



QuartzDesk JVM Agent Installation and Upgrade Guide for Standalone Java Applications

QuartzDesk Version: 2.x

April 24, 2017



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>Java</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	JDBC DRIVER	7
4.3	JVM AGENT WORK DIRECTORY	8
4.4	JVM AGENT CONFIGURATION	8
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	10
4.7	ENABLE LOG MESSAGE INTERCEPTION	11
4.8	STOP APPLICATION.....	11
4.9	START APPLICATION	11
5.	UPGRADING	12
5.1	STOP APPLICATION.....	12
5.2	BACKUP	12
5.3	UPGRADE JVM AGENT.....	12
5.4	UPGRADE PUBLIC API LIBRARY	12
5.5	START APPLICATION	12



1. Purpose

This document describes the installation and upgrade process for the QuartzDesk JVM Agent 2.x in standalone Java applications.

QuartzDesk JVM Agent is a Java Virtual Machine (JVM) plugin that must be installed in all JVMs powering applications with embedded Quartz schedulers managed by QuartzDesk. The QuartzDesk JVM Agent enables the following QuartzDesk features:

- Execution History
- Execution Notifications
- Execution Statistics
- Job Chaining
- Health Indicators
- Scheduler / Job / Trigger Monitoring
- Misfired Triggers

Please note that the installation of the QuartzDesk JVM Agent is required only by the QuartzDesk Standard and Enterprise editions. The QuartzDesk Lite edition does not provide access to any of the above features and therefore it does not require the installation of the QuartzDesk JVM Agent.



If the QuartzDesk GUI detects the QuartzDesk JVM Agent is not installed / enabled in a remote JVM it connects to, it displays a warning message and the above listed features are disabled in the QuartzDesk GUI.

If you experience any problems installing or upgrading the 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</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 XP, Windows Vista, Windows 7, Windows 8, Windows 10.

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

Solaris 11.x and above.

3.1.2 Java

Sun/Oracle Java (JDK) 6, 7, 8.

IBM Java (JDK) 6, 7, 8.

OpenJDK 6, 7, 8.

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 the 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 classpath. The library must be loaded by the same classloader 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 refer to the database engine documentation for details on how to perform the above database operations as they are all database-specific.



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 the QuartzDesk JVM Agent during its first run.

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 3.1.4.

In the standalone Java application's startup script add the JDBC driver JAR file(s) to the JVM's boot classpath. The JDBC driver JAR file(s) must be added to the boot classpath so that QuartzDesk JVM Agent, that is loaded by the JVM's system classloader, can access JDBC driver classes. Please use the following JVM command line option:

```
-Xbootclasspath/p:JDBC DRIVER JAR FILE(S)

For example:

Windows:
-Xbootclasspath/p:c:\jdbc\db2\db2jcc_license_cu.jar;c:\jdbc\db2\db2jcc4.jar

Unix/Linux:
-Xbootclasspath/p:/opt/db2/db2jcc_license_cu.jar:/opt/jdbc/db2/db2jcc4.jar
```


Make sure the JDBC driver JAR file(s) are readable by the user the standalone Java application process runs under.

¹ DB2 restricts the database name length to the maximum of 8 characters. Please adjust the database name accordingly (e.g. qdagent).

4.3 JVM Agent Work Directory


Create the QuartzDesk JVM Agent work directory (AGENT_WORK_DIR) anywhere on the local file system. The directory should 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.

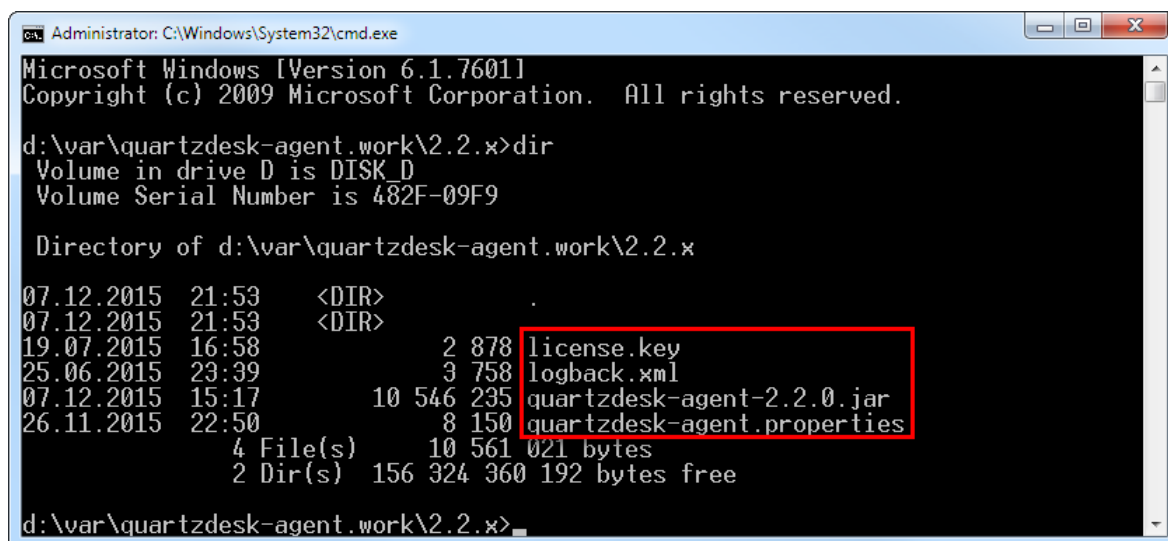
 You can obtain a free 30-day trial license key at www.quartzdesk.com (open the Try / Purchase menu).

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.



```

Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

d:\var\quartzdesk-agent.work\2.2.x>dir
Volume in drive D is DISK D
Volume Serial Number is 482F-09F9

Directory of d:\var\quartzdesk-agent.work\2.2.x

07.12.2015  21:53    <DIR>          .
07.12.2015  21:53    <DIR>          ..
19.07.2015  16:58                2 878 license.key
25.06.2015  23:39                3 758 logback.xml
07.12.2015  15:17           10 546 235 quartzdesk-agent-2.2.0.jar
26.11.2015  22:50                8 150 quartzdesk-agent.properties
           4 File(s)          10 561 021 bytes
           2 Dir(s)         156 324 360 192 bytes free

d:\var\quartzdesk-agent.work\2.2.x>
    
```

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.maxActive` JDBC pool configuration parameter to be 10-20% higher than the maximum number of **concurrently executing** Quartz jobs on the JVM the 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 the QuartzDesk JVM Agent starts. Please refer to the [Logback Manual](#) for Logback configuration details.

4.5 Install JVM Agent

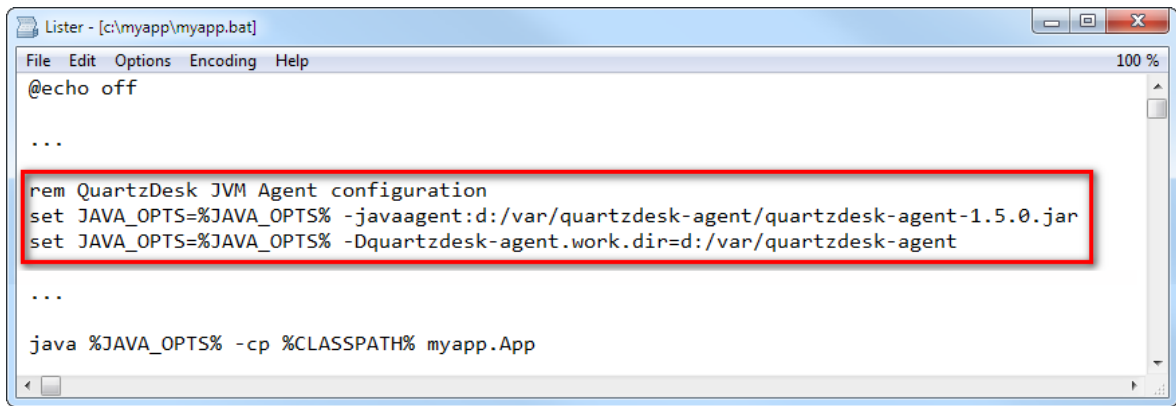
To manage Quartz schedulers embedded in a standalone Java application, you must first enable remote JMX access to the application. Please refer to the **How to Enable Remote JMX Access to Quartz Schedulers** document available at www.quartzdesk.com/documentation/how-tos. Once the remote JMX access has been enabled, continue with the steps below.

4.5.1 Windows

Edit standalone Java application's startup script (typically with `bat`, or `cmd` extension) and add the following two parameters to the list of standalone application's JVM startup parameters.

```
-javaagent:AGENT_WORK_DIR/quartzdesk-agent-x.y.z.jar -Dquartzdesk-agent.work.dir=AGENT_WORK_DIR
```

In the following figure you can see an example of a standalone application's startup script with all required JVM parameters.



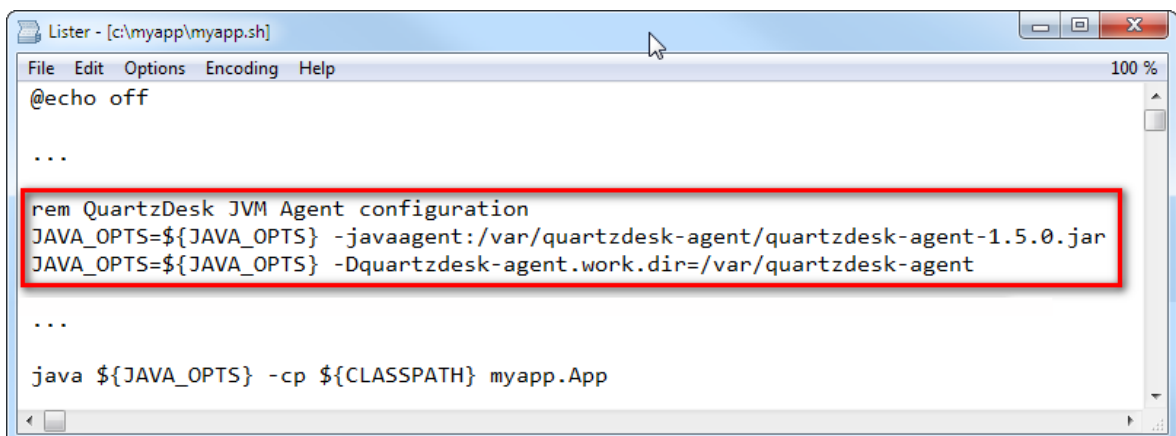
```
Lister - [c:\myapp\myapp.bat]
File Edit Options Encoding Help 100 %
@echo off
...
rem QuartzDesk JVM Agent configuration
set JAVA_OPTS=%JAVA_OPTS% -javaagent:d:/var/quartzdesk-agent/quartzdesk-agent-1.5.0.jar
set JAVA_OPTS=%JAVA_OPTS% -Dquartzdesk-agent.work.dir=d:/var/quartzdesk-agent
...
java %JAVA_OPTS% -cp %CLASSPATH% myapp.App
```

4.5.2 Unix / Linux

Edit standalone Java application's startup script (typically with bat, or cmd extension) and add the following two parameters to the list of standalone application's JVM startup parameters.

```
-javaagent:AGENT_WORK_DIR/quartzdesk-agent-x.y.z.jar -Dquartzdesk-agent.work.dir=AGENT_WORK_DIR
```

In the following figure you can see an example of a standalone application's startup script with all required JVM parameters:



```
Lister - [c:\myapp\myapp.sh]
File Edit Options Encoding Help 100 %
@echo off
...
rem QuartzDesk JVM Agent configuration
JAVA_OPTS=${JAVA_OPTS} -javaagent:/var/quartzdesk-agent/quartzdesk-agent-1.5.0.jar
JAVA_OPTS=${JAVA_OPTS} -Dquartzdesk-agent.work.dir=/var/quartzdesk-agent
...
java ${JAVA_OPTS} -cp ${CLASSPATH} myapp.App
```

4.6 Install Public API Library

The QuartzDesk Public API Library (quartzdesk-api-<version>.jar) works as an interface between the Quartz library (typically distributed as quartz-<version>) used by an application and the QuartzDesk JVM Agent. **The QuartzDesk Public API Library must be loaded by the same Java classloader that loads the Quartz library.**

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

No application code changes are required to install the QuartzDesk Public API Library.

4.7 Enable Log Message Interception

To enable interception of log messages produced by executed Quartz jobs, it is necessary to modify the logging configuration of the standalone Java application. The required configuration changes are simple and vary for individual logging frameworks.

For details please refer to the **How to Enable Log Message Interception in Applications** document available at www.quartzdesk.com/documentation/how-tos.

Please note that this step is optional. When the log message interception is not enabled, the following QuartzDesk features will not be available:

1. Viewing logs in the Execution History panels.
2. Viewing logs of currently executing jobs in the Currently Executing Jobs panel.
3. Accessing and analyzing log messages in JavaScript expressions in Execution Notification rules.
4. Attaching logs to messages sent by Execution Notification rules.

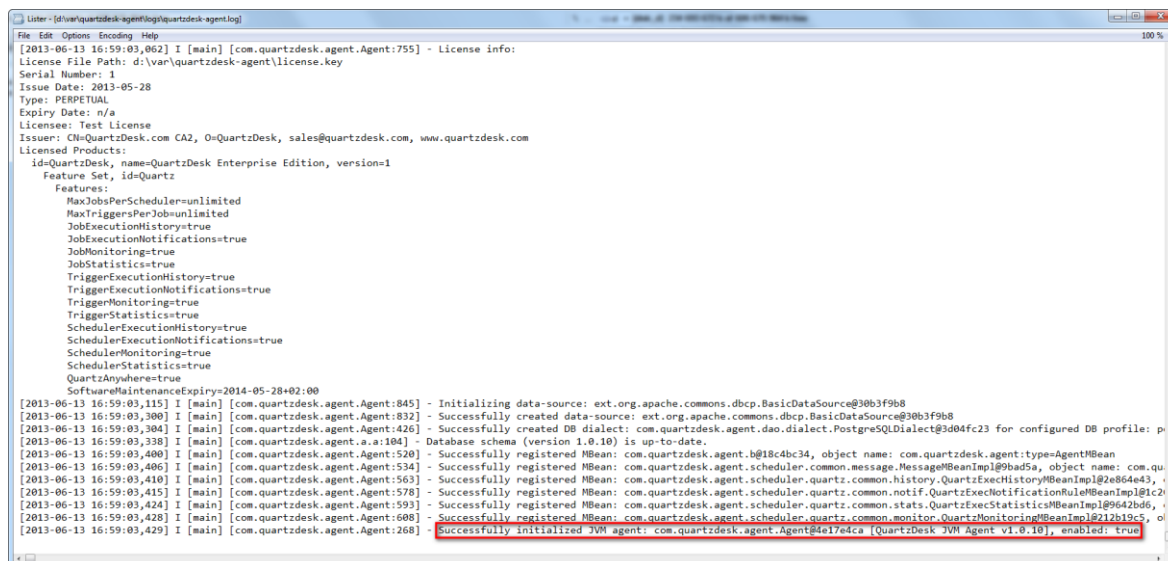
4.8 Stop Application

Stop the standalone Java application if it is running.

4.9 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 QuartzDesk JVM Agent.



```
File Edit Options Encoding Help
[2013-06-13 16:59:03,062] I [main] [com.quartzdesk.agent.Agent:755] - License info:
License File Path: d:\var\quartzdesk-agent\license.key
Serial Number: 1
Issue Date: 2013-05-28
Type: PERPETUAL
Expiry Date: n/a
License: Test License
Issuer: CN=QuartzDesk.com CA2, O=QuartzDesk, sales@quartzdesk.com, www.quartzdesk.com
Licensed Products:
id=QuartzDesk, name=QuartzDesk Enterprise Edition, version=1
Feature Set, id=Quartz
Features:
MaxJobsPerScheduler=unlimited
MaxTriggersPerJob=unlimited
JobExecutionHistory=true
JobExecutionNotifications=true
JobMonitoring=true
JobStatistics=true
TriggerExecutionHistory=true
TriggerExecutionNotifications=true
TriggerMonitoring=true
TriggerStatistics=true
SchedulerExecutionHistory=true
SchedulerExecutionNotifications=true
SchedulerMonitoring=true
SchedulerStatistics=true
QuartzAnywhere=true
SoftwareMaintenanceExpiry=2014-05-28+02:00
[2013-06-13 16:59:03,115] I [main] [com.quartzdesk.agent.Agent:845] - Initializing data-source: ext.org.apache.commons.dbcp.BasicDataSource@30b3f9b8
[2013-06-13 16:59:03,300] I [main] [com.quartzdesk.agent.Agent:832] - Successfully created data-source: ext.org.apache.commons.dbcp.BasicDataSource@30b3f9b8
[2013-06-13 16:59:03,304] I [main] [com.quartzdesk.agent.Agent:426] - Successfully created DB dialect: com.quartzdesk.agent.dao.dialect.PostgreSQLDialect@3d04fc23 for configured DB profile: p
[2013-06-13 16:59:03,338] I [main] [com.quartzdesk.agent.a.a:184] - Database schema (version 1.0.10) is up-to-date.
[2013-06-13 16:59:03,400] I [main] [com.quartzdesk.agent.Agent:520] - Successfully registered MBean: com.quartzdesk.agent.b@18c4bc34, object name: com.quartzdesk.agent.type=AgentMBean
[2013-06-13 16:59:03,406] I [main] [com.quartzdesk.agent.Agent:534] - Successfully registered MBean: com.quartzdesk.agent.scheduler.common.message.MessageMBeanImpl@9bad5a, object name: com.qu
[2013-06-13 16:59:03,410] I [main] [com.quartzdesk.agent.Agent:565] - Successfully registered MBean: com.quartzdesk.agent.scheduler.quartz.common.history.QuartzExecHistoryMBeanImpl@2e864e43,
[2013-06-13 16:59:03,415] I [main] [com.quartzdesk.agent.Agent:578] - Successfully registered MBean: com.quartzdesk.agent.scheduler.quartz.common.notif.QuartzExecNotificationRuleMBeanImpl@1c2
[2013-06-13 16:59:03,424] I [main] [com.quartzdesk.agent.Agent:593] - Successfully registered MBean: com.quartzdesk.agent.scheduler.quartz.common.stats.QuartzExecStatisticsMBeanImpl@9642bd6,
[2013-06-13 16:59:03,428] I [main] [com.quartzdesk.agent.Agent:608] - Successfully registered MBean: com.quartzdesk.agent.scheduler.quartz.common.monitor.QuartzMonitoringMBeanImpl@212b19c5, o
[2013-06-13 16:59:03,429] I [main] [com.quartzdesk.agent.Agent:268] - Successfully initialized JVM agent: com.quartzdesk.agent.Agent@4e17e4ca [QuartzDesk JVM Agent v1.0.10], enabled: true
```

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 archive (`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 4.4 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 classpath. Make sure the new library is included on the application's classpath. Please refer to 4.6 for details.

5.5 Start Application

Start the standalone Java application by following steps outlined in 4.9.