Struts Tools Reference Guide

Version: 3.1.1.GA

1. Introduction	. 1
1.1. Key Features of Struts Tools	. 1
1.2. Other relevant resources on the topic	. 2
2. Projects	3
2.1. Creating a New Struts Project	. 3
2.2. Importing an Existing Struts Project with Any Structure	. 8
2.3. Adding Struts Capability to an Existing Web Application	. 9
2.4. Relevant Resources Links	16
3. Editors	17
3.1. Web.xml Editor	17
3.2. Graphical Editor for Struts Configuration Files	17
3.2.1. Diagram View	17
3.2.2. Tree View	20
3.2.3. Source View	22
3.3. Graphical Editor for Tiles Files	24
3.3.1. Create New Tiles File	24
3.3.2. Tree View	25
3.3.3. Diagram View	27
3.3.4. Source	29
3.4. Graphical Editor for Struts Validation Files	31
4. Modules	39
4.1. When Importing a Struts Project	39
4.2. Editing Modules in an Existing Project	
4.3. Adding New Modules	41
5. Code Generation	43
6. Struts Configuration File Debugger	47
7. Customizable Page Links Recognizer	49
8. Struts Project Verification	
9. Relevant Resources Links	55

Introduction

JBoss Developer Studio provides a number of editors and wizards that allow Struts developers to quickly and efficiently build high quality web applications.



Note:

Note that JBoss Tools support the Struts 1.1, 1.2.x versions.

In this guide you will learn how to take advantage of Struts support that is provided by JBoss Developer Studio.

1.1. Key Features of Struts Tools

The table below lists the main features provided by Struts Tools:

Feature	Benefit	Chapter
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.	Chapter 2, Projects
Support for Struts Configuration File	Working on files using three modes: diagram, tree and source. Synchronization between the modes and full control over the code. Easily move around the diagram using the Diagram Navigator. Working with struts projects that have multiple modules. The option to use thee Struts configuration file debugger, which allows break points to be set on a Struts diagram and then launch the server in debug mode.	Section 3.2, "Graphical Editor for Struts Configuration Files" Chapter 6, Struts Configuration File Debugger
Support for Struts modules	A Struts module (struts-config.xml) is automatically created when creating a new project. There is also the option to add new or edit already existing modules in your existing project or while importing Struts project.	Chapter 4, Modules
Verification and Validation	All errors will be immediately reported by verification feature, no matter which view you are working in. Constant validation and errors checking allows developers to catch many of the errors during development process, which significantly reduces development time.	Chapter 8, Struts Project Verification

Table 1.1. Key Functionality for Struts Tools

1.2. Other relevant resources on the topic

All JBoss Developer Studio/JBoss Tools documentation you can find on *JBoss Tools project page* [http://docs.jboss.org/tools/].

The latest documentation builds are available on *nightly docs page* [http://download.jboss.org/ jbosstools/nightly-docs/].

Projects

JBoss Tools provides the following functionality when working with Struts:

- Create new Struts projects
- Import (open) existing Struts projects. You can import any project structure
- Add Struts capabilities to any existing Eclipse project
- Import and add Struts capabilities to any existing project created outside Eclipse.

Now, we'll focus on all these points more fully.

2.1. Creating a New Struts Project

JBoss Tools provides a **New Struts Project** Wizard that significantly simplifies the process of creating a new Struts project. You just need to follow these steps:

Select File → New → Project... from the menu bar. Then, select JBoss Tools Web → Struts
 → Struts Project from the dialog box. Click the Next button:

	New	×
Select a wizard		
Create a Struts Project		
<u>W</u> izards:		
type filter text		
Properties Hie		<u>^</u>
🕉 TLD File		
💾 Web Descriptor		
🚵 XHTML File		
👂 🗁 JSF		
👂 🗁 Portlet		
▽ 🗁 Struts		
🛋 Struts Config		
🕉 Struts Project		~
(?) < Back	Next > Cancel	Einish
		L'IIIII

Figure 2.1. Selecting Struts Wizard

• On this form, provide the Project Name. You can also select where to create the project, or use the default path.

The Struts Environment sets the Struts version that will be used.

•	New Struts Project 🗙
Create Struts Proje	ct 🚳
a brand new project. If	t Wizard is used for creating you already have a pre-existing port Project Wizard to start working
Project Name*	StrutsProject
	✓ Use default path*
Location*	opt/work/workspace_jbds/StrutsProject
Struts Environment*	Struts 1.1
Template*	Struts 1.2 Didlik
0	<u>Back</u> <u>N</u> ext > <u>F</u> inish Cancel

Figure 2.2. Creating Struts Project



You can select the KickStart template, which results in a project that includes a simple Hello World type application that is ready to run.

	New Struts Project 🗙
Create Struts Proje	et 🚳
a brand new project. If	ct Wizard is used for creating f you already have a pre-existing nport Project Wizard to start working
Project Name*	StrutsProject
	✓ Use default path*
Location*	opt/work/workspace_jbds/StrutsProject
Struts Environment*	Struts 1.1
Template*	Blank
	KickStart
0	< Back Next > Einish Cancel

Figure 2.3. Choosing Struts Template

• Next, you register this application with the current servlet container defined for your workspace (JBoss AS, by default) in order to allow you to easily test your application while still developing it. A new entry will be added in the servlet container configuration file to enable the application to be run in-place (called null deployment or link deployment). Uncheck the *"Target Server"* check box if you prefer not to register your application at this point.

¢	New Struts Project	×
Web		
Servlet Version	2.4	
Context Path*	strutsApplication	
Runtime:*	JBoss 4.2 Runtime	New
Target Server:	JBoss Application Server 4.2	New Select All
		Deselect All
0	< <u>B</u> ack <u>N</u> ext > <u>F</u> inish	Cancel

Figure 2.4. Registering The Project at Server

• On the next form, you can select the TLD files to include in this project:

Rew Struts Project
Tag Libraries Select tag library files you want to use in your project
TLDs
 ✓ struts-html.tld ✓ struts-bean.tld □ struts-nested.tld □ c.tld □ x.tld □ struts-tiles.tld □ sql.tld □ frmt.tld ✓ struts-logic.tld
<pre></pre>

Figure 2.5. Selecting Tag Libraries

After the project is created, you should have the following project structure (if you used the KickStart template):

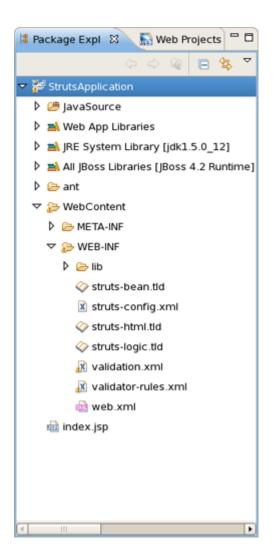


Figure 2.6. Project Structure



Tip:

If you want to hide the JAR files from Web App Libraries in view, select the downpointing arrow in the upper right corner, select **Filters...**, check the box next to Name filter patterns (matching names will be hidden), and type *.jar into the field. Then, click the **OK** button.

2.2. Importing an Existing Struts Project with Any Structure

For detailed information on migration projects to JBoss Developer Studio see the Migration Guide.

2.3. Adding Struts Capability to an Existing Web Application

This section will describe how to add Struts functionality (Struts libraries, tag libraries and a Struts configuration file) to any existing Web application project in your Eclipse workspace.

By adding a Struts Nature to your project, you can now edit files using JBoss Tools editors, such as the *Struts configuration editor* and the JBoss Tools JSP editor. To take advantage of this just right-click the project and select **JBoss Tools** \rightarrow **Add Struts Capabilities** from the context menu. This will start the process of adding all the necessary Struts libraries and files to your existing project.

Package Exp	N 😫 🔚 Web Projects 🗖 🗖		
	수 수 @ 🖪 😫 🏹		
> 🐸 JSFHello	Ne <u>w</u>	>	
> 🐉 StrutsApp	Go <u>i</u> nto		
	Open in <u>N</u> ew Window		
	Ope <u>n</u> Type Hierarchy	F4	
	Sho <u>w</u> In	Shift+Alt+W 🕻	
	<u>[]</u> <u>С</u> ору	Ctrl+C	
	🗎 Copy Qualified Name		
	💼 <u>P</u> aste	Ctrl+V	
	💢 <u>D</u> elete	Delete	
	le Remove from Context	Shift+Ctrl+Alt+Down	
	<u>B</u> uild Path	>	
	<u>S</u> ource	Shift+Alt+S 🕻	
	Refactor	Shift+Alt+T 🕻	
	🖮 Import		
	✓ Export		
	🖑 Re <u>f</u> resh	F5	
	Clo <u>s</u> e Project		
	Close <u>U</u> nrelated Projects		
	Assign Working Sets		
	😯 Convert to Drools Project		
	<u>R</u> un As	>	
	Debug As	>	
	<u>P</u> rofile As	>	
	Validate		
	m2 Maven	>	
	T <u>e</u> am	>	
	Comp <u>a</u> re With	>	
	Restore from Local History		🔇 Add Struts Capabilities
	Instrument <u>a</u> tion Source	>	Add Seam support
	JBoss Tools	>	Add Custom Capabilities
	Java EE Tools	>	Remove JSF Capabilities
	Configure	<u> </u>	Add CDI support
	Properties	Alt+Enter	Add JavaScript support Convert to Plug-in Projects.

Figure 2.7. Adding Struts Capabilities

In the wizard you should point to the location of your deployment descriptor file web.xml and name of the project.

•	Add Struts Capabilities	×
Project Location Please select web.xn	nl location	
Project Name*	JSFHello	
web.xml Location*	/opt/work/jbds_GA/workspace/JSFHello/Web	B <u>r</u> owse
?	ack Next > Cancel	Einish

Figure 2.8. Choosing Project Location

After clicking the **Next** button you will see the following screen. This screen simply indicates that you need to add at least one Struts module to your project to make this project a Struts project. Adding a Struts module means that a new struts-config.xml will be added to your project and registered in the web.xml file. In addition, all required Struts libraries will be added. To add a Struts module, click the **Add Struts Support** button.

lame	URI	
ame 🗌		
u 🗌		
th on Disk		Browse
urces Path		Browse
odule Root		Browse
	Add Struts Support	

Figure 2.9. Project Modules

Here you can select which Struts Version, Servlet Class, URL Pattern and TLDs to add to this project.

•	Add Struts Support 🗙
Struts Project Please select stru	uts support options
Version*	1.2
Servlet Class:*	org.apache.struts.action.ActionServlet Browse
URL Pattern:*	*.do
TLD Files:*	 ✓ struts-html.tld ✓ struts-bean.tld □ struts-nested.tld □ c.tld □ x.tld □ struts-tiles.tld □ sql.tld □ fmt.tld ✓ struts-logic.tld
?	Cancel <u>F</u> inish

Figure 2.10. Selecting Struts Support Options

When done, you will see the default Struts module configuration information. See how to Edit *Struts modules*.

•	Add Struts Capabilities	
Project Mode Configure Proje		
Name	URI	
<default></default>	/WEB-INF/struts-config.xml	
Name	<default></default>	
URI	/WEB-INF/struts-config.xml	
Path on Disk	pt/work/jbds_GA/workspace/JSFHello/WebContent/WEB-INF/struts-config.xml	Browse
Sources Path	/opt/work/jbds_GA/workspace/JSFHello/JavaSource	Browse
Web Root	/opt/work/jbds_GA/workspace/JSFHello/WebContent	Browse
0	< <u>B</u> ack <u>N</u> ext > Cancel	Einish

Figure 2.11. Project Configuration Information

On the last screen you can set the different folders for your project as well as register this application with a servlet container. If you want the libraries (.jar files) to be automatically added to your project, tick the **Add Libraries** checkbox.

•	Import Struts Project	X
Project Folders Select Project Fol		
Classes Folder Lib Folder Ant Build File	/opt/work/jbds_GA/workspace/JSFHello/WebContent/WEB-INF/classes Browse /opt/work/jbds_GA/workspace/JSFHello/WebContent/WEB-INF/lib Browse Browse	
Environment Servlet Version:		•
Context Path* Runtime:*	JSFHello JBoss 4.2 Runtime New	
Target Server:	Choice list is empty. New Select Al Deselect Al	2
0	< <u>B</u> ack <u>Next</u> > Cancel <u>Finish</u>	

Figure 2.12. Registering the Project at Server

When done, you can open and edit the struts-config.xml file using useful Struts configuration file editor provided by JBDS. (The Struts configuration is shown below in the Tree viewer).

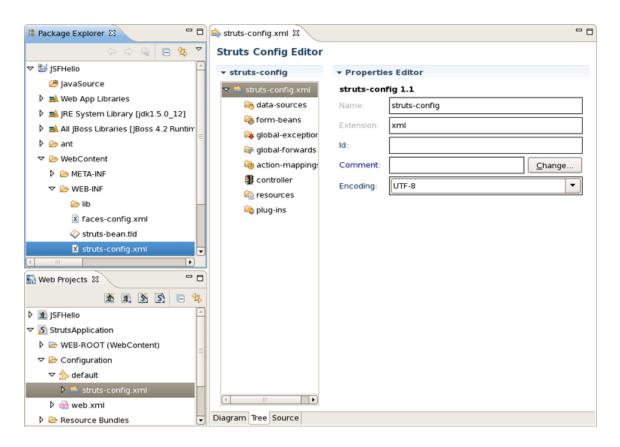


Figure 2.13. Struts-config.xml File

2.4. Relevant Resources Links

You can find more in-depth explanation on how to work with special wizards, editors and views that can be used in various scenarios while developing Struts applications in our Visual Web Tools Guide.

Editors

This chapter will cover the graphical editors provided for editing specific Struts files such as Struts Configuration files, Tiles files, Struts Validation files and web.xml.

3.1. Web.xml Editor

The web.xml file inside the WEB-INF folder is a deployment descriptor file for a Web Application. It describes the servlets, other components and deployment properties that make up your application.

JBoss Tools adds the web.xml file to a Struts project automatically and provides a special editor through which the file can be edited. See the Visual Web Tools guide that gives a descriptive information on the web.xml editor.

3.2. Graphical Editor for Struts Configuration Files

First, let's dwell on the Struts Configuration file editor.

This editor provides three views of the struts-config.xml file: Diagram, Tree and Source. The views can be selected using the tabs at the bottom of the editor. Any changes made in one view are immediately visible when you switch to any other view.

Now, we'll consider each of these three views in more detail.

3.2.1. Diagram View

The Diagram view graphically displays the Web flow of the application defined in the Struts configuration file.

struts-config.x			
🤗 getNa	me		
	/page s	/inputname.jsp	
		∑	
	≥	_	
		/greeting	
		SetNameForm	
		sayHello	
			/pages/greeting.jsp
			/pages/greeting.jsp
gram Tree So			

Figure 3.1. Diagram View

The Diagram view provides a way to edit the navigation in your Struts application. By right-clicking anywhere on the diagram, you can use a context menu to create the building blocks of a Struts application:

- Actions
- Global forwards
- Global exceptions
- JSP Pages

R	ruts-config.xml 업				
	裬 getName				
*		Add	•	👋 Action	
ø		Create Comment		💍 Global Forward	
		👔 Paste	Ctrl + V	Global Exception Page	
		Select Element Auto-Layout			
		🤒 Verify		/pages/greeting.jsp	
		Properties			
		Senerate Java Code			
Diag	ram Tree Source]	

Figure 3.2. Diagram Context Menu

Along the upper-left side of the editor is a stack of seven icons for changing the behavior of the cursor in the diagram.



Figure 3.3. Editor Icons

The first icon switches to the default regular selection cursor, the second to the marquee selection cursor and the third to the new connection cursor. The last four icons switch the cursor to an insert cursor for each type of Struts build block listed above (and in the order listed).

For instance, clicking on the first of these four icons (the one with the gears) will switch the cursor to insert actions. Clicking anywhere in the diagram with this cursor has the same effect as right-click and selecting $Add \rightarrow Action...$ from the context menu with the regular cursor active. It's just more efficient to use this cursor if you're adding more than one action at once.

3.2.2. Tree View

The Tree view represents the different elements of the Struts application that are organized into functional categories on the left-hand side and a form for editing the properties of currently selected items on the right-hand side.

🕏 struts-config.xml ध			- 8
Struts Config Editor			
▼ struts-config	• Properties Ed	itor	<u>^</u>
マ 🚔 struts-config.xml	forward		
🛵 data-sources	ld:		
li form-beans li i i i i i i i i i i i i i i i i i i	ClassName:	Brow	vse
v 🖗 global-forwards	ContextRelative:		-
🥐 getName	Module:		=
	Name:	sayHello	
	Path:	/pages/greeting.js; Char	nge
<pre>controller</pre>	Redirect:		-
kan resources kan	Small-Icon:		
	Large-Icon:		
(III)	Display-Name:		
Diagram Tree Source			

Figure 3.4. Tree View

You can right-click on any node in the category tree to perform a number of operations through a context menu. For instance, by right-clicking on the action-mappings category node, you can add new actions to the application.

🕏 struts-config.xml 🛙			- 8
Struts Config Editor			
	• Properties	Editor	
 ✓ i⇒ struts-config.xml ia data-sources ia form-beans ia global-exceptions ia global-forwards ia getName 	action-map ld: [Type: [Comment: [pings	<u>C</u> hange
✓ ¼ /greeting	Create Action Sort		
controller c	Paste Properties Verify	Ctrl + V	
Diagram Tree Source			

Figure 3.5. Tree Context Menu

Let's consider the tree on the left more closely.

- Under the *data-sources* node you can create a Data Source object that will be configured and made available as a servlet context attribute.
- The *form-beans* node is meant for creating a set of form bean descriptors for this module. Every element under this node is a JavaBean that implements the org.apache.struts.action.ActionForm class. Use the Properties editor on the right to adjust the properties specific to each form-bean.
- The *global-exceptions* node is intended for registering handlers for the exceptions that might be thrown by an Action object.
- Use the *global-forwards* node to add, edit or delete a global forwards that represent ActionForward objects available to all Action objects as a return value.
- The controller node allows you to configure the controller properties.
- Under the resources node you can add, delete, or edit message resources.
- Under the *plug-ins* node you can define a Struts plug-in. Right-click the node, select **Create Plug-in** and specify the plug-in *Id* and *ClassName* by pointing to the Java class which implements the org.apache.struts.action.PlugIn interface.

3.2.3. Source View

In the Source view, you have complete editing control of the underlying XML coding.

```
- 8
농 struts-config.xml 😫 🗋
 <?xml version="1.0" encoding="ISO-8859-1"?>
 <struts-config>
  <data-sources/>
  <form-beans/>
     <form-beans name="GetNameForm" type="demo.GetNameForm"/>
  </form-beans>
  <global-exceptions/>
  <global-forwards>
   <forward name="getName" path="/pages/inputname.jsp"/>
  </global-forwards>
  <action-mappings>
   <action name="GetNameForm" path="/greeting"score="request" type="de</pre>
    <forward name="sayHello" path="/pages/greeting.jsp"/>
   </action>
  </action-mappings>
  <controller/>
  </struts-config
                                                              Diagram Tree Source
```

Figure 3.6. Source View

When working in Source view, you always have all the following features available:

- Content Assist
- Open On Selection
- File Folding

You can take advantage of code assist.

🗟 struts-config.xml 🛙	
<pre><?xml version="1.0" encoding="ISO-8859-1"?> <!DOCTYPE struts-config PUBLIC "-//Apache Software</td> <td>Foundation//DTD St</td></pre>	Foundation//DTD St
<struts-config></struts-config>	
<pre><data-sources></data-sources></pre>	Element : display-name
<> action-mappings <> controller	The "display-name" element contains a short (one line) description of the surrounding element, suitable for use in GUI
<> data-sources	tools.
<> description	Data Type : #PCDATA
«» display-name	
<> form-beans	
<> global-exceptions	
<> global-forwards	
<> message-resources	
<> plug-in	
# comment - xml comment	
	V
	Þ
Diagram Tree Source	

Figure 3.7. Code Assist

The editor will also immediately flag any errors.

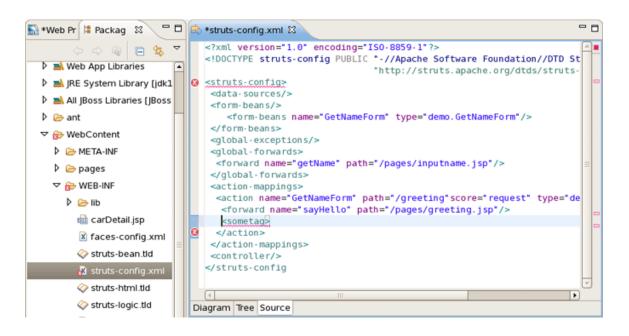


Figure 3.8. Errors in Source View

Finally, you can use the Outline view with the editor to easily navigate through the file.

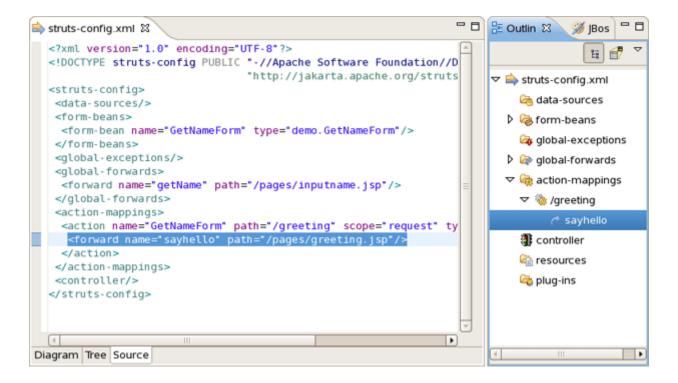


Figure 3.9. Outline View

Find more information about editor features in the editor features chapter.

3.3. Graphical Editor for Tiles Files

Here, you'll know how to make use of the special graphical editor for Tiles configuration files.

The editor has three main views: Tree, Diagram and Source. The views can be selected via the tabs at the bottom of the editor. Any changes made in one view are immediately visible when you switch to any other view.

Before we consider each view of the editor, let's look at the way of creating new Tiles files.

3.3.1. Create New Tiles File

To create new Tiles files, right click any folder and select $\textbf{New} \rightarrow \textbf{Tiles File}.$

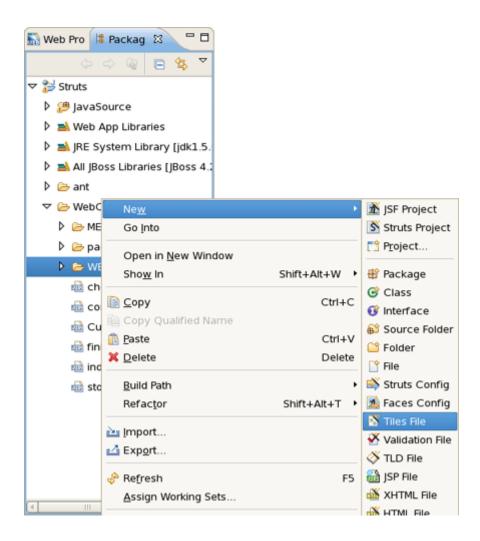


Figure 3.10. Creating a New Tiles File

3.3.2. Tree View

The Tree view represents the different elements of the Tiles file organized into functional categories on the left-hand side and a form for editing the properties of currently selected items on the right-hand side.

tiles-defs	- Tiles Confi	g Description		
 ■ tiles-defs.xml ▼ ■ tiles.layout ■ title 		les-defs		
action	- Definitions	1		
🖫 tiles.head	name	extends	path	<u>A</u> dd
🔚 tiles.action	tiles.layout		tiles/layout.jsp	Remove
🔚 tiles.foot	tiles.head		tiles/head.jsp	
🔚 tiles.error	tiles.action		tiles/action.jsp	<u>E</u> dit
🔚 tiles.layout.errors	tiles.foot		tiles/foot.jsp	<u>Ц</u> р
	tiles.error		tiles/error.jsp	Down
	tiles.layout.e	rror tiles.layout		Down

Figure 3.11. Tree View

To edit the file, simply right click any node and select from the available actions.

🕒 tiles-defs.xml 😫					٥
Tiles Editor					
		▼ Basic		F	*
▽ 🗟 tiles-defs.xml		Name:	tiles.layout		
 tiles.layout title 	🖹 Add P	`ut		•	=
action	{NAdd P	utList	tiles/layout.jsp	<u>C</u> hange	
🔚 tiles.head	Renar	me		Browse	
tiles.action	[Сору			<u>C</u> hange	
tiles.error	👘 Paste				
🔚 tiles.layout.	X Delet	e Delete			
	Prope	rties			
		Description:		<u>C</u> hange	
		Role:		•	
(4		Small-Icon:			•
Tree Diagram Source					

Figure 3.12. Editing in Tiles Editor

3.3.3. Diagram View

The Diagram view allows you to create complex Tiles files in the form of a diagram.

🔚 tiles-defs.xml 🕱	- 8
tiles.layout tiles.layout.errors	
tiles.error	
tiles.foot	
tiles.action	
tiles.head	
Tree Diagram Source	

Figure 3.13. Diagram Mode

To create new definitions, simply right click anywhere in the diagram.

tiles-defs.xml 🕴			- 8
tiles.layout	layout.errors		
tiles.error			
tiles.foot	New Definition.		
	of Cut	Ctrl + X	
tiles.action	🕞 Сору	Ctrl + C	
	👘 Paste	Ctrl + V	
tiles, head	🗱 Delete	Delete	
	Preferences		
	Input Methods	•	
Tree Diagram Source			

Figure 3.14. Creating New Definition

You can also use the Diagram toolbar to make editing easier.



Figure 3.15. Diagram Toolbar

It contains four icons for changing the cursor state. The first one is the default cursor state for selecting existing nodes. The second icon is marquee selector. The third is used for creating new connections and the last one is for adding definition template to the content.

3.3.4. Source

The other view of the Tiles editor is the Source view that gives you full control over the source. Any changes here will immediately appear in other modes when you switch to them.

When working in Source view, you always have all following features available:

Content Assist

Open On Selection

```
- 8
🖥 tiles-defs.xml 🕴
  <?xml version="1.0"?>
  <!DOCTYPE tiles-definitions PUBLIC *-//Apache Software Foundation//DTD Tiles Conf
                                      http://jakarta.apache.org/struts/dtds/tiles-c
 <tiles-definitions>
   <definition name="tiles.layout" path="tiles/layout.jsp">
   <put name="title"/>
   <put name="action"/>
   </definition>
   <definition name="tiles.head" path="tiles/head.jsp"/>
   <definition name="tiles.action" path="tiles/action.jsp"/>
   <definition name="tiles.foot" path="tiles/foot.jsp"/>
   <definition name="tiles.error" path="tiles/error.jsp"/>
   <definition extends="tiles.layout" name="tiles.layout.errors"/>
  </tiles-definitions>
                                                                                   .
Tree Diagram Source
```

Figure 3.16. Source View

Code assist is available in the Source mode.

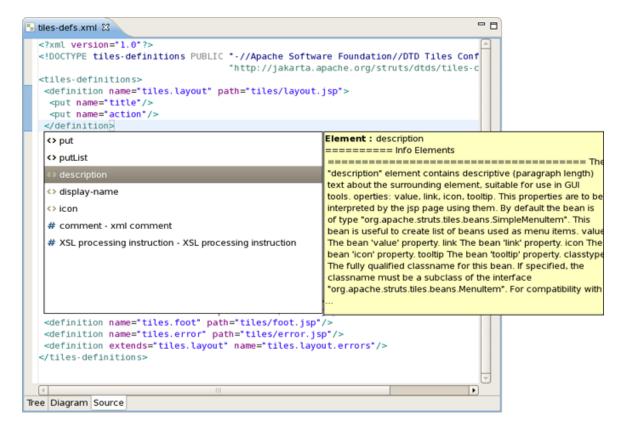


Figure 3.17. Code Assist

Any errors are immediately reported as shown below:

😫 Packag 🕴 🔝 Web Pro 🗖	🗖 🖫 tiles-defs.xml 🕱	• 6
수 수 👰 🖻 🐐	<pre></pre>	F
VEB-INF	<pre>*http://jakarta.apache.org/struts/dtds/tiles-</pre>	
🕨 😂 lib	<pre><tiles-definitions> <definition name="tiles.layout" path="tiles/layout.jsp"></definition></tiles-definitions></pre>	c
📾 carDetail.jsp	<put name="title"></put>	
🗴 faces-config.xml	<pre><put name="action"></put> </pre>	
🗇 struts-bean.tid	<sometag></sometag>	
🗴 struts-config.xml	<pre></pre> definition name="tiles.head" path="tiles/head.jsp"/>	
🗇 struts-html.tld	<pre><definition name="tiles.foot" path="tiles/foot.jsp"></definition></pre>	
🗇 struts-logic.tld	<pre><definition name="tiles.error" path="tiles/error.jsp"></definition> <definition extends="tiles.layout" name="tiles.layout.errors"></definition></pre>	
🐼 tiles-defs.xml		
📾 web.xml	<pre>0 </pre>	-

Figure 3.18. Errors Reporting

You can also use the Outline view together with the editor's Source mode. It provides a way to more easily navigate through the file.



Figure 3.19. Outline View

3.4. Graphical Editor for Struts Validation Files

JBoss Tools comes with a visual validation editor that provides full support for the development of Struts applications. To open the editor double-click on the validation file, or create a new one if it doesn't exist.

To create a new validation file, right click any folder in Project Explorer and select **File** \rightarrow **New** \rightarrow **Other...** from the context menu and then select **JBoss Tools Web** \rightarrow **Struts** \rightarrow **Validation File**.

C New	×
Select a wizard	
<u>W</u> izards:	
type filter text	
🗢 JBoss Tools Web	•
🚳 CSS File	
📸 HTML File	
🛸 JS File	
🛃 JSP File	
📸 Properties File	
🔆 TLD File	
🔭 Web Descriptor	
📸 XHTML File	
D 🗁 JSF	
🔊 Struts Config	
Struts Project	
🖄 Tiles File	
🗙 Validation File	
D 🗁 JPA	•
⑦ < Back Next > Enish	Cancel

Figure 3.20. Creating New Validation File

The validation editor works with five modes: Formsets, Validators, Constants and standard Tree and Source that you can easily switch between using tabs at the bottom of the editor.

The Formsets view shows forms and their elements on the left side and the dialogue for defining their validation rules on the right side.

MyValidation.xml 🕴				- 8
Formsets	Page	1		Edit
	Indexed List Prope			Edit
formset (default) Constants	Msg - Correspond	ed Message Template		
AlertWizard	name	key	resource	Add
Ø dashboardManagerForm				Edit
	4			Delete
▽ 🥥 name		nt Value for Message Te		
🥵 required	name	arg key	resource	Add
📽 mask	required	arg0 Nmae	false	Edit
👂 🥔 dashboardWizardNew				Delete
	 ∵Var - Validator Par	ameter	•	Delete
	var-name	var-value		Add
	mask	^[()A-Za-z0-9()_]+[[A-Za-z0-9']	Edit
				Delete
(III)	1			
Formsets Validators Constants T	ree Source			

Figure 3.21. Formsets View

The Constants view let you set constant values for your validation rules.

*validation.xml 🛙		- C
Current Global Sectio	n globa 🗢 🛛 🖺 🍿	
constant-name	constant-value	bb <u>A</u>
standartName	^[()A-Za-z0-9()_]+[A-Za-z0-9']	Edit
		Delete
ormsets Validators	Constants Tree Source	

Figure 3.22. Constansts View

The validation file can also be viewed in a Tree view.

MyValidation.xml 🕴	- 8
💌 🗹 MyValidation.xml	
🗢 💖 formset (default)	
AlertWizard	
🕨 🥔 dashboardManagerForm	
🗢 🥏 dashboardVeiwForm	
▽ 🧕 name	
🎲 mask	
0 arg0 for required	
🕨 🥔 dashboardWizardNew	
Formsets Validators Constants Tree Source	

Figure 3.23. Tree View

At any point you have full control over the source by switching to the Source view. Any editing in this view will immediately be available in other views of the editor.

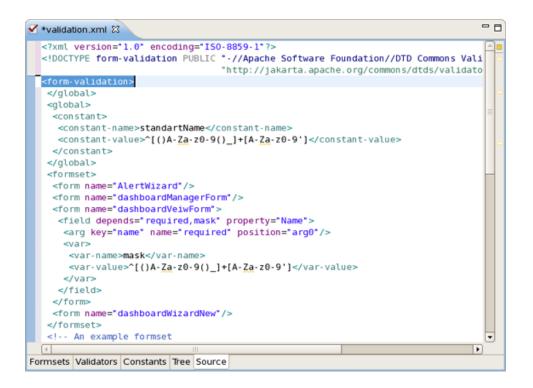


Figure 3.24. Source View

You can also open your own custom or Struts-standard validation-rules.xml file.

The Validators view shows the validation rules for a selected validator. You can of course add your own rules.

Current Global Section globa 🖨) 🗅 🍵	
Validators	Depends	Change
name	Message Key errors.timeornumber	Change
long	Java	
float	Class Name com.echopass.provisioning.rra.validator.TimeOrNumbe	Change
double	Method validateTimeOrNumber	Change
date	Method Param java.lang.Object, org.apache.commons.\	Change
intRange	JavaScript	
creditCard	Function Name	Change
email	Function Body	
url	var timeMask = "^([0-9]){1,3}(:[0-5][0-9]){0,2}\$";	ŀ
time_number	var numberMask = "^[0-9]+.?[0-9]?\$";	
time_numberforthresholds	var booleanMask = "^active inactive\$";	
	var anyMask = *^.*\$*; var percentMask = *^[0-9]+.?[0-9]?%?\$";;	
	var statistic = new Array();	
		>

Figure 3.25. Validators View

Here are the validation rules shown in the Source mode.

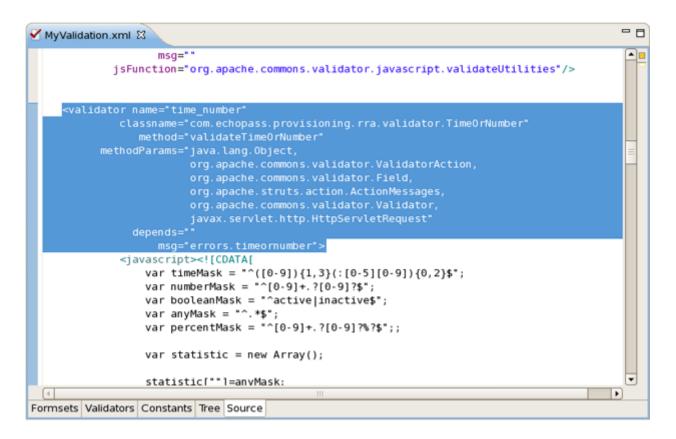


Figure 3.26. Validation Rules

Modules

JBoss Tools supports working with Struts projects that have multiple modules. You can easily do the following:

- Add new modules
- · Edit modules for an existing project or during Struts project import

Now, let's discuss this functionality in more detail.

4.1. When Importing a Struts Project

During Struts project import, if the project has multiple modules, you will see a screen with all existing modules. You can select each module and edit its details.

e	Import Struts Project	×
Project Modu	les	
Configure Proje	ct Modules	89
Name	URI	
<default></default>	/WEB-INF/struts-config.xml	
Name	<default></default>	
URI	/WEB-INF/struts-config.xml	
Path on Disk	/opt/work/jbds_GA/workspace/Struts/WebContent/WEB-INF/struts-config.xml	Browse
Sources Path	/opt/work/jbds_GA/workspace/Struts/JavaSource	Browse
Web Root	/opt/work/jbds_GA/workspace/Struts/WebContent	Browse
0	< <u>B</u> ack <u>N</u> ext > Cancel	Einish

Figure 4.1. Configuring Project Modules

4.2. Editing Modules in an Existing Project

To edit modules in an existing project, right click the project in Web Project view and select **JBoss Tools Struts** \rightarrow **Modules Configuration** (or right click the project in Package Explorer view and select **Configure** \rightarrow **Modules Configuration**).

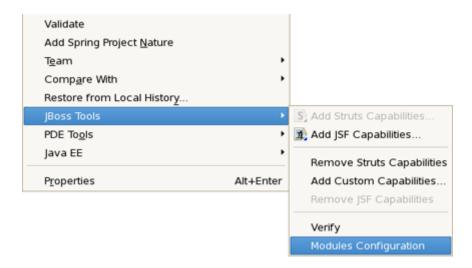


Figure 4.2. Choosing Modules Configuration

You will see the same screen as above where you will be able to select a module and edit its details.

	Modules Configuration	
truts Projec	t	
Name	URI Dele	eted Add
<default></default>	/WEB-INF/struts-config.xml	Delete
Name	<default></default>	
URI	/WEB-INF/struts-config.xml	
Path on Disk	A/workspace/Struts/WebContent/WEB-INF/struts-config	g.xml B <u>r</u> owse
Sources Path	/opt/work/jbds_GA/workspace/Struts/JavaSource	Browse
Web Root	/opt/work/jbds_GA/workspace/Struts/WebContent	Browse
	Finis	ih Cancel

Figure 4.3. Modules Configuration

4.3. Adding New Modules

Adding a new module is very simple. First switch to Web Project view. Expand your project to the Configuration folder. Under that folder you should see the current modules. Right click on Configuration and select **New** \rightarrow **Struts Config**.

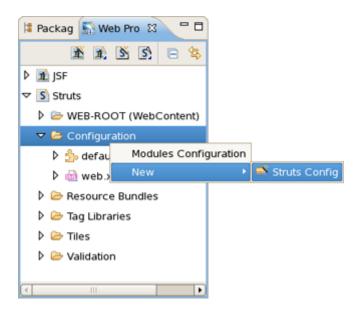


Figure 4.4. Adding New Modules

You will see the screen below. You can specify a new module name and also add the new Struts configuration file to the web.xml file.

۲	Struts Config 🛛 🗙
Struts Co	enfig 🛞
Folder:*	/Struts/WebContent/WEB-INF Browse
Name:*	struts-config-mod1
Version:	1.1 \$
Module:	/mod1
	Register in web.xml
0	Cancel Finish

Figure 4.5. Adding New Modules

Code Generation

JBoss Tools comes with a code generation feature which can be used to generate stub code for Struts Actions, FormBeans, Forwards and Exceptions.

The code generation that JBoss tooling provides is based on Velocity templates which can be modified for your use. The templates are located at {JBossStudioHome} \rightarrow templates \rightarrow codegeneration.

There are a number of ways to invoke code generation. One way is to simply right-click the Struts diagram and select **Generate Java Code...**

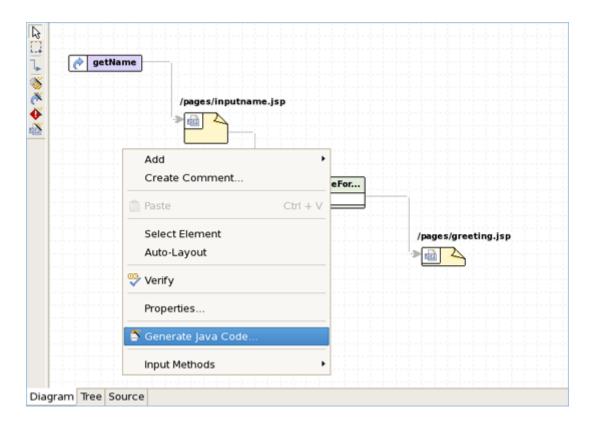
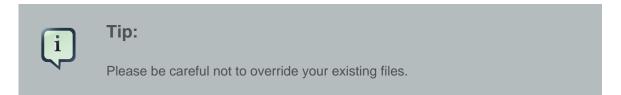


Figure 5.1. Selecting Generate Java Code

On this screen you can select for which elements to generate code. If you click the **Next** button you will be able to specify more options for each of the categories you selected.

	Generate - Step 1
Java Code Ste	p1
	Generate classes for Actions
	Generate classes for FormBeans
	Generate classes for Forwards
	Generate classes for Exceptions
	Overwrite existing files
Base Package:	
	Next >> Generate Cancel

Figure 5.2. Generate - Step 1



When generation is complete, a result window will appear letting you know how many classes were generated:

ê	Generate - Finish	×
Java Code Step 7 Generation finished.		
Message:		
Generated classes: 1 Form beans: 1		
()	III	Þ
		Finish

Figure 5.3. Generation Finished

You don't always have to generate code for all elements at once. You can generate code for an individual Struts artifact as well. Right-click an element on the diagram of the Struts configuration file and select **Generate Java Code...** from the context menu.

occimin	eFor
ayhello	Open Declaration
	Open Form-bean Source
	Oreate Exception
	💑 Create Forward
	📸 Create Property
	Create Comment
	Set as unknown
	Select Referring Item
	🍧 Generate Java Code

Figure 5.4. Generation For Individual Struts Artifact

🕏 struts-config.xml 🛙		- C
Struts Config Editor		
	• Propertie	as Editor
🗢 🛸 struts-config.xml	struts-con	fig 1.1
🛵 data 🛛 Rename		struts-config
👂 🦂 form 🛛 🖓 🖉	► on:	xml
🔯 glot 🔉 🕞 glot	va Code	
Ø Got Ø Got Ø Properties		Change
<pre>controller</pre>	Encoding:	UTF-8
a resources	Literating.	
P 🧠 plug-ins		
Diagram Tree Source		

The same can be done from within the Tree viewer for the editor of the Struts configuration file.

Figure 5.5. Generation in Struts Config Editor

Struts Configuration File Debugger

JBoss Tools come with Struts configuration file debugger. It allows you to set break points on Struts diagram and then simply launch the server in debug mode.

	pages/inputname.jsp		
getName >	Den Page		
	Create Link Create Comment Show/Hide Links Select Referring Iten	n	
	Run on Server		
	Copy Cut Paste	Ctrl + C Ctrl + X Ctrl + V	/pages/greeting.jsp
	X Delete	Delete	
	Add Breakpoint		
	Input Methods	,	

Simply right click an Action or a page and select Add Breakpoint.

Figure 6.1. Adding Breakpoint

Customizable Page Links Recognizer

Custom page links allow you to define custom Struts page links that will be recognizable in the Struts application diagram. You can define these links by selecting **Window** \rightarrow **Preferences** from the menu bar and then selecting **JBoss Tools** \rightarrow **Web** \rightarrow **Struts** \rightarrow **Customization** from the Preferences dialog box.

Preferences 🗙						
type filter text		Customizatio	n			⇔• ⇔-
General	<u>^</u>	Link Recognize	er			
Ant		Tag	Attribute	Refer to	Link Type	Add
Connectivity FreeMarker Editor		html:link	action page	action 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		forward	Struts	
		logic:redirect		forward	Struts	
JBoss Servers		logic.realiect	Torward	Iorward	50005	
Packaging Archives						
⊽ Web						
Editors						
▷ JSF	Ц					
Seam						
Automation						
Customization						
Project						
Struts Pages						
Verification		4				
XDoclet	•			Restr	ore <u>D</u> efaults	Apply
(III)				rieste	ne <u>D</u> eruald	
٢					ок	Cancel

Figure 7.1. Customization Panel

Struts Project Verification

This section covers the Struts project verification functionality provided by JBoss Tools.

To configure Struts project verification select **Window** \rightarrow **Preferences** from the menu bar, select **JBoss Tools** \rightarrow **Web** \rightarrow **Verification** from the Preferences dialog box, and then expand the Struts Rules node.

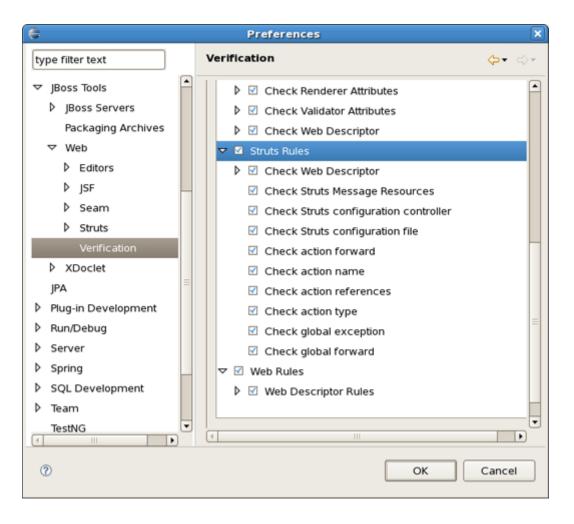


Figure 8.1. Struts Rules

Suppose you are working in the Source viewer for a Struts configuration file as shown below:



Figure 8.2. Struts Configuration File

While typing a class name or entering it from the graphical editor, you might make a minor typo (like *"sample.GreetingAction1"* instead of *"sample.GreetingAction"*). After saving the file, it is verified to make sure everything is correct and finds the error shown below:



Figure 8.3. Error Reporting

Notice that the Package Explorer View shows a marked folder and a marked file to indicate where the error is.

You can place the cursor over the line with the error to view a detailed error message:



Figure 8.4. Error Message

The verification process also checks to make sure you have specified the correct JSP page for the forward:



Figure 8.5. JSP Page Verification

Once you place the cursor over the line, you can see the error message:

	<pre></pre>
--	-------------

Figure 8.6. Error Message

You can always invoke the verification process by switching to the Diagram viewer, right-clicking and selecting **Verify** from the context menu:

🕏 *struts-config.xml 🛛		- 0
↓ ↓ ↓ </th <th>/pages/inputname.jsp</th> <th></th>	/pages/inputname.jsp	
	Add Create Comment Paste Ctrl + Select Element Auto-Layout	ages/greeting.jsp
	Properties	
Diagram Tree Source	Senerate Java Code	

Figure 8.7. Verify Command

Relevant Resources Links

More information on Struts can be found on the Struts website [http://struts.apache.org/].

This reference has been desogned to help you become familiar with those parts of JBoss Tools which assist in developing using Struts technology. It covers how to create and import Struts projects, enable Struts capabilities for an existing web project, as well as organize and edit all necessary the necessary files in your Struts application using the views and editors provided by JBoss Tools.

Feedback is always appreciated. You can leave your questions and suggestions on our *Forum* [http://www.jboss.com/index.html?module=bb&op=viewforum&f=201].