



QuartzDesk JVM Agent Installation and Upgrade Guide for Standalone Java Applications

QuartzDesk Version: 4.x

March 3, 2020



Table of Contents

1.	PURPOSE	3
2.	DEFINITIONS	4
3.	REQUIREMENTS.....	5
3.1	SOFTWARE REQUIREMENTS	5
3.1.1	<i>Operating System</i>	5
3.1.2	<i>JVM</i>	5
3.1.3	<i>Database</i>	5
3.1.4	<i>Database JDBC Driver</i>	5
3.1.5	<i>QuartzDesk JVM Agent Library</i>	6
3.1.6	<i>QuartzDesk Public API Library</i>	6
3.2	HARDWARE REQUIREMENTS.....	6
4.	INSTALLATION.....	7
4.1	DATABASE.....	7
4.2	JVM AGENT WORK DIRECTORY	7
4.3	JDBC DRIVER	8
4.4	JVM AGENT CONFIGURATION	9
4.5	INSTALL JVM AGENT.....	9
4.5.1	<i>Windows</i>	9
4.5.2	<i>Unix / Linux</i>	10
4.6	INSTALL PUBLIC API LIBRARY	11
4.7	STOP APPLICATION.....	11
4.8	START APPLICATION	12
5.	UPGRADING	13
5.1	STOP APPLICATION.....	13
5.2	BACKUP	13
5.3	UPGRADE JVM AGENT.....	13
5.4	UPGRADE PUBLIC API LIBRARY	13
5.5	START APPLICATION	13
6.	QUARTZDESK 2.X TO 3.X MIGRATION NOTES	14
6.1	MINIMUM REQUIRED JAVA VERSION.....	14
6.2	CONFIGURATION PROPERTIES CHANGES	14
6.3	UPGRADE STEPS	14
7.	QUARTZDESK 3.X TO 4.X MIGRATION NOTES	15
7.1	MINIMUM REQUIRED JAVA VERSION.....	15
7.2	CONFIGURATION PROPERTIES CHANGES	15
7.3	UPGRADE STEPS	15



1. Purpose

This document describes the installation and upgrade process for QuartzDesk JVM Agent 4.x in standalone Java applications.

QuartzDesk JVM Agent is a Java Virtual Machine (JVM) plugin that must be installed in all JVMs powering Quartz-based applications managed by QuartzDesk Standard or Enterprise editions.

If you experience any problems installing or upgrading QuartzDesk JVM Agent, please let us know at support@quartzdesk.com.



2. Definitions

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

Acronym / Shortcut	Definition
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.

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

Location / Property	Example	Description
AGENT_WORK_DIR	<code>/var/quartzdesk-agent.work</code>	QuartzDesk JVM Agent work directory.
DB_HOST	<code>localhost</code>	QuartzDesk JVM Agent database server host.
DB_PORT	<code>5432</code>	QuartzDesk JVM Agent database server port.
DB_NAME	<code>quartzdesk_agent</code>	QuartzDesk JVM Agent database name.
DB_SCHEMA	<code>quartzdesk_agent</code>	QuartzDesk JVM Agent database schema.
DB_USER	<code>quartzdesk_agent</code>	QuartzDesk JVM Agent database user.
DB_PASSWORD	<code>quartzdesk_agent</code>	QuartzDesk JVM Agent database user password.



3. Requirements

3.1 Software Requirements

3.1.1 Operating System

Windows 7, Windows 8, Windows 10.
Linux (any distribution) with kernel 2.6.x and above.
Solaris 11.x and above.

3.1.2 JVM

Oracle JDK 8–13.
IBM JDK 7.
OpenJDK 8–13.

3.1.3 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	9.1

3.1.4 Database JDBC Driver

Database	JDBC Driver
DB2	IBM DB2 JDBC 4.0 driver available at http://www-01.ibm.com/support/docview.wss?uid=swg21363866 .
H2	Database engine including the JDBC driver is available at http://www.h2database.com .
Microsoft SQL Server	<p>Microsoft JDBC driver 4.0 for SQL Server available at http://msdn.microsoft.com/en-us/sqlserver/aa937724.aspx.</p> <p>We strongly advise against using the alternative JTDS JDBC driver, because it currently does not support the datetime2 data type. As a result, all datetime values written by the QuartzDesk application would end up rounded up, or down. For datetime data type rounding details, please refer to http://msdn.microsoft.com/en-us/library/ms187819.aspx.</p>
MySQL	Connector/J JDBC driver available at http://dev.mysql.com/downloads/connector/j/ .

Oracle	Oracle JDBC driver available at http://www.oracle.com/technetwork/database/features/jdbc/index-091264.html . For a comprehensive overview of JDBC driver versions vs. supported database versions, please refer to http://www.oracle.com/technetwork/database/enterprise-edition/jdbc-faq-090281.html#01_02 .
PostgreSQL	JDBC4 PostgreSQL driver available at http://jdbc.postgresql.org/ .

3.1.5 QuartzDesk JVM Agent Library

To install QuartzDesk JVM Agent, you need to obtain the QuartzDesk JVM Agent JAR. The latest version can be downloaded at www.quartzdesk.com (click Downloads → Latest Release → View files → quartzdesk-agent-x.y.z.jar).

3.1.6 QuartzDesk Public API Library

QuartzDesk JVM Agent requires all applications with embedded Quartz schedulers deployed on the given JVM to have the QuartzDesk Public API Library on their class path. The library must be loaded by the same class loader that loads the Quartz API.

The latest version of the QuartzDesk Public API Library can be downloaded at www.quartzdesk.com (click Downloads → Latest Release → View files → quartzdesk-api-x.y.z.jar). It is also available in the Maven Central repository – see <http://search.maven.org/#search|ga|1|quartzdesk-api>.

3.2 Hardware Requirements

QuartzDesk JVM Agent runs on any physical or virtualized hardware that supports the above software requirements.



4. Installation

4.1 Database

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

Create a new QuartzDesk JVM Agent database named `quartzdesk_agent1` (`DB_NAME`) owned by the `DB_USER`.

If the database supports database schemas, create a new schema named `quartzdesk_agent` (`DB_SCHEMA`). The schema must be owned by the `DB_USER`. Make the created `DB_SCHEMA` the default schema of the `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 other database objects (tables, keys, indices etc.) in the QuartzDesk JVM Agent database. These objects will be automatically created by QuartzDesk JVM Agent during its first run.

4.2 JVM Agent Work Directory

Create the QuartzDesk JVM Agent work directory (`AGENT_WORK_DIR`) anywhere on the local file system. The directory must be readable and writeable by the user the standalone Java application process runs under.

Copy your QuartzDesk license key file (`license.key`) to `AGENT_WORK_DIR`.



You can obtain a free 30-day trial license key at www.quartzdesk.com (open the Try / Purchase menu).

Copy the QuartzDesk JVM Agent JAR file (`quartzdesk-agent-x.y.z.jar`) to `AGENT_WORK_DIR`.

Open the QuartzDesk JVM Agent JAR file and copy all files from the `extras/work` directory into `AGENT_WORK_DIR`.



If you cannot open the JAR file directly, rename it to `*.zip` and then open it. Do not forget to rename the file back to `*.jar` once you have extracted the required files.

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

¹ If you use DB2, the database name length is restricted to the maximum of 8 characters. Please adjust the database name accordingly (e.g. `qdagent`).

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.18363.592]
(c) 2019 Microsoft Corporation. All rights reserved.

d:\var\quartzdesk-agent.work\4.0.x>dir
Volume in drive D is DISK_D
Volume Serial Number is 7A4F-989B

Directory of d:\var\quartzdesk-agent.work\4.0.x

2020-02-12  15:24    <DIR>          .
2020-02-12  15:24    <DIR>          ..
2016-11-02  13:29             4,259 license.key
2015-06-25  23:39             3,758 logback.xml
2020-02-11  20:48          12,301,128 quartzdesk-agent-4.0.0.jar
2018-05-23  14:48             9,256 quartzdesk-agent.properties
           4 File(s)          12,318,401 bytes
           2 Dir(s)    2,864,885,121,024 bytes free

d:\var\quartzdesk-agent.work\4.0.x>
```

4.3 JDBC Driver

Download and install the JDBC driver for the created database. For a list of supported JDBC drivers please refer to chapter 3.1.4.

Copy the JDBC driver JAR file to AGENT_WORK_DIR. Make sure the JAR file is readable by the user the standalone Java application process is started under.

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.18363.592]
(c) 2019 Microsoft Corporation. All rights reserved.

d:\var\quartzdesk-agent.work\4.0.x>dir
Volume in drive D is DISK_D
Volume Serial Number is 7A4F-989B

Directory of d:\var\quartzdesk-agent.work\4.0.x

2020-02-12  15:45    <DIR>          .
2020-02-12  15:45    <DIR>          ..
2016-11-02  13:29             4,259 license.key
2015-06-25  23:39             3,758 logback.xml
2018-07-04  09:54          594,994 postgresql-9.3-1104.jdbc41.jar
2020-02-11  20:48          12,301,128 quartzdesk-agent-4.0.0.jar
2018-05-23  14:48             9,256 quartzdesk-agent.properties
           5 File(s)          12,913,395 bytes
           2 Dir(s)    2,864,884,486,144 bytes free

d:\var\quartzdesk-agent.work\4.0.x>
```

In the standalone Java application's startup script add the JDBC driver JAR file(s) to the JVM's boot class path (for Java 8) or to the JVM's module path (for Java 9–13).

4.4 JVM Agent Configuration

Open the QuartzDesk JVM Agent configuration file `AGENT_WORK_DIR/quartzdesk-agent.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	>= 5.6	mysql
MySQL (Inno)	>= 5.6	mysql_inno
Oracle	== 8i	oracle8
Oracle	>= 9i	oracle9
PostgreSQL	== 8.1	postgres81
PostgreSQL	>= 8.2	postgres82

Uncomment the Agent JDBC pool configuration section based on the QuartzDesk JVM Agent database type. Make sure the JDBC pool configuration sections for other database types are commented out (prefixed with '#'). The default sample `quartzdesk-agent.properties` file assumes the use of a PostgreSQL database.

Adjust values of the JDBC pool configuration parameters to match your database configuration. You will typically need to change the default host value (`localhost`) in the `jdbc.url` parameter to point to `DB_HOST`. Please refer to the JDBC driver manual for a description of the JDBC URL format and related details.

Set the value of the `jdbc.pool.maxTotal` JDBC pool configuration parameter to be 10-20% higher than the maximum number of **concurrently executing** Quartz jobs on the JVM QuartzDesk JVM Agent will be installed on.

To adjust QuartzDesk JVM Agent logging parameters, edit the `AGENT_WORK_DIR/logback.xml` configuration file. The default sample `logback.xml` configuration file creates the QuartzDesk JVM Agent log under the `AGENT_WORK_DIR/logs` directory that is automatically created when QuartzDesk JVM Agent starts. Please refer to the [Logback Manual](#) for Logback configuration details.

4.5 Install JVM Agent

4.5.1 Windows

Edit standalone Java application's startup script (typically with `bat`, or `cmd` extension) and add the following three JVM command line arguments to the list of JVM command line arguments already present in the script.

Java 8

```
-javaagent:AGENT_WORK_DIR/quartzdesk-agent-x.y.z.jar -Dquartzdesk-agent.work.dir=<AGENT_WORK_DIR> -Xbootclasspath/a:<JDBC_DRIVER_JAR_FILE_PATH>
```

Java 9–13

```
-javaagent:AGENT_WORK_DIR/quartzdesk-agent-x.y.z.jar -Dquartzdesk-agent.work.dir=<AGENT_WORK_DIR> --module-path <JDBC_DRIVER_JAR_FILE_PATH>
```

Example of a standalone application's Windows startup script with all required JVM parameters.

```
@echo off
set "JAVA_HOME=d:/Java/Jdk13"
set "QD_JVM_AGENT_WORK_DIR=d:/var/quartzdesk-agent.work/4.0.x"
set "JAVA_OPTS="

rem Set application class path here.
set "CLASSPATH=."

rem Set JVM command line arguments here.
set "JAVA_OPTS=%JAVA_OPTS% -Xms250m"
set "JAVA_OPTS=%JAVA_OPTS% -Xmx500m"

rem
rem QuartzDesk JVM Agent JAR
rem
set "JAVA_OPTS=%JAVA_OPTS% -javaagent:%QD_JVM_AGENT_WORK_DIR%/quartzdesk-agent-4.0.0.jar"

rem
rem QuartzDesk JVM Agent work directory
rem
set "JAVA_OPTS=%JAVA_OPTS% -Dquartzdesk-agent.work.dir=%QD_JVM_AGENT_WORK_DIR%"

rem
rem JDBC driver used by QuartzDesk JVM Agent
rem

rem For Java 8
rem set "JAVA_OPTS=%JAVA_OPTS% -Xbootclasspath/a:%QD_JVM_AGENT_WORK_DIR%/postgresql-9.3-1104.jdbc41.jar"

rem For Java 9-13
set "JAVA_OPTS=%JAVA_OPTS% --module-path %QD_JVM_AGENT_WORK_DIR%/postgresql-9.3-1104.jdbc41.jar"

%JAVA_HOME%/bin/java.exe %JAVA_OPTS% -cp %CLASSPATH% myapp.App
```

4.5.2 Unix / Linux

Edit standalone Java application's startup script (typically with sh extension) and add the following three JVM command line arguments to the list of JVM command line arguments already present in the script.

Java 8

```
-javaagent:AGENT_WORK_DIR/quartzdesk-agent-x.y.z.jar -Dquartzdesk-agent.work.dir=<AGENT_WORK_DIR> -Xbootclasspath/a:<JDBC_DRIVER_JAR_FILE_PATH>
```

Java 9–13

```
-javaagent:AGENT_WORK_DIR/quartzdesk-agent-x.y.z.jar -Dquartzdesk-agent.work.dir=<AGENT_WORK_DIR> --module-path <JDBC_DRIVER_JAR_FILE_PATH>
```

Example of a standalone application's Unix/Linux startup script with all required JVM parameters.

```
#!/bin/sh
JAVA_HOME="/usr/lib/jvm/java"
QD_JVM_AGENT_WORK_DIR="/opt/quartzdesk-agent.work/4.0.x"
JAVA_OPTS=

# Set application class path here.
CLASSPATH="."

# JVM command line arguments here.
JAVA_OPTS="${JAVA_OPTS} -Xms250m"
JAVA_OPTS="${JAVA_OPTS} -Xmx500m"

#
# QuartzDesk JVM Agent JAR
#
JAVA_OPTS="${JAVA_OPTS} -javaagent:${QD_JVM_AGENT_WORK_DIR}/quartzdesk-agent-4.0.0.jar"

#
# QuartzDesk JVM Agent work directory
#
JAVA_OPTS="${JAVA_OPTS} -Dquartzdesk-agent.work.dir=${QD_JVM_AGENT_WORK_DIR}"

#
# JDBC driver used by QuartzDesk JVM Agent
#

# For Java 8
# JAVA_OPTS="${JAVA_OPTS} -Xbootclasspath/a:${QD_JVM_AGENT_WORK_DIR}/postgresql-9.3-1104.jdbc41.jar"

# For Java 9-13
# JAVA_OPTS="${JAVA_OPTS} --module-path ${QD_JVM_AGENT_WORK_DIR}/postgresql-9.3-1104.jdbc41.jar"

${JAVA_HOME}/bin/java ${JAVA_OPTS} -cp ${CLASSPATH} myapp.App
```

4.6 Install Public API Library

QuartzDesk Public API Library² works as an interface between the Quartz library³ used by a Java application and QuartzDesk JVM Agent. **QuartzDesk Public API Library must be loaded by the same Java class loader that loads the Quartz library.**

For standalone applications, the Quartz library is typically placed on the application's class path in its startup script. To install QuartzDesk Public API Library, it is only necessary to add the QuartzDesk Public API Library JAR on the class path.

Please note that the installation of QuartzDesk Public API Library requires no application code changes.

4.7 Stop Application

Stop the standalone Java application if it is running.

² JAR file name: quartzdesk-api-<version>.jar

³ JAR file name name: quartz-<version>.jar or quartz-all-<version>.jar

4.8 Start Application

Start the standalone Java application, e.g. by executing its startup script. Wait for the application to start and check the application log for errors.

Check the QuartzDesk JVM Agent logs (in AGENT_WORK_DIR/logs directory) for errors and verify the version number of the installed QuartzDesk JVM Agent.

```
File Edit Options Encoding Help
Network Interface: MAC-00-00-24-2B-86-00, IP=1
Network Interface: MAC-00-00-24-2B-86-00, IP=1
Network Interface: MAC-00-00-24-2B-86-00, IP=1
Network Interface: MAC-00-00-24-2B-86-00, IP=1
Network Interface: MAC-00-00-00-00-00-00, IP=1
Network Interface: MAC-00-00-00-00-00-00, IP=1
Network Interface: MAC-00-00-00-00-00-00, IP=1
Network Interface: MAC-00-00-00-00-00-00, IP=1
Network Interface: MAC-00-00-00-00-00-00, IP=1
Network Interface: MAC-00-00-00-00-00-00, IP=1
[2020-02-13 16:34:09,824] I [main] [com.quartzdesk.agent.Agent:459] - Successfully created license manager: com.quartzdesk.agent.shaded.com.quartzdesk.license.v1.0.impl.LicenseManagerImpl825f4d54
[2020-02-13 16:34:09,821] I [main] [com.quartzdesk.agent.Agent:545] - License info:
Serial Number: 1
Issue Date: 2016-11-02
Type: PERPETUAL
Expiry Date: n/a
License: Test License
Issuer: CH-QuartzDesk.com CA2, O-QuartzDesk, sales@quartzdesk.com, www.quartzdesk.com
Licensed Products:
ID-QuartzDesk, name=QuartzDesk Enterprise Edition
[2020-02-13 16:34:09,824] I [main] [com.quartzdesk.agent.Agent:574] - Licensed product features:
Feature Set, id=General
Features:
  AuditTrail=true
  Customization=true
  WSAuditTrailService=true
  WSConnectionService=true
  WSSecurityService=true
  WSMessageChannelProfileService=true
  ODM=true
  SoftwareMaintenanceExpiry=2020-12-31+01:00
Feature Set, id=Quartz
Features:
  MaxJobsPerScheduler=unlimited
  MaxTriggersPerJob=unlimited
  ExecutionHistory=true
  PlannedJobExecutions=true
  MisfireTrigger=true
  ExecutionNotifications=true
  PollMonitoring=true
  ExecutionStatistics=true
  JobChaining=true
  WSQuartzService=true
  WSQuartzExecHistoryService=true
  WSQuartzNotificationSubscriberService=true
  WSQuartzJobChainService=true
[2020-02-13 16:34:09,833] I [main] [com.quartzdesk.agent.Agent:426] - Successfully created DB dialect: com.quartzdesk.agent.db.dialect.PostgresJdbiDialect@30c69952 for configured DB profile: postgres02
[2020-02-13 16:34:09,833] I [main] [com.quartzdesk.agent.Agent:4130] - Detected QuartzDesk JVM Agent database schema version: 4.0.0. Desired database schema version: 4.0.0
[2020-02-13 16:34:09,877] I [main] [com.quartzdesk.agent.Agent:4142] - QuartzDesk JVM Agent database schema is up-to-date.
[2020-02-13 16:34:09,871] I [main] [com.quartzdesk.agent.Agent:627] - Successfully registered MBean: com.quartzdesk.agent.AgentMBeanImpl875d951a, object name: com.quartzdesk.agent:type=AgentMBean
[2020-02-13 16:34:09,872] I [main] [com.quartzdesk.agent.Agent:641] - Successfully registered MBean: com.quartzdesk.agent.DataSynchronizationMBeanImpl87c84687, object name: com.quartzdesk.agent:type=DataSynchronizationMBean
[2020-02-13 16:34:09,873] I [main] [com.quartzdesk.agent.Agent:655] - Successfully registered MBean: com.quartzdesk.agent.scheduler.common.message.MessageMBeanImpl875f5f58, object name: com.quartzdesk.agent:type=MessageMBean
[2020-02-13 16:34:09,875] I [main] [com.quartzdesk.agent.Agent:669] - Successfully registered MBean: com.quartzdesk.agent.scheduler.quartz.history.QuartzExecHistoryMBeanImpl86f2cb53, object name: com.quartzdesk.agent:type=Quartz
[2020-02-13 16:34:09,876] I [main] [com.quartzdesk.agent.Agent:683] - Successfully registered MBean: com.quartzdesk.agent.scheduler.quartz.notify.QuartzExecutionNotificationMBeanImpl8e7d75f, object name: com.quartzdesk.agent:type=
[2020-02-13 16:34:09,881] I [main] [com.quartzdesk.agent.Agent:697] - Successfully registered MBean: com.quartzdesk.agent.scheduler.quartz.start.QuartzExecStartMBeanImpl887605d7, object name: com.quartzdesk.agent:type=Quartz
[2020-02-13 16:34:09,882] I [main] [com.quartzdesk.agent.Agent:711] - Successfully registered MBean: com.quartzdesk.agent.scheduler.quartz.chain.QuartzJobChainMBeanImpl85c69b8d, object name: com.quartzdesk.agent:type=QuartzJobCh
[2020-02-13 16:34:09,883] I [main] [com.quartzdesk.agent.Agent:725] - Successfully registered MBean: com.quartzdesk.agent.scheduler.quartz.misfire.QuartzMisfireTriggerMBeanImpl8305a68a, object name: com.quartzdesk.agent:type=Qu
[2020-02-13 16:34:09,885] I [main] [com.quartzdesk.agent.Agent:739] - Successfully registered MBean: com.quartzdesk.agent.scheduler.quartz.monitor.QuartzMonitoringMBeanImpl83435123, object name: com.quartzdesk.agent:type=QuartzM
[2020-02-13 16:34:09,885] I [main] [com.quartzdesk.agent.Agent:753] - Successfully registered MBean: com.quartzdesk.agent.scheduler.quartz.monitor.QuartzMonitoringMBeanImpl83435123, object name: com.quartzdesk.agent:type=QuartzM
[2020-02-13 16:34:09,885] I [main] [com.quartzdesk.agent.Agent:767] - Successfully registered MBean: com.quartzdesk.agent.scheduler.quartz.monitor.QuartzMonitoringMBeanImpl83435123, object name: com.quartzdesk.agent:type=QuartzM
[2020-02-13 16:34:09,890] I [main] [com.quartzdesk.agent.Agent:781] - Starting RMI connector listening on registry port: 11171 and server port: 11171, SSL: false
[2020-02-13 16:34:10,026] I [main] [com.quartzdesk.agent.Agent:8106] - Starting JNDI connector listening on port: 11170, SASL: false
[2020-02-13 16:34:10,054] I [main] [com.quartzdesk.agent.Agent:8207] - Successfully started JNDI connector: java.management.remote.rmi.RMISocketFactory@8405224f listening on port: 11170
[2020-02-13 16:34:10,055] I [main] [com.quartzdesk.agent.Agent:248] - Successfully initialized JVM Agent com.quartzdesk.agent.Agent@32464f44 [QuartzDesk JVM Agent v4.0.0], enabled: true
[2020-02-13 16:34:12,515] I [ServerService Thread Pool -- 62] [com.quartzdesk.agent.scheduler.quartz.scheduler.v4.0.QuartzSchedulerImpl@94] - Successfully instrumented org.quartz.impl.jdbcjobstore.JobStoreImpl class loaded from: vfi...
```

Verify that the application works as expected.

5. Upgrading

5.1 Stop Application

Stop the standalone Java application if it is running.

5.2 Backup

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

Backup the contents of the QuartzDesk JVM Agent work directory.

Store the backups in a safe place so that you can restore the original QuartzDesk JVM Agent version if the need arises.

5.3 Upgrade JVM Agent

Delete the old QuartzDesk JVM Agent JAR file in `AGENT_WORK_DIR`. Copy the new `quartzdesk-agent-x.y.z.jar` to `AGENT_WORK_DIR`.

Rename the `AGENT_WORK_DIR/quartzdesk-agent.properties` configuration file to `quartzdesk-agent.properties.old`.

Open the QuartzDesk JVM Agent JAR file (`quartzdesk-agent-x.y.z.jar`) and copy the `extras/work/quartzdesk-agent.properties` configuration file to `AGENT_WORK_DIR`.



If you cannot open the JAR file directly, rename it to `*.zip` and then open it. Do not forget to rename the file back to `*.jar` once you have extracted the required files.

Adjust the values of the configuration properties in the new configuration file `AGENT_WORK_DIR/quartzdesk-agent.properties` to match your system setup. You can use the old configuration file as a reference.

Please refer to 4.3 for a description of the configuration parameters that you need to adjust.

5.4 Upgrade Public API Library

Upgrade the QuartzDesk Public API library on the standalone Java application's class path. Make sure the new library is included on the application's class path. Please refer to 4.6 for details.

5.5 Start Application

Start the standalone Java application by following steps outlined in 4.8.

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

6.1 Minimum Required Java Version

QuartzDesk JVM Agent 3.x requires Java 7 or higher.

6.2 Configuration Properties Changes

The following two `quartzdesk-agent.properties` configuration properties have been deprecated in QuartzDesk JVM Agent 3.x and will be removed in 4.x. Make sure your `quartzdesk-agent.properties` file uses the new property names.

Deprecated Configuration Property Name	New Configuration Property Name
<code>jdbc.pool.maxActive</code>	<code>jdbc.pool.maxTotal</code>
<code>jdbc.pool.maxWait</code>	<code>jdbc.pool.maxWaitMillis</code>

6.3 Upgrade Steps

To upgrade QuartzDesk JVM Agent 2.x to 3.x, apply upgrade steps outlined in 5.

7. QuartzDesk 3.x to 4.x Migration Notes

7.1 Minimum Required Java Version

QuartzDesk JVM Agent 4.x requires Java 8 or higher.

7.2 Configuration Properties Changes

The following two `quartzdesk-agent.properties` configuration properties have been deprecated in QuartzDesk JVM Agent 3.x and removed in QuartzDesk JVM Agent 4.x. Make sure your `quartzdesk-agent.properties` file uses the new property names.

Removed Configuration Property Name	New Configuration Property Name
<code>jdbc.pool.maxActive</code>	<code>jdbc.pool.maxTotal</code>
<code>jdbc.pool.maxWait</code>	<code>jdbc.pool.maxWaitMillis</code>

7.3 Upgrade Steps

To upgrade QuartzDesk JVM Agent 3.x to 4.x, apply upgrade steps outlined in 5.

