



# **Red Hat JBoss BRMS 6.2**

## **Oracle Weblogic Installation and Configuration Guide**

For Red Hat JBoss BRMS



# Red Hat JBoss BRMS 6.2 Oracle Weblogic Installation and Configuration Guide

---

For Red Hat JBoss BRMS

PressGang CCMS Build System

Red Hat Engineering Content Services

## Legal Notice

Copyright © 2015 Red Hat, Inc.

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](#). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## Abstract

A guide to installing and configuring Red Hat JBoss BRMS on Oracle Weblogic.

---

## Table of Contents

<b>CHAPTER 1. INTRODUCTION</b>	<b>3</b>
1.1. ABOUT RED HAT JBOSS BRMS FOR ORACLE WEBLOGIC SERVER	3
<b>CHAPTER 2. DOWNLOAD AND EXTRACT</b>	<b>4</b>
2.1. DOWNLOAD RED HAT JBOSS BRMS FOR ORACLE WEBLOGIC SERVER	4
2.2. EXTRACT RED HAT JBOSS BRMS FOR ORACLE WEBLOGIC SERVER	4
<b>CHAPTER 3. CONFIGURE</b>	<b>5</b>
3.1. SETTING ENVIRONMENT VARIABLES	5
3.2. CONFIGURING SECURITY SETTINGS	5
3.3. CREATING A DATA SOURCE	6
3.4. CONFIGURING JAVA MESSAGE SERVICE (JMS)	7
<b>CHAPTER 4. INSTALL</b>	<b>10</b>
4.1. INSTALL BUSINESS CENTRAL	10
4.2. INSTALL REALTIME DECISION SERVER	10
<b>CHAPTER 5. NEXT STEPS</b>	<b>12</b>
<b>APPENDIX A. ADDITIONAL NOTES</b>	<b>13</b>
<b>APPENDIX B. REVISION HISTORY</b>	<b>14</b>



# CHAPTER 1. INTRODUCTION

## 1.1. ABOUT RED HAT JBOSS BRMS FOR ORACLE WEBLOGIC SERVER

Red Hat JBoss BPM Suite for Oracle WebLogic Server is provided as two deployable web application archives:

- **business-central.war** - The main business rules management application.
- **kie-server.war** - Application for executing rules through REST, JMS or a Java client side application.

Installation of JBoss BRMS on Oracle WebLogic Server is supported from the 6.1 version of JBoss BRMS. In this guide, you will explore how it can be installed on a full profile version of Oracle WebLogic Server.

Before installation, several configuration steps have to be performed to enable a successful setup. This guide will outline these steps.

Before you proceed, make sure you have access to the server on Oracle WebLogic Server and that you are able to successfully access Oracle WebLogic Server's administrative console through a web browser (Usually at <http://target-server:7001/console>).

As noted earlier, JBoss BRMS for Oracle WebLogic Server is distributed as a WAR file. It is then deployed as an exploded archive and then configured as any other web application.

## CHAPTER 2. DOWNLOAD AND EXTRACT

### 2.1. DOWNLOAD RED HAT JBOSS BRMS FOR ORACLE WEBLOGIC SERVER

You can download the deployable Red Hat JBoss BRMS package file for Oracle WebLogic Server from Red Hat Customer Portal:

1. Go to the [Red Hat Customer Portal](#) and log in.
2. Click **Downloads**.
3. In the **Product Downloads** page that opens, click **Red Hat JBoss BRMS**.
4. From the **Version** drop-down menu, select 6.2.
5. In the table, navigate to **Red Hat JBoss BRMS 6.2 Deployable for Oracle WebLogic Server** row and then click **Download**.

### 2.2. EXTRACT RED HAT JBOSS BRMS FOR ORACLE WEBLOGIC SERVER

The installation zip file for JBoss BRMS that you have downloaded contains all necessary WAR deployables mentioned in [Section 1.1, “About Red Hat JBoss BRMS for Oracle WebLogic Server”](#). Copy the installation zip to your WebLogic Server host and extract it to a location the server can access.

```
$ mkdir ~/apps
$ cd ~/apps
$ unzip ~/jboss-brms-6.2.0-deployable-wls12c.zip .
```



## CHAPTER 3. CONFIGURE

### 3.1. SETTING ENVIRONMENT VARIABLES

Certain environment variables on your Oracle WebLogic Server require configuration before deploying the application.

#### JVM Memory Size

With the default JVM memory size, the WebLogic Server freezes and/or causes deployment errors when deploying Business Central. Increase the memory size by setting the following environment variable:

```
USER_MEM_ARGS=-Xms512m -Xmx1024m -XX:MaxPermSize=512m
```

#### JVM Custom Properties

The following custom properties require configuration on the WebLogic Server:

##### **kie.server.jms.queues.response**

JNDI name of the response queue for the Decision Server. Set to **jms/KIE.SERVER.RESPONSE**.

##### **javax.xml.bind.context.factory**

The initialization class for the JAXB Context Factory. Set to **com.sun.xml.bind.v2.ContextFactory**

##### **org.uberfire.start.method**

Defines startable beans for Uberfire. Set to **ejb** (Enterprise JavaBeans).

##### **org.uberfire.domain**

Sets the domain for Uberfire to use. Set to **OracleDefaultLoginConfiguration**

##### **com.sun.jersey.server.impl.cdi.lookupExtensionInBeanManager**

Uses BeanManager to look up extensions. This avoids conflicts across multiple instances. Set to **true**.

Set all of these custom properties in a single environment variable:

```
JAVA_OPTIONS="-Dkie.server.jms.queues.response=jms/KIE.SERVER.RESPONSE -
Djavax.xml.bind.context.factory=com.sun.xml.bind.v2.ContextFactory -
Dorg.uberfire.start.method=ejb -
Dorg.uberfire.domain=OracleDefaultLoginConfiguration -
Dcom.sun.jersey.server.impl.cdi.lookupExtensionInBeanManager=true"
```

### 3.2. CONFIGURING SECURITY SETTINGS

Several security settings on Oracle WebLogic Server must be set for the Business Central application to work. The following settings enable the container managed authentication mechanisms provided by the WebLogic server.

1. In the WebLogic administrative console, click on **Security Realms**.

2. Choose your desired security realm, or click on **New** to create a new security realm.
3. Navigate to **Users and Groups** → **Groups** to get to the Groups list for your security realm.
4. Click on **New** to create a new group. Create the following three new groups: **admin**, **analyst**, and **kie-server**. Also create the REST API groups if you will use API. For further information about API roles, refer to [Chapter 17. Remote API](#) from Red Hat JBoss BPM Suite Development Guide.
5. Click on the **Users** tab and click **New** to create a new user. Provide this new user with a **Name** (For example, **business-central-admin**) and a **Password**. Click **OK** to save.

**NOTE**

Make sure usernames you create are not the same as the roles defined in the steps above.

6. Click on the newly created user, then click the **Groups** tab. Use the selection tool to move the **admin** group from the **Available** field to the **Chosen** field. Click **Save** to save.

**NOTE**

You may assign this user to any of the groups previously created and in actual production systems you are likely to create separate users for separate groups that align with business roles. The admin group is all encompassing and is therefore useful for the purposes of this setup.

### 3.3. CREATING A DATA SOURCE

The Business Central application requires a data source which must be created prior to the deployment of the actual WAR file. This also means that you must have access to an underlying database that the datasource connects to. Whatever your underlying database, make sure you have the datasource ready.

1. Navigate to **Services** → **Data Sources**, which takes you to your JDBC Data Sources list.
2. Click **New** → **Generic Data Source** to start creating a new data source.
3. Provide your data source with the following details:
  - **Name** - Enter a name for your data source.
  - **JNDI Name** - Set to **jdbc/jbpm**.
  - **Database Type** - Set to **MySQL**.



## NOTE

You can use alternative database types by editing **WEB-INF/classes/META-INF/persistence.xml** in the Business Central WAR archive. Edit the **hibernate.dialect** property to your preferred database. For example, to change to OracleDB, edit the **hibernate.dialect** to the following:

```
<property name="hibernate.dialect"
value="org.hibernate.dialect.Oracle9Dialect"/>
```

Click **Next** to advance to the next configuration screen.

4. Select your **Database Driver** in the drop-down menu. Click **Next** to advance to the next configuration screen.
5. Leave the **Transaction Options** as the defaults and click **Next** to advance to the next configuration screen.
6. Provide the following **Connection Properties** for your data source:
  - **Database Name** - The name of the database to use on your data source.
  - **Host Name** - The hostname or IP address of the system containing the database.
  - **Port** - The port used to connect to the database. Unless you have configured the database to use a different port, use the default port provided in this field.
  - **Database User Name** - The database user that interacts with the database. Make sure the chosen user has the required permissions to access and write to the chosen database.
  - **Password** - The password for the chosen database user.

Click **Next** to advance to the next configuration screen.

7. The **Test Database Connection** page provides a means to test and confirm your database connection. Click **Test Configuration** and the page will refresh with a valid connection message. Click **Finish** to complete the data source configuration.
8. Click on the name of the new data source and navigate to **Targets** tab. Click the checkbox for the server chosen to host the Business Central deployment. Click **Save** to save your selection.

## 3.4. CONFIGURING JAVA MESSAGE SERVICE (JMS)

Oracle WebLogic Server must be configured to send and receive JMS messages through Decision Server. This requires a JMS Server. Follow the steps below to create a JMS Server.

1. Navigate to **Services → Messaging → JMS Servers**.
2. Click **New** to start creating a new JMS Server.
3. Provide your JMS Server with a **Name**. Click **Next** to advance to the next configuration screen.
4. Select the **Target** server chosen for the Business Central deployment. Click **Finish** to complete the JMS Server creation.

## Create JMS Module

A JMS Module stores your JMS resources, such as connection factories and queues. Use the following steps to create a new JMS Module.

1. Navigate to **Services** → **Messaging** → **JMS Modules**.
2. Click **New** to start creating a new module.
3. Provide your module with a **Name** and click **Next** to advance to the next configuration screen.
4. Select the **Target** server chosen for the Business Central deployment. Click **Finish** to complete the JMS Module creation.
5. Click on the newly created module's name, then click on **Subdeployments**.
6. Click **New** to create a subdeployment for your module.
7. Provide your subdeployment with a **Name** and click **Next** to advance to the next configuration screen.
8. Choose the previously created JMS Server by marking the checkbox. Click **Finish** to complete the Module's subdeployment configuration.

## Create JMS Connection Factories

To send and receive messages from Decision Server, you will need to create the JMS connection factories - one for receiving messages and one for sending them. The following connection factories are required:

- **KIE.SERVER.REQUEST** - for all requests to the Decision Server.  
Default value: `jms/cf/KIE.SERVER.REQUEST`.
- **KIE.SERVER.RESPONSE** - receiving all responses produced by the Decision Server.  
Default value: `jms/cf/KIE.SERVER.RESPONSE`.

Use the following procedure to create each connection factory.

1. If you are not there already, navigate to **Services** → **Messaging** → **JMS Modules** to see your list of JMS Modules.
2. Click on your previously created module, then click **New** to start creating a new JMS resource.
3. Select **Connection Factory** and click **Next**.
4. Enter the **Name** of the connection factory (for example, **KIE.SERVER.REQUEST**) and the **JNDI Name** (for example, `jms/cf/KIE.SERVER.REQUEST`).  
  
Click **Next** to advance to the next configuration screen.
5. The connection factory automatically selects the servers assigned to the JMS Module as the default. Click **Finish** to complete the connection factory creation.

Repeat the above procedure for each connection factory.

## Create JMS Queues

You will now need to create the JMS Queues. These queues are the destination end points for point-to-point messaging. You will create:

- **KIE.SERVER.REQUEST** - for all requests to the Decision Server.

Default value: `jms/KIE.SERVER.REQUEST`.

- **KIE.SERVER.RESPONSE** - for the Decision Server responses.

Default value: `jms/KIE.SERVER.RESPONSE`.

Use the following procedure to create each queue.

1. If you are not there already, navigate to **Services** → **Messaging** → **JMS Modules** to see your list of JMS Modules.
2. Click on your previously created module, then click **New** to start creating a new JMS resource.
3. Select **Queue** and click **Next**.
4. Enter the **Name** of the queue (for example, **KIE.SERVER.REQUEST**) and the **JNDI Name** (for example, `jms/KIE.SERVER.REQUEST`).

Click **Next** to advance to the next configuration screen.

5. Choose the JMS Module's subdeployment that connects to the JMS Server. Click **Finish** to complete the queue creation.

Repeat the above procedure for each queue.

## CHAPTER 4. INSTALL

Now that the basic configuration is done the Oracle WebLogic Server is set to deploy JBoss BRMS.

As noted earlier, the JBoss BRMS zip file for Oracle WebLogic Server contains the deployable WAR file for Business Central.

Let's install Business Central for JBoss BRMS on Oracle WebLogic Server.

### 4.1. INSTALL BUSINESS CENTRAL

Business Central is uploaded as a Web archive and then accessed by a familiar through a web browser. Start this deployment by installing the Business Central exploded WAR as a WebLogic Application.

1. In your Oracle WebLogic Server administrative console, click on **Deployments**. This will show you all the existing applications in the system and allow you to install a new one.
2. Click the **Install** button to start the process.
3. Navigate to the exploded archive location for the Business Central WAR (For example, `/apps/business-central.war/`) and select it. Click **Next** to continue.
4. Select **Install this deployment as an application** as the targeting style, then click **Next**.
5. Set the application's Name to *business-central* and set the security model to **DD Only**. Leave the remaining options as default and click **Next** to continue.
6. In the **Additional Configuration** section, choose **No, I will review the configuration later** and click **Finish**.

You have now successfully installed Business Central on Oracle WebLogic Server. To access the application, navigate to the following location:

`http://target-server:7001/business-central`

### 4.2. INSTALL REALTIME DECISION SERVER

The Realtime Decision Server is distributed as a deployable WAR. To install it, please follow these steps:

1. Navigate to **Deployments**. This will show you all the existing applications in the system and allow you to install a new one.
2. Click the **Install** button to start the process.
3. Navigate to the exploded archive location for the Decision Server WAR (For example, `/apps/kie-server.war/`) and select it. Click **Next** to continue.
4. Select **Install this deployment as an application** as the targeting style, then click **Next**.
5. Set the application's Name to *kie-server* and set the security model to **DD Only**. Leave the remaining options as default and click **Next** to continue.

6. In the **Additional Configuration** section, choose **No, I will review the configuration later** and click **Finish**.

You can now access the Decision Server using this URL:

**`http://target-server:7001/kie-server`**

## CHAPTER 5. NEXT STEPS

Now that you have completed the installation, use the following guides to start using Red Hat JBoss BRMS:

- *Red Hat JBossBRMS 6.2 Getting Started Guide*- Provides an introductory tutorial on the core features of BRMS.
- *Red Hat JBossBRMS 6.2 Users Guide*- Provides steps on how to start creating your business rules.
- *Red Hat JBossBRMS 6.2 Administration and Configuration Guide*- Provides steps on how to configure aspects of your JBoss BRMS deployment, including migration, data management, imports and exports, integration, and monitoring.



## APPENDIX A. ADDITIONAL NOTES

- The WebLogic class loading mechanism does not provide access to the application jars located in Business Central's **WEB-INF/lib** directory, which can cause problems for developers looking to inject KIE-CDI classes. As a workaround for this issue, JBoss BRMS includes a CDI extension that temporarily swaps class loaders to load the application. This class is located in **WEB-INF**-

**classes/org/kie/workbench/backend/weblogic/SwapClassLoaderExtension**  
**.java** and enabled in **WEB-INF/classes/META-INF/services/javax.enterprise.inject.spi.Extension**:

```
org.kie.workbench.backend.weblogic.SwapClassLoaderExtension
```

- If using a non-exploded archive, Weblogic packs contents of Business Central's **WEB-INF/classes** into **WEB-INF/lib/\_wl\_cls\_gen.jar**. If a developer aims to create KieBase and KieSession from resources, KIE-Spring cannot find the classes. For this reason, the recommended deployment method is to use the exploded archives contained within the product's ZIP file.

## APPENDIX B. REVISION HISTORY

Note that revision numbers relate to the edition of this manual, not to version numbers of Red Hat JBoss BRMS.

<b>Revision 6.2.0-4</b> Updated with latest fixes.	<b>Thu Apr 28 2016</b>	<b>Tomas Radej</b>
<b>Revision 6.2.0-3</b> Build for release update 2 of JBoss BRMS.	<b>Tue Mar 29 2016</b>	<b>Tomas Radej</b>
<b>Revision 6.2.0-2</b> Added note about versions in Revision History, fixed changelog dates.	<b>Mon Nov 30 2015</b>	<b>Tomas Radej</b>
<b>Revision 6.2.0-1</b> Initial build for release 6.2.0 of JBoss BRMS.	<b>Mon Nov 30 2015</b>	<b>Tomas Radej</b>