

JBossServer

A JBoss application server

Table of contents

1 Overview.....	2
2 Design.....	2
3 Constraints.....	2
4 Attributes.....	4
5 Commands.....	5
6 Related Types.....	5

1. Overview

JBossServer: *A JBoss application server*

This type manages an instance of a JBoss Server.

A [JavaZip](#) object should be added as a child dependency to automate deployment of the jboss package distribution.

An example of managing JBoss server instances is here: [JBoss Library](#).

See: [JBoss Community](#).

2. Design

Super Type Service

Role	Concrete. (Objects can be created.)
Instance Names	Unique
Notification	false
Template Directory	<code>\${modules.dir}/JBossServer/templates</code>
Data View	Children, proximity: 1
Logger Name	JBossServer

3. Constraints

3.1. Allowed Child Dependencies

- [AppTimezone](#) 1
- DefaultAllowMultiplePackageMatches
- DefaultPackageName1
- DefaultPackageType1
- [JBBindAddr](#) 1
- [JBConfiguration](#) 1
- [JBInpPort](#) 1
- [JBJrmpRmiObjectPort](#) 1
- [JBNamingRmiPort](#) 1
- [JBossDefaultInstance](#) 1
- [JBossDocumentRoot](#) 1

- [JBossEar](#)
- [JBossExtractEar](#) 1
- [JBossHashDataSourcePasswords](#) 1
- [JBossHost](#) 1
- [JBossJavaHome](#) 1
- [JBossJavaOpts](#) 1
- [JBossPidfile](#) 1
- [JBossPortConfig](#) 1
- [JBossZip](#) 1
- [JBPooledServerBindPort](#) 1
- [JBTCajpPort](#) 1
- [JBTCHttpPort](#) 1
- [JBUIL2ServerBindPort](#) 1
- [JBWSPort](#) 1
- [JVMMMaxHeap](#) 1
- [JVMMMaxPermSize](#) 1
- [JVMMMinHeap](#) 1
- [JVMPPermSize](#) 1
- [Rdb](#)

Support datasource generation within the jboss server instance's deploy directory.

1: These types have a *Singleton* constraint. Only one instance may be added as a resource.

Restrict JBossServer resources to JBossZip platform package, JBossEar packages, arbitrary jboss settings, and Rdb objects.

3.2. Allowed Parent Dependencies

- [JavaServiceWrapper](#)

In support of windows java service capability.

- Node
- Site

Restrict JBossServer to Node, Site, and optionally a JavaServiceWrapper. for java/windows service support.

3.3. Allowed Property Values

Property	Allowed Values	Default	Enforced
deployment-basedir	• <code>\${entity.attrib</code>	• <code>\${entity.attrib</code>	false

deployment-install-root	• <code>\${user.home}/jboss</code>	• <code>\${user.home}/jboss</code>	false
-------------------------	------------------------------------	------------------------------------	-------

4. Attributes

4.1. Exported Attributes

Name	Property	Description
<code>jboss_basedir</code>	<code>deployment-basedir</code>	associate the <code>jboss_basedir</code> attribute property with the standard <code>deployment-basedir</code> property.
<code>jboss_install_root</code>	<code>deployment-install-root</code>	associate the <code>jboss_install_root</code> attribute property with the standard <code>deployment-install-root</code> property.

4.2. Defaults for Imported Attributes

Name	Default	Description
<code>defaultAllowMultiplePack</code>	true	Default is true which allows us to assign more than one ear during Change-Dependencies operation.
<code>deployDir</code>	<code>\${opts.basedir}/deploy</code>	defaults to typical deploy directory
<code>documentRoot</code>	<code>\${entity.attribute.jboss_basedir}/OT.war</code>	identify the DocumentRoot based on the integrated jboss tomcat webapp for potential deployable static content.
<code>hashDataSourcePasswords</code>	false	default to not hashing passwords
<code>java_home</code>	<code>\${env.JAVA_HOME}</code>	set a reasonable JAVA_HOME by using the user's environment.
<code>javaOpts</code>		optional JAVA_OPTS to be passed to the java runtime at startup

jboss_extract_ear	false	set ear extraction behavior to false within the jboss instance deploy directory.
jboss_host	0.0.0.0	sets the jboss bind address to all interfaces as a resonable operational default. As of JBoss 4.2, the default behavior changes which is the loopback unless specified.
jbosDefaultInstance	default	set the jboss reference instance to the "default" instance.
jbossPidFile	\${entity.attribute.jboss_basedir}/c	Default location of the JBoss process-identifier file.
jvm_maxheap	-Xmx512m	use a small maximum heap setting in order not to break modest virtual machines, etc
jvm_minheap	-Xms256m	use a small minimum heap setting.
port_config	ports-default	use the default port configuration in the jboss service bindings configuration.
timezone	US/Pacific	set the jboss timezone to US Pacific time

5. Commands

View the available commands in the [User Reference](#).

6. Related Types

The following types are defined for use with JBossServer.

6.1. AppTimezone

6.1.1. Overview

AppTimezone: *A appserver timezone*

if set, applies the `-Duser.timezone=<value>` java runtime property at Configure time.

6.1.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.1.3. Attributes

attribute containing the java runtime `user.timezone` property value

6.1.3.1. Exported Attributes

Name	Property
timezone	settingValue

6.2. JBBindAddr

6.2.1. Overview

JBBindAddr: *JBoss Bind Address*

Deprecated, see `#JBossPortConfig`

6.2.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.2.3. Attributes

6.2.3.1. Exported Attributes

Name	Property
bindAddr	settingValue

6.3. JBConfiguration

6.3.1. Overview

JBConfiguration: *The name of a JBoss Configuration*

6.3.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.3.3. Attributes

6.3.3.1. Exported Attributes

Name	Property
jboss_configuration	settingValue

6.4. JBInpPort

6.4.1. Overview

JBInpPort: *The name of a JBoss Configuration*

Deprecated, see #JBossPortConfig

6.4.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.4.3. Attributes

6.4.3.1. Exported Attributes

Name	Property
jnpPort	settingValue

6.5. JBJrmpRmiObjectPort

6.5.1. Overview

JBJrmpRmiObjectPort: *JBoss JRMP RMI Object Port*

Deprecated, see #JBossPortConfig

6.5.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.5.3. Attributes

6.5.3.1. Exported Attributes

Name	Property
jboss_jrmp_rmi_object_port	settingValue

6.6. JBNamingRmiPort

6.6.1. Overview

JBNamingRmiPort: *JBoss Naming RMI Object Port*

Deprecated, see #JBossPortConfig

6.6.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.6.3. Attributes

6.6.3.1. Exported Attributes

Name	Property
jboss_naming_rmi_port	settingValue

6.7. JBossDefaultInstance

6.7.1. Overview

JBossDefaultInstance: *configure new jboss instance from a reference instance*

The jboss instance directory, i.e. the JBossServer object's basedir, is configured from scratch based on a reference jboss instance. By default, this reference instance is the "default" instance residing under the JBOSS_HOME/server directory. At Deploy time, the Configure workflow will auto generate this server instance from the reference instance using recursive copy and filtering techniques. This assumes that the JBossServer base directory is not the same as the reference instance, otherwise this will not occur since we cannot copy a directory onto itself and in this case the server instance will be untouched. If the jboss "default" instance is not the desired reference instance, this can be overridden by this setting object. For example, this settingValue can be another instance such as "all" or "minimal".

6.7.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.7.3. Attributes

attribute setting identifying the jboss reference instance to copy from when provisioning at Configure time.

6.7.3.1. Exported Attributes

Name	Property
jbossDefaultInstance	settingValue

6.8. JBossDocumentRoot

6.8.1. Overview

JBossDocumentRoot: *JBoss Document Root*

6.8.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.8.3. Attributes

6.8.3.1. Exported Attributes

Name	Property
documentRoot	settingValue

6.9. JBossExtractEar

6.9.1. Overview

JBossExtractEar: *modify ear deployment behavior for the JBossServer to explode or not explode the ear within the jboss deploy directory*

6.9.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.9.3. Attributes

6.9.3.1. Exported Attributes

Name	Property	Description
------	----------	-------------

jboss_extract_ear	settingValue	"true" or "false" to respectively explode or not explode the ear
-------------------	--------------	--

6.10. JBossHashDataSourcePasswords

6.10.1. Overview

JBossHashDataSourcePasswords: *determines if we hash database passwords into the login_config.xml or generate clear text passwords directly into datasource file*

6.10.2. Design

Super Type
[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.10.3. Attributes

6.10.3.1. Exported Attributes

Name	Property	Description
hashDataSourcePasswords	settingValue	true/false value determining whether datasources should be hashed or not

6.11. JBossHost

6.11.1. Overview

JBossHost: *Defines which interface jboss will bind to. 0.0.0.0 will bind to all interfaces*

6.11.2. Design

Super Type
[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.11.3. Attributes

6.11.3.1. Exported Attributes

Name	Property	Description
jboss_host	settingValue	specifies interface jboss will bind to. JBoss will bind to the loopback unless specified. This configuration will override this behavior to specify all interfaces via "0.0.0.0".

6.12. JBossJavaHome

6.12.1. Overview

JBossJavaHome: *JAVA_HOME* for JBoss

Identifies the location of the java (jdk or jre) installation root typically referred to as JAVA_HOME.

6.12.2. Design

Super Type
[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.12.3. Attributes

6.12.3.1. Exported Attributes

Name	Property	Description
java_home	settingValue	attribute setting containing the fully qualified path to the java installation root (i.e. JAVA_HOME).

6.13. JBossJavaOpts

6.13.1. Overview

JBossJavaOpts: *JAVA_OPTS* for JBoss

To manage the allowed options (standard and non standard) that need to be passed to the java runtime when starting the jboss server.

6.13.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.13.3. Attributes

6.13.3.1. Exported Attributes

Name	Property	Description
javaOpts	settingValue	attribute setting containing the allowed java runtime options when starting jboss.

6.14. JBossPidfile

6.14.1. Overview

JBossPidfile: *jboss pid file location*

Used to set the location of the JBoss process identifier file used to manage JBoss on Linux/Unix systems.

6.14.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.14.3. Attributes

6.14.3.1. Exported Attributes

Name	Property
jbossPidFile	settingValue

6.15. JBossPortConfig

6.15.1. Overview

JBossPortConfig: *JBoss service bindings port configuration*

6.15.1.1. Intent

Since JBoss has many different configuration files that refer to the various ports required to run a jboss server instance, ControlTier automates the JBossServer port configuration via the [Service Binding Manager](#). This approach allows centralized management of the ports (which there are many of them) that conform to a standard port numbering convention. This unique set of ports is then indexed by a configuration name (i.e ports-default, ports-01, etc).

6.15.1.2. Discussion

For example, the "ports-default" configuration setting refers to the standard JBoss HTTP Listen port, 8080, HTTPS Listen Port, 8443, and the AJP Port, 8009, etc.

Another example, the "ports-01" configuration setting refers to JBoss HTTP Listen port, 8180, HTTPS Listen Port, 8543, and the AJP Port, 8109, etc.

Therefore, by accepting the conventions that JBoss encourages via the use of their Service Binding Manager, managing sets of ports through a single setting is achieved.

6.15.2. Design

Super Type
[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.15.3. Attributes

6.15.3.1. Exported Attributes

Name	Property	Description
port_config	settingValue	attribute containing the desired port configuration within the Service Binding Manager.

6.16. JBossSetting

6.16.1. Overview

JBossSetting: *A JBoss Setting*

6.16.1.1. Intent

Supertype to encapsulate related subtypes to support JBossServer service.

6.16.2. Design

Super Type Setting

Role	Abstract. (Objects cannot be created.)
Instance Names	Unique

6.16.3. Constraints

6.16.3.1. Allowed Parent Dependencies

- [JBossServer](#)

Restrict setting objects of this type to be referred to by a JBossServer object.

6.17. JBPoolServerBindPort

6.17.1. Overview

JBPoolServerBindPort: *JBoss Pooled Server Bind Port*

Deprecated, see #JBossPortConfig

6.17.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.17.3. Attributes**6.17.3.1. Exported Attributes**

Name	Property
jboss_pooled_server_bind_port	settingValue

6.18. JBTcAjpPort**6.18.1. Overview**

JBTcAjpPort: *Embedded Tomcat/JBoss AJP Port*

Deprecated, see #JBossPortConfig

6.18.2. Design

Super Type
[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.18.3. Attributes**6.18.3.1. Exported Attributes**

Name	Property
ajpPort	settingValue

6.19. JBTcHttpPort**6.19.1. Overview**

JBTcHttpPort: *JBoss HTTP Port*

Deprecated, see #JBossPortConfig

6.19.2. Design

Super Type
[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.19.3. Attributes

6.19.3.1. Exported Attributes

Name	Property
httpPort	settingValue

6.20. JBUIL2ServerBindPort

6.20.1. Overview

JBUIL2ServerBindPort: *JBoss UIL2 ServerBind Port*

Deprecated, see #JBossPortConfig

6.20.2. Design

Super Type
[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.20.3. Attributes

6.20.3.1. Exported Attributes

Name	Property
jboss_uil2_server_bind_port	settingValue

6.21. JBWSPort

6.21.1. Overview

JBWSPort: *JBoss WebServices Port*

Deprecated, see #JBossPortConfig

6.21.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.21.3. Attributes

6.21.3.1. Exported Attributes

Name	Property
jboss_ws_port	settingValue

6.22. JVMMaxHeap

6.22.1. Overview

JVMMaxHeap: *JVM maximum heap size*

If set, the java runtime maximum heap, i.e. -Xmx512m, will be applied at Configure time

6.22.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.22.3. Attributes

attribute containing the java runtime maximum heap parameter.

6.22.3.1. Exported Attributes

Name	Property
jvm_maxheap	settingValue

6.23. JVMMaxPermSize

6.23.1. Overview

JVMMaxPermSize: *JVM Max Perm Size memory setting*

If set, the java runtime "Max Perm Size" setting is applied, i.e. -XX:MaxPermSize=256m, at Configure time

6.23.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.23.3. Attributes

attribute containing the java runtime max perm size parameter

6.23.3.1. Exported Attributes

Name	Property
jvm_max_perm_size	settingValue

6.24. JVMMinHeap

6.24.1. Overview

JVMMinHeap: *JVM minimum heap size*

If set, the java runtime minimum heap, i.e. -Xms256m, will be applied at Configure time

6.24.2. Design

Super Type

[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.24.3. Attributes

attribute containing the java runtime mimimum heap parameter.

6.24.3.1. Exported Attributes

Name	Property
jvm_minheap	settingValue

6.25. JVMPermSize

6.25.1. Overview

JVMPermSize: *JVM Perm Size memory setting*

If set, the java runtime "Perm Size" setting is applied, i.e. -XX:PermSize=256m, at Configure time

6.25.2. Design

Super Type
[JBossSetting](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique

6.25.3. Attributes

attribute containing the java runtime permsize parameter

6.25.3.1. Exported Attributes

Name	Property
jvm_perm_size	settingValue