JBoss Server Manager Reference Guide

Version: 2.0.0.GA

Copyright © 2007 Red Hat

Table of Contents

1. Runtimes and Servers in the JBoss AS plugin	1
1.1. WTP Runtimes	1
1.1.1. Installing a new Runtime	1
1.2. WTP Servers	5
1.2.1. Creating a New Server	5
2. JBoss AS Perspective	9
2.1. The JBoss Servers View	9
2.2. Project archives View	19
2.2.1. Creating an Archive	20
2.2.1.1. Creating a Folder	21
2.2.1.2. Creating a FileSet	21
2.2.2. View Actions	22
2.2.3. Publish to Server	22
2.3. Deploy to Server	23
3. WTP Projects	25
3.1. Description	25
3.2. Faceted Project Wizards	25
4. Deploying Modules	29
4.1. WTP Projects	29
4.1.1. Run On Server	29
4.1.2. Add and Remove Projects	32
4.1.3. JBoss Servers View, Publish	32
4.1.4. Project Archives View	33

1

Runtimes and Servers in the JBoss AS plugin

In this chapter we will discuss how to install runtimes and servers

The JBoss AS plugin makes use of WTP. This includes starting and stopping servers in run or debug mode. It also includes targeting WTP projects, such as dynamic web projects, to certain server runtimes in order to ensure that the proper jars from a specific server are added to the project's classpath properly.

In order to get started creating, running, and debugging J2EE applications, we must start with creating our runtime and server instances.

1.1. WTP Runtimes

In JBoss Tools, the main purpose of Server Runtimes is to point to a server installation somewhere on disk. In our case, this will be a JBoss installation, and it can than be used for two primary purposes:

- it provides classpath additions to WTP projects that require them.
- For JBoss server at least, it provides information necessary for the starting and stopping of the server, it tells which jars to run and which configuration to use.

1.1.1. Installing a new Runtime

You can install runtimes into eclipse from the *Window* > *Preferences*... menu, and then select *Server* > *Installed Runtimes* from the categories available.

•			Preferences		×
ty	pe filter text		Installed Server Runt	⇔ ∙ ⇔∗	
⊳	General	<u> </u>	Add, remove, or edit insta	alled server runtime defir	nitions.
⊳	Ant		Installed server runtimes:		
⊳	Connectivity		Name	Type	Add
	FreeMarker Editor		JBoss 4 2 Buntime	IBoss 4 2 Buntime	
⊳	Help		JBoss 4 2 Runtime 1	Boss 4.2 Runtime	Edit
	HQL editor		, jooss 4.2 Raname 1	J0000 4.2 Runame	Remove
⊳	Install/Update				
⊳	Internet				<u>S</u> earch
⊳	Java				
⊳	JBoss jBPM	=			
⊳	JBoss Tools				
	JPA				
⊳	Mylyn				
⊳	Plug-in Development				
⊳	Run/Debug				
∇	Server				
	Audio				
	Installed Runtimes				
	Launching				
⊳	SQL Development				
⊳	Team				
	Validation	•			
4					
G	D			ОК	Cancel

Figure 1.1. Installed Runtimes

From this preference page you can see what runtimes are declared, and what type they are. In the image shown above, there are two declared runtimes, including a JBoss 4.2 instance.

To create a JBoss runtime, we begin by clicking the *Add* button. This will open another dialog that allows us to choose what type of runtime we want to create. Most of the runtime options are provided by WTP, but those

provided by JBoss Tools are the ones we will focus on.

E New Server Runtime
New Server Runtime
Define a new installed server runtime environment
Runtimes are used at build time to compile projects. Download additional server adapters
Select the <u>r</u> untime type:
type filter text
✓ ➢ JBoss, a division of Red Hat ↓ JBoss 3.2 Runtime
JBoss 4.0 Runtime
🤪 JBoss 4.2 Runtime
🤪 JBoss Deploy-only
🕨 🗁 Macromedia
JBoss Application Server 4.2
Image: Second

Figure 1.2. Adding a Runtime

As seen above, there appear to be two JBoss categories. The first is contributed by WTP, and is a generic adapter that is not upkept very well. For this reason, JBoss Tools provides updated and supported adapters of our own. There is one for each of JBoss 3.2, 4.0, and 4.2. You'll also note a deploy-only runtime type. This type provides no classpath for WTP projects. It is used solely by it's server type for the purpose of setting up a deploy directory for users who don't wish to make use of starting, stopping, or debugging their projects inside eclipse.

🖨 New Server Runtime 🗙
JBoss Runtime Wizard
A JBoss Server runtime references a JBoss installation directory. It can be used to set up classpaths for projects which depend on this runtime, as well as by a "server" which will be able to start and stop instances of JBoss.
Name
JBoss 4.2 Runtime 2
Home Directory
/opt/work/jboss-4.2.0.GA Browse
JRE
jdk1.5.0_12 \$ JRE
Configuration
🖏 minimal
🦉 default
Image: Section of the section of t

Figure 1.3. Adding a JBoss 4.2 Runtime

As shown above, all you need to do to create the runtime is to name it, browse to it's install directory, select a Java Runtime Environment, and select which configuration you want. As you browse to a valid installation folder, the list of configurations will update allowing you to select the configuration of your choice.

Once the runtime is created, the configuration is an unchanging property of that runtime. This is because many of the jars necessary to provide for classpaths, such as the ejb3 jar locations or the servlet jar locations, are located in deploy directories of each configurations (depending on which version of JBoss is being used). Because of this, to compile against a different configuration's jars, you will need to create a new runtime from that configuration.

Also, because of the open-source nature of JBoss, it is likely that a user may want to modify and repackage some of the configuration-specific jboss jars and create their own configuration using those modified jars. Rather than forcing the user to copy his entire JBoss installation, this structure allows them to create only a new configuration instead.

As a result of having each runtime represent a specific configuration rather than the server installation as a whole, it is very likely you'll create several different runtimes to test each of your configurations. It becomes important to ensure your runtimes, and later your servers, are given descriptive names that help you remember which is which. It will do no good to try to remember if "JBoss-runtime 5" is the 4.0 install with ejb3? Or the 4.2 install's custom configuration you decided to create.

After pressing finish, you'll see that your new runtime has been added to the list and can now be targeted by WTP type projects or servers, both of which we'll get to later.

1.2. WTP Servers

WTP servers are eclipse-representations of a backing server installation. They are used to start or stop servers, deploy to servers, or debug code that will run on the server. They keep track of what modules (jars, wars, etc) you deploy to the server and also allow you to undeploy those modules.

Servers can be started or stopped with different command-line arguments. They are often backed by a runtime object representing that server's location.

1.2.1. Creating a New Server

There are many ways to get to the new server wizard. One way is to use the old standard *File -> New -> Other...* wizard, and type in *Server*. This should show the screen below, which does not look that different from the initial screen when creating a new runtime.

e	New Server
Define a New Server	
Choose the type of server to create	
Server's <u>h</u> ost name: localhost	•
	Download additional server adapters
Select the server type:	
type filter text	
Boss	
🗢 🗁 JBoss, a division of Red Hat	
JBoss AS 3.2	
🤪 JBoss AS 4.0	
💝 JBoss AS 4.2	
🤪 JBoss System Copy Server	
👂 🗁 Macromedia	
A CHINADAN	•
JBoss Application Server 4.2	
Server runtime: IBoss 4.2 Runtime	Installed Buntimes
Boss 4.2 Runtime	1
0	< Back Next > Einish Cancel

Figure 1.4. Adding a JBoss Server

Because the server object is what keeps track of things like command line arguments when starting or stopping, and runtimes keep track of the location of the installation, each server instance must be backed by an appropriate runtime.

Because there may be many runtimes of each type declared, the wizard allows you to select which runtime you want your server to be backed by. The combo box below the view lets you select which declared runtime to use. For example, if there were already multiple JBoss 4.2 runtimes declared, the combo box would list all of the 4.2 runtimes available.

If none of the runtimes declared are one you want to use, for example if you declared a default and a minimal runtime before but now want your server to be backed by the ALL configuration, then you can click on the *Installed Runtimes...* button to bring up the preference page shown at the beginning of this chapter.

0	Preferences (Filte	red)	×				
type filter text	Installed Server Runt	ime Environments	<i>⇔</i> ≠ <i>⇔</i> ≠				
▼ Server	Add, remove, or edit inst	alled server runtime defir	itions.				
Installed Runtimes	Installed server runtimes:						
	Name	Туре	<u>A</u> dd				
	🤪 JBoss 4.2 Runtime	JBoss 4.2 Runtime	Edit				
	🤪 JBoss 4.2 Runtime 1	JBoss 4.2 Runtime					
			Remove				
			Search				
0		ОК	Cancel				

Figure 1.5. Installed Server Runtime Environments

If the server you want to create doesn't have any installed runtime yet, the combo box and button will disappear, and the next page in the wizard will force you to create the associated runtime first.

Either way, after targeting your server to a runtime, the final screen in this wizard is largely confirmational, giving the user a chance to verify that he's selected the appropriate runtime. It also allows the user to name the server appropriately.

2

JBoss AS Perspective

This chapter tells how to manage installed JBoss AS servers via JBoss AS Perspective

The JBoss AS Perspective is similar to the Java perspective, but it contains a few additional views. Two of the additional views are standard views, specifically the Console view and the Properties view. The other two views that are added are the Project archives view and the JBoss Servers View.

2.1. The JBoss Servers View

This chapter will focus on the the JBoss Server's View. The JBoss Servers View is based on the Webtool's view, Servers View. The top half of the JBoss Servers View essentially embeds the original Servers View directly into it, making slight changes to the context menu. A second half was added to provide additional information about the server selected in the top half. In the image provided, categories in the second half include which modules are currently deployed.



Figure 2.1. The JBoss Servers View

In order to access the view's preferences, you should access Window > Preferences > JBoss Tools > JBoss Servers

> *View*. This preference page allows you to select which view extensions you want on or off, the order they appear in the view, as well as any other extension-specific preferences that may be available.

۲				Pre	ferences	×
ty	ype filter text	View				⇔ - ⇒-
Þ	General	4				
Þ	Ant			1		
Þ	Connectivity		Name	Enabled	Description	
	FreeMarker Editor		Modules		Allow modules to be included in the tree.	
Þ	Help	Move U	p i Event Log		A place to view server-related events	
	HQL editor	Move Do	XML Configuration		An extension that lets you quickly change often-used fields or ports	
Þ	Install/Update	MOVE DO	🛄 🗟 File Filters		A front end to open, edit, or view files in your server's directories.	
Þ	Internet		lnactive Categories	5 🗹	Lists inactive categories and allows you to turn them back on.	
Þ	Java		뷆 JMX Console		A clone of the JMX view in the web browser	
Þ	Boss jBPM					
~	Boss Tools					
	✓ JBoss Servers	Below, yo	u may set or clear any pref	ferences spe	ecific to each view extension above.	
	View	Some ext	ensions may have no prefe	erences to se	et.	
	Packaging Archive					
	Þ Web					
	XDoclet					
	IPA					
Þ	Mvlvn					
Þ	Plug-in Development					
Þ	Run/Debug					
Þ	Server					
Þ	SOL Development					
Þ	Team					
ľ	Validation					
Þ	Web and XMI					
4		•			Restore <u>D</u> efaults	Apply
	0				ОК	Cancel

Figure 2.2. View Preferences

Extender is meant to provide additional functionality relevant to the server selected in the top half of the view. If a standard server element is selected from above, some of the extensions may still provide the additional information. Others may not. So, let's look at the currently available extensions to the JBoss Server's View.



Figure 2.3. View Extensions

The Modules section shows what modules are currently deployed to the server, and allows you to remove them from the server, or force a full republish upon them. It only shows which modules have been deployed through Eclipse, not any and all modules that happen to be in the deploy directory.



Figure 2.4. Modules Action

The Event Log will show relevant information to your server's startup, shutdown, and publish processes. This allows you to keep an eye on what's going on (such as automatic incremental deployment if you have it enabled). The only action available is to clear the event log. However if the properties view is opened, you can receive further information on each event log item (when available).

🤪 JBoss Server View Σ	3 🚜 Se	rvers	٢	Con	sole	•	- 0
			莎	0	ø		ĘD
Server	State		Stat	tus			
🤯 JBoss 4.2 Server	🖡 Started		Syr	chro	onize	d	
🗢 i Event Log		Clear	r Ev	ent I	oa		
🗢 🗒 Starting the Se	rver						
🖏 Server is st	ill starting	Refre	esh l	tem			
🖏 Server is st	ill starting	Disat	ble (Cate	gory		
🖡 Server has	completed	l startı	up				
🚡 startup suc	ceeded						
뉮 Publishing to s	erver						•
							Þ

Figure 2.5. Event Log Actions



Figure 2.6. Stopping the Server

The XML Configuration category allows you to quickly browse to descriptor files in your server's deploy directory

and check or change the values. Its use requires the Properties view. Basically, XML Configuration are XML XPaths where a xpath is a path used to access some specific part of an xml document.



Figure 2.7. XML Configuration and Properties View

The view itself contains only a list of categories. By right-clicking on *XML Configuration*, you can create a new category. Ports are provided by default and is filled with many of the most commonly used ports in the JBoss Server.

🖓 JBoss Server View 🕅	😽 Server	s 📮 Console	- 0
		参 🖸 🖉	= £
Server	Status	Status	
🤯 JBoss 4.2 Server	🚡 Stopped	Synchronized	
Modules			
i Event Log			
🗢 📓 XML Configuration		10104	
🗁 Ports	New Categ	jory	
🗟 File Filters	Refresh Ite	m	
🗟 Inactive Categorie	Disable Ca	ategory	
👂 🏭 JMX Console			

Figure 2.8. Adding New Category

By right-clicking on *Ports*, you can create a new XPaths.



Figure 2.9. Adding New XPath

After that, the dialog shown below will appear.

e	New XPath	×						
An XPath is a way to find a specific XML element inside an xml file. This dialog will help you create one. These XPaths' values can then be modified by using the JBoss Servers View with the Properties View.								
Name:	test] [
XPath Pattern:	/server/mbean]						
Attribute Name:	/server/mbean[@code=	_						
Location	/server/mbean[@name=							

Figure 2.10. Adding New XPath

As you type, the fields autocomplete to help you locate exactly what xpath you're looking for. The goal here is to get an end result where the xpath matches up with an easily changed property. With that in mind, if the property you want to change is the text of an element, then the final field Attribute Name should be left blank and your xpath should end with port. If, on the other hand, your desired field is the port attribute of <fieldName port="35">, then your xpath will end with fieldName and your Attribute Name will be "port". When finished, you can click *Preview* to see how many matches you have for that particular xpath, as shown below.

e	New XPath	×
An XPath is a way This dialog will he by using the JBoss	to find a specific XML element inside an xml file. Ip you create one. These XPaths' values can then be modified Servers View with the Properties View.	
Name:	test	
XPath Pattern:	/server/mbean[@name='jboss.ejb:service=EJBDeployer']/xmbean/description	
Attribute Name:		
Location	Value	
	OK Cancel Preview	

Figure 2.11. XPath Preview

As part of the JBoss Server view there is a *JMX Console* section which allows you to browse and use the JMX exposed beans on the server.

Properties 🛿			⊽ ⊡		🖹 Probl	le 🛛 🔕 Tasks	🛞 Seam C 🚱	JBoss S 🕄	🔄 Consol	e - 🗆
Attributes			•					*	0 🖉	• 🌮
for MBean jboss.deployme	ent:flavor=URL,type=Deploym	entScanne	r (15 operations, 10 attributes)		Server		State	Status		
Name Ty	pe	Acces	Value	A	e 🤪 (oss Applicatio	n 🥉 Started	Synchronize	d	
URLCompar jav	va.lang.String	RW	org.jboss.deploymen							
Filter jav	va.lang.String	RW	org.jboss.deploymen							
StateString jav	va.lang.String	R	Started							
State int	t	R	3	=	۸ز 👑 🔻	IX Console				-
StopTimeOu lor	ng	RW	60000		Þ	jboss.consol	e			=
ScanEnable bo	olean	RW	true			jboss.deploy	ment			
URLList jav	va.util.List	RW	[file:/opt/work/rhds_re		~	flavor=UR	L.type=Deployn	nentScanner		
RecursiveSe bo	olean	RW	true	HI		 Attribut 	es			
Name jav	va.lang.String	R	URLDeploymentScan	•		suspen	dDeployment			
4	11		Þ			start				-

Figure 2.12. JMX Console

Also you can see the server's publish status in the JBoss Servers view that allows you to see if changes are awaiting (Republish), are being updated (Publishing...) or are in-sync (Synchronized).

🖹 Proble	🙆 Tasks	🕖 Seam	😲 JBoss	x	🗐 (Conso	ا ا	- 0
				参	0	Þ		₿D
Server		State	SI	tatus				
🤯 JBos	s 4.2 Serv	er 🖡 Starl	ted R	epublish	n			

Figure 2.13. Server's Publish Status

The Server section of the view lists all declared servers and their current states, that is, whether they are started or stopped. By Double-clicking on each server, an editor window will appear allowing you to edit parts of that server. The editor will also link you to modify that server's launch configuration, set command line arguments, and other things that are relevant to launching the server.

A username/password is available in the UI when configuring the server. If you get an SecurityException when trying to launch the server it is most likely because your server is protected and hence you need to fill the username/ password fields with appropriate values.

Jeessrappieda	on Server 4.2 🕄		,
0verview			-
General Infor Specify the hos Server name: Host name: Runtime: Open launch	mation t name and other common settings. JBoss Application Server 4.2 localhost <u>¢ Edit</u> configuration	Automatic Publishing	
- Deploymen	t		
The temporary be copied to be This folder sho as the deploy f Deploy Directo	deployment location is where files will afore being renamed into the deploy directory. uld be on the same file system older.		
The temporary be copied to be This folder sho as the deploy f Deploy Directo Temporary Dep	deployment location is where files will efore being renamed into the deploy directory. uld be on the same file system older. TY /opt/work/rhds_rc1/jboss-ea Browse Noy Directory /opt/work/rhds_r Browse		
The temporary be copied to be This folder sho as the deploy f Deploy Directo Temporary Dep	t deployment location is where files will efore being renamed into the deploy directory. uld be on the same file system older. TY /opt/work/rhds_rc1/jboss-ea Doloy Directory /opt/work/rhds_r Browse Browse Browse		
The temporary be copied to be This folder sho as the deploy for Deploy Directo Temporary Dep - Login Cred Set the login ar This will ensure User Name Password	e deployment location is where files will efore being renamed into the deploy directory. uid be on the same file system older. TY /opt/work/rhds_rc1/jboss-ea Browse bloy Directory /opt/work/rhds_r Browse entials ad password for your server. a it starts and stops properly.		
The temporary be copied to be This folder sho as the deploy f Deploy Directo Temporary Dep - Login Cred Set the login and This will ensure User Name Password	e deployment location is where files will efore being renamed into the deploy directory. uld be on the same file system older. ry /opt/work/rhds_rc1/jboss-ea bloy Directory /opt/work/rhds_r Browse entials ad password for your server. e it starts and stops properly.		

Figure 2.14.

The view's toolbar shown below provides quick access to starting a server (in debug mode, run mode, or profile mode), restarting the server, stopping the server, or publishing to the server.



Figure 2.15. The JBoss Server View Toolbar

In order to debug your applications or EJB's that are deployed to the server, you must start the server in debug mode. By starting the server in debug mode, eclipse will allow you to set breakpoints on code in your workspace and step through the code.

The publish icon, on the extreme right, will republish any modules where it has determined the workspace is out of sync with the server. It will attempt to do an incremental publish if it turns out that the module in question is capable of doing one.

Because the JBoss servers have few properties to edit in the editor, a shortcut to the launch configuration has been

provided in the context menu when right-clicking on a server. This menu also provides the functionality of starting the server in run or debug mode, stopping it, or synching the publish information between the server and work-space.

🤪 JBoss Server Vie	w 🖾 🖳 Console	莎	0	${\rm I}^{\rm e}$	ED	- 0
Server	State Status					
🤯 JBoss Applica	🕆 New Server					
😲 JBoss 4.2 Ser	💢 Delete	ed				
	🚺 Start					
	棼 Debug					
	🔳 Stop					
👂 🦲 Modules	🗐 Publish					
i Event Log	Clean					
XML Configur File Filters Inactive Cate	Twiddle Server Edit Launch Configuration					
JMX Console	add and Kernove Projects	·				

Figure 2.16. Launch Configuration

Near the bottom you'll also see the Add and Remove Projects... item, which allows you to publish a new project to the server (if its type is supported). The last menu item is the one labeled Twiddle Server.

The Twiddle Server item will open a dialog and allow you to run Twiddle commands against it. Twiddle is a JMX library that comes with JBoss, and you can use it to access any variables that exposed via the JBoss JMX interfaces.

2.2. Project archives View

Every application, whether plain old Java, J2EE, or some other language altogether, needs to be packaged in some way. In Java-related projects, many people use ANT. JBoss Tools comes with our own archives tool with a bit easier and less-verbose XML and a handy user interface.

The Project Archives plugin consists primarily of a view to set up each packaging configuration. Each project can enable or disable its builder, or depend on the global setting.

The packaging configuration for each project is stored in that project's root folder, and is in a file named .packages, which has a fairly simple XML structure. Modifying the file by hand is neither required nor recommended, and using the UI is the official way of modifying your packaging structure.

Aside from the builder, the other preferences for the plugin are mostly cosmetic, allowing you to show full or truncated paths, show the project at the root, etc. None of these has any effect on the functionality of the packaging plugin.

2.2.1. Creating an Archive

When creating a new archive, you have some different options at your disposal. If the project has no .packages file, your options will be presented to you all at once for you to choose from (as above). Otherwise you will right-click inside the view and select *New Archive* to see your archive type options.

C Project archives 🛿		6	- 8
New Archive 🔸	G WAR G EJB JAR EAR M JAR		

Figure 2.17. Create an Archive

JAR is the standard archive type, and does very little configuration, leaving most of the work up to you. You can customize the name, add folders, filesets, and inner jars to it.

The other types, for the most part, simply start off with a default setting, usually the jar with some specific children based on an expected structure of the project. For example, if the project is a dynamic web project, and you create a WAR archive, the archive will be created with a few filesets relevant to the known structure of the project.

Because the first page of all new archive wizards are the same, and it is also the only page in the New Jar Wizard, that page is shown below.

New JAR X
Create a new archive Create a new archive consisting of filesets in the workspace.
Archive information Archive name: SomeProject.jar Destination: SomeProject Workspace Archive type Standard archive using zip compression Standard archive resulting in a folder (no compression)
⑦ <u>Einish</u> Cancel

Figure 2.18. New JAR Wizard

The page is pretty simple. All it asks is for the name of your new archive, a destination, which we'll get to in a moment, and whether the archive is exploded or packaged up.

The destination of an archive can be anywhere on the filesystem, anywhere in the workspace, inside some other archive, or inside a folder declared inside an archive. You can browse to workspace or filesystem destinations by clicking on their respective buttons. To select a destination inside some other archive, you'll need to press the Workspace... button. At the bottom of the list, you'll see archives that have been declared in the workspace.

2.2.1.1. Creating a Folder

Creating a folder is much easier. You simply right-click on an archive or folder you want your new folder to be a child under. The only piece of required information is naming the file.

2.2.1.2. Creating a FileSet

To create a new fileset, you click on an available target location such as an archive, a nested archive, or a folder within an archive, and select New Fileset. The New Fileset Wizard requires a destination (where the files will go), and a root directory (or where the files are coming from). The source can be anywhere in the workspace or from the filesystem at large.

Below that, the fileset requires only an includes pattern and an excludes pattern. As you type in either of these fields, the preview viewer should update itself with which files are matched.

2.2.2. View Actions



Figure 2.19. Context Menu on the Item

The context menu on the items in the view is extendable, but there are several that come standard. The first is the Build Archive action, enabled only on top-level archives, which initiates a full build on that archive. Editing and deleting nodes are also standard actions, with deletion not needing an explanation. The edit action brings up the wizard associated with that particular node type and allows the details to be changed. The final action contribution here is the ability to publish to a declared server.

2.2.3. Publish to Server

e		×
Publish a	rchive to a server	
Select the	server to publish the archive to.	
	JBoss 4.2 Server	
	Always publish to these servers	
	Auto-deploy to selected servers after builds	
0	Einish	Cancel

Figure 2.20. Context Menu on the Item

The dialog above appears after selecting Publish To Server. To simply publish once, you just select the server(s) that you want, and finish. If you want the Publish to Server action on that particular Archive to always publish to that set of servers, then check the appropriate checkbox. And finally, to enable automatic publishing upon build events, check the last checkbox.

The automatic publishing feature is nice if, for example, your package's destination (where it is built) is a temporary folder and you want the archive published to several servers. If you only really want your archive published to one server, it might be easier to have the archive's destination folder be the deploy folder of the server.

2.3. Deploy to Server

In the context menu of files there is a Deploy To Server option that allows a single file deployment. To deploy these non-WTP files/projects right click on the file (-ds.xml, .ear, .jar etc.) and select *Deploy To server* and it will be automatically deployed.



Figure 2.21. Deploy to Sever

The deployed files are listed side-by-side with other modules that are deployed to the server.

3

WTP Projects

This chapter covers how to select installed runtime/server in your web project

3.1. Description

WTP provides what are called "faceted" projects. Their most popular of these projects are their J2EE projects, such as their Dynamic Web Project, their EJB Project, or their EAR project. Web projects of JBoss Tools are Stuts, JSF and Seam projects.

The idea behind faceted projects is that each project can accept units of functionality, or facets, which can be added or removed by the user. Some examples of these facets are adding a webdoclet facet to a web project, or an ejbdoclet to an EJB Project.

Most often, these "facets" either add to the project's classpath, enable a builder, or watch the project in some other fashion.

WTP projects have undergone some criticism as being *over-engineered* or too restrictive in their design. WTP projects are set up in a tree-relationship to each other, where one project can be a child of another. For example, an EAR project may have a Web Project child, an EJB project child, or other types.

The benefit of this is that the structure of your projects is then known, and packaging it up *should* be trivial. However, if your project is non-standard, or you feel too confined by such rigid structural requirements, you can still choose to package your project using the Archives plugin

3.2. Faceted Project Wizards

To create a new Dynamic Web Project select File > New > Other... then Web > Dynamic Web Project

0	New
Select a wizard	
Create a Dynamic Web project	
<u>W</u> izards:	
type filter text	
User Assistance	
🗢 🗁 Web	
🚰 CSS	
📑 Dynamic Web Project	
5 HTML	
😭 JavaScript	
jsp	
💕 Servlet	
Ctatic Wab Draiget	
⑦ < <u>B</u> ack	Next > Enish Cancel

Figure 3.1. New Dynamic Web Project

Click Next and you will see Dynamic Web Project page

New Dynamic Web Project X
Dynamic Web Project
Create a standalone Dynamic Web project or add it to a new or existing Enterprise Application.
Project name: SomeWebProject
Project contents:
Use <u>d</u> efault
Directory: /opt/work/workspace_BTools/SomeWebProject Browse
Target Runtime
JBoss 4.2 Runtime ↓ New
Configurations
Default Configuration for JBoss 4.2 Runtime
A good starting for working with JBoss 4.2 Runtime runtime. Additional facets can later be installed to add new functionality to the project.
EAR Membership
Add project to an EAR
EAR Project Name: EAR
⑦ < Back Next > Finish Cancel

Figure 3.2. Faceted Project Wizard: First Page

The first page of most WTP projects allows you to target a specific runtime, representing a server's library location. It will also provide you the ability to add this project to an EAR project, and select a pre-selected default set of facets, called a configuration, rather than manually select each facet you might want.

Selecting the runtime, again, allows the project to install the proper classpaths to the project so it knows what code to compile against.

(🖨 New Dynamic Web Project 🗙								
	Project Facets Select the facets that should be enabled for this project.								
Configurations: Default Configuration for JBoss 4.2 Runtime							Delete		
	Project Facet			Version					
	🕨 🔲 🙆 Axis2 W	leb Services							
	🗹 👼 Dynami	ic Web Module		2.5				•	
	🗹 🔬 Java			6.0				•	
	🗆 🚸 Java Pe	rsistence		1.0					
	🗆 📄 JavaSer	ver Faces		1.1				•	
	🗆 📄 Seam			2.0				•	
	🗆 📄 WebDo	clet (XDoclet)		1.2.3				-	
						(<< Show	<u>R</u> untimes	
_									
	0			< <u>B</u> ack	<u>N</u> ext >	Ein	ish	Cancel	

Figure 3.3. Faceted Project Wizard: Second Page

The second page of the wizard allows you to enable or disable specific facets, as described above. Some facets may require others, and some may conflict with others, but on the whole this page allows you to add any number of facets that don't conflict with each other.

Further pages are specific to either the project type, or the facets selected.

4

Deploying Modules

In this chapter it will be described how to deploy modules onto the server.

First of all it is necessary to say that deploying to a server is mostly painless. There are several ways to do it provided by WTP, and some additional methods provided by JBoss Tools. These methods are discribed further in this chapter

4.1. WTP Projects

4.1.1. Run On Server

The first WTP method is to right-click on a WTP project, such as a dynamic web project, ejb project, or ear project and then select Run As > Run on Server. The resulting dialog allows you to select which supporting server the project can be published to.

Run On Server	×
Define a New Server	
Choose the type of server to create	
	_
How do you want to select the server?	
Ochoose an existing server	
Manually define a new server	
Select the server that you want to use:	
type filter text	
▽ 🗁 localhost	
🤯 JBoss 4.2 Server	
JBoss Application Server 4.2	
Always use this server when running this project	
⑦ < <u>Back</u> <u>Next</u> > <u>Finish</u> Ca	ancel

Figure 4.1. Define a New Server

Click Next button to see add or remove projects page where you can choose projects to configure them on server.

ŧ	Run On Server	X						
Add and Remove Project	ts configured on the ser	ver						
Move projects to the right to configure them on the server								
<u>Available projects:</u>		Configured projects:						
		🐻 JSF						
	A <u>d</u> d >	🔓 SomeWebProject						
	< <u>R</u> emove							
	Add All >>							
	<< Remove All							
⑦ < <u>B</u> ack	<u>k</u> <u>N</u> ext >	<u>Finish</u> Cancel						

Figure 4.2. Add or Remove Projects

For the JBoss AS Server Adapters, publishing using this method will force a default, best-guess, packaging configuration for your project. This best-guess does *not* publish incrementally, but instead repackages your entire project into a .war, .jar, or .ear as appropriate, and then copies that file into the proper deploy directory. For quicker smarter deployment, you will need to create archives using the *Project Archives* view and customize packaging yourself.

4.1.2. Add and Remove Projects

Another method is in either the Servers View, or the JBoss Servers View, to right click on a server and select the *Add and Remove Projects* menu item. This will bring up a dialog allowing you to either publish projects or modules to a server, or remove them from the server.



Figure 4.3. Add and Remove Projects

If the selected module is a WTP project, it will be published as in the *Run on Server* option, with a best-guess full package. If, however, the selected element is an archive from the *Project Archives View*, it will be published according to the rules of that module type.

4.1.3. JBoss Servers View, Publish

In the JBoss Server's View, in the bottom section, is a category called *Modules* which should display all currentlypublished modules on the server. By right-clicking on the desired module and selecting *Full Publish*, it will force a full rebuild of the entire module.

~	🗢 🦲 Modules								
	i Event XML C File Fil Inactiv	 Remove Full Publish Incremental Publish Refresh Item 							

Figure 4.4. Full Publish

4.1.4. Project Archives View

In the Project Archives View, you can right-click on any declared archive and select the *Publish To Server* element, as described in the JBoss AS Perspective chapter.



Figure 4.5. Publish to Server

The only way to ensure an *Incremental Build*, such as changes to one jsp, html, or .class file, is to enable the builder for that project. This is done by either changing the global preferences for the Archives View, or in enabling project-specific preferences and ensuring the builder is on.