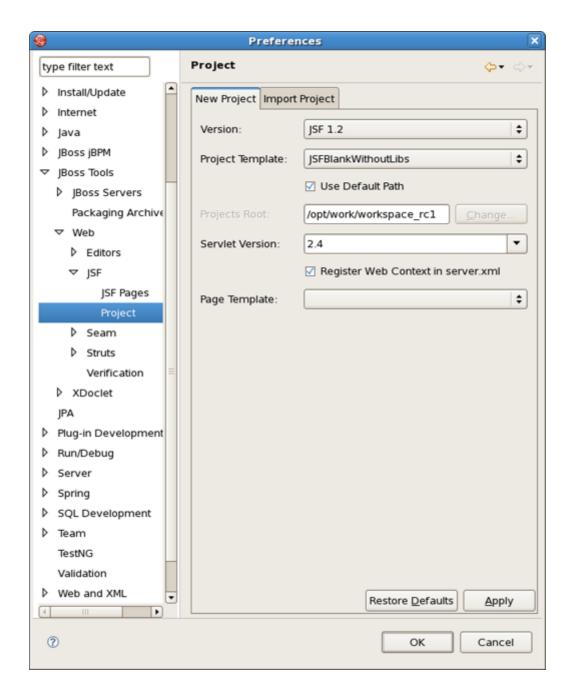
9	Preferences		×
type filter text	Variables		<b>⇔</b> ∙ ⇔∗
▷ Help	Definition of the variab	les used in templates	
HQL editor	Classname:	^(\w*)\$	
Install/Update			
▷ Internet	Classname Subset:	^(\w*)Bean\$	
⊅ Java	Packagename:	^(\S*)\$	
▷ JBoss jBPM	Parent Packagename:	^(\S*)\.\w*\$	
	-		
JBoss Servers			
Packaging Archives			
VWeb			
Templates			
Variables			
JPA			
Plug-in Development			
Run/Debug			
▷ Server			
▷ Spring			
SQL Development			
▶ Team			
TestNG			
Validation			
· (       )		Restore Defaults	Apply
0		ок	Cancel

Figure 6.38. XDoclets Variables

# 6.27. Changing Default Environment During Project Creation

To change the default environment and project template for either JSF or Struts new project creation:

- 1. Select Window > Preferences > JBoss Tools > Web > {JSF or Struts} > Project
- 2. For Version set the environment you want to be the default one
- 3. For Project Template set the template you want to be the default one



#### Figure 6.39. Changing Enviroment Template

# 6.28. Changing Default Project Template During Project Creation

To change the default project template for either JSF or Struts new project creation:

1. Select Window > Preferences > JBoss Tools > Web > {JSF or Struts} > Project

2. For Project Template set the template you want to be the default one

## Changing Default Project Template During

**Project Creation** 

Preferences         X		
type filter text	Project	<b>⇔</b> • ⇔∗
Install/Update Install/Update	New Project Import	Project
▷ Internet ▷ Java	Struts Version:	Struts 1.1
▷ JBoss jBPM	Project Template:	Blank
⊐ JBoss Tools	rioject lemplate.	
JBoss Servers		✓ Use Default Path
Packaging Archive	Projects Root:	/opt/work/workspace_rc1 Change
⊽ Web	Servlet Version:	2.3
Editors		
▷ JSF		Register Web Context in server.xml
▷ Seam Struts	Page Template:	
Automation		
Customizati		
Project		
Struts Pages		
Verification		
♦ XDoclet		
JPA		
Plug-in Development		
▶ Run/Debug		
Server		
Spring		
SQL Development		
Team TestNG		
		Restore Defaults Apply
0		OK Cancel

Figure 6.40. Changing Project Template