



# QuartzDesk Web Application Installation and Upgrade Guide for RedHat JBoss AS 6.1.0 and 7.x Community, 6.x and 7.x EAP

QuartzDesk Version: 3.x

January 21, 2019



## Table of Contents

<b>1.</b>	<b>PURPOSE .....</b>	<b>3</b>
<b>2.</b>	<b>DEFINITIONS .....</b>	<b>4</b>
<b>3.</b>	<b>REQUIREMENTS.....</b>	<b>5</b>
3.1	SOFTWARE REQUIREMENTS .....	5
3.1.1	Browser .....	5
3.1.2	Operating System .....	5
3.1.3	Java .....	5
3.1.4	Application Server.....	5
3.1.5	Database .....	5
3.1.6	Database JDBC Driver .....	6
3.1.7	QuartzDesk Web Application Archive.....	6
3.2	HARDWARE REQUIREMENTS.....	6
<b>4.</b>	<b>INSTALLATION.....</b>	<b>7</b>
4.1	DATABASE.....	7
4.2	JDBC DRIVER .....	7
4.3	JDBC DATASOURCE.....	7
4.3.1	DB2 .....	9
4.3.2	H2.....	11
4.3.3	Microsoft SQL Server.....	13
4.3.4	MySQL.....	15
4.3.5	Oracle.....	17
4.3.6	PostgreSQL.....	19
4.4	TEST JDBC DATASOURCE.....	21
4.5	APPLICATION WORK DIRECTORY .....	22
4.6	APPLICATION CONFIGURATION.....	23
4.7	DEPLOY APPLICATION.....	23
4.8	START APPLICATION .....	24
<b>5.</b>	<b>UPGRADING .....</b>	<b>26</b>
5.1	STOP EXISTING APPLICATION .....	26
5.2	BACKUP .....	27
5.3	REMOVE EXISTING APPLICATION .....	28
5.4	DEPLOY NEW APPLICATION.....	28
5.5	START NEW APPLICATION .....	28
<b>6.</b>	<b>QUARTZDESK 2.X TO 3.X MIGRATION NOTES .....</b>	<b>29</b>
6.1	MINIMUM REQUIRED JAVA VERSION.....	29
6.2	RENAME CONFIGURATION FILE .....	29
6.3	RENAME LOG FILES .....	29
6.4	ACCESS TO MONITORING URLS (REST API) .....	30
6.5	ACCESS TO JAX-WS ENDPOINTS .....	31
6.6	REMOVE UNUSED SECURITY DOMAIN.....	32
<b>7.</b>	<b>CLUSTER DEPLOYMENT NOTES.....</b>	<b>33</b>
7.1	HTTP SESSION REPLICATION AND AFFINITY .....	33
7.2	SHARED WORK DIRECTORY .....	33
7.3	LOGGING CONFIGURATION .....	33
7.3.1	Using Shared Log Files.....	34
7.3.2	Using Separate Log Files.....	35
7.4	INTERNAL QUARTZ SCHEDULER .....	36

## 1. Purpose

This document describes the installation and upgrade process for the QuartzDesk web application 3.x on RedHat JBoss Application Server 6.1.0 and 7.x Community, 6.x and 7.x EAP running in the **standalone** mode.

The installation and upgrade process in the **domain** mode is similar and is not described herein.

If you experience any problems installing or upgrading the QuartzDesk web application, please let us know at [support@quartzdesk.com](mailto:support@quartzdesk.com).



## 2. Definitions

The following table lists all acronyms and shortcuts used throughout this document.

Acronym / Shortcut	Definition
AS	Application Server.
EAR	Enterprise Application Archive. A file with <code>.ear</code> extension.
JAR	Java Application Archive. A file with <code>.jar</code> extension.
JVM	Java Virtual Machine.
JAC	JBoss Administrative Console.
JAS	JBoss Application Server.
WAR	Web Application Archive. A file with <code>.war</code> extension.

The following table lists all locations and properties used throughout this document.

Location / Property	Example	Description
DB_HOST	localhost	QuartzDesk web application database server host.
DB_PORT	5432	QuartzDesk web application database server port.
DB_NAME	quartzdesk	QuartzDesk web application database name.
DB_SCHEMA	quartzdesk	QuartzDesk web application database schema.
DB_USER	quartzdesk	QuartzDesk web application database user.
DB_PASSWORD	quartzdesk	QuartzDesk web application database user password.
JAS_INSTALL_ROOT	<code>/usr/local/jboss</code>	JBoss Application Server installation directory.
JAS_CONFIG	standalone	JBoss Application Server configuration.
JAS_HTTP_HOST	localhost	JBoss HTTP listener host.
JAS_HTTP_PORT	8080	JBoss HTTP listener port.
WORK_DIR	<code>/var/quartzdesk-web.work</code>	QuartzDesk web application work directory.

## 3. Requirements

### 3.1 Software Requirements

#### 3.1.1 Browser

The QuartzDesk web application GUI requires a modern JavaScript-enabled browser. Please make sure JavaScript is enabled and not blocked by third party anti-virus/anti-malware software.

The QuartzDesk web application has been tested with the following browser versions. These are also the minimum browsers versions required.

Browser	Minimum Version
Chrome	17
FireFox	10
Internet Explorer	8
Opera	12
Safari	6

#### 3.1.2 Operating System

Windows XP, Windows Vista, Windows 7, Windows 8, Windows 10.

Linux (any distribution) with kernel 2.6.x and above.

Solaris 11.x and above.

#### 3.1.3 Java

Sun/Oracle Java (JDK) 7, 8, 9, 10.

IBM Java (JDK) 7, 8, 9.

OpenJDK 7, 8, 9, 10.

#### 3.1.4 Application Server

JBoss Application Server 6.1.0 Community.

JBoss Application Server 7.x Community.

JBoss Application Server 6.x EAP.

JBoss Application Server 7.x EAP.

#### 3.1.5 Database

Database	Minimum Version
DB2	10.1
H2	1.3.174
Microsoft SQL Server	2008 R2 SP1
MySQL	5.6.4
Oracle	10.2 (10g R2)
PostgreSQL	8.1

### 3.1.6 Database JDBC Driver

Database	JDBC Driver
<b>DB2</b>	IBM DB2 JDBC 4.0 driver available at <a href="http://www-01.ibm.com/support/docview.wss?uid=swg21363866">http://www-01.ibm.com/support/docview.wss?uid=swg21363866</a> .
<b>H2</b>	Database engine including the JDBC driver is available at <a href="http://www.h2database.com">http://www.h2database.com</a> .
<b>Microsoft SQL Server</b>	Microsoft JDBC driver 4.0 for SQL Server available at <a href="http://msdn.microsoft.com/en-us/sqlserver/aa937724.aspx">http://msdn.microsoft.com/en-us/sqlserver/aa937724.aspx</a> .  We strongly advise against using the alternative JTDS JDBC driver because it does not support the datetime2 data type at this time. As a result, all datetime values written by the QuartzDesk web application would end up rounded up, or down. For datetime data type rounding details, please refer to <a href="http://msdn.microsoft.com/en-us/library/ms187819.aspx">http://msdn.microsoft.com/en-us/library/ms187819.aspx</a> .
<b>MySQL</b>	Connector/J JDBC driver available at <a href="http://dev.mysql.com/downloads/connector/j/">http://dev.mysql.com/downloads/connector/j/</a> .
<b>Oracle</b>	Oracle JDBC driver available at <a href="http://www.oracle.com/technetwork/database/features/jdbc/index-091264.html">http://www.oracle.com/technetwork/database/features/jdbc/index-091264.html</a> .  For a comprehensive overview of JDBC driver versions vs. supported database versions, please refer to <a href="http://www.oracle.com/technetwork/database/enterprise-edition/jdbc-faq-090281.html#02_02">http://www.oracle.com/technetwork/database/enterprise-edition/jdbc-faq-090281.html#02_02</a> .
<b>PostgreSQL</b>	JDBC4 PostgreSQL driver available at <a href="http://jdbc.postgresql.org/">http://jdbc.postgresql.org/</a> .

### 3.1.7 QuartzDesk Web Application Archive

To install QuartzDesk web application, you need to obtain the quartzdesk-web-x.y.z.war file. The latest version can be downloaded at [www.quartzdesk.com](http://www.quartzdesk.com) (click Downloads → Latest Release → View files → quartzdesk-web-x.y.z.war).

## 3.2 Hardware Requirements

QuartzDesk web application runs on any physical or virtualized hardware that supports the above software requirements.

## 4. Installation

This chapter describes the standard QuartzDesk installation. If you are only evaluating QuartzDesk, you may be interested in the **one-step installation mode** to dramatically reduce the number of required installation steps. For details, please refer to our [FAQs](#) (search for "one-step installation").

### 4.1 Database

Create a new database user named `quartzdesk` (`DB_USER`) with an arbitrary password (`DB_PASSWORD`).

Create a new QuartzDesk web application database named `quartzdesk1` (`DB_NAME`) owned by `DB_USER`.

In the `quartzdesk` database create a new schema named `quartzdesk` (`DB_SCHEMA`). The schema must be owned by `DB_USER`. Make the created `DB_SCHEMA` the default schema of `DB_USER` and/or add the schema to the `DB_USER`'s schema search path.

Please contact your DBA, or refer to the database engine documentation for instructions on how to complete the above database-specific tasks.



Please note that you do not have to create any database objects (tables, keys, indices etc.) in the `quartzdesk` database / schema. These objects will be automatically created by the QuartzDesk web application during its first start.

### 4.2 JDBC Driver

Download and install the JDBC driver for the created database. For a list of supported JDBC drivers please refer to chapter 3.1.6.

Copy the JDBC driver JAR file(s) to `JAS_INSTALL_ROOT/JAS_CONFIG/deployments` directory and restart the application server.



To install the H2 JDBC driver, do not copy it to `JAS_INSTALL_ROOT/JAS_CONFIG/deployments` directory because the H2 driver is already installed as a module (`JAS_INSTALL_ROOT/modules/.../com/h2database/h2`). To update the existing driver module JAR, copy the new H2 driver JAR to the module's main directory and edit the module descriptor file `module.xml` to update the JAR name.

### 4.3 JDBC Datasource

In JAC go to:

**JBoss 7.x Community:** Profile → Connector → Datasources

<sup>1</sup> DB2 restricts the database name length to the maximum of 8 characters. Please adjust the database name accordingly (e.g. `qdesk`).

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → Subsystems → Connector → Datasources

**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → Subsystems → Connector → Datasources

Click the Add button to create a new JDBC datasource.

**JBoss 7.x EAP:** Configuration → Subsystems → Datasources → Non-XA

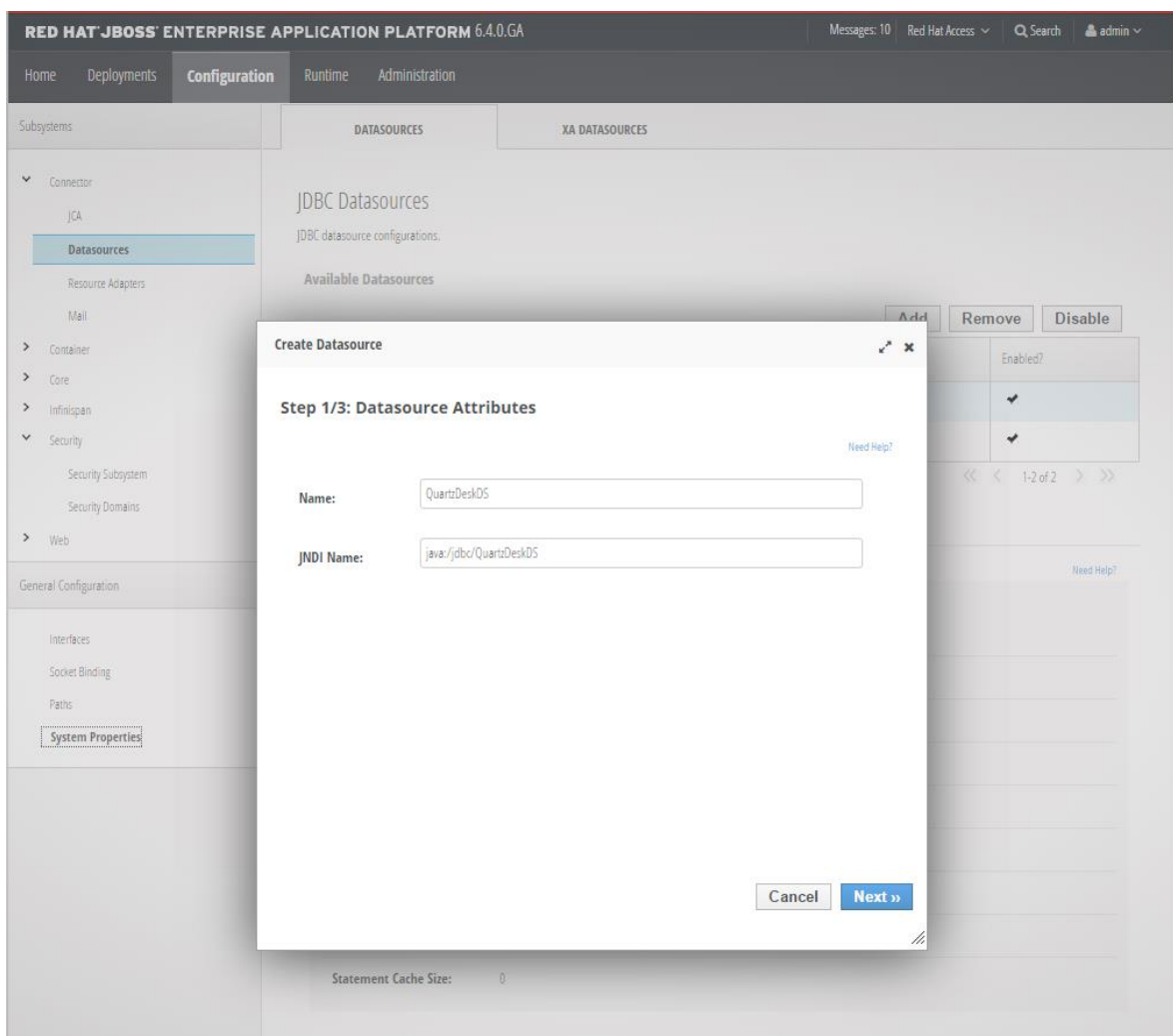
Click the Add button to create a new JDBC datasource.

Select the new JDBC datasource type. Click Next.

Enter these datasource names:

Name: QuartzDeskDS

JNDI Name: java:/jdbc/QuartzDeskDS



Click Next.



The following steps depend on the QuartzDesk database type and are described in the following sub-chapters.

### 4.3.1 DB2

In Step 2, select the DB2 JDBC driver. Click Next.

In Step 3, enter the following values:

Connection URL: jdbc:db2://DB\_HOST:DB\_PORT/DB\_NAME

Username: DB\_USER

Password: DB\_PASSWORD

Security Domain: leave empty

**JBoss 7.x Community, JBoss 6.x EAP:** Click Done.

**JBoss 7.x EAP:** Click Next and Finish.

In JAC go to:

**JBoss 7.x Community:** Profile → Connector → Datasources

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → Subsystems → Connector → Datasources

**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → Subsystems → Connector → Datasources

Select the QuartzDeskDS datasource.

**JBoss 7.x EAP:** Configuration → Subsystems → Datasources → Non-XA

Select the QuartzDeskDS datasource. Click View.

Modify the datasource configuration under the following tabs:

#### Attributes / Statements

Click the Edit button/link and enter the following values:

Statement Cache Size: 100

Click Save.

#### Properties

Add the following properties:

Key: clientApplicationInformation

Value: QuartzDesk

#### Pool

Click the Edit button/link and enter the following values:

Min Pool Size: 2  
Max Pool Size: 10  
Prefill enabled / Prefill: check

Click Save.

#### Validation

Click the Edit button/link and enter the following values:

Valid Connection Checker:

`org.jboss.jca.adapters.jdbc.extensions.db2.DB2ValidConnectionChecker`

State Connection Checker:

`org.jboss.jca.adapters.jdbc.extensions.db2.DB2StaleConnectionChecker`

Exception Sorter:

`org.jboss.jca.adapters.jdbc.extensions.db2.DB2ExceptionSorter`

Click Save.

In JAC go to:

**JBoss 7.x Community:** Profile → Connector → Datasources

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → Subsystems → Connector → Datasources

**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → Subsystems → Connector → Datasources

Select the QuartzDeskDS datasource. Click the Enable button.

**JBoss 7.x EAP:** Configuration → Subsystems → Datasources → Non-XA

Select the QuartzDeskDS datasource. Open the datasource menu and click Enable.

Click the Confirm button in the confirmation dialog.

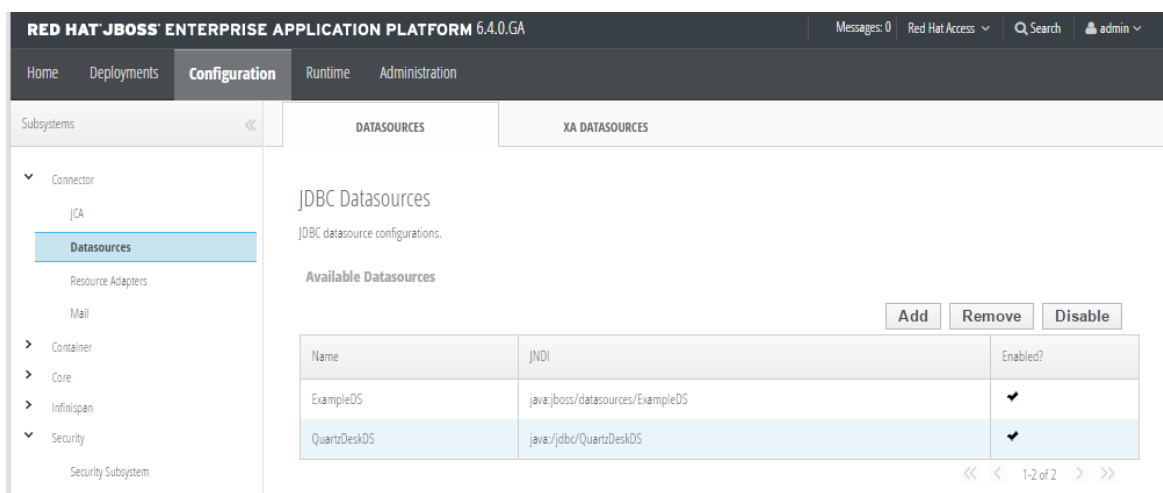
The QuartzDeskDS datasource should now be enabled.

## JDBC Datasources

JDBC datasource configurations.

### Available Datasources

Name	JNDI	Enabled?
ExampleDS	java:jboss/datasources/ExampleDS	<input checked="" type="checkbox"/>
QuartzDeskDS	java:/jdbc/QuartzDeskDS	<input checked="" type="checkbox"/>



RED HAT JBOSS ENTERPRISE APPLICATION PLATFORM 6.4.0.GA

Messages: 0 Red Hat Access Search admin

Home Deployments **Configuration** Runtime Administration

Subsystems << DATASOURCES XA DATASOURCES

JDBC Datasources  
JDBC datasource configurations.

Available Datasources

Name	JNDI	Enabled?
ExampleDS	java:jboss/datasources/ExampleDS	<input checked="" type="checkbox"/>
QuartzDeskDS	java:/jdbc/QuartzDeskDS	<input checked="" type="checkbox"/>

### 4.3.2 H2



We recommend using H2 for evaluation and/or experimental purposes only. We strongly discourage using H2 in production environments.

In Step 2, select the H2 JDBC driver. Click Next.

In Step 3, enter the following values:

Connection URL: jdbc:h2:file:<H2\_DB\_FILE\_PATH>

Username: DB\_USER

Password: DB\_PASSWORD

Security Domain: leave empty

Please note that H2 can be configured to run in various operating modes by adjusting the Connection URL value. For details, please refer to the H2 documentation at

[http://www.h2database.com/html/features.html#database\\_url](http://www.h2database.com/html/features.html#database_url).

**JBoss 7.x Community, JBoss 6.x EAP:** Click Done.

**JBoss 7.x EAP:** Click Next and Finish.

In JAC go to:

**JBoss 7.x Community:** Profile → Connector → Datasources

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → Subsystems → Connector → Datasources

**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → Subsystems → Connector → Datasources

Select the QuartzDeskDS datasource.

**JBoss 7.x EAP:** Configuration → Subsystems → Datasources → Non-XA

Select the QuartzDeskDS datasource. Click View.

Modify the datasource configuration under the following tabs:

### Attributes / Statements

Click the Edit button/link and enter the following values:

Statement Cache Size: 100

Click Save.

### Properties

Add the following properties:

Key: applicationName

Value: QuartzDesk

### Pool

Click the Edit button/link and enter the following values:

Min Pool Size: 2

Max Pool Size: 10

Prefill enabled / Prefill: check

Click Save.

### Validation

Click the Edit button/link and enter the following values:

Valid Connection Checker:

org.jboss.jca.adapters.jdbc.extensions.novendor.JDBC4ValidConnectionChecker

Exception Sorter:

org.jboss.jca.adapters.jdbc.extensions.novendor.NullExceptionSorter

Click Save.

In JAC go to:

**JBoss 7.x Community:** Profile → Connector → Datasources

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → Subsystems → Connector → Datasources

**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → Subsystems → Connector → Datasources

Select the QuartzDeskDS datasource. Click the Enable button.

**JBoss 7.x EAP:** Configuration → Subsystems → Datasources → Non-XA

Select the QuartzDeskDS datasource. Open the datasource menu and click Enable.

Click the Confirm button in the confirmation dialog.

The QuartzDeskDS datasource should now be enabled.

### JDBC Datasources

JDBC datasource configurations.

#### Available Datasources

Name	JNDI	Enabled?
ExampleDS	java:jboss/datasources/ExampleDS	<input checked="" type="checkbox"/>
QuartzDeskDS	java:/jdbc/QuartzDeskDS	<input checked="" type="checkbox"/>

Add Remove Disable  
<< < 1-2 of 2 > >>

### 4.3.3 Microsoft SQL Server

In Step 2, select the Microsoft SQL Server JDBC driver. Click Next.

In Step 3, enter the following values:

Connection URL: jdbc:sqlserver://DB\_HOST:DB\_PORT;databaseName=DB\_NAME

Username: DB\_USER

Password: DB\_PASSWORD

Security Domain: leave empty

**JBoss 7.x Community, JBoss 6.x EAP:** Click Done.

**JBoss 7.x EAP:** Click Next and Finish.

In JAC go to:

**JBoss 7.x Community:** Profile → Connector → Datasources

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → Subsystems → Connector → Datasources

**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → Subsystems → Connector → Datasources

Select the QuartzDeskDS datasource.

**JBoss 7.x EAP:** Configuration → Subsystems → Datasources → Non-XA

Select the QuartzDeskDS datasource. Click View.

Modify the datasource configuration under the following tabs:

#### **Attributes / Statements**

Click the Edit button/link and enter the following values:

Statement Cache Size: 100

Click Save.

#### **Properties**

Add the following properties:

Key: applicationName

Value: QuartzDesk

#### **Pool**

Click the Edit button/link and enter the following values:

Min Pool Size: 2

Max Pool Size: 10

Prefill enabled / Prefill: check

Click Save.

#### **Validation**

Click the Edit button/link and enter the following values:

Valid Connection Checker:

org.jboss.jca.adapters.jdbc.extensions.mssql.MSSQLValidConnectionChecker

Click Save.

In JAC go to:

**JBoss 7.x Community:** Profile → Connector → Datasources

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → Subsystems → Connector → Datasources

**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → Subsystems → Connector → Datasources

Select the QuartzDeskDS datasource. Click the Enable button.

**JBoss 7.x EAP:** Configuration → Subsystems → Datasources → Non-XA

Select the QuartzDeskDS datasource. Open the datasource menu and click Enable.

Click the Confirm button in the confirmation dialog.

The QuartzDeskDS datasource should now be enabled.

## JDBC Datasources

JDBC datasource configurations.

### Available Datasources

Name	JNDI	Enabled?
ExampleDS	java:jboss/datasources/ExampleDS	<input checked="" type="checkbox"/>
QuartzDeskDS	java:jdbc/QuartzDeskDS	<input checked="" type="checkbox"/>

Add Remove Disable  
<< < 1-2 of 2 > >>

## 4.3.4 MySQL

In Step 2, select the PostgreSQL JDBC driver. Click Next.

In Step 3, enter the following values:

Connection URL: jdbc:mysql://DB\_HOST:DB\_PORT/DB\_NAME

Username: DB\_USER

Password: DB\_PASSWORD

Security Domain: leave empty

**JBoss 7.x Community, JBoss 6.x EAP:** Click Done.

**JBoss 7.x EAP:** Click Next and Finish.

In JAC go to:

**JBoss 7.x Community:** Profile → Connector → Datasources

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → Subsystems → Connector → Datasources

**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → Subsystems → Connector → Datasources

Select the QuartzDeskDS datasource.

**JBoss 7.x EAP:** Configuration → Subsystems → Datasources → Non-XA

Select the QuartzDeskDS datasource. Click View.

Modify the datasource configuration under the following tabs:

### Attributes / Statements

Click the Edit button/link and enter the following values:

Statement Cache Size: 100

Click Save.

### Properties

Add the following properties:

Key: cachePrepStmts

Value: true

### Pool

Click the Edit button/link and enter the following values:

Min Pool Size: 2

Max Pool Size: 10

Prefill enabled / Prefill: check

Click Save.

### Validation

Click the Edit button/link and enter the following values:

Valid Connection Checker:

org.jboss.jca.adapters.jdbc.extensions.mysql.MySQLValidConnectionChecker

Exception Sorter:

org.jboss.jca.adapters.jdbc.extensions.mysql.MySQLExceptionSorter

Click Save.

In JAC go to:

**JBoss 7.x Community:** Profile → Connector → Datasources

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → Subsystems → Connector → Datasources

**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → Subsystems → Connector → Datasources

Select the QuartzDeskDS datasource. Click the Enable button.

**JBoss 7.x EAP:** Configuration → Subsystems → Datasources → Non-XA

Select the QuartzDeskDS datasource. Open the datasource menu and click Enable.



Click the Confirm button in the confirmation dialog.

The QuartzDeskDS datasource should now be enabled.

## JDBC Datasources

JDBC datasource configurations.

### Available Datasources

Name	JNDI	Enabled?
ExampleDS	java:jboss/datasources/ExampleDS	<input checked="" type="checkbox"/>
QuartzDeskDS	java:jdbc/QuartzDeskDS	<input checked="" type="checkbox"/>

Add Remove Disable

1-2 of 2

## 4.3.5 Oracle

In Step 2, select the Oracle JDBC driver. Click Next.

In Step 3, enter the following values:

Connection URL: jdbc:oracle:thin:@DB\_HOST:DB\_PORT: ORACLE\_SERVICE\_NAME

Username: DB\_USER

Password: DB\_PASSWORD

Security Domain: leave empty

**JBoss 7.x Community, JBoss 6.x EAP:** Click Done.

**JBoss 7.x EAP:** Click Next and Finish.

In JAC go to:

**JBoss 7.x Community:** Profile → Connector → Datasources

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → Subsystems → Connector → Datasources

**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → Subsystems → Connector → Datasources

Select the QuartzDeskDS datasource.

**JBoss 7.x EAP:** Configuration → Subsystems → Datasources → Non-XA

Select the QuartzDeskDS datasource. Click View.

Modify the datasource configuration under the following tabs:

### Attributes / Statements

Click the Edit button/link and enter the following values:

Statement Cache Size: 100

Click Save.

### Properties

Add the following properties:

Key: driverType

Value: thin

Key: databaseName

Value: DB\_NAME

### Pool

Click the Edit button/link and enter the following values:

Min Pool Size: 2

Max Pool Size: 10

Prefill enabled / Prefill: check

Click Save.

### Validation

Click the Edit button/link and enter the following values:

Valid Connection Checker:

org.jboss.jca.adapters.jdbc.extensions.oracle.OracleValidConnectionChecker

State Connection Checker:

org.jboss.jca.adapters.jdbc.extensions.oracle.OracleStaleConnectionChecker

Exception Sorter:

org.jboss.jca.adapters.jdbc.extensions.oracle.OracleExceptionSorter

Click Save.

In JAC go to:

**JBoss 7.x Community:** Profile → Connector → Datasources

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → Subsystems → Connector → Datasources

**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → Subsystems → Connector → Datasources

Select the QuartzDeskDS datasource. Click the Enable button.

**JBoss 7.x EAP:** Configuration → Subsystems → Datasources → Non-XA

Select the QuartzDeskDS datasource. Open the datasource menu and click Enable.

Click the Confirm button in the confirmation dialog.

The QuartzDeskDS datasource should now be enabled.

## JDBC Datasources

JDBC datasource configurations.

### Available Datasources

Name	JNDI	Enabled?
ExampleDS	java:jboss/datasources/ExampleDS	<input checked="" type="checkbox"/>
QuartzDeskDS	java:jdbc/QuartzDeskDS	<input checked="" type="checkbox"/>

Add Remove Disable

<< < 1-2 of 2 > >>

## 4.3.6 PostgreSQL

In Step 2, select the PostgreSQL JDBC driver. Click Next.

In Step 3, enter the following values:

Connection URL: jdbc:postgresql://DB\_HOST:DB\_PORT/DB\_NAME

Username: DB\_USER

Password: DB\_PASSWORD

Security Domain: leave empty

**JBoss 7.x Community, JBoss 6.x EAP:** Click Done.

**JBoss 7.x EAP:** Click Next and Finish.

In JAC go to:

**JBoss 7.x Community:** Profile → Connector → Datasources

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → Subsystems → Connector → Datasources

**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → Subsystems → Connector → Datasources

Select the QuartzDeskDS datasource.

**JBoss 7.x EAP:** Configuration → Subsystems → Datasources → Non-XA

Select the QuartzDeskDS datasource. Click View.

Modify the datasource configuration under the following tabs:

### Attributes / Statements

Click the Edit button/link and enter the following values:

Statement Cache Size: 100

Click Save.

### Properties

Add the following properties:

Key: applicationName

Value: QuartzDesk

### Pool

Click the Edit button/link and enter the following values:

Min Pool Size: 2

Max Pool Size: 10

Prefill enabled / Prefill: check

Click Save.

### Validation

Click the Edit button/link and enter the following values:

Valid Connection Checker:

org.jboss.jca.adapters.jdbc.extensions.postgres.PostgreSQLValidConnectionChecker

Exception Sorter:

org.jboss.jca.adapters.jdbc.extensions.postgres.PostgreSQLExceptionSorter

Click Save.

In JAC go to:

**JBoss 7.x Community:** Profile → Connector → Datasources

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → Subsystems → Connector → Datasources

**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → Subsystems → Connector → Datasources

Select the QuartzDeskDS datasource. Click the Enable button.

**JBoss 7.x EAP:** Configuration → Subsystems → Datasources → Non-XA

Select the QuartzDeskDS datasource. Open the datasource menu and click Enable.

Click the Confirm button in the confirmation dialog.

The QuartzDeskDS datasource should now be enabled.

## JDBC Datasources

JDBC datasource configurations.

### Available Datasources

Name	JNDI	Enabled?
ExampleDS	java:jboss/datasources/ExampleDS	<input checked="" type="checkbox"/>
QuartzDeskDS	java:jdbc/QuartzDeskDS	<input checked="" type="checkbox"/>

Add Remove Disable  
<< < 1-2 of 2 > >>

## 4.4 Test JDBC Datasource

In JAC go to:

**JBoss 7.x Community:** Profile → Connector → Datasources

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → Subsystems → Connector → Datasources

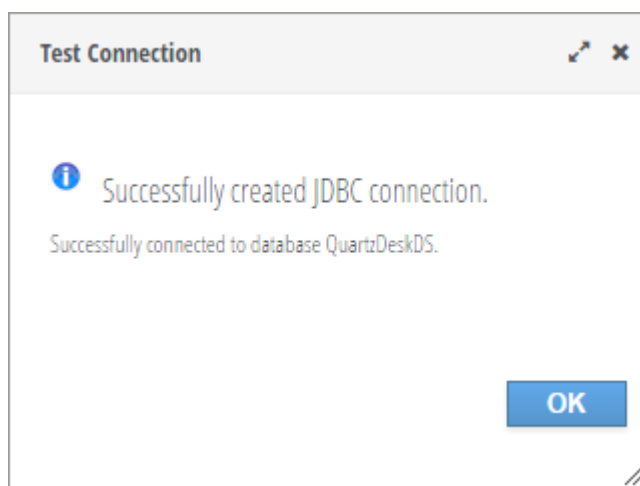
**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → Subsystems → Connector → Datasources

Select the QuartzDeskDS datasource. Open the Connection tab and click the Test Connection button.

**JBoss 7.x EAP:** Configuration → Subsystems → Datasources → Non-XA

Select the QuartzDeskDS datasource. Open the local menu and click Test Connection.

If the JDBC connection pool test is successful, a dialog with “Successfully created JDBC connection” message is displayed.



If the JDBC connection pool test fails, no dialog is displayed and an error exception is logged in the JAS log (JAS\_INSTALL\_ROOT/JAS\_CONFIG/log/server.log).

## 4.5 Application Work Directory

Create a QuartzDesk web application work directory (WORK\_DIR) anywhere on the local file system. The directory must be readable and writable by the user the JAS process is running under.

Copy your QuartzDesk license key file (license.key) to WORK\_DIR.



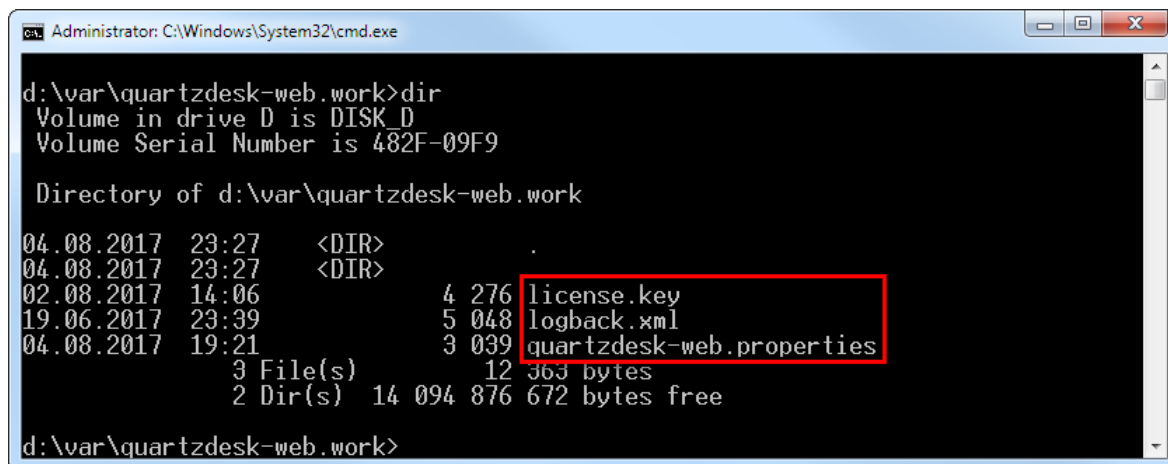
You can obtain a free 30-day trial license key at [www.quartzdesk.com](http://www.quartzdesk.com) (go to Try / Purchase > Get Trial License Key).

Open the QuartzDesk web application archive (quartzdesk-web-x.y.z.war) and copy all files from the extras/work directory into WORK\_DIR.



If you cannot open the WAR file directly, rename it to \*.zip. Do not forget to rename the file back to \*.war once you have extracted the required files.

In the following figure you can see an example of a QuartzDesk web application work directory correctly set up on a Microsoft Windows machine.



```
Administrator: C:\Windows\System32\cmd.exe
d:\var\quartzdesk-web.work>dir
Volume in drive D is DISK D
Volume Serial Number is 482F-09F9

Directory of d:\var\quartzdesk-web.work

04.08.2017  23:27    <DIR>          .
04.08.2017  23:27    <DIR>          ..
02.08.2017  14:06             4 276 license.key
19.06.2017  23:39             5 048 logback.xml
04.08.2017  19:21             3 039 quartzdesk-web.properties
               3 File(s)              12 363 bytes
               2 Dir(s)  14 094 876 672 bytes free

d:\var\quartzdesk-web.work>
```

In JAC go to:

**JBoss 7.x Community:** Profile → General Configuration → System Properties

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → General Configuration → System Properties

**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → General Configuration → System Properties

Click the Add button.

**JBoss 7.x EAP:** Configuration → System Properties

Click the View button. Click the Add button.

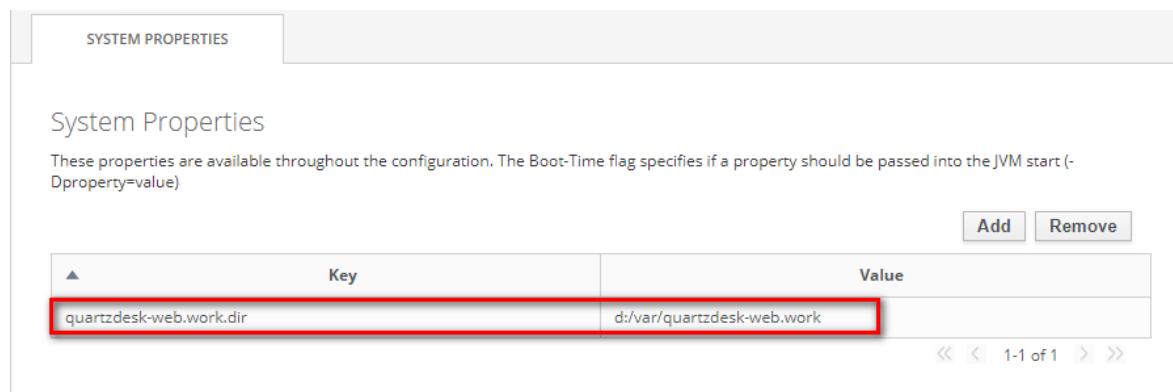
Add a new boot-time system property.

Name: quartzdesk-web.work.dir

Value: WORK\_DIR



JAC requires all backslashes in the WORK\_DIR to be escaped as two consecutive backslash characters (\\). Alternatively, use forward slash characters (/).



## 4.6 Application Configuration

Open the QuartzDesk web application configuration file `WORK_DIR/quartzdesk-web.properties`.

Based on the type and version of the database created in step 4.1, change the value of the `db.profile` configuration property according to the following table.

Database	Database Version	db.profile Value
DB2	>= 10.0	db2
H2	>= 1.3.170	h2
Microsoft SQL Server	>= 2008	mssql
MySQL (MyISAM)	>= 5.6	mysql
MySQL (InnoDB)	>= 5.6	mysql_innodb
Oracle	== 8i	oracle8
Oracle	>= 9i	oracle9
PostgreSQL	== 8.1	postgres81
PostgreSQL	>= 8.2	postgres82

Optionally, you can adjust the QuartzDesk web application logging parameters by editing the `WORK_DIR/logback.xml` configuration file. The default sample `logback.xml` configuration file makes QuartzDesk web application log under the `WORK_DIR/logs` directory that is automatically created when the web application starts. Please refer to the [Logback Manual](#) for Logback configuration details.

## 4.7 Deploy Application

In JAC go to:

**JBoss 7.x Community:** Runtime → Deployments → Manage Deployments. Click the Add Content button.

**JBoss 6.1 EAP, JBoss 6.2 EAP, JBoss 6.3 EAP:** Runtime → Server → Manage Deployments. Click the Add button.

**JBoss 6.4 EAP:** Deployments. Click the Add button.

In Step 1/2 (Deployment Selection) click on the Choose File button to select the `quartzdesk-web-x.y.z.war` file. Click Next.

In Step 2/2 (Verify Deployment Name(s)) make no changes and click Save.

**JBoss 7.x EAP:** Deployments. Click the Add button.

Select “Upload a new deployment” option and click Next.

Click on the Choose File button to select the `quartzdesk-web-x.y.z.war` file. Click Next.

In the next step (Verify Upload), uncheck the Enable option. Make no changes to the deployment names and click Finish.

## 4.8 Start Application

In JAC go to:

**JBoss 7.x Community:** Runtime → Deployments → Manage Deployments.

Click the Enable button next to the QuartzDesk web application in the Deployments list.

**JBoss 6.1 EAP, JBoss 6.2 EAP, JBoss 6.3 EAP:** Runtime → Server → Manage Deployments

**JBoss 6.4 EAP:** Deployments.

Select the QuartzDesk web application and click the Enable button. Confirm this action in the dialog window that gets opened.

**JBoss 7.x EAP:** Deployments.

In the Deployments list, open local menu for the QuartzDesk web application and choose the Enable option. Confirm this action in the dialog window that gets opened.

The application should be starting now.

Monitor the `JAS server.log` log file under `JAS_INSTALL_ROOT/JAS_CONFIG/logs` for errors and wait for the startup procedure to complete.



You can safely ignore the following warning messages:

```
16:35:37,714 WARN [org.jboss.as.server.deployment] (MSC service  
thread 1-5) JBAS015960: Class Path entry lib/<library>.jar in  
/d:/Java/jboss-eap-6.4/bin/content/quartzdesk-web-<version>.war  
does not point to a valid jar for a Class-Path reference.
```

Check the QuartzDesk web application logs (by default located in the `WORK_DIR/logs` directory) for errors.

If there are no errors, point your browser to [http://JAS\\_HTTP\\_HOST:JAS\\_HTTP\\_PORT/quartzdesk/](http://JAS_HTTP_HOST:JAS_HTTP_PORT/quartzdesk/) and verify that the QuartzDesk web application GUI is accessible.

Check the version number of the deployed QuartzDesk web application.



To log in, use the default administrator login credentials:

Username: admin  
Password: admin123

Once logged in, you can go to Settings > Users to manage users with access to the QuartzDesk web application GUI. Users can be assigned different access permissions based on their intended roles.

In Settings > Groups, you can manage groups and assign access permissions to these groups. A group can contain users (members) who inherit access permissions of the group. A user can be a member of any number of groups.

Effective access permissions of a user are permissions associated directly with the user plus access permissions of all groups the user is a member of.

## 5. Upgrading

### 5.1 Stop Existing Application

In JAC go to:

**JBoss 7.x Community:** Runtime → Deployments → Manage Deployments

Click the Disable button next to the QuartzDesk web application in the Deployments list. Confirm this action in a dialog window that opens and wait for the action to complete.

Upon successful disabling, the Enabled flag, shown next the existing QuartzDesk web application in the Deployments list, indicates that the application is no longer enabled.

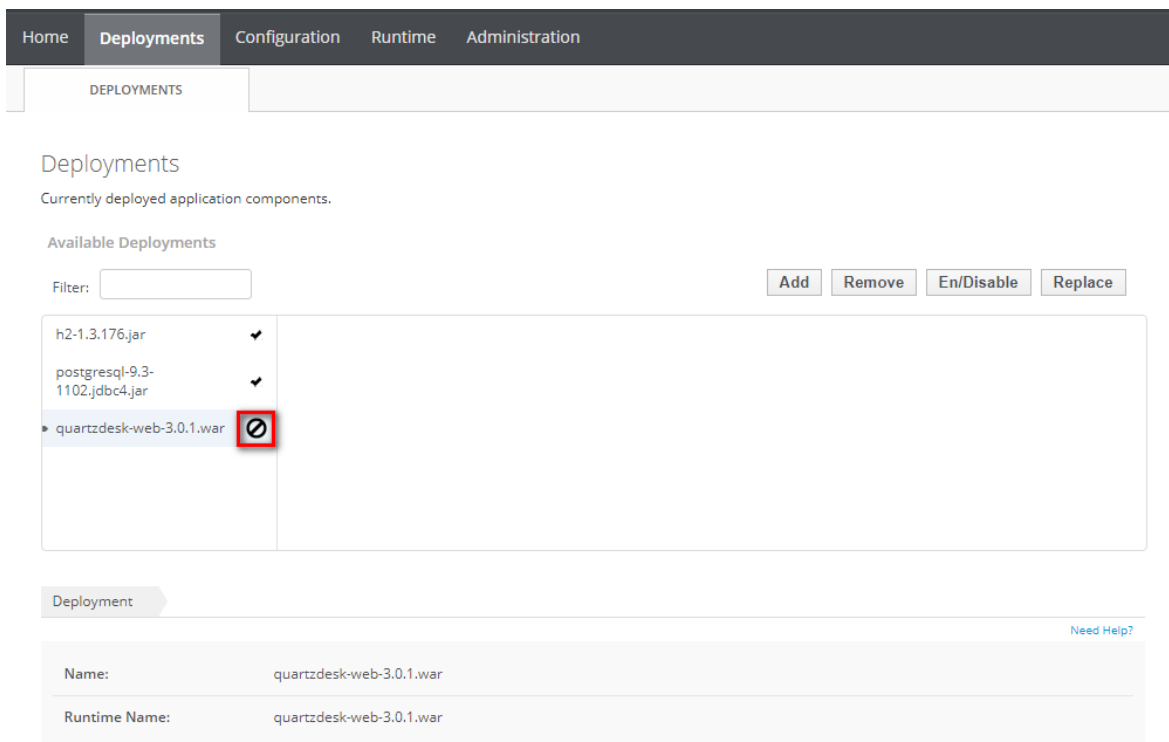


**JBoss 6.1 EAP, JBoss 6.2 EAP, JBoss 6.3 EAP:** Runtime → Server → Manage Deployments

**JBoss 6.4 EAP:** Deployments

Select the QuartzDesk web application in the Deployments list and click the En/Disable button. Confirm this action in a dialog window that opens and wait for the action to complete.

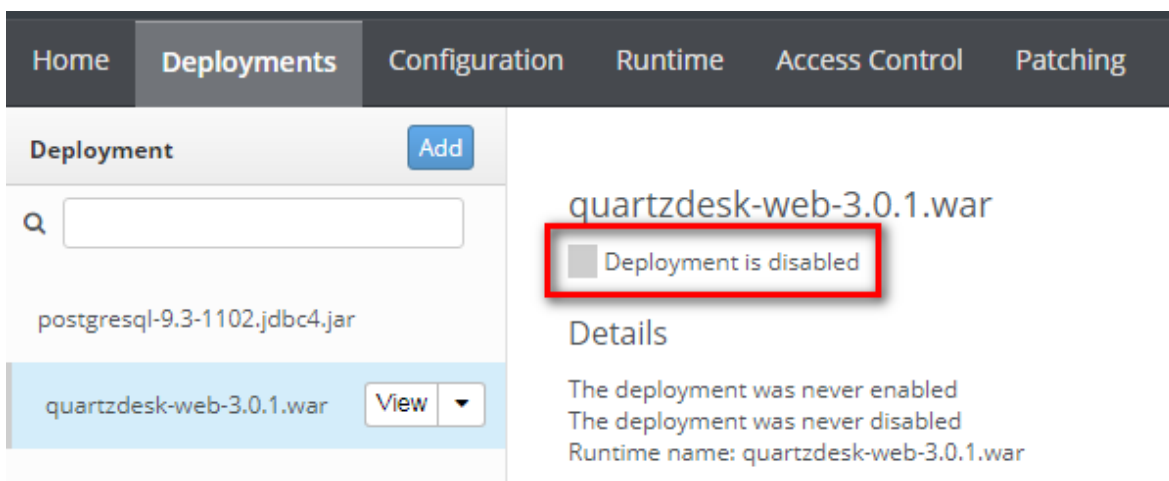
Upon successful disabling, the Enabled flag, shown next the existing QuartzDesk web application in the Deployments list, indicates that the application is no longer enabled.



## JBoss 7.x EAP: Deployments

In the Deployments list, open local menu for the QuartzDesk web application and choose the Disable option. Confirm this action in a dialog window that opens and wait for the action to complete.

Upon successful disabling, the flag, shown next the existing QuartzDesk web application in the Deployments list, indicates that the application is no longer enabled.



## 5.2 Backup

Backup your QuartzDesk database. We recommend performing a **full database backup**.

Backup the contents of the QuartzDesk work directory.

Make sure you have the WAR file of the existing QuartzDesk web application.

Store the backup files in a safe place so you can restore the original QuartzDesk web application version if the need arises.

### 5.3 Remove Existing Application

In JAC go to:

**JBoss 7.x Community:** Runtime → Deployments → Manage Deployments

Click the Remove button next to the existing QuartzDesk web application. Confirm this action in a dialog window that opens and wait for the action to complete.

Upon successful removal, the QuartzDesk web application disappears from the Deployments list.

**JBoss 6.1 EAP, JBoss 6.2 EAP, JBoss 6.3 EAP:** Runtime → Server → Manage Deployments

**JBoss 6.4 EAP:** Deployments

Select the deployed QuartzDesk web application and click the Remove button. Confirm this action in a dialog window that opens and wait for the action to complete.

**JBoss 7.x EAP:** Deployments

In the Deployments list, open the local menu for QuartzDesk web application and choose the Remove option. Confirm this action in a dialog window that opens and wait for the action to complete.

Upon successful removal, the QuartzDesk web application disappears from the Deployments list.

### 5.4 Deploy New Application

Deploy the new version of the QuartzDesk web application by following the deployment steps outlined in 4.7.



Some JAS releases contain a bug that prevents the application server from invalidating and recompiling the JSP cache of redeployed web applications. Therefore, we recommend that you to stop the WFAS instance and manually purge the QuartzDesk web application JSP cache located at `JAS_INSTALL_ROOT/JAS_CONFIG/tmp/work/jboss.web/[host]/quartzdesk`.

### 5.5 Start New Application

Start the new version of the QuartzDesk web application by following the steps outlined in 4.8.

## 6. QuartzDesk 2.x to 3.x Migration Notes

To upgrade QuartzDesk web application 2.x to 3.x, follow the upgrade steps outlined in 5.

Before deploying the new QuartzDesk web application WAR file (`quartzdesk-web-x.y.z.war`), as outlined in step 5.4, make sure you have implemented changes described in this chapter.

### 6.1 Minimum Required Java Version

QuartzDesk web application 3.x requires Java 7 or higher. Java 6 is no longer supported.

Make sure JAS is configured to use Java 7 or higher.

### 6.2 Rename Configuration File

The name of the QuartzDesk web application 3.x configuration file has changed from `quartzdesk.properties` to `quartzdesk-web.properties`.

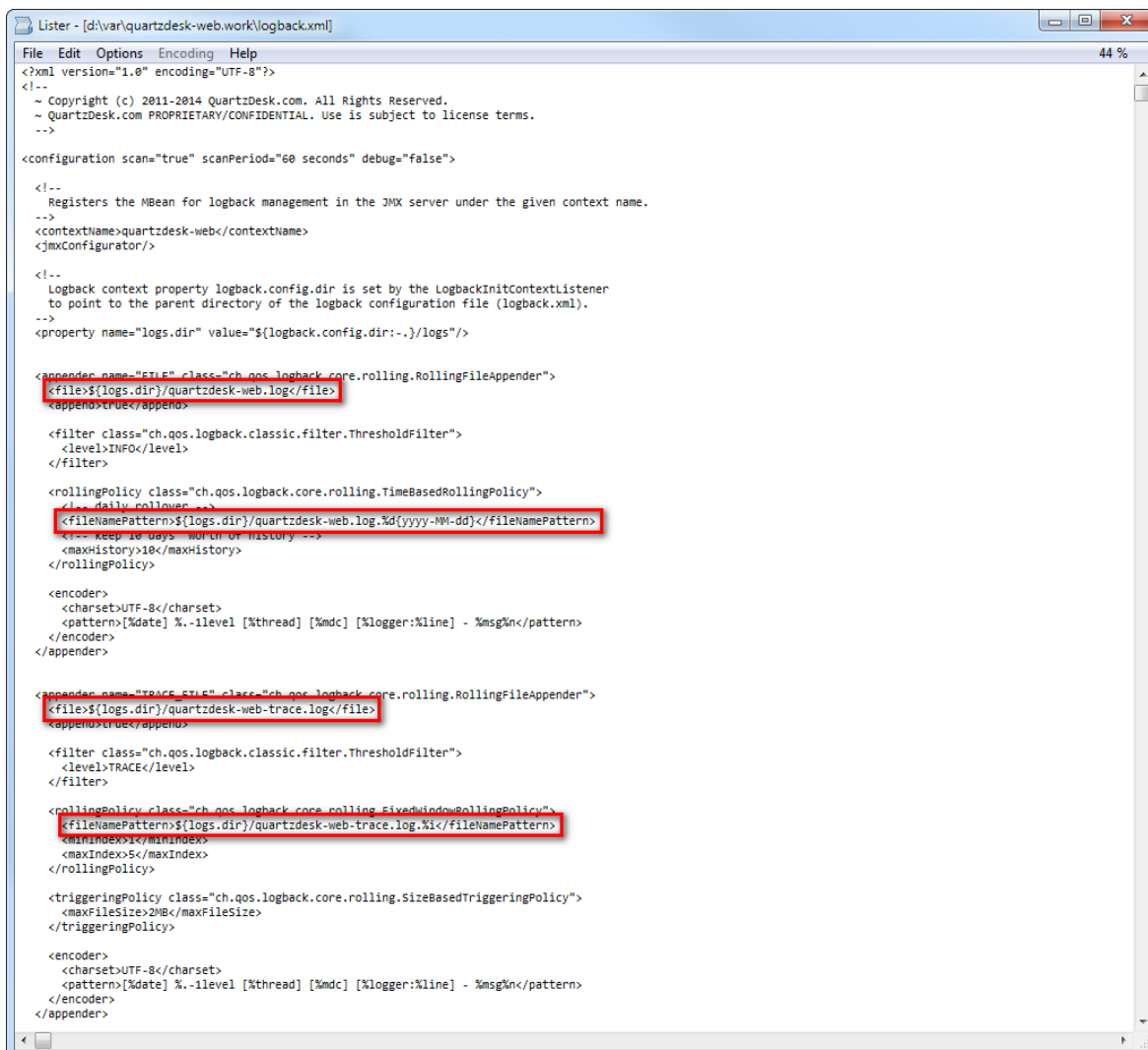
Rename the existing configuration file `quartzdesk.properties` located in the QuartzDesk web application work directory.

### 6.3 Rename Log Files

The names of QuartzDesk web application 3.x log files have changed.

Original Log File Name (2.x)	New Log File Name (3.x)
<code>quartzdesk.log</code>	<code>quartzdesk-web.log</code>
<code>quartzdesk-trace.log</code>	<code>quartzdesk-web-trace.log</code>

To use these new log file names, edit the QuartzDesk web application logging configuration file (`WORK_DIR/logback.xml`) and change the following lines:



```
File Edit Options Encoding Help
<?xml version="1.0" encoding="UTF-8"?>
<!--
~ Copyright (c) 2011-2014 QuartzDesk.com. All Rights Reserved.
~ QuartzDesk.com PROPRIETARY/CONFIDENTIAL. Use is subject to license terms.
-->

<configuration scan="true" scanPeriod="60 seconds" debug="false">

  <!--
  Registers the MBean for logback management in the JMX server under the given context name.
  -->
  <contextName>quartzdesk-web</contextName>
  <jmxConfigurator/>

  <!--
  Logback context property logback.config.dir is set by the LogbackInitContextListener
  to point to the parent directory of the logback configuration file (logback.xml).
  -->
  <property name="logs.dir" value="${logback.config.dir:-.}/logs"/>

  <appender name="FILE" class="ch.qos.logback.core.rolling.RollingFileAppender">
    <file>${logs.dir}/quartzdesk-web.log</file>
    <append>true</append>

    <filter class="ch.qos.logback.classic.filter.ThresholdFilter">
      <level>INFO</level>
    </filter>

    <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">
      <class>daily rollover -->
      <fileNamePattern>${logs.dir}/quartzdesk-web.log.%d{yyyy-MM-dd}</fileNamePattern>
      <!-- Keep 10 days worth of history -->
      <maxHistory>10</maxHistory>
    </rollingPolicy>

    <encoder>
      <charset>UTF-8</charset>
      <pattern>[%date] %.-level [%thread] [%mdc] [%logger:%line] - %msg%n</pattern>
    </encoder>
  </appender>

  <appender name="TRACE_FILE" class="ch.qos.logback.core.rolling.RollingFileAppender">
    <file>${logs.dir}/quartzdesk-web-trace.log</file>
    <append>true</append>

    <filter class="ch.qos.logback.classic.filter.ThresholdFilter">
      <level>TRACE</level>
    </filter>

    <rollingPolicy class="ch.qos.logback.core.rolling.FixedWindowRollingPolicy">
      <fileNamePattern>${logs.dir}/quartzdesk-web-trace.log.%i</fileNamePattern>
      <minIndex>1</minIndex>
      <maxIndex>5</maxIndex>
    </rollingPolicy>

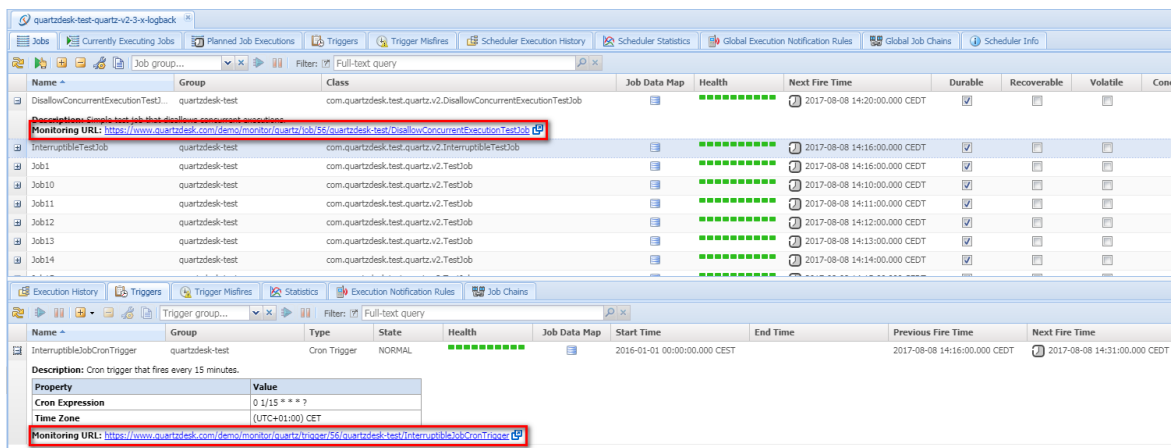
    <triggeringPolicy class="ch.qos.logback.core.rolling.SizeBasedTriggeringPolicy">
      <maxFileSize>2MB</maxFileSize>
    </triggeringPolicy>

    <encoder>
      <charset>UTF-8</charset>
      <pattern>[%date] %.-level [%thread] [%mdc] [%logger:%line] - %msg%n</pattern>
    </encoder>
  </appender>
```

Alternatively, extract the default `logback.xml` configuration file from the QuartzDesk web application 3.x WAR (`quartzdesk-web-x.y.z.war/extras/work/logback.xml`) and copy it to `WORK_DIR`.

## 6.4 Access to Monitoring URLs (REST API)

In QuartzDesk web application 2.x, the monitoring REST API URLs could be accessed by users with the QuartzDeskMonitor J2EE security role. In QuartzDesk web application 3.x, these monitoring URLs can be accessed by all authenticated users.



We recommend that you create a dedicated user account to access these monitoring URLs. The user account can be created in Settings → Users in the QuartzDesk GUI.



All monitoring URLs in QuartzDesk 3.x support the HTTP Basic authentication scheme where the user's authentication credentials are passed in the `Authorization` HTTP header. Please note that the same authentication scheme was used by monitoring URLs in QuartzDesk 2.x.

## 6.5 Access to JAX-WS Endpoints

In QuartzDesk web application 2.x, all JAX-WS web service endpoints could be accessed by users with the QuartzDeskService J2EE security role. In QuartzDesk web application 3.x, these web service endpoints can only be accessed by authenticated users with particular access permissions.

The following table lists all JAX-WS web services and the security permissions that are required to access these web services.

JAX-WS Service	Required Permission
Connection Service	WS_CONNECTION
Security Service	WS_SECURITY
Quartz Service	WS_QUARTZ
Quartz Execution History Service	WS_QUARTZ_EXEC_HISTORY
Quartz Execution Notification Rule Service	WS_QUARTZ_EXEC_NOTIF_RULE
Quartz Job Chain Service	WS_QUARTZ_JOB_CHAIN

We recommend that you create a dedicated user account to access these JAX-WS endpoints. The user account can be created in Settings → Users in the QuartzDesk GUI. Do not forget to assign the user the relevant permission(s).



All JAX-WS web service endpoints in QuartzDesk 3.x support the HTTP Basic authentication scheme where the user's authentication credentials are passed in the `Authorization` HTTP header. Please note that the same authentication scheme was used by JAX-WS endpoints in QuartzDesk 2.x.

## 6.6 Remove Unused Security Domain

In JAC go to:

**JBoss 7.x Community:** Profile → Security → Security Domains

**JBoss 6.1 EAP, JBoss 6.2 EAP:** Profile → Subsystems → Security → Security Domains

**JBoss 6.3 EAP, JBoss 6.4 EAP:** Configuration → Subsystems → Security → Security Domains

**JBoss 7.x EAP:** Configuration → Subsystems → Security

Select **quartzdesk** security domain.

Click the Remove button to remove the selected security domain.

### Security Domains

Registered security domains. Please select a domain to edit the security policies.

<a href="#">Add</a> <a href="#">Remove</a>	
Name	Option
jboss-ejb-policy	<a href="#">View &gt;</a>
jboss-web-policy	<a href="#">View &gt;</a>
other	<a href="#">View &gt;</a>
<b>quartzdesk</b>	<a href="#">View &gt;</a>

**No longer needed**

<< < 1-4 of 4 > >>

#### Details

[Attributes](#)

<a href="#">Need Help?</a>	
<a href="#">Edit</a>	
<b>Cache Type:</b>	default
<b>Name:</b>	quartzdesk

Remove the users and roles properties files that were used by the removed security domain:

JAS\_INSTALL\_ROOT/JAS\_CONFIG/quartzdesk-users.properties

JAS\_INSTALL\_ROOT/JAS\_CONFIG/quartzdesk-roles.properties



## 7. Cluster Deployment Notes

When deploying the QuartzDesk web application to a JBoss cluster you need to follow the configuration steps described in preceding chapters. In addition to these, there are several extra configuration steps that must be performed for a cluster deployment.

### 7.1 HTTP Session Replication and Affinity

The QuartzDesk web application makes use of HTTP sessions and to store some short-lived and user-specific data. To achieve high-availability (HA), it is necessary to make the session data available on all application cluster members so that when one cluster member becomes unavailable, the remaining cluster members can take over and handle user requests without the user noticing any service interruption. To make the session data available on all application cluster members, the HTTP session replication process must be enabled on the cluster.



The amount of data stored by the QuartzDesk web application in an HTTP session is kept at the absolute minimum to reduce the session replication overhead. The total size of data stored in the session does not exceed 1KB.

When configuring session replication, we recommend that you also enable session affinity (sticky-sessions) on the load-balancer so that all user requests are preferably passed to the JBoss instance that handled the first user request that established the session.

Please refer to the JBoss and load-balancer documentation for details on how to configure session replication and session affinity because the actual steps may vary depending on the JBoss cluster topology and configuration.

### 7.2 Shared Work Directory

We recommend that you put the QuartzDesk web application work directory, described in chapter 4.3, on a shared drive and make this work directory available to all cluster members. Not only does this make application and configuration upgrading easier, it is actually required by all “Save” (for example, Save Log, Save Chart etc.) actions provided by the QuartzDesk web application GUI. These actions trigger two subsequent HTTP requests where the first request prepares the data and stores it in the `WORK_DIR/tmp` directory and the second request downloads the data and makes the browser open the Save As dialog.

During a fail-over or if the session affinity is not enabled, it can easily happen that the first request is handled by cluster member A and the second request is handled by cluster member B. If A and B are not configured to use the same `WORK_DIR/tmp` directory, then B will fail to serve the data prepared by A during the preceding request because the data will not be found.

### 7.3 Logging Configuration

If you set up your cluster to use a shared QuartzDesk web application work directory, as described in the previous chapter, you will need to edit the QuartzDesk web application logging configuration file `WORK_DIR/logback.xml` and decide where QuartzDesk web application instances running on individual cluster members should log. There are two options:

- 1) Logging into the same (shared) log files.
- 2) Logging into separate log files.

The QuartzDesk web application uses two log files – `quartzdesk-web.log` and `quartzdesk-web-trace.log` that are stored in `WORK_DIR/logs` directory. The following chapters discuss these two options.

### 7.3.1 Using Shared Log Files

In order to make individual QuartzDesk web application instances log into the same log files, you must enable the prudent mode on both file appenders used in the `WORK_DIR/logback.xml` configuration file:

```
...

<appender name="FILE"
class="ch.qos.logback.core.rolling.RollingFileAppender">
  <file>${logs.dir}/quartzdesk-web.log</file>
  <append>true</append>
  <prudent>true</prudent>
  ...
</appender>

<appender name="TRACE_FILE"
class="ch.qos.logback.core.rolling.RollingFileAppender">
  <file>${logs.dir}/quartzdesk-web-trace.log</file>
  <append>true</append>
  <prudent>true</prudent>
  ...

<!--
  We must use the TimeBasedRollingPolicy because the
  FixedWindowRollingPolicy is not supported in prudent mode!
-->
<rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">
  <!-- daily rollover -->
  <fileNamePattern>${logs.dir}/quartzdesk-web.log.%d{yyyy-MM-
dd}</fileNamePattern>
  <!-- keep 10 days' worth of history -->
  <maxHistory>10</maxHistory>
</rollingPolicy>

<!--
  The SizeBasedTriggeringPolicy removed because it is used only in
  conjunction with the FixedWindowRollingPolicy.
-->

<encoder>
  <charset>UTF-8</charset>
  <pattern>[%date] %.-1level [%thread] [%mdc] [%logger:%line] -
msg%n</pattern>
</encoder>
</appender>

...
```

For details on the Logback prudent mode, please refer to <http://logback.qos.ch/manual/appenders.html#FileAppender>.



Because prudent mode relies on exclusive file locks to manage concurrent access to the log files and these locks can have negative impact on the QuartzDesk web application's performance, we generally discourage using the prudent mode and shared log files.

### 7.3.2 Using Separate Log Files

In order to make individual QuartzDesk web application instances log into separate log files, you can use a JVM system property set on all cluster member JVMs. The value of this property must be unique for all cluster members. The property can be referred to from the `WORK_DIR/logback.xml` logging configuration file.

The following examples assume the use of the `cluster.member.instanceId` JVM system property, but any JVM system property name can be used.

There are two common approaches as to where the separate log files produced by individual QuartzDesk web application instances are stored:

- 1) Log files created under a common log root directory.

```
...
<appender name="FILE"
class="ch.qos.logback.core.rolling.RollingFileAppender">
  <file>${logs.dir}/quartzdesk-web-_${cluster.member.instanceId}.log</file>
  <append>true</append>
...
</appender>

<rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">
  <!-- daily rollover -->
  <fileNamePattern>${logs.dir}/quartzdesk-web-
_${cluster.member.instanceId}.log.%d{yyyy-MM-dd}</fileNamePattern>
  <!-- keep 10 days' worth of history -->
  <maxHistory>10</maxHistory>
</rollingPolicy>
...
</appender>

<appender name="TRACE_FILE"
class="ch.qos.logback.core.rolling.RollingFileAppender">
  <file>${logs.dir}/quartzdesk-web-_${cluster.member.instanceId}-
trace.log</file>
  <append>true</append>
...
</appender>

<rollingPolicy
class="ch.qos.logback.core.rolling.FixedWindowRollingPolicy">
  <fileNamePattern>${logs.dir}/quartzdesk-web-
_${cluster.member.instanceId}-trace.log.%i</fileNamePattern>
  <minIndex>1</minIndex>
  <maxIndex>5</maxIndex>
</rollingPolicy>
...
</appender>
...
```

2) Log files created in separate (cluster member specific) log root directories.

```

...
<!--
  Logback context property logback.config.dir is set by the
  LogbackInitContextListener to point to the parent directory of the Logback
  configuration file (logback.xml).
-->
<property name="logs.dir" value="${logback.config.dir:-
  .}/${cluster.member.instanceId}/logs"/>
...
  
```

## 7.4 Internal Quartz Scheduler

The QuartzDesk web application ships with an embedded Quartz scheduler to periodically execute its internal jobs. When deploying the QuartzDesk web application to a cluster, it is necessary to **assign unique instance IDs to Quartz scheduler instances** running in the clustered QuartzDesk web application instances.

For these purposes the QuartzDesk web application configuration (`quartzdesk-web.properties` file) provides the `scheduler.org.quartz.scheduler.instanceIdGenerator.class` configuration property. The value of this property must be a fully-qualified class name of a Java class that implements the `org.quartz.spi.InstanceIdGenerator` Quartz API interface. Quartz API provides two out of the box implementations suitable for clustered QuartzDesk web application deployments:

Implementation	Description
<code>org.quartz.simpl.HostnameInstanceIdGenerator</code>	<p>This implementation is suitable for QuartzDesk web application deployments where individual clustered QuartzDesk web application instances run on distinct hosts and each of these hosts is assigned a unique hostname.</p> <p>This is the default implementation used by the QuartzDesk web application. No configuration changes are necessary to use this instance ID generator.</p>
<code>org.quartz.simpl.SystemPropertyInstanceIdGenerator</code>	<p>This implementation is suitable for QuartzDesk web application deployments where some of the clustered QuartzDesk web application instances run on the same host.</p> <p>This implementation extracts the Quartz scheduler instance ID from the <code>org.quartz.scheduler.instanceId</code> JVM system property that must be explicitly set.</p> <p>Please refer to the JBoss documentation for details on how to add a new JVM system property.</p>

Please refer to the table above and optionally modify the value of the `scheduler.org.quartz.scheduler.instanceIdGenerator.class` configuration property according to the cluster configuration.

