vmware

HQ Installation Guide

Marie McGarry

Version: 83
Date: 9/8/10
Creator: Marie McGarry

Table of Contents

1. Installation Requirements

Topics marked with relate to HQ Enterprise-only features.*

*Feedback is welcome. Click **Add Comment** at the bottom of the page.*

1.1. HQ Server Requirements

HQ supports only one HQ Server on a single host. The host must have a static IP address for server communications.

1.1.1. HQ Server JRE Requirements

HQ Server can run with a 1.5 or 1.6 JRE. For HQ Server 4.2, Hyperic recommends a 1.6 JRE. In HQ 4.2, platform-specific HQ Server installers include a 1.6 JRE.

Use the no-JRE server installer if you prefer to use an existing JRE in your environment, or if no architecture-specific HQ Server package is available.

HQ Server can run with either a 32-bit or 64-bit JRE, except on Windows, as noted below.

> **Note for Windows Environments**
>
> HQ Server does **not** support 64-bit JREs under Windows. Use the 32-bit installer package for Windows, or the platform-independant installer if you prefer to use a pre-existing JRE in your environment.
>
> If you do use a JRE of your own, see HQ Components and JRE Location for information about how to ensure that HQ Server uses the right JRE.

Note for Solaris Sparc Environment

As of HQ 4.2 there is *no* platform-specific HQ Server package for Solaris SPARC. Install a no-JRE HQ Server package in Solaris SPARC environments. For information about how to ensure that HQ Server can determine what JRE to use, see HQ Components and JRE Location.

A platform-specific HQ Agent package with a bundled JRE is available for SPARC Solaris in a tarball.

Note: You **must** use GNU Tar to unpack HQ tarballs.

1.1.2. HQ Server Resource Requirements

For small to medium deployments, up to 50 managed platforms:

- 1 GHz or higher Pentium 4, or equivalent (2 x 2.4GHz Pentium Xeon or equivalent recommended)
- 1 GB RAM (4 or more GB recommended)
- 1-5 GB Free Disk Space

1.1.3. HQ Server Operating System Requirements

- Linux
- Solaris 10 or higher
- Mac OS X (Intel x86)
- Windows 2003 Server

Notes:

Although Hyperic does not support HQ Server running under Windows XP in production environments, the configuration works; you can run small evaluation deployments under Windows XP.

For Unix-based platforms, the libXp.so.6 X library is required. See below.

1.1.4. X Libraries on Unix-Based Platforms

On Unix-based platforms, HQ Server requires the libXp.so.6 X library to create charts and other graphics in the HQ Portal. The location of this library varies by version and provider:

- Enterprise Linux---As of Red Hat Enterprise Linux 4 and CentOS 4, libXp.so.6 is in the xorg-x11-deprecated-libs RPM.
- Debian---install the libxp6, libxt6, libxtst6, and libx11-6 packages.
- Fedora Core 5---as of Fedora Core 5, the libXp.so library has been split out to its own package; install the libXp RPM.
- Other distributions---The required libraries can be found in either the XFree86-libs or the xorg-x11-libs package.

1.1.5. HQ Server Supported Databases

HQ is packaged with a built-in PostgreSQL V8.2.5 database, which is suitable for evaluation or very small HQ deployments.
Hyperic does not support deployments that use the built-in database for production deployments with more than 25 managed platforms.

For production deployments, Hyperic recommends running the HQ database MySQL or Oracle. For large (100+ platforms) Hyperic requires the database to be located on a different host than the HQ Server and not be shared.

Hyperic supports the use of these databases for an external HQ database:

- MySQL
  - MySQL Enterprise Server and MySQL Community Server, v5.1.x (recommended)
  - MySQL Enterprise Server and MySQL Community Server, v5.0.45 or higher
- Oracle:
  - 9i or 10g for HQ 4.0
  - 10g or 11g for HQ 4.1 and later
- PostgreSQL
  - PostgreSQL 8.3
  - **Exception**: Hyperic does not support the use of PostgreSQL running under Windows in production deployments, and for production environments recommends the use of MySQL or Oracle.

1.1.6. Supported Browsers

Supported Browsers in HQ 4.3

HQ 4.3 supports these browser versions:

- Firefox 2.x
- Firefox 3.x
- Internet Explorer 7
- Internet Explorer 8, except if running on Windows 2008

> **Firefox Skype plugin causes problems**
>
> The Skype plugin for Firefox causes unexpected behavior in the HQ user interface. Disable the plugin to work around this problem.

Supported Browsers in HQ 4.2.x

HQ 4.2 supports these browser versions:

- Firefox 1.5.x

- Firefox 2.0.x
- Internet Explorer 6
- Internet Explorer 7

Firefox is recommended.

1.2. HQ Database Requirements

- 1 GHz or higher Pentium 4, or equivalent (2 x 2.4GHz Pentium Xeon or equivalent recommended)
- 2 GB RAM (4 or more GB recommended)
- 25 GB Free Disk Space
- Physical hardware (not virtualized)

1.3. HQ Agent Requirements

Hyperic supports only one HQ Agent on a host. The host must have a static IP address for agent communications.

1.3.1. Agent System Resource Requirements

- 500 MHz Celeron or higher, or equivalent
- 256 MB RAM
- 500 MB Free Disk Space

1.3.2. Agent Operating System Requirements

- Linux
- Windows XP
- Windows XP Pro
- Windows 2000
- Windows 2003 Standard Edition
- Windows 2003 Enterprise Edition
- Windows 2008 Standard Edition
- Windows 7
- Solaris 8 or higher
- Mac OS X
- HP-UX 11.11 or higher

- AIX 5.2 or higher
- FreeBSD

Host operating systems should employ a method of time sync(NTP). This is required in order to ensure accuracy of metric data reporting and alerts.

1.3.3. Agent JRE Requirements

The HQ Agent can run with either a 1.5 or 1.6 JRE. Hyperic recommends a 1.5 JRE, which is included in platform-specific agent installers.

Use the platform-independent agent installer if you prefer to use an existing JRE in your environment.

1.3.4. Drivers for Monitoring Databases

In Hyperic HQ 4.2 and later, the plugins packaged with the HQ Agent do for MSSQL, Oracle, Informix, DB2, and Sybase do not include the database vendor's JDBC plugin. After installing or upgrading to Hyperic HQ 4.2 you must download and install the vendor-provided JDBC drivers for these plugins to work.

Note: The database plugins in HQ Enterprise 4.2 still include the JDBC drivers.

2. Choose and Download an Installation Package

Hyperic HQ and Hyperic Enterprise are available for download at http://www.hyperic.com/downloads.html.

This section describes the different types of installation packages.

2.1. Full Installation vs. Agent-Only Packages

The "HQ Server Package" contains the HQ Server, including a built-in database you can use for evaluation purposes, and the HQ Agent. You can use this package to install the server, the agent, or both.

The "HQ Agent Package" contains only the agent.

2.2. Platform-Specific Versus Platform-Independent

Operating system-specific packages and platform-independent packages are available.

Platform-specific installers include a JRE, and the server package includes a built-in PostgreSQL database, suitable for use in evaluations.

In HQ 4.2 and later, platform-specific HQ Agent installers include a 1.5 JRE, and platform-specific HQ Server installers include a 1.6 JRE.

Platform-independent installers do not include a JRE, or, in the case of the HQ Server installer, the built-in PostgreSQL database.

> **Note for Solaris Sparc Environment**
>
> As of HQ 4.2 there is *no* platform-specific HQ Server package for Solaris SPARC. Install a no-JRE HQ Server package in Solaris SPARC environments. For information about how to ensure that HQ Server can determine what JRE to use, see HQ Components and JRE Location.
>
> A platform-specific HQ Agent package with a bundled JRE is available for SPARC Solaris in a tarball.
>
> **Note**: You **must** use GNU Tar to unpack HQ tarballs.

Select a platform-independent installer if you are installing components on a platform for which there is no platform-specific installer, or you want to use a JRE that is already installed on the host.

2.3. Package Formats

HQ installation packages are provided in these formats:

- Tarball archive - Tarball packages are available for Linux-like systems. HQ tarballs are archived with GNU Tar and must be unpacked to GNU Tar.

```
tar zxvf hyperic-hq-installer-4.x.y-xxx.tgz
```

> **Unpack Tarballs with GNU Tar Only**
>
> Use GNU Tar to unpacking HQ tarballs. Use of proprietary Unix Tar utilities will result in warnings. GNU Tar is available at http://www.gnu.org

- Windows MSI package - The full installer is available as an MSI package.
- Zip archive - The full installer and an agent-only distribution are available in .zip form for use on Windows platforms that do not have a tar-compatible utility. Platform-independent packages are also available in .zip format. These packages are archived with GNU Zip. Hyperic recommends the use of GNU Zip to unpack zip archives.
- RPM package - Agent-only and server-only RPM packages are available for RedHat Linux. RPM is a method of distributing software that makes it easy to install, upgrade, query and delete. RPM files contain information on the package's name, version, other file dependency information (if applicable), platform information (such as Intel or Alpha, etc.), as well as default file install locations.

Note: The RPM package for the HQ Agent does not include a JRE.

3. Set Up HQ Database

This page has links to instructions for setting up an external HQ database. Follow the instructions for the database you are using.

3.1. Set Up MySQL

*Topics marked with*relate to HQ Enterprise-only features.*
This section provides instructions for setting up MySQL as your external HQ database. It is assumed that you have already installed MySQL and are either familiar with MySQL or have the support of someone who is.

Note: If you are installing HQ for evaluation, you can use HQ's built-in PostgreSQL, rather than set up an external database.

If you are new to MySQL, the introduction to MySQL at http://dev.mysql.com/tech-resources/articles/mysql_intro.html may be of interest.

3.1.1. Create a MySQL Database Instance

Run these commands at the mysql prompt, as the root user:

```
mysql> create user 'hqadmin'@'<hq_server_host>' identified by '<passwd>';
mysql> create database HQ CHARACTER SET utf8 COLLATE utf8_bin;
mysql> grant all on HQ.* to 'hqadmin'@'<hq_server_host>';
```

UTF8 is required for encoding.

3.1.2. Configure MySQL Startup Options and System Variables

In this step, you configure the MySQL database, by editing the settings in its configuration file. In Unix and Linux, the file is /etc/my.cnf. In Windows the file is my.ini, located in the MySQL installation base directory.

For more information about InnoDB startup options and system variables, see http://dev.mysql.com/doc/refman/5.0/en/innodb-parameters.html

1. Enable the full query log. Every query (even ones with incorrect syntax) that the database server receives will be logged. This is useful for debugging, but it is usually disabled in production use. Be sure to change the paths given here to match your environment.

```
[mysqld]
log-error = /home/mysql/log/mysqld.err
log = /home/mysql/log/mysql_general.log
```

2. Print warnings to the error log file. If you have any problem with MySQL, you should enable logging of warnings and examine the error log for possible explanations.

```
log_warnings
server-id       = 1
```

3. Configure buffer pool size. The size of the MySQL buffer pool is has a significant impact on MySQL performance. If your database is on a dedicated machine, make the buffer pool about 80% of total memory.

```
innodb_buffer_pool_size = 256M
```

4. Configure the frequency with which the log buffer is written to the log, and the log is flushed to the disk. Setting this value to 0 dramatically increases MySQL performance, but with this setting, you are likely to lose data in the event of a server crash. If loss of data is unacceptable, use a value of 2 instead. Hyperic does not recommend setting the value to 1.

```
innodb_flush_log_at_trx_commit = 2
innodb_log_buffer_size = 64M
innodb_log_file_size = 256M
```

5. Configure innodb as the default storage engine---this is required.

```
default-storage_engine=innodb
bulk_insert_buffer_size = 32M
join_buffer_size = 8M
max_heap_table_size = 256M
tmp_table_size = 256M
max_tmp_tables = 48
myisam_sort_buffer_size = 256M
```

6. Configure the sort buffer size. MySQL recommends a sort_buffer_size larger than the one suggested her.

```
sort_buffer_size = 64K
```

An article on experimenting with sort buffer size is available here.

http://www.mysqlperformanceblog.com/2007/08/18/how-fast-can-you-sort-data-with-mysql

7. Configure the read buffer size. Because Hyperic does a significant volume of sequential reads, a large read buffer improve performance.

```
read_buffer_size = 1M
read_rnd_buffer_size = 10M
table_cache = 2048
set-variable = max_connections=400
key_buffer_size = 256M
thread_cache_size = 32
```

8. Configure the number of threads that can run in the InnoDB kernel. A starting point for setting this value is to to set a value equal to 2 times the number of CPUs times the number of disks.

```
innodb_thread_concurrency = 8
```

9. Set the method that is used to flush data and log files. For battery-backed-up storage with write-back cache mode on Linux OSs, the O_DIRECT flush method is good. Learn about other innodb flush methods.

```
innodb_flush_method=O_DIRECT
innodb_rollback_on_timeout=1
```

In this situation, tune your Linux OS (version 2.6 or higher) to favor the use of main memory rather than file caches:

```
# sysctl -w vm.swappiness=30
```

or

```
# echo 30 >/proc/sys/vm/swappiness
```

10. Set the size of the query cache. Generally, the higher this value, the better the performance. However, in MySQL versions older than 5.0.50, beware of setting this variable too high, as it may cause the database to pause. For more information, see the bug description at http://bugs.mysql.com/bug.php?id=21074.

```
query_cache_size = 0
```

11. The default value here is 1M. If the qcache_hits-to-qcache_inserts ratio is low, raise this value.

```
query_cache_limit = 8M
```

12. HQ requires a char encoding of utf-8.

```
default-character-set=utf8
collation_server=utf8_bin
```

3.1.3. Install the HQ Server

For instructions, see Section 4, “Installing HQ”.

3.1.4. Tune the Batch Aggregate Inserter for MySQL

NOTE: Perform these steps only after installing the HQ Server.

These tuning recommendations are based on a performance tuning exercise in an environment with 700 HQ Agents reporting to an HQ Server on an 8 way / 16 GB host with an MySQL database running on an 8 way / 8 GB host, each running CentOS 5, with

- Workers: 4
- QueueSize: 4000000
- BatchSize: 2000

With 7 hours of backfilled data the server peaked out at 2.2 million rows inserted.

This intent of the strategy was to keep the Batch Aggregate Inserter (BAI) on "cruise control", instead throwing threads at the queued metrics all at once and causing CPU spikes.

It was found that the BAI workers had no trouble keeping up with the "normal" incoming load, and in a catchup scenario (after backfilling) the high Queue Size allowed them plenty of time to catch up.

For a smaller deployment, consider only tweaking the number of workers down to 1 or 2. This will ease random CPU spikes and MySQL should have no problem keeping up with the incoming traffic.

Please NOTE these settings may not be applicable to PostgreSQL and Oracle since MySQL handles catchup scenarios much more gracefully.

To update the Batch Aggregate Inserter settings for MySQL run these commands at the mysql prompt as the hqadmin user:

```
mysql> update HQ.EAM_CONFIG_PROPS set propvalue = 4 where propkey =
 'BATCH_AGGREGATE_WORKERS';
mysql> update HQ.EAM_CONFIG_PROPS set propvalue = 2000 where propkey =
 'BATCH_AGGREGATE_BATCHSIZE';
mysql> update HQ.EAM_CONFIG_PROPS set propvalue = 4000000 where propkey =
 'BATCH_AGGREGATE_QUEUE';
```

3.1.5. Solve Problems with MySQL Configuration

If MySQL fails to start and issues a message similar to this:

```
InnoDB: Error: log file ./ib_logfile0 is of different size 0 5242880 bytes
InnoDB: than specified in the .cnf file 0 268435456 bytes!
080403  8:06:13 ERROR Default storage engine (InnoDB) is not available
080403  8:06:13 ERROR Aborting
```

the actual log size does not match the configured log size.

Delete the log files in /var/lib/mysql/ and restart MySQL.

3.1.6. MySQL Maintenance Examples

Here are examples of regular maintenance for mysql

1. Simple MySQL Backup Script

```
#!/bin/sh

START=`date '+%A %Y/%m/%d %H:%M:%S'`
DAY=`date +%A`
MYSQLADMIN="/usr/bin/mysqladmin"
MYSQLDUMP="/usr/bin/mysqldump"
USER="root"
PASSWORD="mysql"
DBNAME="hqdb"
DEST="/home/mysql/dumps/$DBNAME-$DAY.sql.gz"
flushCmd="$MYSQLADMIN -u $USER -p$PASSWORD flush-logs"
dumpCmd="$MYSQLDUMP -u $USER -p$PASSWORD --quick --single-transaction $DBNAME"
gzip="gzip"
echo "Starting backup: $START"
echo "$flushCmd && $dumpCmd | $gzip > $DEST"
$flushCmd && $dumpCmd | $gzip > $DEST
END=`date '+%A %Y/%m/%d %H:%M:%S'`
echo "Backup completed: $END"
```

2. Simple Log Rollover Scheme. This may be done with error files, log files, etc.

```
cp /path/to/mysql/log/mysqld.err /path/to/mysql/log/mysqld-`date '+%w'`.err ;
cp /dev/null /path/to/mysql/log/mysqld.err
```

3. Sample Unix Cron Entries (empty lines will fail in cron, beware)

```
#
#       Field 1: (0-59) minute
#       Field 2: (0-23) hour
#       Field 3: (1-31) day  of  the month
#       Field 4: (1-12) month of the year
#       Field 5: (0-6)  day of the week - 1=Monday
# ------------------------------------------------------------------------
```

```
#
0 2 * * * backup.sh
0 1 * * * cp /path/to/mysql/log/mysqld.err /path/to/mysql/log/mysqld-`date '+%w'`.err ;
cp /dev/null /path/to/mysql/log/mysqld.err
```

3.2. Set Up Oracle

Topics marked with relate to HQ Enterprise-only features.*
This section has instructions for configuring an Oracle database for an HQ installation.

*Feedback is welcome. Click **Add Comment** at the bottom of the page.*

3.2.1. Before Installing HQ Server

Before installing HQ Server, you create the HQ database in Oracle, configure the database, and create the database user that HQ will use to access the database. The sections that follow have instructions.

Installation Requirements

Hyperic supports:

- Oracle 10g 10.2.0.1 or higher patch level. Preferred version is 10.2.0.4
- Oracle 11g

Hyperic recommends that the Oracle server for the Hyperic database run a dedicated system, with at least 8 GB RAM allocated to SGA.

Create an Oracle instance

Install Oracle on the machine to be used, and create a database.

The database can be created with Oracle Database Configuration Assistant.

Select **New Database** (Includes datafiles = No). To save space, decline to install the Example Schemas.

Because you are running Oracle on a dedicated host, you can select the "Typical Memory" configuration.

Select "OLTP" as the type of database sizing to use. Allocate as high a percentage of system resources as you can afford: 70-90%, ideally in the higher range. See Configure Initialization Parameters for Oracle for SGA and PGA size.

Create Tablespaces

1. Create the TEMP_HQDB temporary tablespace, 2 GB in size
2. Create the TS_HQDB tablespace, 25 GB in size. This tablespace will be used to store HQ_METRIC_DATA_*D_*S tables.

Create the Database User for HQ

In this step you create the database user account that HQ will use to access the Oracle database.

There are multiple methods for creating a user in Oracle. To do it using SQL*Plus, log into the Oracle instance as the system user with SQL*Plus, and issue the create user command:

```
SQL> CREATE USER _HQUSER_ IDENTIFIED BY _HQPASSWORD_ DEFAULT TABLESPACE TS_HQDB;
```

replacing *HQUSER* and *HQPASSWORD* with desired values.

Grant Permissions to the Database User

You can grant the database user the necessary permissions in SQL*Plus with the grant command:

```
SQL> GRANT CONNECT, RESOURCE, CREATE VIEW TO HQUSER;
```

Verify the permission setting:

```
SQL> SELECT GRANTED_ROLE, DEFAULT_ROLE FROM dba_role_privs WHERE grantee = 'HQUSER';
```

Make sure that you see the following rows for CONNECT and RESOURCE roles:

```
GRANTED_ROLE DEFAULT_ROLE
CONNECT YES
RESOURCE YES
```

If that is not the case, update the permissions:

```
ALTER USER HQUSER DEFAULT ROLE RESOURCE, CONNECT;
```

Obtain Oracle JDBC Driver (Hyperic HQ only)

The HQ Server in Hyperic HQ does not include the Oracle JDBC driver. Perform these steps before installing HQ Server.

To obtain the driver:

1. Go to http://www.oracle.com/technology/software/tech/java/sqlj_jdbc/index.html
2. In the **JDBC Driver Downloads** section, select the link for your version of Oracle.
3. On the drivers download page, select the driver for your JDK version.
   - JDK 5.0 — ojdbc5.jar
   - JDK 6.0 — ojdbc6.jar
     - **Note:** JDeveloper 11g JDBC drivers support JDBC 4.0 features. Use ojdbc6.jar with JDK 6.0 for JDBC 4.0 features.
4. On the page that appears accept the licensing agreement.
5. Download the jar file.
6. Copy the driver jar file to `hyperic-hq-installer/installer-4.x.y/lib`.
7. Proceed with the HQ installation process.

3.2.2. Install HQ Server

Install HQ Server. See Section 4, “Installing HQ” for instructions.

Note: The HQ database must be up for the HQ Server installation process to succeed.

3.2.3. Enable Row Movement for Oracle 10g and 11g

After installing HQ Server, run the enable_row_movement.sql procedure attached to this page - this is necessary to enable the routine maintenance described in Periodic Oracle 10g and 11g Database Maintenance. It is only necessary to enable row movement once; you do not need to do it on a recurring basis.

3.2.4. Periodic Oracle 10g and 11g Database Maintenance

Perform the following maintenance after running Hyperic for about a week, and monthly thereafter:

1. Run the table_maintenance.sql procedure attached to this page to compact and shrink space on all HQ tables (except HQ_METRIC* tables).

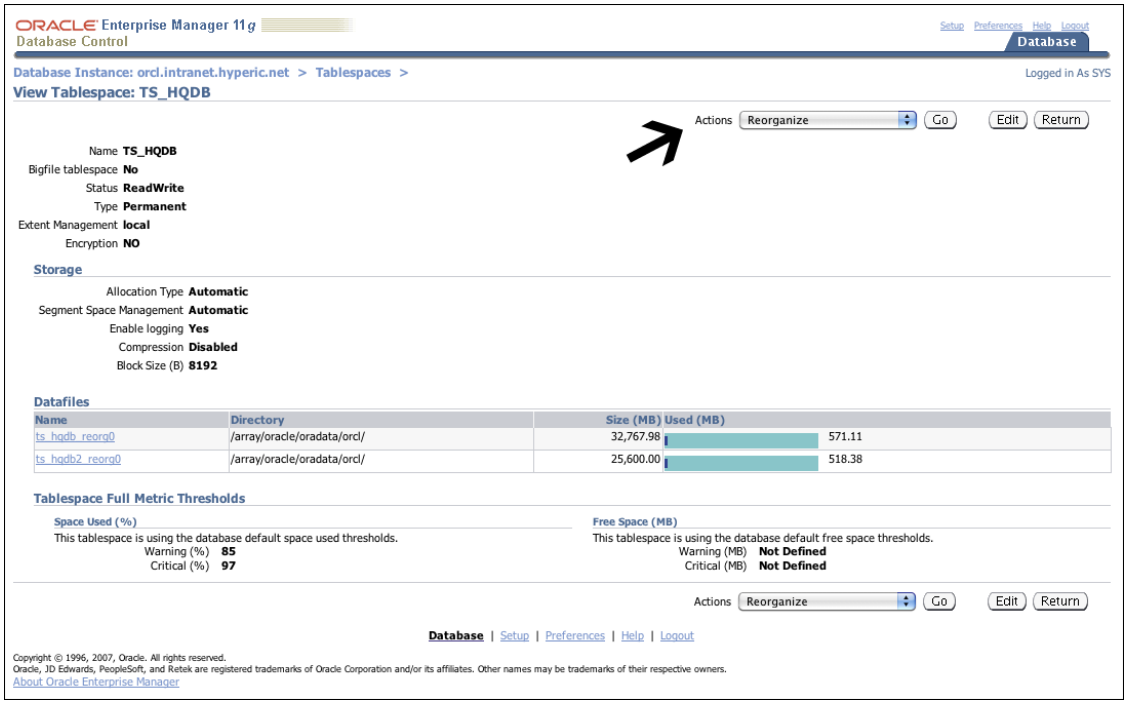
2. Run the rebuild_index.sql procedure attached to this page to rebuild the index on all HQ tables.

3. Run tablespace maintenance as appropriate:

   a. Perform the following query to determine how fragmented a tablespace is. The query returns the tablespace name and total count of holes in it.

```
select count(*), tablespace_name
from dba_free_space
group by tablespace_name
order by 1,2;
```

   b. If the hole count is in the order of thousands, run **Reorganize**.



3.2.5. Tuning HQ Database on Oracle for Medium to Large Environments

If you manage more than 100 platforms, follow the steps in this section to tune your Oracle-hosted HQ database.

Create the TS_HQDB_16K Tablespace

Create the TS_HQDB_16K tablespace, 25 GB in size, with 16 K blocksize. Here is an example of the syntax to create a table space with 16 K blocksize, where 'datafile_name.dbf' is a file where the tablespace data will physically reside:

```
CREATE TABLESPACE TS_HQDB_16K
datafile 'datafile_name.dbf' SIZE 25000M AUTOEXTEND OFF
ONLINE
```

```
PERMANENT
EXTENT MANAGEMENT LOCAL UNIFORM SIZE 128K
BLOCKSIZE 16K
SEGMENT SPACE MANAGEMENT AUTO
;
```

Configure REDO Logs

Redo logs are transaction journals. Each transaction is recorded in the redo logs. Redo logs are used in a serial fashion with each transaction queuing up in the redo log buffers and being written one at a time into the redo logs.

Configure at least three REDO logs, each 2048 MB in size.

Locate REDO logs on a separate disk spindle from datafiles.

Do not locate REDO logs on a RAID array.

Configure Initialization Parameters for Oracle

Set following Oracle initialization parameters:

```
DB_WRITER_PROCESS = 4
SGA_MAX = 8G
SGA_TARGET = 7G
SHARED_POOL_SIZE=700M
PGA_AGGREGATE_TARGET=1500M
DB_16K_CACHE_SIZE=1000M
DB_KEEP_CACHE_SIZE = 500M
FILESYSTEMIO_OPTIONS=SetAll
DB_FILE_MULTIBLOCK_READ_COUNT=16
OPEN_CURSORS=300
PROCESSES=500
```

Configure Batch Aggregate Inserter

1. Make sure that the HQ Server is shut down.
2. Log into the Oracle instance as HQ user with SQL*Plus, and issue these commands to increase the data aggregate inserter batch size to 8000 and number of workers to 10:

```
UPDATE EAM_CONFIG_PROPS SET PROPVALUE=8000 where PROPKEY='BATCH_AGGREGATE_BATCHSIZE';
UPDATE EAM_CONFIG_PROPS SET PROPVALUE=10 where PROPKEY='BATCH_AGGREGATE_WORKERS';
UPDATE EAM_CONFIG_PROPS SET PROPVALUE=5000000 where
PROPKEY='BATCH_AGGREGATE_QUEUE';
```

Move Database Tables

While still logged into the Oracle instance as the system user, run these commands to move metric and measurement tables to the TS_HQDB_16K tablespace:

```
alter table HQADMIN.HQ_METRIC_DATA_0D_0S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_0D_1S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_1D_0S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_1D_1S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_2D_0S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_2D_1S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_3D_0S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_3D_1S move tablespace TS_HQDB_16K;
```

```
alter table HQADMIN.HQ_METRIC_DATA_4D_0S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_4D_1S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_5D_0S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_5D_1S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_6D_0S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_6D_1S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_7D_0S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_7D_1S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_8D_0S move tablespace TS_HQDB_16K;
alter table HQADMIN.HQ_METRIC_DATA_8D_1S move tablespace TS_HQDB_16K;
alter table HQADMIN.EAM_MEASUREMENT_DATA_1D move tablespace TS_HQDB_16K;
alter table HQADMIN.EAM_MEASUREMENT_DATA_1H move tablespace TS_HQDB_16K;
alter table HQADMIN.EAM_MEASUREMENT_DATA_6H move tablespace TS_HQDB_16K;
```

Rebuild Indexes for Moved Tables

Run these commands to rebuild the indexes for the moved tables:

```
alter index HQADMIN.METRIC_DATA_0D_0S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_0D_1S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_1D_0S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_1D_1S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_2D_0S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_2D_1S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_3D_0S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_3D_1S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_4D_0S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_4D_1S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_5D_0S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_5D_1S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_6D_0S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_6D_1S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_7D_0S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_7D_1S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_8D_0S_MID_IDX rebuild;
alter index HQADMIN.METRIC_DATA_8D_1S_MID_IDX rebuild;
alter index HQADMIN.MEASUREMENT_DATA_1H_MID_IDX rebuild;
alter index HQADMIN.MEASUREMENT_DATA_6H_MID_IDX rebuild;
alter index HQADMIN.MEASUREMENT_DATA_1D_MID_IDX rebuild;
```

Configure Tables for High Concurrency

Run these commands for any HQ 4.x version:

```
alter table <schema>.HQ_METRIC_DATA_0D_0S initrans 15;
alter index <schema>.METRIC_DATA_0D_0S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_0D_1S initrans 15;
alter index <schema>.METRIC_DATA_0D_1S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_1D_0S initrans 15;
alter index <schema>.METRIC_DATA_1D_0S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_1D_1S initrans 15;
alter index <schema>.METRIC_DATA_1D_1S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_2D_0S initrans 15;
alter index <schema>.METRIC_DATA_2D_0S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_2D_1S initrans 15;
alter index <schema>.METRIC_DATA_2D_1S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_3D_0S initrans 15;
alter index <schema>.METRIC_DATA_3D_0S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_3D_1S initrans 15;
alter index <schema>.METRIC_DATA_3D_1S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_4D_0S initrans 15;
alter index <schema>.METRIC_DATA_4D_0S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_4D_1S initrans 15;
```

```
alter index <schema>.METRIC_DATA_4D_1S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_5D_0S initrans 15;
alter index <schema>.METRIC_DATA_5D_0S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_5D_1S initrans 15;
alter index <schema>.METRIC_DATA_5D_1S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_6D_0S initrans 15;
alter index <schema>.METRIC_DATA_6D_0S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_6D_1S initrans 15;
alter index <schema>.METRIC_DATA_6D_1S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_7D_0S initrans 15;
alter index <schema>.METRIC_DATA_7D_0S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_7D_1S initrans 15;
alter index <schema>.METRIC_DATA_7D_1S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_8D_0S initrans 15;
alter index <schema>.METRIC_DATA_8D_0S_MID_IDX initrans 15;
alter table <schema>.HQ_METRIC_DATA_8D_1S initrans 15;
alter index <schema>.METRIC_DATA_8D_1S_MID_IDX initrans 15;
```

Configure Hibernate Batch Size

Update the HQ Server's hibernate.jdbc.batch_size from the default of 30 to to 200, in:

`HQ_SERVER_HOME/hq-engine/server/default/deploy/hq.ear/META-INF/hibernate.cfg.xml`

Restart HQ Server

Restart the HQ Server.

3.3. Set Up PostgreSQL

Hyperic recommends that this setup be performed by or with the support of your database administrator.

These instructions assume that you are performing a new installation of an RPM package of PostgreSQL, using Yum, an automatic RPM package installer.

If you do not have an RPM package, you can obtain the source from the PostgreSQL website.

3.3.1. Example Configuration

The instructions that follow show commands for setting up PostgreSQL in an environment with these characteristics:

Operating system	Red Hat Enterprise Linux 4
Database	PostgreSQL 8.3
Database IP address	192.168.1.4
HQ Server IP address	192.168.1.6
Database user name	admin
Database password	hqadmin
Database location	/var/lib/pgsql/data/
PostgreSQL configuration file	/var/lib/pgsql/data/postgresql.conf
PostgreSQL authorization file	/var/lib/pgsql/data/pg_hba.conf

3.3.2. Install and Initialize PostgreSQL

Log in as root to the target Linux machine, and enter this command:

```
yum install postgresql postgresql-server
```

PostgreSQL will be installed in:

```
/etc/init.d/
/usr/bin/
/usr/share/doc/
/var/lib/pgsql/
```

3.3.3. Create PostgreSQL User

1. Change user to PostgreSQL and connect to the database locally.

```
# su postgres
```

   - The psql prompt is displayed.

2. Create a user named admin withlogin and createdb privileges.

```
create role admin with login createdb password 'hqadmin';
```

3. Create a default database for Hyperic. Place quotes around the string HQ so that the database name will be uppercase.

```
CREATE DATABASE "HQ" OWNER admin;
```

3.3.4. Configure PostgreSQL Server Options

In this step, you configure PostgreSQL Server options in the PostgreSQL server configuration file, postgresql.conf.

1. Open the postgresql.conf file.

2. The default database permissions allow local connections only. To configure PostgreSQL to listen on all network interfaces, uncomment the listen address entry and change its value as shown below.

```
listen_addresses = '*'
```

3. Add the following settings to optimize HQ performance:

```
##performance changes for HQ
shared_buffers=10000
work_mem=2048
statement_timeout=30000
```

Note: In PostgreSQL 8.3, the PostgreSQL parameter that enables HQ to monitor the database - track_counts - is enabled, because the PostgreSQL autovacuum daemon needs the collected information. Only superusers can change this setting.

3.3.5. PostgreSQL Configuration Tips for Large HQ Deployments

These changes and additions to postgresql.conf can improve HQ performance in large environments, assuming that you have at least 2GB RAM available for the database.

```
shared_buffers = 20000
commit_delay = 10000
checkpoint_segments = 15
work_mem = 8192
maintenance_work_mem = 32768
max_fsm_pages = 40000
effective_cache_size = 5000
```

In particular, increasing effective_cache_size is beneficial, given sufficient RAM.

You may need to refine your database configuration based on the performance you experience. Review of the database log by a database administrator should indicate whether further adjustments to checkpoint_segments or max_fsm_pages are appropriate.

3.3.6. Configure Client Authentication

In this step, you configure PostgreSQL to allow connections from other users and from the HQ Server.

PostgreSQL client authentication is defined in the pg_hba.conf file, which contains lines, referred to as records, that specify allowed connection types, users, client IP addresses, and authentication method. Locate this line in the file:

```
# TYPE  DATABASE USER  CIDR-ADDRESS AUTH-METHOD
```

and add these lines below it:

```
local   all      all   ident           sameuser
host    all      all *192.168.1.6/32* password
```

For more information about pg_hba.conf see [[http://www.postgresql.org/docs/8.2/interactive/auth-pg-hba-conf.html

3.3.7. Install PostgreSQL Client on the HQ Server

The purpose of installing the PostgreSQL client is so that you can verify connectivity between the server and the database. To install the client, enter these commands:

```
yum -y install postgresql
psql -d postgres -h 192.168.1.4 -U admin -W
```

Once the connectivity is tested, you can remove the PostgreSQL client with this command:

```
yum -y remove postgresql
```

3.3.8. Install the HQ Server

Install the HQ Server. For instructions, see Section 4, “Installing HQ”.

Note: Do not start the HQ Server until after completing the steps in the following section.

3.3.9. Create PLPG Language for PostgreSQL 8 Installations

Note: This step is necessary only for v8.0 of PostgreSQL.

HQ usually automatically creates a language in the PostgreSQL database. HQ is not able to create the language automatically in PostgreSQL 8.0, so for that version, you must run the following command on the HQ database before starting the HQ Server:

```
createlang plpgsql [ICG:DATABASE NAME]
```

For example:

```
createlang plpgsql HQ
```

The createlang executable is located in the bin directory of your PostgreSQL installation.

3.3.10. Start HQ Server

Start the HQ Server. For instructions, see Configure and Run the HQ Server. If the server fails to start up, there may be problems with your PostgreSQL configuration. Check the PostgreSQL logs for connection failures or errors.

3.3.11. Troubleshooting Database Connection Problems

If network connections to the database fail, you can troubleshoot the issue in PostgreSQL log files, using the UNIX® tail command with the -f parameter

tail -f displays the lines at the end of a file, and displays additional log messages that follow to the terminal. This is useful for watching log files, or any other file which may be appended over time. The following log files will have information about failed connections.

- /var/lib/pgsql/data/pg_log/postgresql-day.log
- /var/lib/pgsql/pgstartup.log

3.3.12. Useful PostgreSQL Commands

```
\h     help  with SQL commands
?      help with psql commands
\du    list roles/users
\l     list databases
\c     to choose a database
\d     to list tables once in a database
\q    quit
```

4. Installing HQ

This section has instructions for installing the HQ Server and the HQ Agent. See the tips in Section 4.1, “Before You Start”, then refer to the instructions for the package you are using.

Note: Unless you are using the internal HQ database, you must create a database for HQ before installing the HQ Server.

4.1. Before You Start

Review these topics before starting the installation process.

4.1.1. If You Are Using an External Database

If you are using an external database, setup the HQ database before installing HQ components. Refer to the instruction for the database server you use:

- Section 3.1, “Set Up MySQL”
- Section 3.2, “Set Up Oracle”
- Section 3.3, “Set Up PostgreSQL”

Make a note of the JDBC URL, database username, and database password---these values are required when you set up the HQ Server.

4.1.2. HQ Components and JRE Location

If you are installing an HQ component from a package with with a bundled JRE on a system that doesn't not have a pre-existing JRE, the component should have no problem resolving its JRE. In other circumstances, you may need to set the HQ_JAVA_HOME environment variable.

HQ Server JRE

The order of preference when resolving the HQ Server's JRE is:

1. HQ_JAVA_HOME environment variable
2. embedded JRE
3. JAVA_HOME environment variable

When installing the HQ Server with a platform-independent installer, make sure that your JAVA_HOME environment variable points to your pre-existing JRE.

If you install the HQ Server from a package with an embedded JRE, and you want the server to use a different JRE on the host, set HQ_JAVA_HOME to point to the desired JRE.

HQ Agent JRE

On Unix-based platforms, the order of preference when resolving the agent's JRE is:

1. HQ_JAVA_HOME environment variable
2. Embedded JRE
3. JAVA_HOME

On Windows, the order of preference when resolving the agent's JRE is:

1. HQ_JAVA_HOME **system** variable
2. Embedded JRE

Note: Under Windows, neither the JAVA_HOME nor the HQ_JAVA_HOME environment variables are honored when resolving the JRE for HQ 4.x Agents. To run a 4.x agent under windows you must set HQ_JAVA_HOME as as a system variable:

```
My Computer > Properties > Advanced > Environment Variables > System variables > New
```

4.1.3. To Understand Agent - Server Communications

For those new to configuring HQ components, the Understand Agent Environment and Operation summarizes key facts about how the agent and server communicate with each other.

4.2. Installing the Agent and Server from a Tarball or Zip Archive

This section has instructions for installing HQ components using the full installer.

4.2.1. About the Setup Script

The setup script, `setup.bat` for Windows or `setup.sh` for non-Windows, is in the HQ installation package. You can use to install the HQ Server, the HQ Agent, or both.

When you run the setup script, you can supply an argument that sets the installation mode.

Mode Argument	Associated Installation "Mode"
none	Quick install; the HQ components you choose to install will be installed with default settings for most configuration options---you supply installation directories only. If you install the server, it will be configured to use its built-in PostgreSQL database. This is the quickest and easiest way to install HQ.
-full	Full install; installer will prompt you to supply values for all installation options.
-upgrade	Server upgrade only; installer will prompt you for the path of the HQ server to upgrade. Upgrade to the HQ 4.0 server is supported from versions 3.x. For information on the upgrade process, see Section 5, “Upgrade HQ Components”
-postgresql	Quick install when using a standalone (not the HQ built-in) PostgreSQL database; installer will prompt you for database connection information and use defaults for other configuration settings.
-oracle	Quick install mode for Oracle; installer will prompt you for database connection information and use defaults for other configuration settings.
-mysql	Quick install mode for MySQL; installer will prompt you for database connection information and take defaults for everything else.

4.2.2. Run the Setup Procedure

Unpack the installation package if you have not already done so. These steps below are for an installation run in -full mode. This section describes the dialog that the installation script presents if you run it in -full mode, and select optional configuration options, such as an database and LDAP servers. Depending on the installation mode you select, some of the prompts described below will not appear.

1. Create a directory for HQ.

The installation dialog assumes your HQ installation directory is:

```
/home/hyperic
```

2. Unpack the tarball or zip archive.

Unpack Tarballs with GNU Tar Only

Use GNU Tar to unpacking HQ tarballs. Use of proprietary Unix Tar utilities will result in warnings. GNU Tar is available at http://www.gnu.org

3. Open a command shell.

Opening Windows Command Shell

On Windows, if you open your command shell from the **Start** menu, you must run as "Administrator", like this:

1. Start button - All Programs - Accessories.
2. Right click "Command Prompt".
3. Select "Run as".
4. Choose "Administrator" option.

On Unix-based platforms, enter:

```
PathToInstaller/setup.sh -mode
```

On Windows platforms, enter:

```
PathToInstaller\setup.exe -mode
```

where *mode* is one of the values in the table above.

Installation Prompt	Notes
Choose which software to install *1: Hyperic HQ Server* *2: Hyperic HQ Agent*	To install both the server and the agent, enter: 1,2
HQ server installation path [default '/home/hyperic']:	Accept the default, or enter a directory location. You must have write access the the location.
What port should the HQ server's web-based GUI listen on for http communication? [ICG:default '7080']:	
What port should the HQ server's web-based GUI listen on for secure https communication? [default '7443']	
What port should the HQ server use for the jnp service? [default '2099']:	
What port should the HQ server use for the mbean server? [default '9093']:	
Enter the base URL for the HQ server's web-based GUI [default...]	The URL used to access the HQ Server. This value is used in alert notification emails. This value can be changed on the HQ Server Administration page in the HQ Portal.
SMTP server information [default 'localhost]	The installer will look for an SMTP server running on the localhost. If it finds one, this question is skipped. If no SMTP server is found, it will ask you to specify one that it can use to send outbound alert-notification

Installation Prompt	Notes
	emails. In this case, if you do not specify an SMTP server, alert notifications will not be sent. Alert functionality is still enabled. Note: If you use an external SMTP server, after completing the HQ Server installation, perform the steps in Configuring HQ Server for SMTP Server.
Enter the email address that HQ will use as the sender for email messages [default...]	The email address of the HQ Administrator. Note that most mail servers will not deliver mail without a valid domain name in the From field.
What backend database should the HQ server use? [default '1']: *Choices:* *1: HQ Built-in Database* *2: Oracle 9i/10g* *3: Oracle 11i* *4: PostgreSQL* *5: MySQL 5.x*	
If in the previous step, you chose the HQ Built-in Database, this prompt appears: *What port should HQ's built-in database use? [default '9432']:*	

If instead you selected an external database, these prompts appear:
• *Enter the JDBC connection URL.* The prompt supplies a default URL, which assumes the external database is on local host.
• *Enter the username to use to connect to the database:*
• *Enter the password to use to connect to the database:*
	Supply:
	• a URL in the form shown, editing as appropriate to identify the connection details, such as host and name.
	Note: If you use MySQL for the HQ database, Hyperic recommends you add these parameters to the database URL:
	`connectTimeout=60000&amp;socketTimeout=60000`
	resulting in an URL like:
	`jdbc:mysql://localhost:3306/hqdb?connectTimeout=60000&amp;socketTimeout=60000`
	• the database username that was set up when the HQ database was created.
	• the database password that was set up when the HQ database was created.

What should the username be for the initial admin user? [default 'hqadmin']:	
What should the password be for the initial admin user?:	The installer will not echo the password but will prompt for it twice so it can be verified.
**What should the email address be for the initial admin user? [default...]*	

Installation Prompt	Notes
If the installation procedure does not detect a local LDAP server, it offers the option of configuring the use of an external LDAP authorization data source. If you accept that option, you will be prompted for LDAP connection information.	LDAP authentication can also be configured after HQ is installed. For details see on how to configure HQ for LDAP authentication, see Configure LDAP Properties.
HQ agent installation path [:default '/applications/hyperic']:	

The installer indicates the installation was successful, provides the URL for the HQ Portal along with the default username and password, and returns you to the command prompt.

4.3. Installing HQ Using the Windows MSI Installer

This section has instructions for installing components using the HQ Windows MSI installation package. You can install the HQ Server, the HQ Agent, or both. The MSI installer can be run interactively or in silent mode.

4.3.1. Known Issues on Windows Vista

These requirements apply when you use the MSI installer on Windows Vista:

- Installation requires administrator privileges and permissions to install Windows services.
- Uninstallation must be performed by the same user who performed the installation.

4.3.2. Installing HQ Interactively Using the HQ MSI Installer

Follow these steps to run the MSI installer interactively:

1. Double-click the MSI package.
   - The SpringSource license agreement is displayed.
2. Accept the license agreement and click **Next**.
   - The **Setup Type** window is displayed.
3. Choose **Complete** to install both the HQ Server and the HQ Agent, or **Custom** to install one or the other.
   - If you chose a complete installation, the **Destination** window is displayed, and supplies a default installation folder in the root of the Program Files folder.
4. If desired, select a different installation folder, and click **Next**.
   - The **Ready to Install...** window displays the selected installation location, components to install, and a check box for choosing whether or not to install the agent and server as Windows services - uncheck the box if you don't want the components to run a services.
5. Click **Install**.

4.3.3. Solving Service Startup Problems After MSI Install

If you install the HQ Server and the HQ Agent on the same machine, and accept the default "Start Hyperic HQ Services when install completes" option, agent startup problems can result.

In this scenario, as a last step, the installer will issue a server start command, followed by an agent start command.

The agent must contact the server to start up successfully. If the machine the HQ components run on is slow or busy, the HQ Server can take a long time to start. The HQ Agent makes a finite number of attempts to connect to the server, and if it continues to fail, the agent gets stuck. No software will be auto-discovered on the platform and the agent will not appear in the HQ user interface.

To solve this problem, force the agent to repeat the setup process by entering this command in a shell:

```
AGENT_HOME/bin/hq-agent.bat setup
```

4.3.4. Syntax for Running MSI Installer in Silent Mode

You can run the HQ MSI installer in silent mode from the DOS prompt. For example:

- To install Hyperic HQ, type the following command in a terminal window, supplying the desired installation location, correct installer file name, and the desired installation properties - defined below in MSI Silent Mode Properties. The /qn switch turns off the user interface.

```
%Comspec% /c msiexec /i "installer_path\Hyperic HQ 4.1.msi" /qn PROPERTY1=VALUE1
 PROPERTY2=VALUE2 ...
```

- To uninstall HQ Enterprise in silent mode, type the following command in a terminal window. You must uninstall from the same user account that used to perform the installation.

```
%Comspec% /c msiexec /x "installer_path\Hyperic HQ Enterprise 4.1.msi" /qn
```

> **Opening Windows Command Shell**
>
> On Windows, if you open your command shell from the **Start** menu, you must run as "Administrator", like this:
>
> 1. Start button - All Programs - Accessories.
> 2. Right click "Command Prompt".
> 3. Select "Run as".
> 4. Choose "Administrator" option.

4.3.5. Example Silent Mode MSI Invocations

- To silently install HQ Server and the HQ Agent on a local machine under "C:\hyperic":

```
%Comspec% /c msiexec /i "installer_path\Hyperic HQ 4.1.msi" /qn INSTALLDIR="C:\hyperic"
```

- To install (locally) an HQ Agent that will communicate securely with the HQ Server at 69.59.181.106:

```
%Comspec% /c msiexec /i "installer_path\Hyperic HQ 4.1.msi" /qn ADDLOCAL=Agent
AGENT_IS_SECURE=1 AGENT_SERVER_ADDRESS=69.59.181.106 AGENT_SERVER_USER=hqadmin
AGENT_SERVER_PASSWORD=password
```

- To install the HQ Server and the HQ Agent on a local machine using an MSI installer on a remote machine that is accessible on the network:

```
%Comspec% /c msiexec /i "\\network_path\Hyperic HQ 4.0.msi" PROPERTY1=VALUE1
PROPERTY2=VALUE2 ...
```

- To silently install the HQ Server and the HQ Agent to a local machine using an MSI installer on a remote machine that is accessible on the network, include the /qn switch to turn off the user interface:

```
%Comspec% /c msiexec /i "\\network_path\Hyperic HQ 4.0.msi" /qn PROPERTY1=VALUE1
PROPERTY2=VALUE2...
```

4.3.6. MSI Silent Mode Properties

Installation properties for installing HQ in MSI silent mode are described below.

Note that all properties and their values are case-sensitive.

Properties that begin with the strings SERVER and AGENT are server and agent properties, respectively.

Silent Mode Property	Description	Default
INSTALLDIR	Directory where the HQ components will be installed.	C:\Program Files\Hyperic HQ 4.0
ADDLOCAL	Comma-separated list of components to be installed. Allowable case-sensitive values are Agent and Server. Use this property if you want to install only the HQ Server or the HQ Agent.	If you do not specify ADDLOCAL, both agent and server will be installed.
HQ_ENGINE_JNP_PORT	The JNDI listen port. The value assigned will be saved in the `hq-engine.jnp.port` property in `server.conf`.	2099
HQ_ENGINE_PORT	The JRMP listen port. The value assigned will be saved in the `hq-engine.server.port` property in `server.conf`.	9093
SERVER_ADMIN_EMAIL	HQ Server Administrator's email address. Default is based on values of SERVER_ADMIN_USER and SERVER_MAIL_HOST properties.	SERVER_ADMIN_USER@ SERVER_MAIL_HOST
SERVER_ADMIN_USER	Specifies the user name of the original admin user in HQ. Will be used in configuring that account.	hqadmin
SERVER_ADMIN_PASSWORD	Specifies the password of the original admin user in HQ server. Will be used in configuring that account.	hqadmin
SERVER_DATABASE_USER	Defines the username the HQ server will use when connecting to the HQ database. The value assigned will be saved in the `server.database-user` property in `server.conf`.	hqadmin

Silent Mode Property	Description	Default
SERVER_DATABASE_PASSWORD	Defines the password the HQ server will use when connecting to the HQ database. The value assigned will be saved in the `server.database-password` property in the `server.conf` file.	hqadmin
SERVER_MAIL_HOST	The IP address or hostname of the SMTP server that the HQ server will use for sending alerts and other HQ-related emails. Most UNIX platforms have a local SMTP server. If you wish to use a non-local SMTP server, specify the address with this property. The value assigned will be saved in the `server.mail.host` property in `server.conf` file.	127.0.0.1
SERVER_MAIL_SENDER	The 'From' address in email notifications from the HQ Server.	SERVER_ADMIN_EMAIL
SERVER_POSTGRESQL_PORT	HQ Server's embedded database listen port.	9432
SERVER_WEBAPP_HOST	Specifies the HQ Server's listen address for HQ Portal communications. By default, this property and AGENT_SERVER_ADDRESS have the same value. If you wish, you can use these properties to designate different hosts for agent-server and agent-portal communications. The value assigned will be saved in `server.webapp.host` in `server.conf`.	host IP address
SERVER_WEBAPP_PORT	Specifies the HQ Server listen port on which the server listens for HQ Portal communications in non-secure mode. By default, this property and AGENT_SERVER_PORT have the same value. If you wish, you can use these properties to designate different ports for agent-server and agent-portal communications. The value assigned will be saved in the `server.webapp.port` property in `server.conf`. The HQ Portal Dashboard will be located at the URL of the form:	7080

Silent Mode Property	Description	Default
	`http://SERVER_WEBAPP_HOST:SERVER_WEBAPP_PORT`	
SERVER_WEBAPP_SECURE_PORT	Specifies the HQ Server port on which the server listens for HQ Portal communications in secure mode. The value assigned will be saved in the `server.webapp.secure.port` property in `server.conf`.	7443
HQ_START_SERVICES	Property indicating whether the HQ Agent and Server processes should be started as a Windows Service at the end of the installation. 1 indicates true, 0 indicates false.	1
AGENT_ADDRESS	The IP address to which the agent binds at startup. The default value allows the agent to listen on all IP addresses on the the agent host. The value assigned is saved in both the agent.listenIp and the agent.setup.agentIP properties in agent.properties. If there is a firewall between the agent and the server, set AGENT_ADDRESS to the firewall address. After installation is complete, set agent.listenIP in agent.properties to the agent's local IP address, and configure the wall to forward agent-bound traffic to that address.	host IP address
AGENT_IS_SECURE	Indicates whether communications between the HQ Agent and the HQ Server should take place over a secure encrypted channel. 1 indicates secure communications, 0 indicates that communications will not be secured. The setting will be stored appropriately in the agent.setup.camSecure property in agent.properties.	0
AGENT_PORT	The port on the agent's listen address to which the agent binds at startup. This value is saved to both agent.setup.agentPort and agent.listenPort in agent.properties.	2144

Silent Mode Property	Description	Default
AGENT_SERVER_ADDRESS	Specifies the IP address the agent connects to to reach the HQ server. The value is saved to agent.setup.camIP in agent.properties.	host IP address
AGENT_SERVER_USER	The HQ username to use when registering itself with the server. The value is saved to agent.setup.camLogin in agent.properties. Typically this property and AGENT_SERVER_PASSWORD have the same values as SERVER_ADMIN_USER and SERVER_ADMIN_PASSWORD respectively. However, if you are installing a server and an agent on the same host, and the agent will report to a server on a different host, you might specify different credentials for AGENT_SERVER_USER/ AGENT_SERVER_PASSWORD and SERVER_ADMIN_USER/ SERVER_ADMIN_PASSWORD.	hqadmin
AGENT_SERVER_PASSWORD	The password for the user specified by AGENT_SERVER_USER. The value is saved to agent.setup.camPword in agent.properties.	hqadmin
AGENT_SERVER_PORT	Port on server port to use for non-secure communications with the server. The value is saved to agent.setup.camPort in agent.properties.	7080
AGENT_SERVER_SSL_PORT	Port on server to use for SSL communications with the server. The value is saved to agent.setup.camSSLPort in agent.properties.	7443

4.3.7. Silent MSI installation to Multiple Hosts Using Push Techniques

This section describes alternatives for doing silent MSI installs to multiple machines.

Using AT or SOON to start a process on a remote workstation

The AT and SOON commands can be used to schedule commands at a future time. AT, which is built into the command processor, schedules commands and programs to run on a local or remote computer at a specified time. Instead of running processes at a specific time, the SOON command runs them after a specified delay. SOON.EXE is available as a free Microsoft download.

Here are examples of how to run these commands:

```
AT targetPC 10:30 /INTERACTIVE \\myPC\myShare\quietInstall.bat
SOON targetPC 30 /INTERACTIVE \\myPC\myShare\quietInstall.bat
```

Executing processes on a remote system has security implications.

- The local machine must have sufficient privileges to start a batch routine on a remote system.
- You must establish privileges for the remote system to access network resources when running the install batch routine. When the command processor runs your batch routine on the target system, it executes with Local System privileges. It is therefore necessary for the batch routine to open a privilege pipe to the network resource containing the MSI package. A workaround is to add a NET command to your batch routine, as demonstrated in this sample quietInstall.bat:

```
net use * \\myPC\myShare /user:domain\username password /persistent:no
%Comspec% /c msiexec /i "\\myPC\myShare\Hyperic HQ 3.2.msi" /qn
```

Using PsExec to Start a Process on a Remote System

The Windows PsExec utility is a freely distributed light-weight telnet replacement that lets you execute processes on other systems, complete with full interactivity for console applications, without having to manually install client software. PsExec's most powerful uses include launching interactive command prompts on remote systems and remote-enabling tools to show information about remote systems. It can be downloaded as part of the Sysinternal PsToolspackage.

PsExec can be used to run the batch routine at a remote workstation by invoking the following command:

```
psexec targetPC -u domain\username -p password -i -c -f
\\myPC\myShare\quietInstall.bat
```

In the example above, domain\username has local administrative privileges for the targetPC machine. In addition, it should have the necessary privileges to access the myShare folder on the myPC machine.

The batch file quietInstall.bat is used to invoke the MSI installer over the network on the myPC machine. A sample quietInstall.bat might contain the following command:

```
%Comspec% /c msiexec /i "\\myPC\myShare\Hyperic HQ 3.2.msi" /qn
```

Remote Installs with Microsoft Management Console

This section has information on how to automatically install components to a group of machines, using Microsoft Management Control and Active Directory.

With Windows Group Policy, HQ components can be automatically installed on a group of machines by performing the following steps:

1. Log on to the domain controller.

2. Copy the MSI file into a folder that is shared with access granted to all target machines.
3. Open the Microsoft Management Control (MMC) Active Directory Users and Computers snap-in.
4. Navigate to the group of computers onto which an HQ component is to be deployed.
5. Open Properties.
6. Open Group Policies.
7. Add a new polices, and edit it.
8. In Computer Configuration/Software Installation, chose New/Package.
9. Select the MSI file through the network path.
10. Optionally, select that you want HQ to be uninstalled if the computer leaves the scope of the policy.

Propagation of group policy propagation typically takes some time. In order to reliably deploy the HQ MSI package, all machines should be rebooted.

4.4. Installing an Agent-Only Package

This section has instructions for installing a single HQ Agent. If you have multiple agents to install, see Deploying Multiple HQ Agents.

4.4.1. Installing an Agent from an Agent-Only Tarball

On non-Windows systems, the HQ Agent is automatically installed as a daemon.

1. Create a directory for the HQ Agent.
2. Unpack the tarball into the agent directory.
   - Starting the agent will run it as a daemon process.

> **Unpack Tarballs with GNU Tar Only**
>
> Use GNU Tar to unpacking HQ tarballs. Use of proprietary Unix Tar utilities will result in warnings. GNU Tar is available at http://www.gnu.org

For instructions on how to configure the agent, see Configure and Run the HQ Agent.

4.4.2. Installing an Agent from an Agent-Only Zip Archive

To install the HQ Agent as a Windows Service on a Windows system:

1. Create a directory for the HQ Agent.
2. Unpack the archive into the agent directory.
3. Open a command shell and use this command to install the HQ Agent as a Windows Service:

```
AGENT_HOME\bin\hq-agent.bat install
```

For instructions on how to configure the agent, see Configure and Run the HQ Agent.

4.5. Installing an RPM Package

This section has key facts for Linux administrators who will install HQ components from RPM packages. It is assumed that the administrator performing RPM installations is familiar with RPM packages and installation processes.

4.5.1. What You Should Know About HQ RPM Packages

The HQ Agent RPM does not include a JRE. Agent hosts must have the J2RE virtual package installed. A Sun 1.5 JRE is recommended.

The HQ Server RPM includes a 1.6 JRE and the built-in PostgreSQL database. Note that this RPM is based on the standard HQ Server installation script, which is wrapped in an Expect script. The HQ Server RPM is primarily intended for evaluation installations in environments that dictate the use of RPM.

4.5.2. What You Need to Do Before Installing an HQ RPM

- **Configure HQ Agent Properties** - An HQ Agent obtains the settings it needs to connect to and communicate with the HQ Server at first startup, either interactively, or from startup properties that can be specified in the `agent.properties` file. If you wish to automate agent installation and configuration, you must edit `agent.properties` to specify the startup properties. For more information, see Configure Agent - Server Communication in Properties File.
- **Check Path to JRE** - The agent init script is installed as `/etc/init.d/hyperic-hq-agent`, and is added to the appropriate run-levels via **chkconfig**. The script assumes the path to your JRE is /usr/java/jdk1.5.0_12. If this is not the case on the target host, you must modify the init script and specify the path to your JRE in the `HQ_JAVA_HOME` environment variable.
- **Open Firewall Port if Necessary** - If **iptables** (a host-based firewall tool typically enabled by default on Redhat and Fedora installations) is configured, you may need to open up the port for communication from the HQ Server, using a command similar to this:

```
`/sbin/iptables -ARH-Firewall-1-INPUT -p tcp --dport 2144 -j ACCEPT`
```

  Additional configuration may be required if SELinux is enabled.
- **Start Local SMTP Server** - The RPM installer requires that your SMTP server is listening on port 25 on the host where you install HQ Server.

4.5.3. What the RPM Package Does

If the "hyperic" user and "hyperic" group do not exist, they are created.

The /opt/hyperic directory is created if it does not already exist. This is set as the "hyperic" user's home directory if that user did not exist previously, and appropriate permissions are set.

The /opt/hyperic/hq-plugins directory is created to hold custom plugins.

The agent is installed in /opt/hyperic-hq-agent.

4.5.4. RPM Support Files

If you wish to create your own RPMs for installing HQ components, you can download rpm_support_files_EE.tgz, which contains the RPM spec files, init scripts, and other necessary files, from http://www.hyperic.com/downloads/

The spec file is noarch-EE.spec. The init script is hyperic-hq-agent.init.rh.

4.5.5. What to Do After Installing the HQ Server and HQ Agent

If you have installed both the server and an agent, start the server first, and then start the agent.

- For information about starting and configuring the server see Configure and Run the HQ Server.
- For information on starting the agent the first time, see Configure and Run the HQ Agent.

4.6. What to Do After Installing the HQ Server and HQ Agent

Topics marked with relate to HQ Enterprise-only features.*

*Feedback is welcome. Click **Add Comment** at the bottom of the page.*

4.6.1. Obtain and Install Drivers for Monitoring Databases

In Hyperic HQ 4.2 and later, the plugins packaged with the HQ Agent for MSSQL, Oracle, Informix, DB2, and Sybase do not include the database vendor's JDBC plugin. After installing or upgrading to Hyperic HQ 4.2 you must download and install the vendor-provided JDBC drivers for these plugins to work.

Note: The database plugins in HQ Enterprise 4.2 still include the JDBC drivers.

4.6.2. Starting HQ Components

If you have installed both the server and an agent, start the server first, and then start the agent.

For information about starting and configuring your HQ server see Configure and Run the HQ Server.

For information on starting the agent the first time, see Configure and Run the HQ Agent.

4.6.3. Installing Additional Agents

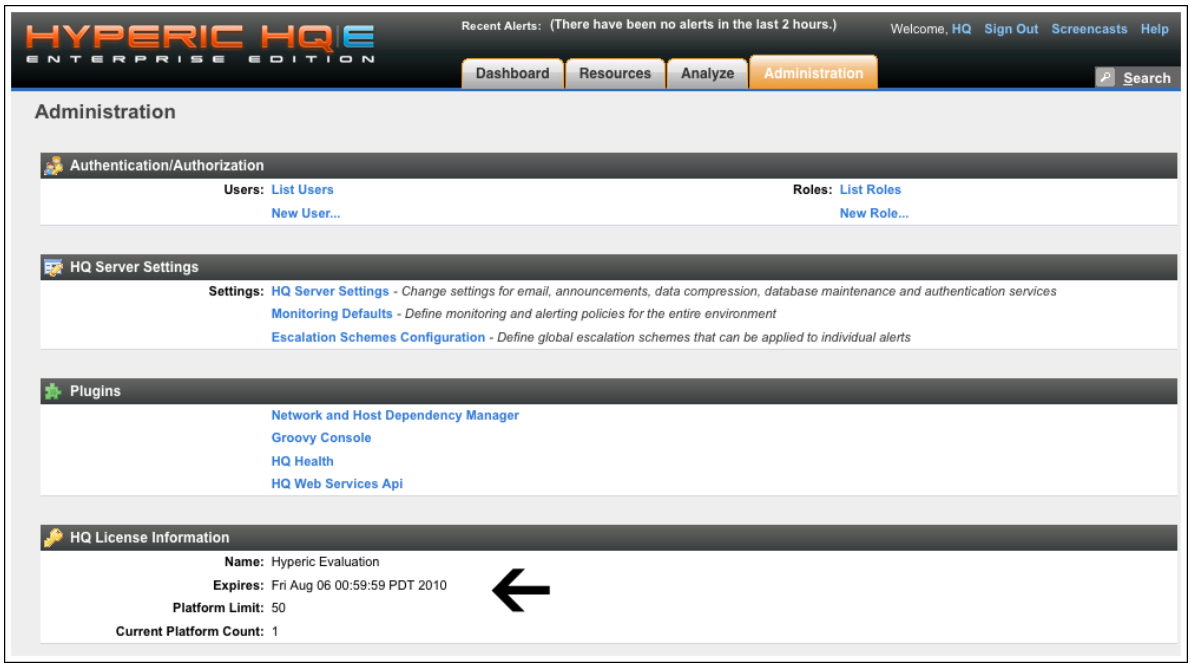
If you have installed the server only, and are going to install one or more agents on other platforms, see Section 4.4, “Installing an Agent-Only Package” or Deploying Multiple HQ Agents.

4.7. Installing an HQ Enterprise License

In HQ Enterprise, the `license.xml` file in the HQ Server's `/conf` directory specifies the number of platforms you are licensed to manage, and the expiration date of the license, as applicable. Perpetual licenses have no expiration date.

HQ Enterprise evaluation distributions include a time-limited license for 50 platforms.

You can view your license terms on the **HQ License Details** section of the HQ **Administration**, as shown in the screenshot below.



After you purchase HQ Enterprise, you download `license.xml` from the SpringSource Portal and install it in `ServerHome/conf`. You need to obtain a new license to increase the number of managed platforms or upon expiration of your license.

HQ sends an email notification of upcoming expiration starting 45 days prior to the expiration date.

4.8. Uninstalling an Agent

If the agent is managed by HQ, remove the platform for the agent before uninstalling it. Then, simply delete the agent's installation folder.

5. Upgrade HQ Components

Topics marked with relate to HQ Enterprise-only features.*

This section has instruction for upgrading your HQ deployment to a new version. You should upgrade both the HQ Server and the HQ Agent to the same version.

*Feedback is welcome. Click **Add Comment** at the bottom of the page.*

5.1. Upgrade HQ Server

You upgrade the HQ Server using the full installer, using the upgrade option. (The installer does not upgrade the HQ Agent.)

5.1.1. What Happens During Server Upgrade

The installer installs a new version of HQ Server; it obtains the configuration information from your previous server installation and configures the new server instance accordingly.

If you use HQ's internal database, the installer creates a new database instance that contains the data from the existing instance. The new instance has an updated schema, but the PostgreSQL server itself in not upgraded to a new version.

If you use an external database, the installer updates the existing instance.

Upgrading HQ Server External Database

When you perform a fresh installation and have an existing HQ database, the installation procedure will offer the option to upgrade the existing database, to overwrite the existing database, or to exit the installation. Note that if your database is quite large, overwriting may take a very long time. In large deployments, you can avoid the problem by dropping and recreating the HQ database schema before installing a new version of HQ Server. The commands to do so are:

```
drop database hqdb;
create database hqdb;
grant all on hqdb.* to 'hqadmin'@'%' identified by 'hqadmin'
grant all on hqdb.* to 'hqadmin'@'localhost' identified by 'hqadmin'
```

5.1.2. Upgrade HQ Server on Unix-Based Platforms

1. Stop the current server instance. For example:

```
/opt/hyperic/server-4.1.1/bin/hq-server.sh stop
```

2. **If you use an external HQ database, back it up before proceeding.**

3. Run the HQ installer in upgrade mode. For example:

```
/opt/hyperic/hyperic-hq-installer/setup.sh -upgrade
```

4. You are prompted to acknowledge the SpringSource license agreement.

5. The installer prompts for the path to the previous HQ Server instance. Enter the path, for example:

```
/opt/hyperic/server-4.1.1
```

6. The installer prompts for the path to the new server instance. Enter the path to the directory under which the new server instance will be installed. For example, to install the new instance under your existing HQ home directory:

```
/opt/hyperic
```

The installer will finish the upgrade.

7. Archive your old HQ Server directory, so that if you want, you can revert to the previous version. For example:

```
tar -zcvf hq-server-4.1.1-archive.tgz hq-server-4.1.1-EE
```

8. Start the new server instance. For example:

```
/opt/hyperic/server-4.2.0/bin/hq-server.sh start
```

5.1.3. Upgrade HQ Server on Windows Platforms

1. Stop the existing server instance using the Windows Services Control Panel.

2. Follow the instructions that apply, depending on whether you use the HQ built-in database or an external database:

- If you use the built-in HQ database, stop the HQ database using the Windows Services Control Panel. The upgrade process will migrate your database schema to the latest edition. Note that PostgreSQL itself is not upgraded to the latest version that ships with the HQ. The database server remains the one installed when you first installed HQ Server.
- **If using an external database, back it up.**

3. Run the HQ installer in upgrade mode:

```
c:\hyperic\hyperic-hq-installer\setup.bat -upgrade
```

4. You are prompted to acknowledge the SpringSource license agreement.

5. The installer prompts for the path to the previous HQ Server instance. Enter the full path to your existing server installation, for instance:

```
c:\hyperic\server-4.1.1
```

5. The installer prompts for the path where the upgrade version should be installed. Enter the path to the directory that will contain the new server installation. For instance, to install the new instance under your existing HQ home directory:

```
c:\hyperic\
```

The installer will finish the upgrade.

6. Archive your previous HQ Server directory so that if you wish you can revert to the previous version.

7. Update the Windows Service with the new version information:

```
c:\hyperic\server-4.2.0\bin\hq-server.exe -i
```

8. Start the upgraded HQ Server using the Windows Services Control Panel.

5.1.4. Solving Problems with Upgraded Servers with an Oracle Database

If you are upgrading an HQ installation with an Oracle backend and you experience any of the following errors during upgrade, follow the steps below to resolve the problem.

```
Error updating EAM_SERVICE.SERVICE_TYPE_ID: java.sql.SQLException:
ORA-02296: cannot enable (HQDBUSER.) - null values found
Error executing statement desc=null SQL=[
ALTER TABLE eam_stat_errors DROP CONSTRAINT rt_errs_fk_rstat CASCADE
] java.sql.SQLException: ORA-02443: Cannot drop constraint - nonexistent
constraint
```

Fix this with these steps:

1. Restore your database from backup.

2. Execute this SQL:

```
DELETE FROM EAM_SERVICE WHERE SERVICE_TYPE_ID IS NULL;
```

3. Re-run the upgrade.

5.1.5. Clear Browser Cache Before Using the Portal

If you are upgrading to 4.x from a 3.2.x or 3.1.x version, before using the HQ user interface, users must clear the browser cache, or reload the Dashboard using the Shift Refresh key sequence.

If you do not clear the browser cache, portions of the HQ user interface may display improperly.

5.2. Upgrade HQ 3.1.x and 3.2.x Agents

These instructions apply to both the open source and enterprise editions of HQ.

You must use an agent-only package to upgrade 3.1.x and 3.2.x agents to 4.x.

1. Stop the 3.1.x or 3.2.x agent.

2. Unpack the 4.x agent into the agent installation directory.

3. To preserve your previous configuration settings, copy property settings that you have customized from the 3.2.x or 3.1.x agent.properties file into the 4.x properties file, in *AgentHome*/conf.

*Note: There are new properties as pf HQ 4.0, so you cannot use a 3.x properties file.

4. Install the HQ Agent service (Windows only):

```
AgentHome\bin\hq-agent.bat install
```

5. Start the agent. For instructions, see Start the HQ Agent.

Notes:

- The first time you start the agent, it will prompt for startup settings. If you prefer, you can supply the settings in the agent properties file before starting the agent the first time. For instructions, see Configure Agent - Server Communication in Properties File.
- When you first upgrade a HQ Enterprise 3.1.x or 3.2.x agent to 4.0 or later, you cannot configure unidirectional communications at first startup. You must configure bidirectional communications at first startup of an agent upgraded from 3.x to 4.x, and then, follow the instructions in Changing Between Unidirectional and Bidirectional Agent Communications.

5.3. Upgrade HQ 4.x Agents

5.3.1. Push Agent Bundle from the HQ Server

Available only in HQ Enterprise

Known upgrade issue in HQ 4.3 on HP-UX

A known problem in HQ 4.3 causes an agent upgrade performed from the HQ user interface to fail, because of incorrect permission settings for a shared library file, `libsigar-pa-hpux-11.sl`. The upgrade fails, and this error is written to `agent.log`:

{{2010-04-26 11:43:49,347 DEBUG [WrapperStartStopAppMain] [Sigar] /local0/hq/hyperic-hq-agent-4.3.0-EE/bundles/agent-4.3.0-EE-1411/pdk/lib/libsigar-pa-hpux-11.sl: Permission denied org.hyperic.sigar.SigarException: /local0/hq/hyperic-hq-agent-4.3.0-EE/bundles/agent-4.3.0-EE-1411/pdk/lib/libsigar-pa-hpux-11.sl: Permission denied}}

To avoid this problem, set execute permissions on `libsigar-pa-hpux-11.sl`:

1. On the HP-UX server where the agent was upgraded, go to the `<hq-agent_install_path>/bundles/<hq-agent_upgraded_version>/pdk/lib` directory.
2. As the directory owner, run the **chmod 755** command on the `libsigar-pa-hpux-11.sl` file.
3. Go to the `<hq-agent_install_path>/bin` directory.
4. Restart the HQ Agent.

You can update one or more HQ Agents by pushing the new bundle to it from the HQ Server, using the HQ user interface. The bundle must reside in the /hq-agent-bundles directory of the HQ Server installation:

```
ServerHome/hq-engine/server-4.x.y-EE/default/deploy/hq.ear/hq-agent-bundles
```

The agent upgrade command is available on the **Views** tab for an HQ Agent (or a group of HQ Agents). For more information, see Agent Control Commands.

Note: When you update an agent bundle, its previous agent configuration is preserved.

5.3.2. Upgrade a 4.x Agent Bundle Manually

Available only in HQ Enterprise

Follow these steps if you wish to manually upgrade the agent bundle in your agent installation, instead of pushing the bundle from the HQ Server.

Note: When you update an agent bundle, your previous agent configuration is preserved.

1. Copy the agent bundle (agent-4.x.y-nnn.tgz or agent-4.x.y-nnn.zip) from

```
ServerHome/hq-engine/server-4.x.y-EE/default/deploy/hq.ear/hq-agent-bundles
```

to

```
AgentHome/bundles
```

2. Unpack the agent bundle.

3. Edit the rollback.properties file in AgentHome/conf to specify the location of the new agent bundle and the bundle it will supersede.

Property	Description	Example
HQ_AGENT_BUNDLE	Name of directory with the new bundle, without full path specification.	agent-4.1.0-EE-875
HQ_AGENT_ROLLBACK_BUNDLE	Name of directory with the old bundle (the one you are upgrading from), without full path specification.	agent-4.1.0-EE-874

4. Restart the agent. For instructions, see Restart the HQ Agent.

Note: The first time you start the agent, it will prompt for start settings. If you prefer, you can supply the settings in the agent properties file before starting the agent the first time. For instructions, see Configure Agent - Server Communication in Properties File.

If the upgrade to the new agent bundle fails, an attempt will be made to start the agent using the old agent bundle.

You can determine whether the upgrade was successful and what version you are running by looking at the log files in AgentHome/logs.

5.3.3. Create a Custom 4.x Agent Upgrade Bundle

Available only in HQ Enterprise
This section describes how to create a custom agent bundle. Pre-configuring the agent eases the process of upgrading multiple agents. For additional information, see Deploying Multiple HQ Agents.

1. Back up an existing agent located in:

```
ServerHome/hq-engine/server-4.x.y-EE/default/deploy/hq.ear/hq-agent-bundles
```

For example:

```
cp ServerHome/hq-engine/server-4.z.y-EE/default/deploy/hq.ear/hq-agent-bundles/agent-4.0.0-
EE-802.tgz
ServerHome/hq-engine/server-4.x.y-EE/default/deploy/hq.ear/hq-agent-bundles/agent-4.0.0-
EE-802.tgz.bak
```

2.; Extract the bundle. For example:

```
tar xzf ServerHome/hq-engine/server-4.x.y-EE/default/deploy/hq.ear/hq-agent-bundles/
agent-4.0.0-EE-802.tgz
```

This results in a new directory corresponding to the agent bundle, like this:

```
ServerHome/hq-engine/server-4.x.y-EE/default/deploy/hq.ear/hq-agent-bundles/agent-4.0.0-
EE-802
```

3. Update the contents of expanded directory. For instance, you could add custom plugins to the plugins directory:

```
ServerHome/hq-engine/server-4.x.y-EE/default/deploy/hq.ear/hq-agent-bundles/agent-4.0.0-
EE-802/pdk/plugins
```

4. Rename expanded directory to the name of custom agent bundle. For example:

```
mv ServerHome/hq-engine/server-4.x.y-EE/default/deploy/hq.ear/hq-agent/bundles/agent-4.0.0-
EE-802
ServerHome/hq-engine/server-4.x.y-EE/default/deploy/hq.ear/hq-agent-bundles/my-bundle
```

5. Pack up agent bundle, using the directory name from from the previous step as the tarball file name. For example:

```
{page-break}
tar cvf ServerHome/hq-engine/server-4.x.y-EE/default/deploy/hq.ear/hq-agent-bundles/my-
bundle.tar
ServerHome/hq-engine/server-4.x.y-EE/default/deploy/hq.ear/hq-agent-bundles/my-bundle;
gzip ServerHome/hq-engine/server-4.x.y-EE/default/deploy/hq.ear/hq-agent-bundles/my-bundle
```

5.3.4. Upgrade a 4.x Agent Using a Full Agent Package

These instructions apply to both the open source and enterprise editions of HQ.

To upgrade an HQ 4.x Agent using full the agent package:

1. Stop the 4.x agent.

2. Preserve the existing agent configuration.

   - Back up the `agent.properties` file from your previous installation. The default location for `agent.properties` in 4.x installations is the `AgentHome/conf` directory.

   - Note: In some HQ environments, `agent.properties` is stored in an alternative location that eases the process of automating the deployment of multiple agents. On Unix-based platforms, that location is the `.hq` subdirectory of the home directory of the user that runs the Agent. If your agent configuration is stored in that location, it will not be over-written by the new installation.

2. If the agent runs on Windows, uninstall the agent service from a command shell in *AgentHome*/bin:

```
hq-agent.bat remove
```

3. Unpack the 4.y agent into the agent installation directory.

4. On Windows, install the new agent service. In a command shell in *AgentHome*/bin enter:

```
hq-agent.bat install
```

5. Start the agent. For instructions, see Start the HQ Agent.

Note: The first time you start the agent, it will prompt for start settings. If you prefer, you can supply the settings in the agent properties file before starting the agent the first time. For instructions, see Configure Agent - Server Communication in Properties File.