

JBossZip

A JBoss Distribution Package

Table of contents

1 Overview.....	2
1.1 Intent.....	2
1.2 Problem.....	2
2 Design.....	2
3 Constraints.....	2
3.1 Allowed Child Dependencies.....	2
3.2 Allowed Parent Dependencies.....	2
3.3 Allowed Property Values.....	3
4 Attributes.....	3
4.1 Exported Attributes.....	3
5 Commands.....	3
5.1 assertPackageIsInstalled.....	3
5.2 extract.....	4

1. Overview

JBossZip: A JBoss Distribution Package

1.1. Intent

To manage the installation of a JBoss Server via its zip based package distribution as part of the [JBossServer](#) deployment lifecycle.

1.2. Problem

A JBossServer deployment has to manage many dependencies. This dependency can be considered as fundamental and depending on operational procedures, the jboss server platform package can be automated thru this package module. Objects of this type can be assigned as a [JBossServer](#) package dependency in this case.

2. Design

Super Type
[PlatformZip](#)

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

3. Constraints

3.1. Allowed Child Dependencies

- Package

3.2. Allowed Parent Dependencies

- Deployment

3.3. Allowed Property Values

Property	Allowed Values	Default	Enforced
package-install-root	<ul style="list-style-type: none"> <code>\${user.home}/jb</code> 	<ul style="list-style-type: none"> <code>\${user.home}/jb</code> 	false
package-repo-url	<ul style="list-style-type: none"> <code>\${framework.pkg}</code> 	<ul style="list-style-type: none"> <code>\${framework.pkg}</code> 	false
package-filetype	<ul style="list-style-type: none"> zip 	<ul style="list-style-type: none"> zip 	false
package-arch	<ul style="list-style-type: none"> noarch 	<ul style="list-style-type: none"> noarch 	false
package-install-rank	<ul style="list-style-type: none"> 10 	<ul style="list-style-type: none"> 10 	false

4. Attributes

4.1. Exported Attributes

Name	Property	Description
jboss_version	package-version	package-version property, the version of the JBoss distribution, maps to the jboss_version attribute. This value is referenced by the referring JBossServer deployment object.
jbossPackageBase	package-base	

5. Commands

Note:

Commandline options displayed in square brackets "[]" are optional. If an option expects arguments, then angle brackets are shown after the option "<>". Any default value is shown within the brackets.

5.1. assertPackageIsInstalled

Confirm the package process is installed.

Simply check for existence of the bin/run.sh and fail if it does not exist.

Usage

```
assertPackageIsInstalled [-basedir <>] [-depot <>]
[-executable <>] [-installroot <>] [-name <>] [-nodeps]
[-nodir] [-nomodule] [-nopackageinstall] [-noproperties]
[-script <>] [-subdirs <>] [-type <>]
```

Options

Option	Description
basedir	<i>instance directory</i>
depot	<i>project depot</i>
executable	<i>executable to run</i>
installroot	<i>package installroot</i>
name	<i>object name</i>
nodeps	<i>do not install dependencies</i>
nodir	<i>do not create directories</i>
nomodule	<i>do not update module</i>
nopackageinstall	<i>do not install package</i>
noproperties	<i>do not update properties</i>
script	<i>script to execute</i>
subdirs	<i>directory list</i>
type	<i>object type</i>

5.2. extract

Extract the package archive.

check for the existence of run.sh and if not, we extract the jboss package. Upon extraction, ensure proper file permissions.

Usage

```
extract [-base <>] -filename <> [-filtersfile <>]
-installroot <> -url <>
```

Options

Option	Description
base	<i>base directory name of the installation root</i>

filename	<i>the installer filename</i>
filtersfile	<i>optional filtersfile passed in by the referring deployment</i>
installroot	<i>the installation root</i>
url	<i>the package repository url of the installer package</i>