



Red Hat JBoss Data Virtualization 6.4

Installation Guide

This guide is for installation teams.

Red Hat JBoss Data Virtualization 6.4 Installation Guide

This guide is for installation teams.

Red Hat Customer Content Services

Legal Notice

Copyright © 2018 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

This document guides you through all of the installation options for Red Hat JBoss Data Virtualization.

Table of Contents

CHAPTER 1. DOWNLOADING THE PRODUCT	3
1.1. BACK UP YOUR DATA	3
1.2. DOWNLOAD JBOSS DATA VIRTUALIZATION INSTALLER	3
1.3. VERIFY DOWNLOADED FILES	4
CHAPTER 2. INSTALLING PREREQUISITE COMPONENTS	5
2.1. INSTALL OPEN JDK ON RED HAT ENTERPRISE LINUX	5
2.2. INSTALL MAVEN	5
CHAPTER 3. INSTALLING RED HAT JBOSS DATA VIRTUALIZATION	8
3.1. GRAPHICAL INSTALLATION	8
3.2. TEXT-BASED INSTALLER	10
3.3. AUTOMATED INSTALLATION	11
CHAPTER 4. INSTALLING THE JBOSS DATA VIRTUALIZATION DEVELOPMENT TOOLS	12
4.1. INSTALLING JBOSS DATA VIRTUALIZATION DEVELOPMENT TOOLS	12
CHAPTER 5. RUNNING RED HAT JBOSS DATA VIRTUALIZATION	13
5.1. STARTING JBOSS DATA VIRTUALIZATION	13
5.2. INSTALLING IN DOMAIN MODE	13
5.3. STARTING JBOSS DATA VIRTUALIZATION AS MANAGED DOMAIN	16
5.4. STOPPING JBOSS DATA VIRTUALIZATION	16
CHAPTER 6. CONFIGURING YOUR MAVEN REPOSITORIES	17
6.1. ABOUT THE PROVIDED MAVEN REPOSITORIES	17
6.2. CONFIGURE MAVEN TO USE THE FILE SYSTEM REPOSITORIES	17
6.3. CONFIGURE MAVEN TO USE THE ONLINE REPOSITORIES	18
6.4. USING MAVEN DEPENDENCIES FOR RED HAT JBOSS DATA VIRTUALIZATION	20
6.5. OFFLINE MODE	22
CHAPTER 7. INTEGRATING RED HAT JBOSS DATA VIRTUALIZATION WITH RED HAT JBOSS DATA GRID	23
7.1. CONFIGURE RED HAT JBOSS DATA GRID CONNECTORS	23
CHAPTER 8. ODBC SUPPORT	24
8.1. INSTALL THE ODBC DRIVER ON RED HAT ENTERPRISE LINUX	24
8.2. CONFIGURE THE ODBC ENVIRONMENT	24
8.3. CONFIGURE THE DSN FOR LINUX INSTALLATION	24
8.4. INSTALL THE ODBC DRIVER ON MICROSOFT WINDOWS	24
8.5. CONFIGURE THE DSN FOR WINDOWS INSTALLATION	25
8.6. CONFIGURE ODBC OPTIONS ON MICROSOFT WINDOWS	25
8.7. INSTALL THE ODBC DRIVER ON SOLARIS	27
8.8. CONFIGURE THE DSN FOR SOLARIS INSTALLATION	27
8.9. CONFIGURE ODBC OPTIONS ON RED HAT ENTERPRISE LINUX	27
8.10. DSN-LESS CONNECTION	28
CHAPTER 9. RUNNING IN CLOUD ENVIRONMENTS	30
9.1. RUN RED HAT JBOSS DATA VIRTUALIZATION IN AN AMAZON AWS CLOUD INSTANCE	30
9.2. RUN RED HAT JBOSS DATA VIRTUALIZATION IN A GOOGLE COMPUTE INSTANCE	30
9.3. RUN RED HAT JBOSS DATA VIRTUALIZATION ON AN AZURE INSTANCE	31
APPENDIX A. REVISION HISTORY	32

CHAPTER 1. DOWNLOADING THE PRODUCT

1.1. BACK UP YOUR DATA



WARNING

Red Hat recommends that you back up your system settings and data before undertaking any of the configuration tasks mentioned in this book.

1.2. DOWNLOAD JBOSS DATA VIRTUALIZATION INSTALLER

The JBoss Data Virtualization installer archive is available on the Red Hat Customer portal at <https://access.redhat.com/>.

Prerequisites

- Set up an account on the Red Hat Customer Portal at <https://access.redhat.com/>.
- Ensure your Red Hat subscriptions are up to date.
- Review the supported configurations and ensure your system is supportable.
- Ensure that you have administration privileges for the installation directory.
- Ensure that JAVA_HOME and PATH have been set in the Environment properties for shortcuts to work on Microsoft Windows servers.
- A Java 7 or 8 JDK is required.



NOTE

If you intend to use Red Hat SSO, you must have at least Java 7. Java 7 is also a minimum requirement for the Impala, Hive and HBase data sources. Java 1.7 or greater is also required if you wish to use the file translator or Data Services Builder. Also note that if you intend to access Red Hat JBoss Data Grid 7.1, Red Hat JDV must be running JDK 1.8.

- Optional: Red Hat JBoss Enterprise Application Platform 6.4.x if you do not want to use the version of Red Hat JBoss EAP that comes bundled with the Data Virtualization installer.

Procedure 1.1. Download JBoss Data Virtualization Installer

1. Go to <https://access.redhat.com/>.
2. Click **Log in** and enter your **Red Hat Login** and **Password** to access the **Customer Portal**.
3. Click **Downloads -> Red Hat JBoss Data Virtualization**.

4. Click **Download** next to the **Red Hat JBoss Data Virtualization [Version] Installer** option.
5. Save the file.

1.3. VERIFY DOWNLOADED FILES

Procedure 1.2. Verify File Checksums on Red Hat Enterprise Linux

1. **Obtain checksum values for the downloaded file**
 - a. Go to <https://access.redhat.com/jbossnetwork/>. Log in if required.
 - b. Select your **Product** and **Version**.
 - c. Select the packages you want to verify. Once you have chosen them, navigate to the **Software Details** page.
 - d. Take note of the **MD5** and **SHA - 256** checksum values.
2. **Run a checksum tool on the file**
 - a. Navigate to the directory containing the downloaded file in a terminal window.
 - b. Run `md5sum downloaded_file`.
 - c. Run `shasum downloaded_file`.

Example output:

```
[localhost]$ md5sum jboss-dv-installer-[VERSION]-redhat-[VERSION].jar
MD5 (jboss-dv-installer-[VERSION]-redhat-[VERSION].jar) =
0d1e72a6b038d8bd27ed22b196e5887f
[localhost]$ shasum jboss-dv-installer-[VERSION]-redhat-[VERSION].jar
a74841391bd243d2ca29f31cd9f190f3f1bdc02d  jboss-dv-installer-[VERSION]-redhat-[VERSION].jar
```

3. Compare the checksum values returned by the `md5sum` and `shasum` commands with the corresponding values displayed on the **Software Details** page.
4. Download the file again if the two checksum values are not identical. A difference between the checksum values indicates that the file has either been corrupted during download or has been modified since it was uploaded to the server. Contact Red Hat Support for assistance if after several downloads the checksum does not successfully validate.



NOTE

No checksum tool is included with Microsoft Windows. Download a third-party MD5 application such as MD5 Summer from <http://www.md5summer.org/>.

CHAPTER 2. INSTALLING PREREQUISITE COMPONENTS

2.1. INSTALL OPEN JDK ON RED HAT ENTERPRISE LINUX

1. Install the OpenJDK package:

```
yum install java-[VERSION]-openjdk-devel
```

2. To install it to alternatives, run these commands:

```
sudo alternatives --install /usr/bin/java java /usr/lib/jvm/java-[VERSION]-openjdk/bin/java 1000
```

```
sudo alternatives --install /usr/bin/javac javac /usr/lib/jvm/java-[VERSION]-openjdk/bin/javac 1000
```

3. As root, run the alternatives command for **java**:

```
/usr/sbin/alternatives --config java
```

4. Select **/usr/lib/jvm/jre-[VERSION]-openjdk/bin/java**.

5. Then do the same for **javac**:

```
/usr/sbin/alternatives --config javac
```

6. Select **/usr/lib/jvm/java-[VERSION]-openjdk/bin/javac**.

2.2. INSTALL MAVEN

Maven is a build system for projects that use the Project Object Model (POM). It downloads package dependencies quickly and easily. If you have an infrastructure team providing your Red Hat JBoss Data Virtualization environment, you can skip this procedure. Otherwise, you will need to follow it to install Maven to build your projects.

Prerequisites

The following software must be installed:

- An archiving tool for extracting the contents of compressed files.
- Open JDK.

Procedure 2.1. Install Maven

1. **Download Maven.**

- a. Go to <http://maven.apache.org/download.cgi>.
- b. Download the **apache-maven-[latest-version]** ZIP file.

2. **Install and configure Maven.**

- o **On Red Hat Enterprise Linux**

- Extract the ZIP archive to the directory where you wish to install Maven.
- Open your `.bash_profile` file:

```
vi ~/.bash_profile
```

- Add the `M2_HOME` environment variable to the file:

```
export M2_HOME=/path/to/your/maven
```

- Add the `M2` environment variable to the file:

```
export M2=$M2_HOME/bin
```

- Add the variable `$JAVA_HOME/bin` to set the path to the correct Java installation.

**NOTE**

Make sure `JAVA_HOME` is pointing to a valid location.

- Add the `M2` environment variable to the file:

```
export PATH=$M2:$PATH
```

- Save the file and exit your text editor.

- Reload your profile:

```
source ~/.bash_profile
```

- Run the following command to verify that Maven is installed successfully on your machine:

```
mvn --version
```

- o **On Microsoft Windows**

- Extract the ZIP archive to the directory where you wish to install Maven. The subdirectory **apache-maven-[latest-version]** is created from the archive.
- Press **Start+Pause|Break**. The **System Properties** dialog box is displayed.
- Click the **Advanced** tab and click **Environment Variables**.
- Under System Variables, select **Path**.
- Click **Edit** and add the two Maven paths using a semicolon to separate each entry.
 - Add the `M2_HOME` variable and set the path to **C:\path\to\your\Maven**.
 - Add the `M2` variable and set the value to `%M2_HOME%\bin`.

- f. Update or create the Path environment variable:
 - Add the %M2% variable to allow Maven to be executed from the command line.
 - Add the variable %JAVA_HOME%\bin to set the path to the correct Java installation.
- g. Click **OK** to close all the dialog boxes including the **System Properties** dialog box.
- h. Open Windows command prompt and run the following command to verify that Maven is installed successfully on your machine:

```
■ mvn --version
```

CHAPTER 3. INSTALLING RED HAT JBOSS DATA VIRTUALIZATION

3.1. GRAPHICAL INSTALLATION

The Graphical Installer allows you to install **JBoss Data Virtualization** on your machine using step-by-step GUI instructions.

Prerequisites

You must have already downloaded the Red Hat JBoss Data Virtualization jar file from the [Customer Portal](#).

Procedure 3.1. Install JBoss Data Virtualization

1. Open a terminal window and navigate to the location where the GUI installer was downloaded.
2. Launch the GUI installer:

```
java -jar jboss-dv-VERSION-installer.jar
```

3. A dialogue box will open followed by the End User License Agreement. If you accept the terms of the agreement, click **I accept the terms of this license agreement** and then click **Next**.
4. Tell Red Hat JBoss Data Virtualization where Red Hat JBoss EAP is installed on your server or specify a new location if you do not have it installed as it comes bundled with the product. (If you have a pre-existing installation of Red Hat JBoss EAP, ensure that it is patched to the latest version of 6.4.x.) Click **Next**.
5. Ensure Teiid Installation and Dashboard Builder are selected. Click Data Services Builder if you want the graphical design tool as well or Modeshape Installation if you want that (you cannot have both Data Services Builder and Modeshape). Click **Next**.
6. You will be prompted to create a new EAP Admin, Dashboard Admin, Teiid data access user and Modeshape user (if Modeshape is selected) and whether you want to enable OData access. Once created, EAP Admin is added to the ManagementRealm and can be used to access the Management Console. The other users are added to the ApplicationRealm and can be used to access specific components of JBoss Data Virtualization. Enter the new username and password in the appropriate fields and click **Next**.

You must ensure that you remember all of these passwords. They give you access to different parts of the system. The EAP account and password allows you to administer the EAP Server, the Dashboard password is for administrative functions related to the web interface, and the Teiid data access user and ModeShape user are for standard user access.

Note that the username and password are not allowed to match and the password must have at least eight characters, with one alphabetical character, one numeric character and one non-numeric character.

7. You can install Red Hat JBoss Data Virtualization either with default configuration or with additional configuration options. In this example installation, we will be using the defaults only, so select **Perform default configuration** to install Red Hat JBoss Data Virtualization with default options. Click **Next**.

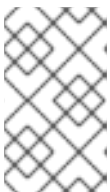
8. The **Configure password vault** screen appears. Input your desired password, which must have no fewer than six characters. Click **Next**.



NOTE

The default H2 database used to store this data is not suitable for production systems. Use it in testing and evaluation environments only.

9. If you selected the Data Services Builder option, you will be prompted for SSL security information. Click **Generate new keystore** or use the option to configure your existing keystore if you already have one you set up previously. Leave the SSL Keystore alias as "jboss". Input a password of six characters. Leave the filepath as the default. Click **Next**.
10. A summary of the installation is displayed. Click **Next** for the installation to commence. This may take a few moments. Once all the components are installed, click **Next**.
11. Click **Generate an automatic installation script** if you wish to generate an automatic script. This allows you to quickly reinstall or mass-deploy the product using the settings you have configured during the initial installation, without having to step through the wizard each time.



NOTE

An automatic installation script created for a given minor release will only work with that release. For example, you cannot use a script created with 6.0 for 6.1 or a script created for 6.3 with 6.4.

12. Click **Done** to exit the installer.



NOTE

After installing JBoss Data Virtualization, if you move the product to another location, you may see some `FileNotFoundException` exceptions. This is because some file paths are hard-coded by the JBoss EAP Server.



WARNING

If you attempt to use a vault with a keystore created with a different JDK than the one in which the data is stored, your server will fail to start. You must consistently use the same JDK when accessing the vault.

Table 3.1. Other Installation Options

Option	Description
Check to use one password for all default passwords.	Allows you to use the default EAP password you create for all of the accounts.

Option	Description
Add a security domain.	If you select this option, the name of the field is mandatory. You can choose between a default and an Infinispan cache type and an authentication module for additional security. You can also add mapping elements to establish roles and credentials. You can also configure a JSSE element using either a keystore or a truststore.
Install JDBC Driver.	The JDBC driver converts application code to the relevant database language. Point to the location of your driver in the address field. You can add more than one.
Configure audit/command logging.	You can activate the logging level for the software to provide audit trails.
Configure JAVA_OPTS.	Configure additional Java options to override the normal defaults for both standalone and domain mode. Input the options you want in the provided fields.
Enable SSL security.	Generate a new keystore or point to an existing one. (The existing location should be that of your SSL certification file.)
Add a security domain.	Adds the server to a secure domain.

3.2. TEXT-BASED INSTALLER

You can install Red Hat JBoss Data Virtualization using the text-based installer. In this mode, you run the installation steps without stepping through the graphical wizard. The GUI installer will run in text mode automatically if no display server is available.

Prerequisites

You must have already downloaded the Red Hat JBoss Data Virtualization jar file from the [Customer Portal](#).

Procedure 3.2. Install JBoss Data Virtualization

1. Open a terminal window and navigate to the location where the GUI installer was downloaded.
2. Start the installation process:

```
java -jar jboss-dv-VERSION-installer.jar -console
```

3. Follow the installation prompts displayed on the terminal. You can either install with default configuration or you can complete additional configuration steps.

4. Finally, generate the automatic installation script. You can use this script to perform headless installation or identical installations across multiple instances.

3.3. AUTOMATED INSTALLATION

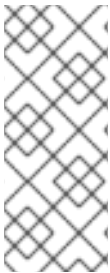
If you need to install a Red Hat JBoss product multiple times with the same configuration, you can save time by using an installation script. By using an installation script with predefined settings, you can perform the entire installation by running a single command, instead of working through the installation step by step each time. You can generate an installation script by running the installer (in graphical or text mode), stepping through with your desired configuration, and then choosing to generate the script when prompted towards the end of the process.

Prerequisites

- You must have downloaded the relevant installer JAR file from <https://access.redhat.com/jbossnetwork/>.
- You must have generated the script and saved it as an XML file during a previous installation.

Procedure 3.3. Installing with a Script

- `java -jar jboss-PRODUCT-installer-VERSION.jar SCRIPT.xml`



NOTE

You can also provide variables using the CLI or automatic installation. Use **java -jar path/to/installer.jar -variablefile /pathtofile** to supply variables using a configuration file or use **java -jar path/to/installer.jar -variables EXAMPLE1=example1,EXAMPLE2=example2** to provide a comma-separated list of variables.

CHAPTER 4. INSTALLING THE JBOSS DATA VIRTUALIZATION DEVELOPMENT TOOLS

4.1. INSTALLING JBOSS DATA VIRTUALIZATION DEVELOPMENT TOOLS

Prerequisites

The following software must be installed:

- Red Hat JBoss Developer Studio (See *Red Hat JBoss Developer Studio Installation Guide* https://access.redhat.com/documentation/en-us/red_hat_jboss_developer_studio/11.0/html/installation_guide/)
- An archiving tool for extracting the contents of compressed files
- Open JDK (or another supported Java Virtual Machine).

Procedure 4.1. Install JBoss Developer Studio Integration Stack

1. Start **Red Hat JBoss Developer Studio**.
2. In **Red Hat JBoss Developer Studio**, click **Help > Install New Software...** from the main menu.
3. On the **Available Software** page, click the **Add ...** button.
4. On the **Add Repository** dialog, enter following details:

Enter JBDSIS (or another unique name) in the **Name** field.

Enter <https://devstudio.redhat.com/11/stable/updates/integration-stack/> in the **Location** field (or <https://devstudio.redhat.com/11/stable/updates/integration-stack/earlyaccess/> if you are using the Early Access version).

5. Click **OK**.
6. In the update site tree view, select the **JBoss Data Virtualization Development** folder and all its children.



NOTE

If JBDSIS is already installed then proceed to next step. To confirm that JBDSIS is installed on your machine, click **what is already installed**.

7. Click **Next**.
8. Accept any additional dependencies and license agreements, then click **Finish**.

CHAPTER 5. RUNNING RED HAT JBOSS DATA VIRTUALIZATION

5.1. STARTING JBOSS DATA VIRTUALIZATION

- You can run **JBoss Data Virtualization** by starting the **JBoss EAP server**. To start the **JBoss EAP server**:

- **Red Hat Enterprise Linux**

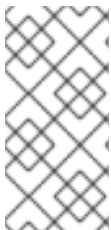
Open a terminal and enter the command:

```
$ EAP_HOME/bin/standalone.sh
```

- **Microsoft Windows**

Open a terminal and enter the command:

```
$ EAP_HOME\bin\standalone.bat
```



NOTE

To verify that there have been no errors, check the server log:

EAP_HOME/LOG/server.log. You can also verify this by opening the [Management Console](#) and logging in using the username and password of a registered JBoss EAP Management User.

5.2. INSTALLING IN DOMAIN MODE

The Domain mode is for multiple server instances. It requires load balancing capable for handling TCP for JDBC connections. (Apache will not work in this case.)

When you have more than one Red Hat JBoss EAP instance in your server farm and you start them all in domain mode, all of the configuration options for this server farm can be centrally managed. For example, you can deploy an artifact or create a data source across all the instances, with one single CLI-based call. Red Hat JBoss Data Virtualization extends this configuration concept to allow you to deploy your VDBs and translators across the whole server farm.

When domain mode is combined with the HA (high availability) profile, you can cluster the Red Hat JBoss Data Virtualization server instances that are deployed. (The HA profile is set as the default in the **domain.xml** file.) When you start the server using the **domain.xml** file, the distributed caching that is used for ResultSet caching and Internal Materialized caching is automatically configured. You use the Admin API the same way in both the standalone mode and the domain mode.

When multiple Red Hat JBoss Data Virtualization instances are available in a cluster, you can make use load balancing and fail-over features.

You must create a server group, then create a server and then deploy all the available resources to the server group you have created.

The following instructions teach you how to install a systemd service on Linux. For other use cases, please refer to https://access.redhat.com/documentation/en-us/red_hat_jboss_enterprise_application_platform/6.4/html-single/administration_and_configuration_guide/index.

1. To start the server in "Domain" mode, install Red Hat JBoss Data Virtualization on all the servers

that are going to be part of the cluster. Select one of the servers as the "master" domain controller. (The rest of the servers will be slaves that connect to the "master" domain controller for all the administrative operations.)

2. On the domain controller, copy the **domain-controller/init.d** directory to **JBOSS_HOME/bin**.
3. On the host controller, copy the **host-controller/init.d** directory to **JBOSS_HOME/bin**.
4. Prepare the directory structure:

```
adduser jboss
mkdir /opt/jdv
mkdir /etc/jboss-as
mkdir /var/log/jboss
mkdir /var/run/jboss
chown -R jboss: /opt/jdv
chown -R jboss: /etc/jboss-as
chown -R jboss: /var/log/jboss
chown -R jboss: /var/run/jboss
```

5. Configure the domain:

```
cat > /etc/jboss-as/jboss-as.conf <<EOF
JBOSS_HOME=/opt/jdv/jboss-data-virtualization
JBOSS_PIDFILE=/var/run/jboss/jboss-as-domain.pid
JBOSS_HOST_CONFIG=host.xml
JBOSS_ADDRESS_MANAGEMENT=10.255.0.82
JBOSS_USER=jboss
JBOSS_CONSOLE_LOG=/var/log/jboss/console.log
EOF
```

6. Configure the host controller:

```
cat > /etc/jboss-as/jboss-as.conf <<EOF
JBOSS_HOME=/opt/jdv/jboss-data-virtualization
JBOSS_PIDFILE=/var/run/jboss/jboss-as-domain.pid
JBOSS_HOST_CONFIG=host-slave.xml
JBOSS_MASTER_ADDRESS=10.255.0.82
JBOSS_PUBLIC_ADDRESS=10.255.0.83
JBOSS_USER=jboss
JBOSS_CONSOLE_LOG=/var/log/jboss/console.log
EOF
```

7. Add this Red Hat JBoss Data Virtualization script to systemd:

```
JBOSS_HOME=/opt/jdv/jboss-data-virtualization
cat > /tmp/teste <<EOF
[Unit]
Description=JBoss Data Virtualization Systemctl script
After=NetworkManager.service
[Service]
Type=forking
ExecStart=$JBOSS_HOME/bin/init.d/jboss-as-domain.sh start
```

```

ExecStop=$JBOSS_HOME/bin/init.d/jboss-as-domain.sh stop
ExecReload=$JBOSS_HOME/bin/init.d/jboss-as-domain.sh restart
PIDFile=/var/run/jboss/jboss-as-domain.pid
[Install]
WantedBy=multi-user.target
EOF

```

8. Configure the firewall ports on both hosts:

```

cat > /etc/firewalld/services/jboss-data-virtualization.xml
<?xml version="1.0" encoding="utf-8"?>
<service version="1.0">
  <short>jboss-data-virtualization</short>
  <port port="31000" protocol="tcp"/>
  <port port="8080" protocol="tcp"/>
  <port port="8443" protocol="udp"/>
  <port port="8009" protocol="tcp"/>
  <port port="4447" protocol="tcp"/>
  <port port="9990" protocol="udp"/>
  <port port="9999" protocol="tcp"/>
</service>
EOF

```

```

firewall-cmd --zone=public --add-service=jboss-data-virtualization
firewall-cmd --permanent --zone=public --add-service=jboss-data-
virtualization
firewall-cmd --zone=public --list-services
firewall-cmd --permanent --zone=public --list-services

```

If you are using port offsets, adjust the values by adding the offset.

9. Once you have configured all the servers, start the "master" node:

```
/bin/domain.sh
```

10. Start the "slave" nodes:

```
/bin/domain.sh
```

The slave nodes fetch their domain configuration settings from the "master" node.

11. Once VDBs have been deployed, users can connect their JDBC applications to Red Hat JBoss Data Virtualization.
12. You can enable Red Hat JBoss Data Virtualization as a service with the operating system, using this command:

```
systemctl enable jboss-data-virtualization
```

You can now start the service like this:

```
systemctl start jdvd
```

To stop it, run this command:

```
systemctl stop jdv
```

To restart, run this command:

```
systemctl restart jdv
```



NOTE

Teiid Designer cannot connect to the Red Hat JBoss Data Virtualization Server in "domain" mode. Red Hat recommends using other types of deployment strategies (such as CLI or web-console) for deploying and testing, as it is expected you will be using domain mode in production environments. Teiid Designer is to aid development-time activities only and should only be used in testing environments.

5.3. STARTING JBOSS DATA VIRTUALIZATION AS MANAGED DOMAIN

Order of Operations

The domain controller must be started before any slave servers in any server groups in the domain. Use this procedure first on the domain controller, and then on each associated host controller and each other host associated with the domain. Please refer to the https://access.redhat.com/documentation/en-us/red_hat_jboss_enterprise_application_platform/6.4/html-single/administration_and_configuration_guide/index for more information.

Procedure 5.1. Start the Platform Service as a Managed Domain

1. For Red Hat Enterprise Linux, run this command: **`EAP_HOME/bin/domain.sh`**
2. For Microsoft Windows, run this command: **`EAP_HOME\bin\domain.bat`**
3. To list all available parameters for the start-up scripts, use the **`-h`** parameter.

5.4. STOPPING JBOSS DATA VIRTUALIZATION

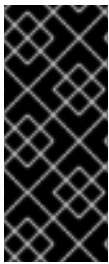
To stop **JBoss Data Virtualization**, you must stop the **JBoss EAP server**. The way you stop JBoss EAP depends on how it was started. You can stop JBoss EAP by pressing **CTRL+C** in the terminal.

CHAPTER 6. CONFIGURING YOUR MAVEN REPOSITORIES

6.1. ABOUT THE PROVIDED MAVEN REPOSITORIES

A set of repositories containing artifacts required to build applications is provided with this release. Maven must be configured to use these repositories and the Maven Central Repository in order to provide correct build functionality.

Two interchangeable sets of repositories ensuring the same functionality are provided. The first set is available for download and is stored in a local file system. The second set is hosted online for use as remote repositories. If you provided the location of Maven's **settings.xml** file during installation, Maven is already configured to use the online repositories.



IMPORTANT

Maven repositories are also subject to patching. After a patch is released, it is applied to the remote repositories. Both original and patched artifacts reside there, only the versions of artifacts are incremented. It is user's responsibility to pick the new version of the patched artifact in their dependency management. For more information see <https://access.redhat.com/site/maven-repository>.

6.2. CONFIGURE MAVEN TO USE THE FILE SYSTEM REPOSITORIES

The Red Hat JBoss DV Maven repository is available online, so it is not necessary to download and install it locally. However, if you prefer to install the JBoss Data Virtualization Maven repository locally, there are three ways to do it: on your local file system, on Apache Web Server, or with a Maven repository manager. This example covers the steps to download the JBoss DV 6 Maven Repository to the local file system. This option can help you become familiar with using Maven for development but is not recommended for team production environments.

Procedure 6.1. Configure Maven

1. Go to <https://access.redhat.com/jbossnetwork/restricted/listSoftware.html?product=data.services.platform>.
2. Download the desired version.
3. Unzip the file on the local file system into a directory of your choosing.
4. Add entries for the unzipped repositories to Maven's **settings.xml** file. The following code sample contains a profile with the repositories and an activation entry for the profile:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/xsd/settings-1.0.0.xsd">
  <localRepository/>
  <profiles>
    <!-- Profile with local repositories required by Data
    Virtualization -->
    <profile>
      <id>dv-local-repos</id>
      <repositories>
        <repository>
```

```

<id>dv-[VERSION]-repository</id>
<name>DV [VERSION] GA Repository</name>
<url>file:///<!-- path to the repository -->/jboss-dv-
[VERSION].redhat-[VERSION]-maven-repository/maven-repository</url>
<layout>default</layout>
<releases>
  <enabled>true</enabled>
  <updatePolicy>never</updatePolicy>
</releases>
<snapshots>
  <enabled>>false</enabled>
  <updatePolicy>never</updatePolicy>
</snapshots>
</repository>
</repositories>
<pluginRepositories>
  <pluginRepository>
    <id>dv-[VERSION].GA-redhat-[VERSION]-repository</id>
    <name>DV [VERSION] GA Repository</name>
    <url>file:///<!-- path to the repository -->/jboss-dv-
[VERSION].redhat-[VERSION]-maven-repository/maven-repository</url>
    <layout>default</layout>
    <releases>
      <enabled>true</enabled>
      <updatePolicy>never</updatePolicy>
    </releases>
    <snapshots>
      <enabled>>false</enabled>
      <updatePolicy>never</updatePolicy>
    </snapshots>
  </pluginRepository>
</pluginRepositories>
</profile>
</profiles>
<activeProfiles>
  <!-- Activation of the Data Virtualization profile -->
  <activeProfile>dv-local-repos</activeProfile>
</activeProfiles>
</settings>

```

6.3. CONFIGURE MAVEN TO USE THE ONLINE REPOSITORIES

The online repositories required for Red Hat JBoss Data Virtualization are located at <https://maven.repository.redhat.com/ga/>. (There is also an early access repository at <https://maven.repository.redhat.com/earlyaccess/all/>.)

If you provided the location of Maven's **settings.xml** file during installation, Maven is already configured to use the online repositories.

Procedure 6.2. Configuring Maven to Use the Online Repositories

1. Add entries for the online repositories to Maven's **settings.xml** file:

```
<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
```

```

    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0
http://maven.apache.org/xsd/settings-1.0.0.xsd">

    <profiles>
      <!-- Profile with online repositories required by Data
Virtualization -->
      <profile>
        <id>dv-online-profile</id>
        <repositories>
          <repository>
            <id>jboss-ga-repository</id>

            <url>http://maven.repository.redhat.com/techpreview/all</url>
            <releases>
              <enabled>true</enabled>
            </releases>
            <snapshots>
              <enabled>false</enabled>
            </snapshots>
          </repository>
        </repositories>
        <pluginRepositories>
          <pluginRepository>
            <id>jboss-ga-plugin-repository</id>

            <url>http://maven.repository.redhat.com/techpreview/all</url>
            <releases>
              <enabled>true</enabled>
            </releases>
            <snapshots>
              <enabled>false</enabled>
            </snapshots>
          </pluginRepository>
        </pluginRepositories>
      </profile>
    </profiles>

    <activeProfiles>
      <!-- Activation of the Data Virtualization profile -->
      <activeProfile>dv-online-profile</activeProfile>
    </activeProfiles>

  </settings>

```

2. If you modified the **settings.xml** file while JBoss Developer Studio was running, you must refresh Maven settings in the IDE. From the menu, choose **Window** → **Preferences**. In the **Preferences** Window, expand **Maven** and choose **User Settings**. Click the **Update Settings** button to refresh the Maven user settings in JBoss Developer Studio.

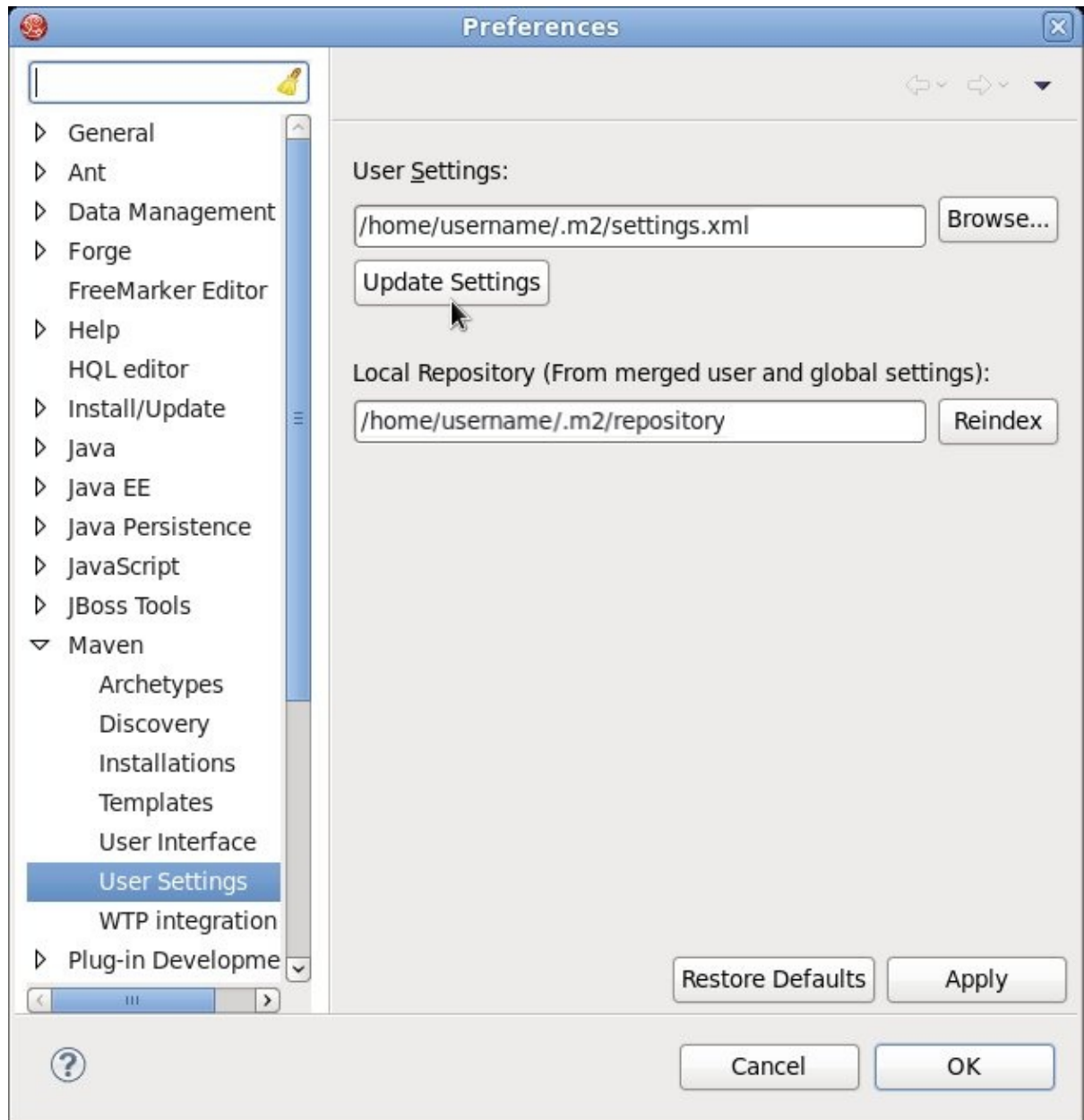


Figure 6.1. Update Maven User Settings

If your cached local Maven repository contains outdated artifacts, you may encounter one of the following Maven errors when you build or deploy your project:

- Missing artifact *ARTIFACT_NAME*
- [ERROR] Failed to execute goal on project *PROJECT_NAME*; Could not resolve dependencies for *PROJECT_NAME*

To resolve the issue, delete the cached local repository – the `~/ .m2/repository/` directory on Linux or the `%SystemDrive%\Users\USERNAME\.m2\repository\` directory on Windows. This will force Maven to download correct versions of required artifacts during the next build.

6.4. USING MAVEN DEPENDENCIES FOR RED HAT JBOSS DATA VIRTUALIZATION

In order to use the correct Maven dependencies in your Red Hat JBoss Data Virtualization project, you must add relevant Bill Of Materials (BOM) and parent POM files to the project's `pom.xml` file. Adding the

BOM and parent POM files ensures that the correct versions of plug-ins and transitive dependencies from the provided Maven repositories are included in the project.

The Maven repository is designed to be used only in combination with Maven Central and no other repositories are required.

The parent POM file to use is **org.jboss.dv.component.management:dv-parent-[VERSION].pom**.

The BOM file to use is **org.jboss.dv.component.management:dv-dependency-management-all-[VERSION].pom**.

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    http://maven.apache.org/maven-v4_0_0.xsd">

  <modelVersion>4.0.0</modelVersion>

  <!-- Example POM file using the DV 6.4.0 and EAP 6.4 component
versions.
  - Parent is set to the DV 6.4.0 parent management POM, which will
  - bring in the correct toolchain (plugin) versions.
  - DependencyManagement dependencies include the DV 6.4.0 and EAP
6.4
  - BOMs - which will bring in the correct compile-time (and other
  - scoped) versions.
-->

  <name>Example POM for DV 6.4.0</name>
  <groupId>org.jboss.dv</groupId>
  <artifactId>dv-example</artifactId>
  <version>0.0.1</version>
  <packaging>pom</packaging>

  <parent>
    <!-- DV version (parent) -->
    <groupId>org.jboss.dv.component.management</groupId>
    <artifactId>dv-parent</artifactId>
    <version>[VERSION]</version>
  </parent>

  <dependencyManagement>
    <dependencies>
      <!-- DV BOM -->
      <dependency>
        <groupId>org.jboss.dv.component.management</groupId>
        <artifactId>dv-dependency-management-all</artifactId>
        <version>[VERSION]</version>
        <type>pom</type>
        <scope>import</scope>
      </dependency>

    </dependencies>
```

```
</dependencyManagement>

</project>
```

6.5. OFFLINE MODE

6.5.1. Using a Custom Offline Repository

When you move from the development phase of a project to the *deployment* phase, it is typically more convenient to pre-install all of the artifacts required by your application, rather than downloading them from the Internet on demand. In this case, the ideal solution is to create a custom offline repository, which contains the artifacts needed for your deployment. Creating a custom offline repository by hand, however, would be difficult, because it would need to include *all* of the transitive dependencies associated with your application bundles and features.

The ideal way to create a custom offline repository is to generate it, with the help of the Apache Karaf **features-maven-plugin** plug-in.

If you have a Maven project and you need to create an offline repository for building this project and its runtime dependencies, you can use the [maven dependency plugin](#).

For example, from the top-level directory of a Maven project (such that the current directory has a **pom.xml** file), you should be able to run the following Maven command:

```
mvn org.apache.maven.plugins:maven-dependency-plugin:2.8:go-offline -
Dmaven.repo.local=/tmp/cheese
```

This downloads all the Maven dependencies and plug-ins required to build the project to the **/tmp/cheese** directory.

To generate the custom offline repository, open a new command prompt, change directory to **ProjectDir/custom-repo**, and enter the following Maven command:

```
mvn generate-resources
```

Assuming that the Maven build completes successfully, the custom offline repository should now be available in the following location:

```
ProjectDir/custom-repo/target/features-repo
```

CHAPTER 7. INTEGRATING RED HAT JBOSS DATA VIRTUALIZATION WITH RED HAT JBOSS DATA GRID

7.1. CONFIGURE RED HAT JBOSS DATA GRID CONNECTORS

The `infinispan-cache`, `infinispan-hotrod` and `infinispan-cache-dsl` translators are not pre-configured to work when the server starts. You must manually configure them if you wish to use Red Hat JBoss Data Grid as a data source.

1. Download the required Java web client and apply overlay to the modules directory.
2. Navigate to the **`docs/teiid/datasources/infinispan/`** directory (and to the **`docs/teiid/datasources/infinispan-hotrod-7.1/`** directory for Hotrod).
3. Execute the appropriate script: **`add-infinispan-cache-translator.cli`**, **`add-infinispan-cache-dsl-translator.cli`** and **`infinispan-hotrod-translator.cli`**.

CHAPTER 8. ODBC SUPPORT

8.1. INSTALL THE ODBC DRIVER ON RED HAT ENTERPRISE LINUX

Prerequisites

- Administrative permissions are required.
1. Download the correct ODBC driver package (**jboss-dv-psqlodbc-[version]-X.rpm**) from <https://access.redhat.com/jbossnetwork/>.
 2. Install the package: **sudo yum localinstall jboss-dv-psqlodbc-[version]-X.rpm**.



NOTE

Installation packages for different operating systems can be downloaded from <https://access.redhat.com/jbossnetwork/>.

8.2. CONFIGURE THE ODBC ENVIRONMENT

- Run the **/opt/redhat/jboss-dv/v6/psqlodbc/etc/setenv.sh** script:

```
[localhost etc]$ ./setenv.sh
```

This script adds the required directories to the **LD_LIBRARY_PATH** and **PATH** environment variables. This script has to be run every time you want to use the driver.

8.3. CONFIGURE THE DSN FOR LINUX INSTALLATION

- Edit the **/opt/redhat/jboss-dv/v6/psqlodbc/etc/odbc.ini** file and update it with the correct username, password, and database. The database name is the VDB name.

ODBC is enabled in JBoss Data Virtualization on port **35432** by default.

8.4. INSTALL THE ODBC DRIVER ON MICROSOFT WINDOWS

Prerequisites

- Administrative permissions are required.

Procedure 8.1. Install the ODBC Driver on Microsoft Windows

1. Download the correct ODBC driver package (**jboss-dv-psqlodbc-[version]-X.zip**) from <https://access.redhat.com/jbossnetwork/>.
2. Unzip the installation package.
3. Double-click the **jboss-dv-psqlodbc-[version]-X.msi** file to start the installer.
4. The installer wizard is displayed. Click **Next**.

5. The End-User License Agreement will be displayed. Click **I accept the terms in the License Agreement** if you accept the licensing terms and then click **Next**.
6. If you want to install in a different directory other than the default directory shown, click the **Browse** button and select a directory. Click **Next**.
7. You are presented with a confirmation screen. Review the choices you have made and click **Next** to begin installation.
8. Click **Finish**.



NOTE

Installation packages for different operating systems can be downloaded from <http://access.redhat.com>.

8.5. CONFIGURE THE DSN FOR WINDOWS INSTALLATION

Procedure 8.2. Configure the DSN for Windows Installation

1. Set the ODBC driver basic options.
2. Set the ODBC driver datasource options.
3. Set the ODBC driver global options.

8.6. CONFIGURE ODBC OPTIONS ON MICROSOFT WINDOWS

Prerequisites

- You must have logged into the workstation with administrative rights.
- You must have used the Control Panel's Data Sources (ODBC) applet to add a new data source name.

Each data source name you configure can only access one VDB within a Teiid System. To make more than one VDB available, you need to configure more than one data source name.

1. From the **Start** menu, select **Settings -> Control Panel**.
2. The **Control Panel** displays. Double-click **Administrative Tools**.
3. Double-click **Data Sources (ODBC)**.
4. The **ODBC Data Source Administrator** applet displays. Click the tab associated with the type of DSN you want to add.
5. The Create New Data Source dialog box displays. In the **Select a driver for which you want to set up a data source** table, select **PostgreSQL Unicode**.
6. Click **Finish**.
7. In the **Data Source Name** edit box, type the name you want to assign to this data source.

8. In the **Database** edit box, type the name of the virtual database you want to access through this data source.
9. In the **Server** edit box, type the host name or IP address of your Teiid runtime.

**NOTE**

If you are connecting via a firewall or NAT address, you must enter either the firewall address or the NAT address.

10. In the **Port** edit box, type the port number to which the system listens for ODBC requests. (By default, Red Hat JBoss Data Virtualization listens for ODBC requests on port 35432.)
11. In the **User Name** and **Password** edit boxes, supply the user name and password for the Teiid runtime access.
12. Leave SSL Mode to disabled. (SSL connections are unsupported at present.)
13. Provide any description about the data source in the **Description** field.
14. Click on the **Datasource** button and configure the options. Tick **Parse Statements**, **Recognize Unique Indexes**, **Maximum Text as LongVarChar** and **Bool as Char** and set **MaxVarChar** to 255, **Max LongVarChar** to 8190, **Cache Size** to 100 and **SysTable Prefixes** to dd_.

On the second page, click **LF, Server side prepare, default, 7.4+** and set the **Extra Opts** to 0x0.

15. Click **Save**.

You can optionally click **Test** to validate your connection if Red Hat JBoss Data Virtualization is running.

Table 8.1. Primary ODBC Settings for Red Hat JBoss Data Virtualization

Name	Description
Updateable Cursors and Row Versioning	Should not be used.
Use serverside prepare and Parse Statements and Disallow Premature	It is recommended that Use serverside prepare is enabled and Parse Statements/Disallow Premature are disabled.
SSL mode	See Security Guide https://access.redhat.com/documentation/en/red-hat-jboss-data-virtualization/6.4/paged/security-guide/
Use Declare/Fetch cursors and Fetch Max Count	Should be used to better manage resources when large result sets are used.

Logging/debug settings can be utilized as needed.

Settings that manipulate datatypes, metadata, or optimizations such as **Show SystemTables**, **True is -1**, **Backend genetic optimizer**, **Bytea as LongVarBinary**, **Bools as Char** are ignored by the server and have no client side effect.

Any other setting that does have a client-side effect, such as **LF to CR/LF conversion**, may be used if desired but there is currently no server-side usage of the setting.

8.7. INSTALL THE ODBC DRIVER ON SOLARIS

Prerequisites

- Administrative permissions are required.
- 1. Download the correct ODBC driver package (**jboss-dv-psqlodbc-[VERSION]-X.zip**) from <https://access.redhat.com/jbossnetwork/>.
- 2. **Unzip the installation package**
Unzip the installation package to **/opt** directory.

- 3. Set the PATH property so that the ODBC binaries are used from the directory where you have unzipped the driver.

```
$ export PATH = $PATH:/opt/redhat/jboss-dv/v6/psqlodbc/bin
```

- 4. Set the ODBCINI environmental variable to match the existing odbc.ini.

```
$ export ODBCINI=/opt/redhat/jboss-dv/v6/psqlodbc/etc/odbc.ini
```



NOTE

If you are using the Bourne Shell as your Solaris terminal, you can add the two export commands above to your `~/.profile` file, so that you do not need to run them every time. Likewise, if you are using Bash, save them in your `~/.bash_profile` file instead.

8.8. CONFIGURE THE DSN FOR SOLARIS INSTALLATION

Edit the `/opt/redhat/jboss-dv/v6/psqlodbc/etc/odbc.ini` file and update it with the correct username, password, and database. The database name is the VDB name.

ODBC is enabled in JBoss Data Virtualization on port **35432** by default.

8.9. CONFIGURE ODBC OPTIONS ON RED HAT ENTERPRISE LINUX

Procedure 8.3. Configure ODBC Options on Red Hat Enterprise Linux

1. Install the driver manager:

```
yum install unixODBC
```

2. Verify that your PostgreSQLdriver has installed correctly:

```
odbcinst -q -d
```

3. To create the DSN, open the configuration file in a text editor:

```
sudo vi /opt/redhat/odbc.ini
```

**NOTE**

You must either use `sudo` or be logged in as root to open this file.

4. Add the following configuration settings to the file:

```
[<DSN name>] Driver = /usr/lib/psqlodbc.so
Description = PostgreSQL Data Source
Servername = <Teiid Host name or ip>
Port = 35432 Protocol = 7.4
UserName = <user-name>
Password = <password>
Database = <vdb-name>
ReadOnly = no
ServerType = Postgres
ConnSettings = UseServerSidePrepare=1
ByteaAsLongVarBinary=1
Optimizer=0
Ksqo=0 Trace =
No TraceFile = /var/log/trace.log
Debug = No DebugFile = /var/log/debug.log
```

5. Save the file and exit the text editor.
6. Run this command to test the DSN:

```
isql <DSN-name> [<user-name> <password>] < commands.sql
```

To connect without DSN, use this DSN-less connection string:

```
ODBC;DRIVER={PostgreSQL};DATABASE=<vdb-name>;SERVER=<host -
name>;PORT=<port>;Uid=<username>;Pwd=<password>
```

If you run `isql` but you encounter an error whereby you see this message: "Can't open lib '/opt/redhat/jboss-dv/v6/psqlodbc/lib64/psqlodbc.so' : file not found" it means that some of the postgres libraries are missing.

To fix this issue, run this command as root: **yum install postgres**

To verify that the packages are now installed, run this command: **rpm -qa|grep post**

You should see the `postgresql` and `postgresql-jdbc` packages listed.

8.10. DSN-LESS CONNECTION

You can also connect to a Red Hat JBoss Data Virtualization VDB using ODBC without explicitly

creating a DSN. However, in these scenarios your application needs a DSN-less connection string. You may want to do this if you are working with multiple computers and do not want to keep distributing the ODBC data source name.

Here is the string for Linux, UNIX, and similar operating systems:

```
ODBC;DRIVER={PostgreSQL};DATABASE=<vdb-name>;SERVER=<host-name>;PORT=
<port>;Uid=<username>;Pwd=<password>;c4=0;c8=1;
```

Here is the string for Windows:

```
ODBC;DRIVER={PostgreSQL Unicode};DATABASE=<vdb-name>;SERVER=<host -
name>;PORT=<port>;Uid=<username>;Pwd=<password>;c4=0;c8=1;
```

CHAPTER 9. RUNNING IN CLOUD ENVIRONMENTS

9.1. RUN RED HAT JBOSS DATA VIRTUALIZATION IN AN AMAZON AWS CLOUD INSTANCE

Procedure 9.1. Running Red Hat JBoss Data Virtualization in an Amazon Cloud

1. Open ports by updating the security group. (At a minimum, you will need to open the TCP, HTTP and SSH ports.)
2. To start the server, add the following parameters to bind the management and host ports: - **Djboss.bind.address.management=0.0.0.0** and **-b 0.0.0.0**



NOTE

-b is a shortcut for -Djboss.bind.address

Here is an example:

```
./standalone.sh -Djboss.bind.address.management=0.0.0.0 -b 0.0.0.0
```

3. To access the AWS instance from Teiid Designer, go to the JBDS preferences and select **General -> Network Connections SSH2**.

Next, under the **Key Management** tab, click **Load Existing Key** to add the key generated by Amazon.

4. To create a server connection, on the **Server Configuration Overview Panel**, under **Server Behavior**, select **Remote System Deployment**. Also ensure you check **Server is externally managed...**

Click the **New Host** button, select the **SSH Only** option and click **Next**.

Set the **Host Name** to match the Amazon public IP address and make the connection name the same.

Click **Finish**.

5. Open the **Remote Systems** tab.

Right mouse click the new connection and click **connect**. Fill in the **User ID**. (You do not need to provide a password if your SSH key is configured.)

6. Go back to the server configuration overview panel and confirm that the **Host** drop-down has selected the new host that you have created.
7. Start the server. (This switches the state of the server you already started.)

9.2. RUN RED HAT JBOSS DATA VIRTUALIZATION IN A GOOGLE COMPUTE INSTANCE

Procedure 9.2. Run Red Hat JBoss Data Virtualization in a Google Compute Instance

1. Open the necessary ports: **Google Developers Console -> Compute -> Compute Engine -> VM Instance -> [name of your instance] -> Network.**
2. Upload your public SSH key: **Google Developers Console - Compute -> Compute Engine -> VM Instance -> [name of your instance] -> SSH Keys.**
3. Bind the management ports (`jboss.bind.address.management`) to an external interface. (The default value for management ports is `localhost`.)

9.3. RUN RED HAT JBOSS DATA VIRTUALIZATION ON AN AZURE INSTANCE

1. Go here to learn how to create an Azure virtual machine: <https://docs.microsoft.com/en-us/azure/virtual-machines/linux/tutorial-manage-vm>.
2. Connect to the Azure virtual machine using SSH.
3. Copy the Red Hat JBoss Data Virtualization installer to that machine over SSH.
4. Run the installer as per normal:

```
java -jar jboss-dv-VERSION-installer.jar
```

5. To start the server, add the following parameters to bind the management and host ports:

```
-Djboss.bind.address.management=0.0.0.0
```

```
-b 0.0.0.0
```

Here is an example:

```
./standalone.sh -Djboss.bind.address.management=0.0.0.0 -b 0.0.0.0
```

APPENDIX A. REVISION HISTORY

Revision 6.40-22 Updates for 6.4.	Wed May 10 2017	David Le Sage
---	------------------------	----------------------