UNIVA, AN ALTAIR COMPANY

GRID ENGINE DOCUMENTATION

Grid Engine Installation Guide

Author:
Altair Engineering

Version:
8.6.17

November 4, 2020

Contents

1 Planning the Installation

Univa Grid Engine is a distributed resource management application that runs on top of other operating systems, including various UNIX based operating systems and Microsoft Windows. For a smooth installation process, the compute resources and the network infrastructure have to be prepared correctly. The following sections describe the necessary prerequisites, provide basic knowledge about Univa Grid Engine, and ask questions that have to be answered by the Univa Grid Engine administrators during or before the installation process.

Univa Grid Engine supports various platforms, operating systems versions and hardware architectures. Find the full list of platforms, operating systems and architectures in the release notes of Univa Grid Engine.

1.1 Basics About the Architecture and Hardware Requirements

All hosts that are available for Univa Grid Engine can either be set up in a single cluster, or they can be split up into multiple groups of hosts where each group defines a cluster. These smaller sub-clusters are named *cells*. Each host can adopt one or more roles, but each host should belong to only one *cell*. The hardware requirements for each host role are listed in the table below.

Host Role	Description
Master Host	The master host is the center of a Univa Grid Engine cluster. This host runs the `sge_qmaster` daemon that stores all configuration data, runtime information provided by all other components, and information about compute jobs started on behalf of Univa Grid Engine system users. The scheduling component also resides on the master host and is responsible for all the planning tasks needed to distribute jobs into the cluster. *Requirements:* * At least 100 MB of free memory must be available. * For very large clusters, 1 GB or more of free memory might be necessary. * 2 CPUs are recommended. * Fast network interface/setup is required. All other host types will communicate with the master host over the network. * Other processing done on that host (database systems, web services, . . .) can affect cluster performance. * Microsoft Windows hosts cannot be used as master hosts.

Host Role	Description
Shadow Master Host	Zero or more shadow master hosts can be setup in each cluster. This host type runs the `sge_shadowd` process. This process provides backup functionality in case the master hosts fails. *Requirements:* * The shadow master host needs read/write permissions for the root or admin user to access the master spool directory and the common directory of the cell (`$SGE_ROOT/$SGE_CELL/common`) * Hardware requirements (memory, CPU) are the same as for the master hosts if the master host fails. * Microsoft Windows hosts cannot be used as shadow master hosts.
Submit Hosts	Submit hosts are used to submit jobs to Univa Grid Engine and to control them. The master host is by default also a submit host. *Requirements:* * Needs access to `$SGE_ROOT/$SGE_CELL/common` directory. * Microsoft Windows Domain controllers are generally not suited for Submit Hosts, therefore installing a Submit Host there is not supported.
Admin Hosts	Operators and managers of Univa Grid Engine can execute administrative commands on admin hosts. As with submit hosts, admin hosts have no special hardware requirements. The master host is by default also an administrative host. *Requirements:* * Needs access to `$SGE_ROOT/$SGE_CELL/common` directory. * Microsoft Windows Domain controllers are generally not suited for Admin Hosts, therefore installing an Admin Host there is not supported.
Execution Hosts	Multiple execution hosts can exist in a cluster. Each of these hosts runs the `sge_execd` process. Hosts running this process provide their compute resources to the corresponding cluster. *Requirements:* * Has to be setup as administrative host before the installation is started. * Should be setup as execution hosts for only one cell, otherwise special cluster setup has to be done so that corresponding resources are not oversubscribed. * Hardware and software requirements are dictated by the types of jobs to be executed on theses hosts.

Host Role	Description
	* Univa Grid Engine has no special requirements concerning memory or CPU resources. * Needs access to `$SGE_ROOT/$SGE_CELL/common` directory. * Microsoft Windows Domain controllers are generally not suited for Execution Hosts, therefore installing an Execution Host there is not supported.

Table 1: TABLE: Memory and CPU Requirements for Different Host Types

Before starting the installation, create the Univa Grid Engine root directory, which is defined by the `$SGE_ROOT` environment variable.

The disk space requirements for that directory depend on the number of hardware architectures available in the cluster and the setup of the Univa Grid Engine system. For an installation on a shared filesystem with spooling under the default locations (`$SGE_ROOT/$SGE_CELL/spool/qmaster` and `$SGE_ROOT/$SGE_CELL/spool/<execution_hostname>/`), the Univa Grid Engine system needs the following:

- 50 MB for the base installation without any binaries
- 60-120 MB for each binary set of hardware architectures
- 50-200 MB for spooling directories of the master host components using classic or database spooling
- 10-200 MB for spooling directories of each execution node, depending on the number of executed jobs and job size

To improve the overall throughput of the cluster, it might be necessary to distribute certain parts of a Univa Grid Engine installation. This will reduce the disk space required on `$SGE_ROOT`, but it will increase the disk space needed on different locations. Here are some examples:

- Binary sets might not be shared. Instead they might be installed on submit/admin/execution hosts to reduce the load on the fileserver, requiring an additional 60-120 MB for each binary set.
- Local execution host spooling is a mandatory requirement for execution hosts running on the Microsoft Windows operating system. Another benefit of execution host local spooling is that it may potentially increase cluster performance. As a result, 10-200 MB might be needed on each execution host instead of the network disk.

1.2 Selecting a File System for Spooling Operations

Univa Grid Engine supports different spooling methods on the master host: classic and databse spooling (PostgreSQL). With classic spooling, the `sge_qmaster` service creates files containing the configuration objects of a Univa Grid Engine installation in human readable format. With PostgreSQL spooling, the data will be spooled to a PostgreSQL database server. All methods have different requirements and characteristics.

Classic spooling can be done on shared filesystems. When using Lustre file shares, disable file striping for Univa Grid Engine directories. PostgreSQL spooling transfers the data directly to the database server.

Note
To make the installation process easier when installing Univa Grid Engine for the first time, use classic spooling, put `$SGE_ROOT` on a network drive (NFS3 or NFS4), and use the default spooling locations. Not using a network share requires the extra step of copying the installation directory to each execution host before continuing with the installation on that host. Shadow master functionality requires classic spooling over an NFS3/NFS4 share or PostgreSQL spooling.

During the installation process, specify both the qmaster spooling directory and the execution host spooling directory. Execution daemons use the host spooling directory to spool dynamic information about jobs started on the corresponding host. By default, all execution hosts use the same spooling location unless this setting is overridden.

Note
On Windows, the execution host must use a local drive for spooling!

1.3 Selecting the Security Mode

Univa Grid Engine can be installed in CSP mode. When the Certificate Security Protocol (CSP) is enabled, data exchanged between Univa Grid Engine components will be encrypted using a secret key, and a public/private key protocol is used to exchange secret keys in the system. The identity of each user who uses the system is checked before requested operations are executed, and each permitted user receives a certificate that will be used during the communication process. Once established, encrypted communication will continue as long as the corresponding session is valid. Once a session becomes invalid, it has to be re-created in a secure manner.

From the user point of view, CSP is completely transparent, but setting up CSP requires additional work during installation and administration of the Univa Grid Engine system:

- With CSP enabled, installation procedures will generate Certificate Authority (CA) system keys and certificates on the master host.
- An administrator must transfer the system keys and certificates to the shadow master hosts, execution hosts, administration hosts, and submit hosts.
- In running installations, keys that have already been created have to be transferred to new hosts that are added to the cluster.

- After the master installation, keys and certificates have to be generated for all users who are permitted to use the system.
- In running installations, new keys and certificates have to be created for new users who are permitted to administer or use the system.

Warning
The security mode is not supported for Windows execution, submit and administration hosts. As it is not possible to run parts of the cluster in secured and parts in unsecured mode, a cluster with Windows hosts currently cannot be run in security mode.

1.4 Selecting Thread Setup of the Master Host Component

When Univa Grid Engine client commands interact with Univa Grid Engine server components then this is done by using an interface named GDI (Grid Engine Data Interface). This interface is used to send client requests to the Univa Grid Engine system that are then handled within the server component and answered by a response message that contains the result for the client request.

This GDI interface is also used for internal Univa Grid Engine communication between components running on execution hosts as well as for internal communication between components within the `sge_qmaster` component itself.

GDI requests can be divided into two categories: Requests that will change the configuration/state of the Univa Grid Engine system (read-write-requests) and requests that will gather information to display the configuration/state of the Univa Grid Engine system (read-only-requests).

Univa Grid Engine 8.2 has been redesigned so that read-write-requests and read-only-requests can be executed completely independently from each other. Furthermore up to 64 read-only requests can work in parallel which is not possible in Sun Grid Engine, Oracle Grid Engine and other open source versions of Grid Engine. This ensures faster response times for all requests and has a huge positive impact on the cluster throughput.

Compared to previous versions of Univa Grid Engine the `sge_qmaster` component allocates 30% more memory permanently. Also peak memory usage increases. It depends on the requests that are executed in parallel and it can be influenced by changing the number of reader-only-threads that are started within `sge_qmaster`. In worst case one read-only-thread can increase the peak memory usage by 25%.

Read-only-threads can be disabled in Univa Grid Engine 8.2. The memory footprint and performance characteristics of `sge_qmaster` will than be the same like it was in previous versions.

Note
Although it is possible to disable all read-only-threads during the installation, this is not recommended. Future versions of Univa Grid Engine will not allow to disable read-only-threads at all.

1.5 Further Univa Grid Engine Configuration

Specifying a range of unused supplementary group IDs is required during installation. These group IDs will be used to tag UNIX processes that are started on behalf of Univa Grid Engine jobs, allowing Univa Grid Engine to identify resources used for each job. These IDs can also be used to enforce the termination of jobs once their defined limits have been exceeded. The ID range has to be big enough so that each job that could be executed at the same time on one execution host would get a unique ID. The default range suggested during the installation is 20000-20100 and would allow 101 concurrent jobs on a compute resource. The range does not need to be the same for each compute node. Individual ranges can also be adjusted after the installation process. Please make sure that no user account uses those supplementary group IDs on the corresponding execution hosts otherwise Univa Grid Engine might delete UNIX processes that belog to those users under the assumption those processes belog to a Univa Grid Engine job. When you intend to share filesystems between execution hosts then take care that the set of supplementary group IDs on the NFS server is disjoint to the sets on other execution hosts. The reason why this is necessary is explained here: Troubleshooting the Installation)

Choose from three scheduling profiles during the installation process. The *normal* scheduling profile is recommended for a fresh installation. When this profile is enabled, the scheduler uses interval scheduling and load adaption. It reports all information gathered during each dispatch cycle. For larger clusters, the *high* profile might be used, enabling the system to better optimize for throughput. The *max* profile can be used in clusters of any size with many short jobs. It disables load adaption and information gathering and instead enables immediate scheduling to further optimize the cluster for throughput.

During installation, all hosts will be added to the *@allhosts* host group, increasing the number of available slots in the *all.q* cluster queue. This setup can be changed once the full Univa Grid Engine cluster is up and running.

2 Installing Univa Grid Engine

2.1 Necessary Information for the Installation

Before starting the installation process, prepare the details for the installation. The table below shows all installation parameters and corresponding descriptions. These parameters must be provided either by creating a configuration file containing these values (automatic installation) or by entering them during an interactive installation.

Parameter	Description	Value
Admin User	User Account for executing all Univa Grid Engine components.	non-root user account is highly recommended to avoid problems in combinations with other products (like Unisight or UGE Rest). This user will own the files of the Univa Grid Engine installation.

Parameter	Description	Value
`$SGE_ROOT`	Base directory of the Univa Grid Engine installation.	
`$SGE_CELL`	Name of the Univa Grid Engine cell to be installed. This name identifies an instance of Univa Grid Engine when several instances run in parallel.	Default value for the first installation is *default*.
`$SGE_CLUSTER_NAME`	Name used by SMF on Solaris architecture to uniquely identify the cluster. It has to start with a letter (a-z or A-Z) followed by letters, digits (0-9), dashes (-) or underscore characters (_).	Default is the character p followed by the `$SGE_QMASTER_PORT` port number(e.g. *p6444*).
`sge_qmaster` Port Number	Port number for the `sge_qmaster` daemon.	Default value is 6444.
`sge_execd` Port Number	Port number for the `sge_execd` daemon.	Default value is 6445.
Spooling Filesystem and Locations	Spooling information for master and execution hosts choose one of: NFS3, NFS4 or Lustre filesystem or on local disk. In case of *postgres* spooling the URL for connecting to the Postgres database.	
Spooling Mechanism	The *classic*, *lmdb or* postgres_ spooling method to be used by Univa Grid Engine.	
Master Hosts	Host on which the main components of the installation process will be started.	
`OPTIONAL` Shadow Hosts	List of candidate hosts eligible to takeover master functionality when the master host fails.	
Execution Hosts	List of hosts configured to execute jobs.	
Administration Hosts	List of host permitted to execute administrative commands.	

Parameter	Description	Value
Submit Hosts	List of hosts from which jobs can be submitted into the system.	
Scheduling Profile	Choose one of *normal*, *high* or *max* scheduling profile.	
Installation Method	Type of installation method used. Interactive text based or automated installation	
OPTIONAL Installation Options	Will the cluster add hosts that run the Windows operating system? Should CSP be enabled? Should JMX functionality be enabled?	
OPTIONAL Windows Administrator User	Name of the Windows administrator account.	

Table 2: TABLE: Necessary Installation Parameters

2.2 Prerequisite Steps

Before starting the installation process, check that all prerequisites have been met.

Preparing the Network Configuration

A proper network setup for all hosts that will be part of a cluster is critical for a successful Univa Grid Engine installation.

IPv4 Network

All service components running on Univa Grid Engine hosts require a IPv4 network that is correctly setup. IPv6 is currently not supported.

Note
Hostname resolution must work properly so that each host integrated into the cluster can be resolved with a valid primary hostname.

TCP Port Setup

Univa Grid Engine requires two unused TCP port numbers. One of these is used for communication with the `sge_qmaster` process and the other for communication with `sge_execds`. The master port needs to be available on the master host and execd port on all execution hosts. When network services are set up with a NIS/NIS+ database, the port numbers can be configured by adding the following lines to the NIS/NIS+ service map:

```
sge_qmaster 6444/tcp
sge_execd 6445/tcp
```

Otherwise, the entries have to be added to `/etc/services` files on each host in cluster.

Note
On Windows hosts, the `services` file is located in `%WINDIR%\system32\drivers\etc\services`, where `%WINDIR%` usually is `c:\Windows`. Up to Windows 7, it does not contain the sge_qmaster and sge_execd ports, i.e. these have to be added manually if they are going to be used.

Password-less *root* Access

Note
`OPTIONAL` Password-less root access is not a requirement for installing Univa Grid Engine. All installation steps can also be done by manually performing necessary installation steps on remote hosts.

Warning
Enabling root login without a password can be a security risk!

Enabling password-less root access to remote hosts makes some installation steps easier for the automated installation. With password-less root access to remote hosts, certain installation steps can be automatically executed from the master host without the need to log in to a remote machine, allowing necessary files to be transferred and components to be started automatically.

Univa Grid Engine supports password-less access via ssh or rsh. Setting up password-less access depends on the operating system version and software installation.

In general, do the following steps:

1. Enable root login on remote hosts.
   - For ssh access, change *PermitRootLogin* to *yes* in the configuration file of sshd (`/etc/ssh/sshd_config`).
   - Remove restrictions that disallow *root* access only from console. On Solaris, this might be done by removing the line `CONSOLE=/dev/console` from the file `/etc/default/login`.
2. Start ssh or rsh service on all remote hosts.
3. Set up access without a password.
   - For ssh, access keys have to be generated.
4. Allow remote access on all remote hosts.
   - For ssh, the public key has to be copied to remote hosts.
   - In case of rsh, an `.rhosts` file that contains the main host name has to be created.

5. Restart the service on the remote hosts.

   - Depending on the operating system and service, it might be necessary to restart the services after configuration changes.

6. Verify that login to remote hosts is functioning.

   - Ability to connect to all remote hosts without being asked for a password indicates that password-less access has been set up correctly.

Shared File Systems

The root directory of a typical Univa Grid Engine installation (`SGE_ROOT`) will be placed on a shared file system to have binaries and utilities available on all hosts of the cluster.

If Univa Grid Engine is installed with a high availability set-up (via `sge_shadowd`), the `sge_qmaster` spool directory also needs to be put on a shared file system.

The spool directory needs to be mounted with the correct mount options:

- For all spool directories, make sure file operations cannot be interrupted. This is default for most operating systems. The `intr` option may not be used as mount option for a shared spool directory, if the default behaviour is unclear, use the `nointr` mount option to explicitly forbid interruption of file operations.
- If `sge_qmaster` is installed with database spooling, the spooling database must be placed on a file system which fully supports standard POSIX filesystem semantics, e.g. NFS version 4.

Downloading the Distribution Files

1. Download the Software.

   - About 300 MB of free disk space is required.
   - Software packages are available in `tar.gz` format for all supported platforms.
   - The distribution is split up into one architecture independent file and multiple platform specific ones. Here is the list of all available files:

Filename	Description
ge-8.6.x-arco.tar.gz	ARCo + dbwriter
ge-8.6.x-bin-aix-ppc64.tar.gz	AIX PPC; 64 bit binaries
ge-8.6.x-bin-aix51.tar.gz	AIX 5.1 and later
ge-8.6.x-bin-darwin-x64.tar.gz	Darwin; 64 bit binaries
ge-8.6.x-bin-hp11-ia64.tar.gz	HP-UX; 64 bit binaries

Filename	Description
ge-8.6.x-bin-lx-amd64.tar.gz	Linux x86; 64 bit binaries
ge-8.6.x-bin-lx-ppc64.tar.gz	Linux PowerPC; 64 bit binaries
ge-8.6.x-bin-lx-x86.tar.gz	Linux x86; 32 bit binaries
ge-8.6.x-bin-sol-amd64.tar.gz	Solaris x86; 64 bit binaries
ge-8.6.x-bin-sol-sparc64.tar.gz	Solaris SPARC platform; 64 bit binaries
ge-8.6.x-bin-sol-x86.tar.gz	Solaris x86; 32 bit binaries
ge-8.6.x-bin-ulx-amd64.tar.gz	Unsupported Linux x86; 64 bit binaries
ge-8.6.x-bin-ulx-x86.tar.gz	Unsupported Linux x86; 32 bit binaries
ge-8.6.x-bin-win-x86.tar.gz	native Windows 32 bit binaries
ge-8.6.x-common.tar.gz	Architecture independent files
ge-8.6.x-doc.tar.gz	Documentation

Table 3: Available Files

- Download the common package and the required binary packages.

2. Prepare the installation directory.

- Log in on the fileserver as user *root*.
- Set the installation directory:

```
# SGE_ROOT=\<installation_path>
# export SGE_ROOT
```

- Create the installation directory:

```
# mkdir $SGE_ROOT
```

3. Unpack the software.

```
# cd $SGE_ROOT
# gzip -dc <download_dir>/ge-8.6.x-common.tar.gz | tar xvpf -
# gzip -dc <download_dir>/ge-8.6.x-bin-lx-amd64.tar.gz | tar xvpf -
# ...
```

4. Correct the file permission.

```
# ./util/setfileperm.sh $SGE_ROOT
```

2.3 Installing with the Command-Line Installation Script

Note

This chapter describes only the fresh installation of Univa Grid Engine systems. For existing installations of Open Source Grid Engine, Sun Grid Engine, or Oracle Grid Engine, check the upgrade matrix to see which systems can be upgraded directly from the existing version of Grid Engine.
This document assumes that Univa Grid Engine will be installed on computers running the Linux operating system. Installations on different operating systems might have slight differences, and if available, documentation concerning those differences can be found in files with the name `$SGE_ROOT/doc/arc_depend_<arch>.asc` where `<arch>` is the architecture name.

There are two options to create a fresh installation of Univa Grid Engine:

- Interactive installation with installation scripts
- Automatic installation with installation scripts

The following sections describe the script-based installations in step-by-step instructions. To automate the installation process, follow the instructions in section the Automated Installation.

2.3.1 Interactive Installation

For a full interactive installation of Univa Grid Engine, run the installation scripts on the master host, the shadow hosts and all execution hosts. The scripts ask a number of questions, and the answers to those questions influence the initial cluster configuration and the daemons that are started.

A fresh installation requires the following steps:

1. Master Host Installation
   - Must be installed first.
   - Installation script must be executed once on the master host.
   - Step-by-step instructions can be found in section [Master host installation][Master host installation].
2. `OPTIONAL` Shadow Master Host Installation
   - Must be installed after the master host installation.
   - Installation script must be executed on all hosts that could act as Shadow Masters.
   - Step-by-step instructions can be found in section [Shadow master host installation][Shadow master host installation].
3. Execution Host Installation
   - Must be installed after the master host installation.

- Installation script must be executed on all hosts that could act as execution hosts.
- Step-by-step instructions can be found in section [Execution host installation][Execution host installation].

Master Host Installation

The step-by-step instructions below show all steps needed for installation. Additional instructions are included for cases when CSP is enabled and when Microsoft Windows execution hosts could be installed. Those who do not want to enable these functionalities can skip corresponding instructions marked with the tags `WIN-only` or `CSP-only`. Installation steps that refer to one of those functionalities will then automatically be skipped by the installation script.

Warning

Univa recommends that first-time installations of Univa Grid Engine should be installed without CSP support to ease the installation and administration of the cluster.

Warning

It is not possible to add Windows hosts to a Univa Grid Engine cluster that uses the CSP mode!

1. Prepare to start.

   - Log in on the master host as *root*.
   - Set necessary environment variables.

   The `$SGE_ROOT` environment variable defines the root directory for the installation.

   ```
   > SGE_ROOT=<installation_path>
   > export SGE_ROOT
   ```

   - Change to the installation directory.

   ```
   > cd $SGE_ROOT
   ```

2. Start the installation.

   - The installation script is named `install_qmaster`.
   - Start this script and optionally provide necessary command line arguments.
   - `CSP-only`: The optional `-csp` flag causes the installation script to enable the security features of the software.

   ```
   > ./install_qmaster -csp
   Welcome to the Grid Engine installation
   ---------------------------------------

   Hit <RETURN> to continue >>
   ```

3. Accept the software license agreement.

   - Read the software license and the support agreement.

     ```
     TERM SOFTWARE LICENSE AND SUPPORT AGREEMENT
     PLEASE READ THIS AGREEMENT BEFORE USING THE SOFTWARE.
     ...
     ```

   - Push *space* or *return* key to reach the end of the text.

     ```
     Do you agree with that license? (y/n)
     ```

   - Enter *y* to accept the license.

     ```
     Welcome to the Grid Engine installation
     ---------------------------------------

     Hit <RETURN> to continue >>
     ```

   - Press *Return* to leave the welcome screen.

4. Set up the admin user account.

   - The installation process prints the installation directory and the current owner.

     ```
     Grid Engine admin user account
     ------------------------------

     The current directory
         <installation_path>
     is owned by user
         <owner>
     ...
     ```

     Do you want to install Grid Engine as admin user >ernst< (y/n)

   - If the owner of that directory is also the administrator user of the installation, then answer with *y*. Installation will continue with the next main step.
   - To choose a different administrator user for the system, answer *n*.

     ```
     Choosing Grid Engine admin user account
     ---------------------------------------

     Do you want to install Grid Engine
     under an user id other than >root< (y/n)
     ```

   - If the administrator user is *root* then answer *n*. Installation will continue with the next main step.
   - Answering *y* will trigger a request to enter the administrator user name.

     ```
     Choosing a Grid Engine admin user name
     --------------------------------------

     Please enter a valid user name
     ```

- Enter the name of the administrator user, and press *return*.

5. Choose the installation location.

   - Check the installation directory.

     ```
     Checking $SGE_ROOT directory
     ----------------------------

     ...

     If this directory is not correct (e.g. it may contain an automounter
     prefix) enter the correct path to this directory or hit <RETURN>
     to use default [<installation_path>]
     ```

   - Press *return* to accept it or enter the correct path and press *return*.

6. Choose the TCP/IP port numbers.

   - Choose the communications ports that should be used for `sge_qmaster` process. The recommended process specified a change to the file `/etc/services` or the addition of corresponding entries to the services NIS/NIS+ map. If the recommended process was followed, the installation will display the corresponding port: press *return* to accept the setting and continue with the selection of the communications ports that should be used for `sge_execd` process.

     ```
     The port for sge_qmaster is currently set as service.

     sge_qmaster service set to port <port_number>

     ...

     Using the >shell environment<:                          [1]
     Using a network service like >/etc/service<, >NIS/NIS+<: [2]

     (default: 2)
     ```

   - In case the service port entry was not already changed, the following screen will appear.

     ```
     Grid Engine TCP/IP communication service
     ---------------------------------------

     The communication settings for sge_qmaster are currently not done.

     (default: 1)
     ```

   - To catch up those changes, start an additional terminal session, login as *root* and change either `/etc/services` or the corresponding services NIS/NIS+ map. Add the following lines, changing the port numbers to the desired ports to use for this installation.

```
sge_qmaster     6444/tcp   # Grid Engine Qmaster Service
sge_execd       6445/tcp   # Grid Engine Execution Service
```

- After the changes are active, enter *2* and press *return*.

```
Grid Engine TCP/IP communication service
----------------------------------------

Using the service

sge_qmaster

...

Hit <RETURN> to continue
```

- Providing the port numbers via environment variables is an alternative to changing the entries in `/etc/services` or the corresponding services NIS/NIS+ map. To enable this alternative, abort the installation process, set the environment variables `$SGE_QMASTER_PORT` and `$SGE_EXECD_PORT`, and restart the installation.

```
# SGE_QMASTER_PORT=6444; export SGE_QMASTER_PORT
# SGE_EXECD_PORT=6445; export SGE_EXECD_PORT
# ./install_qmaster ...

Grid Engine TCP/IP communication service
----------------------------------------

The port for sge_qmaster is currently set by the shell environment.

SGE_QMASTER_PORT = 6444
```

- To accept defined environment variables, choose *1* and press *return*.
- Select the `sge_execd` port the same way ports were selected for the `sge_qmaster`.

7. Choose a unique cell name.

   - Choose a unique cell name. Accept the default value if only one cluster will be installed, giving the cell the name *default*.
   - If other cells are already installed, be sure that the chosen name is different from cell names already in use.

```
Grid Engine cells
-----------------

...

Enter cell name [default]
```

   - Press *return* to continue.

```
Using cell >default<.
Hit <RETURN> to continue >>
```

- Press *return* to continue.

8. Name the cluster.

   - The cluster name uniquely identifies a specific Univa Grid Engine cluster. It must be unique throughout the organization. The name is not related to the cell.

   ```
   Unique cluster name
   -------------------

   ...

   Enter new cluster name or hit <RETURN>
   to use default [p6444]
   ```

   - Press *return* to accept the recommended cluster name that is a combination of the letter 'p' and the `sge_qmaster` port number that has been selected in a previous step.

   ```
   Your $SGE_CLUSTER_NAME: p6444

   Hit <RETURN> to continue
   ```

   - Press *return* to continue.

9. Specify a license file (Univa Grid Engine 8.7.0 and newer).

   - Choose a license file to be installed.

   ```
   Installing Grid Engine cluster license file
   -------------------------------------------

   Please specify the source path of your license file [<default_license_file>] >>
   ```

   - Press *return* to install the `<default_license_file>` shipped with Univa Grid Engine or enter the path to the license file of your choice.
   - The license file will be installed in the default location:

   ```
   $SGE_ROOT/$SGE_CELL/common/uge_license.txt
   ```

10. Select the master daemon spooling directory.

    ```
    Grid Engine qmaster spool directory
    -----------------------------------

    ...

    Enter a qmaster spool directory [<installation_path>/default/spool/qmaster] >>
    ```

- Either accept the default value by pressing *return*, or enter a different directory and press *return*.

```
Using qmaster spool directory ><installation_path>/default/spool/qmaster<.
Hit <RETURN> to continue
```

- Press *return* to continue.

11. Flag Windows execution hosts.

```
Windows Execution Host Support
------------------------------

Are you going to install Windows Execution Hosts? (y/n)
```

- WIN-only: When installing clusters that will include execution hosts running the Windows operating system, answer with *y* and press *return*.

12. Enable/Disable Read-Only Component by Specifying Number of Threads

```
Read-Only Component
-------------------

Univa Grid Engine allows to activate a read-only component as part of the
qmaster process. This component can answer certain requests (read-only
requests like qstat, qhost, ...) in parallel with other activities of qmaster
(read-write requests like configuration changes, scheduling of jobs, ...) at
the same point in time.

An enabled read-only component will improve the responsiveness of qmaster
and also the cluster throughput compared to previous versions of Grid Engine
but it requires that dependent Univa Grid Engine commands are executed in a
session. Consult the Univa Grid Engine documentation for more information

How many reader threads should be started in the read-only component?
0 will disable the read-only component. Higher numbers increase
responsiveness of qmaster but increase also peak memory requirements.
Specify a value between 0 and 64. [2] >>
```

- Either accept the default value by pressing *return*, or enter a different number of threads and press *return*. 0 will disable the read-only component

13. Verify file permissions.

```
Verifying and setting file permissions
--------------------------------------

Did you install this version with >pkgadd< or did you already verify
and set the file permissions of your distribution (enter: y) (y/n)
```

- Answer the question, and press *return*. If the answer to the previous question concerning Windows hosts was *y*, force the verification by answering *n* before continuing.

```
Verifying and setting file permissions
--------------------------------------

We may now verify and set the file permissions of your Grid Engine
distribution.

This may be useful since due to unpacking and copying of your distribution
your files may be not accessible to other users.

We will set the permissions of directories and binaries to

  755 - that means executable are accessible for the world

and for ordinary files to

  644 - that means readable for the world

Do you want to verify and set your file permissions (y/n)
```

- If answering *y*, press *return* to verify the permissions.

```
Verifying and setting file permissions and owner in >3rd_party<
Verifying and setting file permissions and owner in >bin<
Verifying and setting file permissions and owner in >ckpt<
Verifying and setting file permissions and owner in >examples<
Verifying and setting file permissions and owner in >inst_sge<
Verifying and setting file permissions and owner in >install_execd<
Verifying and setting file permissions and owner in >install_qmaster<
Verifying and setting file permissions and owner in >lib<
Verifying and setting file permissions and owner in >mpi<
Verifying and setting file permissions and owner in >pvm<
Verifying and setting file permissions and owner in >qmon<
Verifying and setting file permissions and owner in >util<
Verifying and setting file permissions and owner in >utilbin<
Verifying and setting file permissions and owner in >catman<
Verifying and setting file permissions and owner in >doc<
Verifying and setting file permissions and owner in >include<
Verifying and setting file permissions and owner in >man<

Your file permissions were set

Hit <RETURN> to continue
```

- Press *return* to continue.

14. Choose hostname resolving method and default domain.

- Specify whether all hosts that could be added to the Univa Grid Engine cluster are located in a single DNS domain.

```
Select default Grid Engine hostname resolving method
----------------------------------------------------

...

Are all hosts of your cluster in a single DNS domain (y/n)
```

- Answer 'y' before pressing *return* to whether to specify a default domain.

```
Default domain for hostnames
----------------------------

...

Do you want to configure a default domain (y/n)
```

- Answer *y* again to be able to enter the domain.

```
Please enter your default domain
```

- Specify the domain, and press *return*.

```
Using >univa.com< as default domain. Hit <RETURN> to continue
```

- Press *return* again to continue with the next main installation step.
- If the hosts are not all part of a single domain, then answer the first question with 'n'.

```
The domain name is not ignored when comparing hostnames.

Hit <RETURN> to continue
```

- In this case, domain names will not be ignored.

15. Make directories.

```
Making directories
------------------

creating directory: <installation_path>/default/spool/qmaster
creating directory: <installation_path>/default/spool/qmaster/job_scripts
Hit <RETURN> to continue >
```

- Needed spool directories will be created. Press *return* to continue.

16. Set up the spooling method.

```
Setup spooling
--------------
...

Please choose a spooling method (lmdb|classic|postgres) [classic]
```

- Choose the spooling method: enter either *lmdb or* classic_ or *postgres*, then continue with *return*.
- If choosing LMDB spooling, enter a LMDB spooling directory located either on a local drive or a network filesystem (NFS4, Lustre).

```
LMDB spooling parameters
---------------------------------------

Please enter the database directory now, even if you want to spool locally,
it is necessary to enter this database directory.

Default: [<installation_path>/default/spool/spooldb]
```

- If choosing classic spooling, data will get written to the qmaster spool directory specified earlier. No further input is required.
- If choosing postgres spooling, the connection parameters need to be specified:

```
PostgreSQL Database spooling parameters
---------------------------------------

The spooling parameters define which PostgreSQL database will be used
for spooling and how to connect to this database.

It is a space separated list of key=value pairs, usually it is necessary
to specify the host, dbname and user attributes, e.g.
host=mydbhost dbname=ugespooling user=ugeadmin

If your PostgreSQL Database is configured to require authentication by password
do not specify a password in the connection string but use the .pgpass file mechanism.

See also the PostgreSQL documentation, section libpq - C Library for more information.

Enter the connection string for connecting to your PostgreSQL Database Server >>
```

- The following parameters can be specified. See also the PostgreSQL documentation at http://www.postgresql.org/docs/9.1/static/libpq-connect.html for a full list of possible connection parameters.
- Initial spooling information will be created then.

```
Dumping bootstrapping information
Initializing spooling database

Hit <RETURN> to continue >>
```

- Press *return* to continue.

17. Specify the group ID range.

    CAUTION: Please also have a look here, regarding the impact of a wrong GID_RANGE: Further Univa Grid Engine Configuration

```
Grid Engine group id range
--------------------------

...

Please enter a range [20000-20100]
```

- Enter an additional group ID range that is available on all execution hosts.

```
Using >20000-20100< as gid range. Hit <RETURN> to continue
```

- Press *return* to continue.

18. Set the path of the execution daemon spooling directory.

```
Grid Engine cluster configuration
---------------------------------

...

Default: [<installation_path>/default/spool]
```

- Specify the path of the spooling directory for the execution hosts, and press *return*.

19. Set up administrator mail.

CAUTION: Please, be careful with administrator mails, in huge installations you might receive a high number of e-mails flooding your mail sever.

```
Grid Engine cluster configuration (continued)
---------------------------------------------

...

Default: [none]
```

- Enter an email address for receiving problem reports, and press *return*.

```
The following parameters for the cluster configuration were configured:

execd_spool_dir        <installation_path>/default/spool
administrator_mail     none

Do you want to change the configuration parameters (y/n)
```

- Accept the changes with *y*, or enter *n* to return to a previous installation step.

```
Creating local configuration
----------------------------
Creating >act_qmaster< file
Adding default complex attributes
Adding default parallel environments (PE)
```

```
Adding SGE default usersets
Adding >sge_aliases< path aliases file
Adding >sge_request< default submit options file
Creating >sgemaster< script
Creating >sgeexecd< script
Creating settings files for >.profile/.cshrc<

Hit <RETURN> to continue
```

- Default configuration objects will now be created. Hit *return* to continue.

20. `CSP-only` or `WIN-only`: Initialize security framework.

```
Initializing Certificate Authority (CA) for OpenSSL security framework
------------------------------------------------------------------------
Creating <installation_path>/default/common/sgeCA
Creating /var/sgeCA/port5000/default
Creating <installation_path>/default/common/sgeCA/certs
Creating <installation_path>/default/common/sgeCA/crl
Creating <installation_path>/default/common/sgeCA/newcerts
Creating <installation_path>/default/common/sgeCA/serial
Creating <installation_path>/default/common/sgeCA/index.txt
Creating <installation_path>/default/common/sgeCA/usercerts
Creating /var/sgeCA/port6444/default/userkeys
Creating /var/sgeCA/port6444/default/private

Hit <RETURN> to continue >>
```

- Hit *return* to continue.

```
Creating CA certificate and private key
---------------------------------------
Please give some basic parameters to create the distinguished name (DN)
for the certificates.

We will ask for
    -the two letter country code
    -the state
    -the location, e.g city or your buildingcode
    -the organization (e.g. your company name)
    -the organizational unit, e.g. your department
    -the email address of the CA administrator (you!)

Hit <RETURN> to continue >>
```

- Hit *return* to continue.

```
Please enter your two letter country code, e.g. 'US'
Please enter your state
Please enter your location, e.g city or buildingcode
Please enter the name of your organization
```

```
Please enter your organizational unit, e.g. your department
Please enter the email address of the CA administrator
```

- After entering the requested information, review the summary.

```
You selected the following basic data for the distinguished name of
your certificates:

Country code:         C=DE
State:                ST=BY
Location:             L=RGB
Organization:         O=Univa
Organizational unit:  OU=UGE
CA email address:     emailAddress=geadmin@univa.com

Do you want to use these data (y/n) [y]
```

- Verify the data and accept it with *y*, or press *n* to re-enter the values.

```
Creating CA certificate and private key
Generating a 1024 bit RSA private key
..........++++++
.........................++++++
writing new private key to '/var/sgeCA/port6444/default/private/cakey.pem'
-----

Hit <RETURN> to continue
```

- Hit *return* to continue.

```
Creating 'daemon' certificate and key for SGE Daemon
-------------------------------------------------

...

Creating 'user' certificate and key for SGE install user
-------------------------------------------------------

...

Creating 'user' certificate and key for SGE admin user
-------------------------------------------------

...

Hit <RETURN> to continue
```

- Hit *return* to continue.

21. Specify whether the daemon should be started at boot time.

```
qmaster startup script
----------------------

We can install the startup script that will
start qmaster at machine boot (y/n)
```

- Answer *y* if the daemon should be started at boot time.

```
cp <installation_path>/default/common/sgemaster /etc/init.d/sgemaster.p6444
/usr/lib/lsb/install_initd /etc/init.d/sgemaster.p6444

Hit <RETURN> to continue >>
```

- Hit *return* to continue.

```
Grid Engine qmaster startup
---------------------------

Starting qmaster daemon. Please wait ...
    starting sge_qmaster
Hit <RETURN> to continue
```

- Hit *return* to continue.

22. Identify admin and submit hosts.

```
Adding Grid Engine hosts
------------------------

...

Do you want to use a file which contains the list of hosts (y/n)
```

- Notify Univa Grid Engine about which execution hosts will be installed. These hosts must be added to the configuration as administration hosts before later continuing with the execution host installation. The same hosts will also be configured as submit hosts. If a file containing all those hostnames is available, then answer *y*, enter the filename, and press *return*.

```
Adding admin and submit hosts from file
---------------------------------------

Please enter the file name which contains the host list:
```

- If no file is available, then answer *n*.

```
Adding admin and submit hosts
-----------------------------

Please enter a blank separated list of hosts.

...
```

- In this case, enter a list of hostnames.

```
Host(s):
```

- See messages from Univa Grid Engine when the hosts are added.

```
<hostname> added to administrative host list
<hostname> added to submit host list
Hit <RETURN> to continue >>
```

- Continue entering hostnames until finished.

```
Finished adding hosts. Hit <RETURN> to continue >>
```

- Press *return* to continue.

23. Specify shadow hosts.

```
If you want to use a shadow host, it is recommended to add this host
to the list of administrative hosts.

...

Do you want to add your shadow host(s) now? (y/n)
```

- Also for shadow hosts, specify a file containing the hostnames or enter them manually.

```
Adding Grid Engine shadow hosts
-------------------------------

...

Do you want to use a file which contains the list of hosts (y/n)

Adding admin hosts
------------------

...

Host(s):

Finished adding hosts. Hit <RETURN> to continue
```

- Press enter to return.

24. Add hosts to default objects.

```
Creating the default <all.q> queue and <allhosts> hostgroup
-----------------------------------------------------------

root@<hostname> added "@allhosts" to host group list
root@<hostname> added "all.q" to cluster queue list

Hit <RETURN> to continue
```

- Hit *return* to continue.

25. `CSP-only` or `WIN-only`: Transfer certificate files and public keys.

    - For password-less root access to execution and submit hosts configurations, the installation script will now distribute necessary certificate files. To skip this step, press *n* and *return*. If neither passwordless rcp nor passwordless scp is configured to work between the master host and the execution host, press *n* and *return* here and copy the cert files manually after the installation.

      ```
      Installing SGE in CSP mode
      --------------------------

      Installing SGE in CSP mode needs to copy the cert
      files to each execution host. This can be done by script!

      To use this functionality, it is recommended, that user root
      may do rsh/ssh to the execution host, without being asked for a password!

      Should the script try to copy the cert files, for you, to each
      <execution> host? (y/n) [y]
      ```

    - Answer *y* to transfer necessary files to the execution hosts.

      ```
      You can use a rsh or a ssh copy to transfer the cert files to each
      <execution> host (default: ssh)
      Do you want to use rsh/rcp instead of ssh/scp? (y/n)
      ```

    - Answer *y* to use rsh connection instead of *ssh*.

      ```
      Copying certificates to host <hostname>
      Setting ownership to adminuser ernst
      Installing SGE in CSP mode
      ```

    - Now the installer asks whether or not to copy these files to the submit hosts.

      ```
      You can use a rsh or a ssh copy to transfer the cert files to each
      <submit> host (default: ssh)
      Do you want to use rsh/rcp instead of ssh/scp? (y/n)
      ```

26. Configure the scheduling profile.

    ```
    Scheduler Tuning
    ----------------

    ...

    Enter the number of your preferred configuration and hit <RETURN>!
    Default configuration is [1]
    ```

    - Choose between three predefined scheduler profiles: enter *1*, *2* or *3*, and press return.

```
We're configuring the scheduler with >Normal< settings!
Do you agree? (y/n) [y]
```

- Press *Return* to continue.

27. Summary.

```
Using Grid Engine
-----------------

You should now enter the command:

    source <installation_path>/default/common/settings.csh

if you are a csh/tcsh user or

# . <installation_path>/default/common/settings.sh

if you are a sh/ksh user.

This will set or expand the following environment variables:

    -$SGE_ROOT         (always necessary)
    -$SGE_CELL         (if you are using a cell other than >default<)
    -$SGE_CLUSTER_NAME (always necessary)
    -$SGE_QMASTER_PORT (if you haven't added the service >sge_qmaster<)
    -$SGE_EXECD_PORT   (if you haven't added the service >sge_execd<)
    -$PATH/$path       (to find the Grid Engine binaries)
    -$MANPATH          (to access the manual pages)

Hit <RETURN> to see where Grid Engine logs messages >>
```

- Hit *return* to continue.

```
Grid Engine messages
--------------------

Grid Engine messages can be found at:

    /tmp/qmaster_messages (during qmaster startup)
    /tmp/execd_messages   (during execution daemon startup)

After startup the daemons log their messages in their spool directories.

    Qmaster:     <installation_path>/default/spool/qmaster/messages
    Exec daemon: <execd_spool_dir>/<hostname>/messages

Grid Engine startup scripts
---------------------------
```

```
Grid Engine startup scripts can be found at:

    <installation_path>/default/common/sgemaster (qmaster)
    <installation_path>/default/common/sgeexecd (execd)

Do you want to see previous screen about using Grid Engine again (y/n)
```

- Choose *n*, and hit *return* to continue.

```
Your Grid Engine qmaster installation is now completed
------------------------------------------------------

Please now login to all hosts where you want to run an execution daemon
and start the execution host installation procedure.

If you want to run an execution daemon on this host, please do not forget
to make the execution host installation in this host as well.

All execution hosts must be administrative hosts during the installation.
All hosts which you added to the list of administrative hosts during this
installation procedure can now be installed.

You may verify your administrative hosts with the command

    # qconf -sh

and you may add new administrative hosts with the command

    # qconf -ah <hostname>

Please hit <RETURN> >>
```

- Hit *return* to terminate the installation script and complete the qmaster installation. The `sge_qmaster` process is running, and post-installation tasks can begin.

28. `CSP-only` or `WIN-only`: Transfer certificate files and private keys (manually).

- If the certificates were not transferred automatically to the submit, administrative and execution hosts (which will likely be the case on Windows hosts), this step must be performed manually to continue with the installation of the cluster.
  - The publicly accessible CA and daemon certificates are stored in `$SGE_ROOT/$SGE_CELL/common/sgeCA`.
  - Corresponding private keys are stored in `/var/sgeCA/<dir_name>/cell/private` where `<dir_name>` is either the string *sge_qmaster* or a name starting with *port* followed by the `$SGE_QMASTER_PORT` number.
  - User keys and certificates are stored in `/var/sgeCA/<dir_name>/cell/userkeys/<username>`.
  - Prepare a file containing all private keys and random files.

```
> umask 077
> cd /
> tar cvpf /var/sgeCA/port6444.tar /var/sgeCA/port${SGE_QMASTER_PORT}/$SGE_CELL
```

> **Note**
> WIN-onlyMake sure you use a packer that is available on all of your Windows hosts, too.

- Switch to all execution hosts and copy the file in a secure manner.

```
> umask 077
> cd /
> scp <master_hostname>:/var/sgeCA/port6444.tar .
> umask 022
> tar xfpf /port6444.tar
> rm /port6444.tar .
```

> **Note**
> WIN-onlyUse the appropriate unpacker instead of tar on your Windows hosts.

- WIN-only: The unpacker on the Windows host will not be able to restore the ownership and permissions of the certificate files. The *Administrator* has to be sure that this is done manually.
- Check that the permissions are correct. This can be done using the Windows explorer and checking the `security` settings of the directories and files. Because it's very difficult to describe the necessary settings in Windows notation, here are the corresponding UNIX permissions and ownerships the files must have also on Windows:

```
> ls -lR /var/sgeCA/port6444/
/var/sgeCA/port6444/:
total 2
drwxr-xr-x   4 admin    other        512 Apr  14 11:04 default
/var/sgeCA/port6444/default:
total 4
drwx------   2 admin    staff        512 Apr  14 11:04 private
drwxr-xr-x   4 admin    staff        512 Apr  14 11:04 userkeys
/var/sgeCA/port6444/default/private:
total 8
-rw-------   1 admin    staff        887 Apr  14 11:04 cakey.pem
-rw-------   1 admin    staff        887 Apr  14 11:04 key.pem
-rw-------   1 admin    staff       1024 Apr  14 11:04 rand.seed
-rw-------   1 admin    staff        761 Apr  14 11:04 req.pem
/var/sgeCA/port6444/default/userkeys:
total 4
dr-x------   2 admin    staff        512 Apr  14 11:04 admin
dr-x------   2 root     staff        512 Apr  14 11:04 root
/var/sgeCA/port6444/default/userkeys/admin:
total 16
-r--------   1 admin    staff       3811 Apr  14 11:04 cert.pem
-r--------   1 admin    staff        887 Apr  14 11:04 key.pem
-r--------   1 admin    staff       2048 Apr  14 11:04 rand.seed
```

```
-r--------   1 admin    staff        769 Apr  14 11:04 req.pem
/var/sgeCA/port6444/default/userkeys/root:
total 16
-r--------   1 root     staff       3805 Apr  14 11:04 cert.pem
-r--------   1 root     staff        887 Apr  14 11:04 key.pem
-r--------   1 root     staff       2048 Apr  14 11:04 rand.seed
-r--------   1 root     staff        769 Apr  14 11:04 req.pem
```

29. Review next steps.

    - If shadow master hosts were specified during installation, then continue with the shadow master host installation as described in the next section [Shadow master host installation][Shadow master host installation].
    - Execution nodes can now be set up. Instructions are in the section [Execution Host installation][Execution host installation].

Shadow Master Host Installation

1. Prepare to start.

    - Complete the master host installation as outlined in section [Master host installation][Master host installation] before the installation of a shadow master host. During that installation, specify the name of possible shadow hosts.
    - Log in on a shadow master host as *root*.
    - Set the necessary environment variables by sourcing the settings file.

      ```
      # . <installation_path>/<cell_name>/common/settings.sh
      ```

    - Change into the installation directory.

      ```
      # cd $SGE_ROOT
      ```

    - Check if the current host is already an administration host. If so, the following command will print out information, including the hostname.

      ```
      # qconf -sh
      ...
      ```

    - If the hostname was missing in the output, then make the current host an administration host.

      ```
      # qconf -ah <hostname>
      <hostname> added to administrative host list
      ```

    - CSP-only If the root user does not have write permissions in the `$SGE_ROOT` directory on the shadow master host, then the installation script will ask whether or not it should install the software as the user to whom the directory belongs. To answer *y*, first install the security-related files into that user's `$HOME/.sge` directory before continuing.

```
# su - <admin_user>
# . $SGE_ROOT/default/common/settings.sh
# logout
```

- Make sure that the host you wish to configure as a shadow host has read/write permissions to the qmaster spool and `$SGE_ROOT/$SGE_CELL/common`.

2. Start the shadow master installation.

   - Shadow master installation is done with the `inst_sge` script. Execute the following command to start the installation.

```
# ./inst_sge -sm

Shadow Master Host Setup
------------------------

...

Hit <RETURN> to continue >>
```

   - Press *return* to continue.

3. Specify the admin user.

```
Grid Engine admin user account
------------------------------

The current directory

    <installation_path>

is owned by user

    <owner>

...

Do you want to install Grid Engine as admin user ><username>< (y/n)
```

   - Enter the admin user name, and press *return* to continue.

```
Installing Grid Engine as admin user ><username><
Hit <RETURN> to continue
```

   - Press *return* to continue.

4. Choose the installation location.

```
Checking $SGE_ROOT directory
----------------------------
```

```
...

If this directory is not correct (e.g. it may contain an automounter
prefix) enter the correct path to this directory or hit <RETURN>
to use default [<installation_path>]
```

- Press *return* to accept it, or enter the correct path, and press *return*.

```
Your $SGE_ROOT directory: <installation_path>

Hit <RETURN> to continue
```

- Press *return* to continue.

5. Specify the cell name.

```
Please enter your SGE_CELL directory or use the default [default]
```

- Enter the cell name, and press *return* to continue.

6. Check the hostname resolution.

```
Checking hostname resolving
----------------------------

This hostname is known at qmaster as an administrative host.

Hit <RETURN> to continue >>
```

- Hit *return* to continue.

7. Create local configuration.

```
Creating local configuration
-----------------------------

...

Hit <RETURN> to continue
```

- Hit *return* to continue.

8. Specify whether the daemon should be started at boot time.

```
shadow startup script
----------------------

Hit <RETURN> to continue
```

- Hit *return* to complete the installation.

```
Starting sge_shadowd on host <hostname>

Shadowhost installation completed!
```

9. Review next steps.
   - Continue to install execution hosts.

Execution Host Installation (all architectures except Windows)

> **Note**
> This is the installation guide for all non-Windows execution hosts. Find the guide for Windows execution hosts here: Installing Univa Grid Engine on Microsoft Windows

1. Prepare to start.
   - Log in on a execution host as *root*.
   - Set the necessary environment variables.

     ```
     # SGE_ROOT=<installation_path>
     # export SGE_ROOT
     # . $SGE_ROOT/<cell_name>/common/settings.sh
     ```

   - Change to the installation directory.

     ```
     # cd $SGE_ROOT
     ```

   - Check if the current host is already an administration host. If so, the following command will print out information, including the hostname.

     ```
     # qconf -sh
     ...
     ```

   - If the hostname was missing in the output, then make the current host an administration host.

     ```
     # qconf -ah <hostname>
     <hostname> added to administrative host list
     ```

   - CSP-only If the root user does not have write permissions in the `$SGE_ROOT` directory on the execution host, then the installation script will ask whether or not it should install the software as the user to whom the directory belongs. To answer *y*, first install the security-related files into that user's `$HOME/.sge` directory before continuing.

     ```
     # su - <admin_user>
     # . $SGE_ROOT/default/common/settings.sh
     # logout
     ```

2. Start the execution host installation.
   - The installation script is named `install_execd`.

- Start this script and optionally provide necessary command line arguments. Be sure that certain features enabled during the master host installation are also enabled here.
- CSP-only: The optional `-csp` flag will cause the installation script to enable the security features of the software. To install CSP on an execution host, CSP must already be enabled during the master host installation.

```
Welcome to the Grid Engine execution host installation
------------------------------------------------------

...

Hit <RETURN> to continue
```

- Press *return* to continue.

3. Choose the installation location.

```
Checking $SGE_ROOT directory
----------------------------

The Grid Engine root directory is:

$SGE_ROOT = <installation_path>

If this directory is not correct (e.g. it may contain an automounter
prefix) enter the correct path to this directory or hit <RETURN>
to use default [<installation_path>] >>
```

- Change the directory if necessary and press *return* to continue.

```
Your $SGE_ROOT directory: <installation_path>

Hit <RETURN> to continue
```

- Press *return* again to continue.

4. Specify the cell name.

```
Grid Engine cells
-----------------

Please enter cell name which you used for the qmaster
installation or press <RETURN> to use [default]
```

- Enter the cell name if not *default*, and press *return*.

```
Using cell: >default<

Hit <RETURN> to continue
```

- Press *return* again to continue.

5. Specify the TCP/IP port number.

```
Grid Engine TCP/IP communication service
----------------------------------------

The port for sge_execd is currently set by the shell environment.

    SGE_EXECD_PORT = 5001

Hit <RETURN> to continue
```

   - Press *return* to continue.

6. OPTIONAL : Specify the admin user.

   - The installation script checks to see if the admin user specified during the qmaster installation already exists. If not, then the following screen appears.

```
Local Admin User
----------------

The local admin user <username>, does not exist!
The script tries to create the admin user.
Please enter a password for your admin user >>
```

   - Enter the admin user's password, and press *return*.

```
Creating admin user sgeadmin, now ...

Admin user created, hit <ENTER> to continue!
```

   - Press *return* to continue.

7. Check the hostname resolution.

```
Checking hostname resolving
---------------------------

This hostname is known at qmaster as an administrative host.

Hit <RETURN> to continue
```

   - Press *return* to continue.

8. Choose the local spooling directory.

   - During the master installation, a global spooling directory was specified. Define a local spooling directory now.

```
Execd spool directory configuration
-----------------------------------

...

Do you want to configure a different spool directory
for this host (y/n) [n]
```

- For a *y* answer, specify a local spool directory.

```
Enter the spool directory now!
```

- Enter the directory, and press *return*.

```
Using execd spool directory [<local_execd_spooldir>]
Hit <RETURN> to continue
```

- Press *return* to continue.

```
Creating local configuration
----------------------------

...

Local configuration for host ><hostname>< created.

Hit <RETURN> to continue >>
```

- Press *return* to continue.

9. Specify whether the daemon should be started at boot time.

```
execd startup script
--------------------

We can install the startup script that will
start execd at machine boot (y/n)
```

- Answer *y* if the daemon should be started at boot time.

```
cp <installation_path>/default/common/sgeexecd /etc/init.d/sgeexecd.p6444
/usr/lib/lsb/install_initd /etc/init.d/sgeexecd.p6444

Grid Engine execution daemon startup
------------------------------------

Starting execution daemon. Please wait ...
    starting sge_execd
```

- Hit return to continue.

```
Hit <RETURN> to continue

Add a default queue.

Adding a queue for this host
----------------------------

...

Do you want to add a default queue instance for this host (y/n)
```

- Answer *y* to add the host to the default queue, and press *return*.

```
root@<hostname> modified "@allhosts" in host group list
root@<hostname> modified "all.q" in cluster queue list

Hit <RETURN> to continue >>
```

10. Summary.

```
Using Grid Engine
-----------------

 You should now enter the command:

    source <installation_path>/default/common/settings.csh

 if you are a csh/tcsh user or

    # . <installation_path>/default/common/settings.sh

 if you are a sh/ksh user.

 This will set or expand the following environment variables:

    -$SGE_ROOT         (always necessary)
    -$SGE_CELL         (if you are using a cell other than >default<)
    -$SGE_CLUSTER_NAME (always necessary)
    -$SGE_QMASTER_PORT (if you haven't added the service >sge_qmaster<)
    -$SGE_EXECD_PORT   (if you haven't added the service >sge_execd<)
    -$PATH/$path       (to find the Grid Engine binaries)
    -$MANPATH          (to access the manual pages)

 Hit <RETURN> to see where Grid Engine logs messages
```

- Hit *return* to continue.

```
 Grid Engine messages
 --------------------
```

```
Grid Engine messages can be found at:

   /tmp/qmaster_messages (during qmaster startup)
   /tmp/execd_messages   (during execution daemon startup)

After startup the daemons log their messages in their spool directories.

   Qmaster:     <installation_path>/default/spool/qmaster/messages
   Exec daemon: <execd_spool_dir>/<hostname>/messages


Grid Engine startup scripts
---------------------------

Grid Engine startup scripts can be found at:

   <installation_path>/default/common/sgemaster (qmaster)
   <installation_path>/default/common/sgeexecd (execd)

Do you want to see previous screen about using Grid Engine again (y/n)
```

- Answer *n*, and press *return* to complete the installation.

```
Your execution daemon installation is now completed.
```

11. Review next steps.

- Continue to install the next execution host.

Removing Execution Hosts from Existing Clusters

1. Prepare to uninstall.

- Log in on the master host as user *root*.
- Set the necessary environment variables.

```
# . <installation_path>/default/common/settings.sh
```

- Change to the installation directory.

```
# cd $SGE_ROOT
```

- Be sure that jobs are not currently running on that host nor will any be started during the uninstallation.

2. Start the uninstallation.

- Execute the following command on a execution host as user *root* to uninstall the execution daemon:

```
./inst_sge -ux

Grid Engine uninstallation
--------------------------

You are going to uninstall a execution host <hostname>!
If you are not sure what you are doing, than please stop
this procedure with <CTRL-C>!

Hit <RETURN> to continue
```

- Press *return* to continue.

```
Grid Engine TCP/IP communication service
----------------------------------------

The port for sge_execd is currently set by the shell environment.

   SGE_EXECD_PORT = 6444

Hit <RETURN> to continue
```

- Press *return* to continue.

```
Checking hostname resolving
---------------------------

This hostname is known at qmaster as an administrative host.

Hit <RETURN> to continue
```

- Press *return* to continue.

```
hostname              <hostname>
load_scaling          NONE
complex_values        NONE
load_values           ...

...

Removing execution host <hostname> now!

...

Detected a presence of old RC scripts.
/etc/init.d/sgeexecd.p5000
```

3. Remove startup scripts.

```
Checking for installed rc startup scripts!

Removing execd startup script
-----------------------------

Do you want to remove the startup script
for execd at this machine? (y/n)
```

- Press *y* and *return* to remove the startup script for the execution host.

```
/usr/lib/lsb/remove_initd /etc/init.d/sgeexecd.p5000

Hit <RETURN> to continue
```

- Press *return* to finish the uninstallation.

4. OPTIONAL : Remove admin host privileges.

- If the host is not a shadow host or master host, and if it should not be allowed to execute administrative commands, then the administrator host privileges can be removed with the following command:

```
# qconf -dh <hostname>
```

Removing Shadow Master Hosts from Existing Clusters

1. Prepare to uninstall.

- Log in on the master host as user *root*.
- Set necessary environment variables.

```
# . <installation_path>/default/common/settings.sh
```

- Change to the installation directory.

```
# cd $SGE_ROOT
```

2. Start the uninstallation.

- Execute the following command on a shadow master host as user *root* to uninstall the shadow daemon:

```
./inst_sge -usm

Stopping shadowd!
shutting down Grid Engine shadowd
```

3. OPTIONAL : Remove admin host privileges.

- If the host is not also an execution host, and if it should not be allowed to execute administrative commands, then the administrator host privileges can be removed with the following command:

```
# qconf -dh <hostname>
```

Uninstalling Univa Grid Engine

1. Prepare to uninstall.

- Uninstall all shadow master hosts and execution hosts before continuing.
- Log in on the master host as user *root*.
- Set the necessary environment variables.

```
# . <installation_path>/default/common/settings.sh
```

- Change to the installation directory.

```
# cd $SGE_ROOT
```

2. Start the uninstallation.

```
# ./inst_sge -um

Uninstalling qmaster host
-------------------------
You're going to uninstall the qmaster host now. If you are not sure,
what you are doing, please stop with <CTRL-C>. This procedure will, remove
the complete cluster configuration and all spool directories!
Please make a backup from your cluster configuration!

Do you want to uninstall the master host?
```

- Enter *y* to continue with the uninstallation.

```
Checking Running Execution Hosts

no execution host defined
There are no running execution host registered!

Shutting down qmaster!
root@<hostname> kills qmaster
sge_qmaster is going down ...., please wait!
sge_qmaster is down!
Checking for installed rc startup scripts!
```

3. Remove startup scripts.

```
Removing qmaster startup script
-------------------------------

Do you want to remove the startup script
for qmaster at this machine? (y/n)
```

- Enter *y* and *return* to finish the uninstallation.

2.3.2 Automated Installation

The script `inst_sge` can be used to automate the installation of Univa Grid Engine. Instead of asking questions and expecting answers, this installation method directly reads installation parameters from a template file. Automated installation can be used to install the following host types:

- master host
- shadow host
- execution host
- administration host
- submit host

The `inst_sge` script must be executed on the on each host to install the specified host type.

Note
Windows execution nodes cannot currently be installed automatically using the `inst_sge` script.
Automated installation cannot be used if the administrator user of the cluster is *root*.

Follow these steps to start a fresh automated installation:

1. Prepare a configuration template.

- To be done before any installation is started.

2. Automate the master host installation.

- Requires a configuration template.

3. Automate the shadow master installation.

- Requires a configuration template.
- Complete the automated master host installation before starting the automated shadow master host installation.

4. Automate the execution host installation.

- Requires a configuration template.
- Complete the automated master host installation before starting the automated execution host installation.

Preparing Configuration Templates

1. Change the ownership of the `$SGE_ROOT` directory.

- Automated installation only works correctly if the admin user of the system is not *root*.
- The `$SGE_ROOT` directory, contents and sub-directories must be owned by that admin user. To change the ownership, execute the following command as user root:

```
# SGE_ROOT=<installation_path>
# export SGE_ROOT
# chown -R <admin_user> $SGE_ROOT
```

2. Modify the configuration template.

- Make a copy of the configuration template.

```
# cp $SGE_ROOT/util/install_modules/inst_template.conf $SGE_ROOT/util/
    install_modules/uge_configuration.conf
```

- Modify the copy of the configuration template.

```
# vi $SGE_ROOT/util/install_modules/uge_configuration.conf
```

```
001      #-------------------------------------------------
002      # SGE default configuration file
003      #-------------------------------------------------
004
005      # Use always fully qualified pathnames, please
006
007      # Path to a log file. If the file already exists, the log output
008      # will be appended.
009      # If empty, a log file will be created in <SGE_ROOT>/<SGE_CELL>/common
010      # The file needs to be writable by the admin user
011      LOG_FILE=""
012
013      # enable / disable Systemd startup scripts
014      USE_SYSTEMD="true"
015
016      # SGE_ROOT Path, this is basic information
017      #(mandatory for qmaster and execd installation)
018      SGE_ROOT="Please enter path"
019
020      # SGE_QMASTER_PORT is used by qmaster for communication
021      # Please enter the port in this way: 1300
022      # Please do not this: 1300/tcp
023      #(mandatory for qmaster installation)
024      SGE_QMASTER_PORT="Please enter port"
025
026      # SGE_EXECD_PORT is used by execd for communication
027      # Please enter the port in this way: 1300
028      # Please do not this: 1300/tcp
029      #(mandatory for qmaster installation)
030      SGE_EXECD_PORT="Please enter port"
031
032      # SGE_ENABLE_SMF
033      # if set to false SMF will not control SGE services
034      SGE_ENABLE_SMF="false"
035
036      # SGE_CLUSTER_NAME
037      # Name of this cluster (used by SMF as an service instance name)
038      SGE_CLUSTER_NAME="Please enter cluster name"
039
040      # SGE_JMX_PORT is used by qmasters JMX MBean server
041      # mandatory if install_qmaster -jmx -auto <cfgfile>
042      # range: 1024-65500
043      SGE_JMX_PORT="Please enter port"
044
045      # SGE_JMX_SSL is used by qmasters JMX MBean server
046      # if SGE_JMX_SSL=true, the mbean server connection uses
047      # SSL authentication
048      SGE_JMX_SSL="false"
049
```

```
050      # SGE_JMX_SSL_CLIENT is used by qmasters JMX MBean server
051      # if SGE_JMX_SSL_CLIENT=true, the mbean server connection uses
052      # SSL authentication of the client in addition
053      SGE_JMX_SSL_CLIENT="false"
054
055      # SGE_JVM_LIB_PATH is used by qmasters jvm thread
056      # path to libjvm.so
057      # if value is missing or set to "none" JMX thread will not be installed
058      # when the value is empty or path does not exit on the system, Grid Engine
059      # will try to find a correct value, if it cannot do so, value is set to
060      # "jvmlib_missing" and JMX thread will be configured but will fail to start
061      SGE_JVM_LIB_PATH="Please enter absolute path of libjvm.so"
062
063      # SGE_ADDITIONAL_JVM_ARGS is used by qmasters jvm thread
064      # jvm specific arguments as -verbose:jni etc.
065      # optional, can be empty
066      SGE_ADDITIONAL_JVM_ARGS="-Xmx256m"
067
068      # CELL_NAME, will be a dir in SGE_ROOT, contains the common dir
069      # Please enter only the name of the cell. No path, please
070      #(mandatory for qmaster and execd installation)
071      CELL_NAME="default"
072
073      # ADMIN_USER, if you want to use a different admin user than the owner,
074      # of SGE_ROOT, you have to enter the user name, here
075      # If this is left blank, on UNIX/Linux, the owner of the SGE_ROOT dir
076      # will be used as admin user. On native Windows (win-x86), the user that
077      # starts the installer will be used as admin user.
078      ADMIN_USER=""
079
080      # The ADMIN_PASSWORD is needed on win-x86 to install the UGE Starter Service
081      # properly. The UGE Starter Service is used to start the execution daemon
082      # at system boot time, thus, the ADMIN_PASSWORD is needed only if
083      # ADD_TO_RC is set to true.
084      ADMIN_PASSWORD=""
085
086      # The dir, where qmaster spools this parts, which are not spooled by DB
087      #(mandatory for qmaster installation)
088      QMASTER_SPOOL_DIR="Please, enter spooldir"
089
090      # The dir, where the execd spools (active jobs)
091      # This entry is needed, even if your are going to use
092      # lmdb spooling. Only cluster configuration and jobs will
093      # be spooled in the database. The execution daemon still needs a spool
094      # directory
095      #(mandatory for qmaster installation)
096      EXECD_SPOOL_DIR="Please, enter spooldir"
097
098      # For monitoring and accounting of jobs, every job will get
```

```
099      # unique GID. So you have to enter a free GID Range, which
100      # is assigned to each job running on a machine.
101      # If you want to run 100 Jobs at the same time on one host you
102      # have to enter a GID-Range like that: 16000-16100
103      #(mandatory for qmaster installation)
104      GID_RANGE="Please, enter GID range"
105
106      # If SGE is compiled with -spool-dynamic, you have to enter here, which
107      # spooling method should be used. (lmdb, classic, lmdb or postgres)
108      #(mandatory for qmaster installation)
109      SPOOLING_METHOD="lmdb"
110
118      # The dir, where the DB spools
119      # If LMDB db spooling is used, it must contain the path to
120      # the spooling db. Please enter the full path. (eg. /tmp/data/spooldb)
121      # Remember, this directory must be local on the qmaster host or alternatively
122      # can be a directory on a shared file system supporting full Posix locking
122      # semantics, e.g. NFSv4 or Lustre
123      DB_SPOOLING_DIR="spooldb"
124
125      # The connection parameters used to connect to the Postgres database
126      # if postgres spooling is used.
127      # Space separated list of key=value pairs, e.g.
128      # host=mydbhost dbname=ugespooling user=ugeadmin
129      PG_SPOOLING_ARGS="none"
130
131      # This parameter set the number of parallel installation processes.
132      # To prevent a system overload, or exceeding the number of open file
133      # descriptors the user can limit the number of parallel install processes.
134      # eg. set PAR_EXECD_INST_COUNT="20", maximum 20 parallel execd are installed.
135      PAR_EXECD_INST_COUNT="20"
136
137      # A List of Host which should become admin hosts
138      # If you do not enter any host here, you have to add all of your hosts
139      # by hand, after the installation. The auto-installation works without
140      # any entry
141      ADMIN_HOST_LIST="host1 host2 host3 host4"
142
143      # A List of Host which should become submit hosts
144      # If you do not enter any host here, you have to add all of your hosts
145      # by hand, after the installation. The autoinstallation works without
146      # any entry
147      SUBMIT_HOST_LIST="host1 host2 host3 host4"
148
149      # A List of Host which should become exec hosts
150      # If you do not enter any host here, you have to add all of your hosts
151      # by hand, after the installation. The autoinstallation works without
152      # any entry
153      # (mandatory for execution host installation)
```

```
154     EXEC_HOST_LIST="host1 host2 host3 host4"
155
156     # The dir, where the execd spools (local configuration)
157     # If you want configure your execution daemons to spool in
158     # a local directory, you have to enter this directory here.
159     # If you do not want to configure a local execution host spool directory
160     # please leave this empty
161     EXECD_SPOOL_DIR_LOCAL="Please, enter spooldir"
162
163     # Shell, which should be used for remote installation (rsh/ssh)
164     # This is only supported, if your hosts and rshd/sshd is configured,
165     # not to ask for a password, or promting any message.
166     SHELL_NAME="ssh"
167
168     # This remote copy command is used for csp installation.
169     # The script needs the remote copy command for distributing
170     # the csp certificates. Using ssl the command scp has to be entered,
171     # using  the not so secure rsh the command rcp has to be entered.
172     # Both need a passwordless ssh/rsh connection to the hosts, which
173     # should be connected to. (mandatory for csp installation mode)
174     COPY_COMMAND="scp"
175
176     # Enter your default domain, if you are using /etc/hosts or NIS configuration
177     DEFAULT_DOMAIN="none"
178
179     # If a job stops, fails, finish, you can send a mail to this adress
180     ADMIN_MAIL="none"
181
182     # If true, the rc scripts (sgemaster, sgeexecd) will be added,
183     # to start automatically during boottime
184     ADD_TO_RC="false"
185
186     #If this is "true" the file permissions of executables will be set to 755
187     #and of ordenary file to 644.
188     SET_FILE_PERMS="true"
189
190     # This option is not implemented, yet.
191     # When a exechost should be uninstalled, the running jobs will be rescheduled
192     RESCHEDULE_JOBS="wait"
193
194     # Enter a one of the three distributed scheduler tuning configuration sets
195     # (1=normal, 2=high, 3=max)
196     SCHEDD_CONF="1"
197
198     # The name of the shadow host. This host must have read/write permission
199     # to the qmaster spool directory
200     # If you want to setup a shadow host, you must enter the servername
201     # (mandatory for shadowhost installation)
202     SHADOW_HOST="hostname"
```

```
203
204     # Remove this execution hosts in automatic mode
205     # (mandatory for unistallation of execution hosts)
206     EXEC_HOST_LIST_RM="host1 host2 host3 host4"
207
208     # This option is used for startup script removing.
209     # If true, all rc startup scripts will be removed during
210     # automatic deinstallation. If false, the scripts won't
211     # be touched.
212     # (mandatory for unistallation of execution/qmaster hosts)
213     REMOVE_RC="false"
214
215     # This is a Windows specific part of the auto installation template
216     # If you going to install windows executions hosts, you have to enable the
217     # windows support. To do this, please set the WINDOWS_SUPPORT variable
218     # to "true". ("false" is disabled)
219     # (mandatory for qmaster installation, by default WINDOWS_SUPPORT is
220     # disabled)
221     WINDOWS_SUPPORT="false"
222
223     # Enabling the WINDOWS_SUPPORT, recommends the following parameter.
224     # The WIN_ADMIN_NAME will be added to the list of SGE managers.
225     # Without adding the WIN_ADMIN_NAME the execution host installation
226     # won't install correctly.
227     # WIN_ADMIN_NAME is set to "Administrator" which is default on most
228     # Windows systems. In some cases the WIN_ADMIN_NAME can be prefixed with
229     # the windows domain name (eg. DOMAIN+Administrator)
230     # (mandatory for qmaster installation, if windows hosts should be installed)
231     WIN_ADMIN_NAME="Administrator"
232
233     # If the WIN_ADMIN_PASSWORD is set, the UGE Starter Service will be installed
234     # using the full Administrator credentials.
235     # Setting this parameter makes sense only in conjunction with WIN_ADMIN_NAME.
236     WIN_ADMIN_PASSWORD=""
237
238     # SGE_ROOT Path on the Windows hosts in DOS or UNC notation
239     WIN_SGE_ROOT="Please enter path"
240
241     # local spool directory of the Windows execution host in DOS or UNC notation
242     WIN_EXECD_SPOOL_DIR_LOCAL="Please enter path"
243
244     # The /var/sgeCA path on UNIX in UNIX notation
245     UNIX_SGECA_DIR="Please enter path"
246
247     # The /var/sgeCA path on Windows in DOS or UNC notation
248     WIN_SGECA_DIR="Please enter path"
249
250     # The users home directory on Windows in UNIX notation
251     UNIX_HOME_DIR="Please enter path"
```

```
252
253     # The users home directory on Windows in DOS or UNC notation
254     WIN_HOME_DIR="Please enter path"
255
256     # The local temporary directory on UNIX for all users in UNIX notation
257     UNIX_TMP_DIR="Please enter path"
258
259     # The local temporary directory on Windows for all users in DOS or UNC notation
260     WIN_TMP_DIR="Please enter path"
261
262     # This section is used for csp installation mode.
263     # CSP_RECREATE recreates the certs on each installtion, if true.
264     # In case of false, the certs will be created, if not existing.
265     # Existing certs won't be overwritten. (mandatory for csp install)
266     CSP_RECREATE="true"
267
268     # The created certs won't be copied, if this option is set to false
269     # If true, the script tries to copy the generated certs. This
270     # requires passwordless ssh/rsh access for user root to the
271     # execution hosts
272     CSP_COPY_CERTS="false"
273
274     # csp information, your country code (only 2 characters)
275     # (mandatory for csp install)
276     CSP_COUNTRY_CODE="DE"
277
278     # your state (mandatory for csp install)
279     CSP_STATE="Germany"
280
281     # your location, eg. the building (mandatory for csp install)
282     CSP_LOCATION="Building"
283
284     # your arganisation (mandatory for csp install)
285     CSP_ORGA="Organisation"
286
287     # your organisation unit (mandatory for csp install)
288     CSP_ORGA_UNIT="Organisation_unit"
289
290     # your email (mandatory for csp install)
291     CSP_MAIL_ADDRESS="name@yourdomain.com"
292
293     # Defines the number of initially created read-only threads in qmaster
294     # 0 disables the read-only thread completely. Value has to be in range [0; 64]
295     # Recommended is 2. Values >0 allows to adjust number of threads during runtime.
296     SGE_READER_THREADS="2"
297
298     # -------------------------------------------------------
299     #
300     # Please be aware that changes to the settings below may result in a qmaster that is n
```

```
301      # working correctly. In order to get a fully working qmaster changes to the configurat
302      # may be necessary after the installation (e.g. via Config-API).
303      #
304      # ----------------------------------------------------------
305
306      # Start qmaster during installation.
307      # Some actions (e.g. adding hosts) cannot be performed if the qmaster is
308      # not started.
309      START_QMASTER="true"
310
311      # Start execution daemon during installation.
312      # If set to false, no default queue will be added.
313      START_EXECD="true"
314
315      # Copy common files from the util directory to the directory to
316      # <SGE_ROOT>/<SGE_CELL>/common
317      COPY_COMMON_FILES="true"
318
319      # If CREATE_SETTINGS_FILE is set to false, no settings.sh/.csh will
320      # be created during installation.
321      CREATE_SETTINGS_FILE="true"
322
323      # Path to a file that will be used as bootstrap file.
324      # The file will be copied to <SGE_ROOT>/<SGE_CELL>/common/bootstrap.
325      # If empty, a default bootstrap file will be created.
326      BOOTSTRAP_FILE=""
327
328      # If ADD_DEFAULTS is set to true, a default configuration will
329      # be set during install and the installed qmaster is fully working.
330      # If it is set to false, only a minimal necessary configuration will be
331      # set.
332      ADD_DEFAULTS="true"
333
334      # Path to a configuration file for the qmaster, similiar to
335      # a configuration file provided for qconf -Mconf.
336      # If empty, a default configuration will be set.
337      QMASTER_CONFIG_FILE=""
338
339      # Path to a configuration file for the scheduler.
340      # If empty, a default configuration will be set according to the SCHEDD_CONF setting.
341      SCHEDD_CONFIG_FILE=""
342
343      # Path to a directory with complex entries to be added during qmaster
344      # installation. If empty and ADD_DEFAULTS is set to false,
345      # only a minimal set of complex attributes will be added.
346      CENTRY_DIR=""
347
348      # Path to a folder containing parallel environment entries (PEs) that will
349      # be added during qmaster installation.
```

```
350     # If empty and ADD_DEFAULTS is set to false, no PEs will be added.
351     PE_DIR=""
352
353     # Path to a folder containing usersets that will be added during qmaster
354     # installation.
355     # If empty and ADD_DEFAULTS is set to false, no usersets will be added.
356     USERSETS_DIR=""
357
358     # Path to the license file that will be specified as license file
359     # for the UGE installation.
360     LIC_SRC_PATH="$SGE_ROOT/util/resources/licenses/uge-48core-4gpu.txt"
```

Note

The JMX MBean server functionality is not supported in Univa Grid Engine 8.0; the following parameters can therefore be ignored:

-SGE_JMX_PORT
-SGE_JMX_SSL
-SGE_JMX_SSL_CLIENT
-SGE_JMX_SSL_KEYSTORE
-SGE_JMX_SSL_KEYSTORE_PW
-SGE_JVM_LIB_PATH
-SGE_ADDITIONAL_JVM_ARGS

Note

BSD server spooling is no longer supported after version 6.2u7; therefore, `DB_SPOOLING_SERVER` must be set to *none*.

Note

If execution host local spooling should not be enabled, then set `EXECD_SPOOL_DIR_LOCAL` to an empty string "".

Start the Automated Installation

1. Select parameters for the `inst_sge` script.

- The `inst_sge` script has a number of command line parameters that enable the different hosts installations:

Flag	Description
-auto	Enables the automated installation
-m	Install master host
-x	Install execution host
-sm	Install shadow master host
-s	Install submit host

Flag	Description
-csp	Enables enhanced security features (CSP)

Table 4: Command-line Options for `inst_sge`

- The different flags can be combined.

2. Start the `inst_sge` script.

```
# cd $SGE_ROOT
```

- The command above starts the automated installation on the local host. This will install the master and execution host functionality.

3. Verify the installation result.

- The script creates a log file named `$SGE_ROOT/default/spool/qmaster/install_<hostname>_<date>_<time` where `<hostname>` is the hostname of the local host and `<date>` and `<time>` are the date and time when the automated installation was started. Open that log file to see if any errors occurred.

Automated Uninstallation

1. Select parameters for the `inst_sge` script.

- The `inst_sge` script has a number of command line parameters that enable the different hosts uninstallations.

Flag	Description
-auto	This enables the automated uninstallation.
-um	Uninstall master host.
-ux	Uninstall execution host.
	Note In contrast to the installation, the `EXEC_HOST_LIST_RM` parameter specifies the hosts that will be uninstalled. Do not use the parameter `EXEC_HOST_LIST` during uninstallation.
-usm	Install shadow master host.
-csp	Enables enhanced security features (CSP).

Table 5: Command-line Options for `inst_sge`

- The different flags can be combined.

2. Start the `inst_sge` script.

```
# cd $SGE_ROOT
```

- The command above starts the automated uninstallation on the local host. This will uninstall execution host functionality on the specified hosts.

3. Verify the installation result

- The script creates a log file named `$SGE_ROOT/default/spool/qmaster/install_<hostname>_<date>_<time` where `<hostname>` is the hostname of the local host and `<date>` and `<time>` are the date and time when the automated installation was started. Open that log file to see if any errors occurred.

2.4 Verifying the Installation

In between the main installation steps of the master, shadow master, and execution host installation, verify that the Univa Grid Engine cluster installed so far is running properly. To do so, check if the corresponding daemons are running and if they can be contacted. Simple administrative commands can be executed to see if the daemons respond properly before test jobs should be sent into the cluster.

2.4.1 Verify That Daemons are Running

1. Log in to the host.

- To check if components are running, log in to the hosts to be verified.
- All Univa Grid Engine daemons and clients require that the environment variables `SGE_ROOT`, `SGE_QMASTER_PORT`, `SGE_EXECD_PORT` and `SGE_CELL` are set correctly so that they behave properly. To set those variables, the Bourne shell script `<installation_path/<cell>/common/settings.sh` and the tcsh script `<installation_path/<cell>/common/settings.csh` can be sourced before a Univa Grid Engine is started. Both scripts are created during the installation process. Depending on the host architecture where they are sourced, they also ensure that the shared library path is set correctly.
- The port variables are not necessary if the `/etc/services` file or the corresponding NIS/NIS+ map contains the entries *sge*qmaster_ and *sge*execd_.

2. Find running Univa Grid Engine components.

- Since the Univa Grid Engine daemon processes contain the character sequence *sge* in their names, the following command will show all running daemon processes.

```
# ps -efa | grep sge
```

3. Find the reasons why services are not running.

- When daemons are not running as expected, look in the message file of that component, located in the corresponding spooling directory and named *messages*.

4. (Re)start services.

- To start or restart a daemon, execute the corresponding startup script on the host.
- `$SGE_ROOT/$SGE_CELL/common/sgemaster` will start the master daemon.
- `$SGE_ROOT/$SGE_CELL/common/sgeexecd` will start the execution daemon.
- Startup script accepts the parameter *start* to start a service, but they can also be used to shut down the corresponding component by passing *stop* as the first parameter.

Run Simple Commands

1. Set up the environment.

- Take care that the environment is properly set up as outlined in the previous chapter.

2. Execute client commands.

- The following command can be executed to request the global configuration from the master component.

```
# qconf -sh
```

- If this command displays the global configuration and does not return with an error, then the master component is up and running.
- On submit hosts, the `qstat` command can be used by any user to get response from qmaster if it is running.
- If qmaster is down, then this command will return with the error message.

```
# qstat
error: commlib error: got select error (Connection refused)
```

Start Test Jobs

1. Start test jobs.

- The `$SGE_ROOT` directory contains some example jobs in the directory `$SGE_ROOT/examples/jobs`. Execute the sleeper job to see if the cluster works properly.

```
# qsub $SGE_ROOT/examples/jobs/sleeper.sh 60
```

- This will submit a sleeper job that, when executed, will sleep for 60 seconds.
- Observe the job with the `qstat` command to watch the state changes.

2. Check output and error file.

- After the job has finished, output and/or error files can be found in the users home directory. The names of those files are `<jobname>.e<jobid>` and `<jobname>.o<jobid>`.

2.5 Modifying the License File (Univa Grid Engine 8.7.0 and newer)

A new license can be installed by using the command:

```
# qconf -Mlic <license_file>
```

where `<license_file>` describes the path to the license file to be installed. The old license file will be replaced by the new one.

Alternatively, a license file can be directly copied to

```
# $SGE_ROOT/$SGE_CELL/common/uge_license.txt
```

Univa Grid Engine will detect a newly installed license file after a few seconds.

In order to verify the currently active license file, the command

```
# qconf -slic
```

can be used to show its content.

2.6 Post-Installation Steps

The core Univa Grid Engine installation is now finished. The cluster is now ready for installation of additional components like ARCo, as outlined in the next section, or for configuration of the cluster.

3 Setting Up the Accounting and Reporting Database

For an introduction to Accounting and Reporting, see Concepts and Components -> The Accounting and Reporting Database.

3.1 Prerequisites

Before installing ARCo, make sure the Univa Grid Engine is installed and the `sge_qmaster` component is running.

The `$SGE_ROOT` directory must be available (be mounted) on the host running `dbwriter`. `dbwriter` can run on any host, but running it on the same host as the database server typically results in the best performance.

`dbwriter` requires a database server that is running one of the following supported database systems:

- PostgreSQL >= 8.0
- MySQL >= 5.0
- Oracle >= 10g

`dbwriter` is a Java application that requires the availability of Java Version 1.6 update 4 or newer. To find out which version of Java is running on a machine, execute the following command:

```
$ java -version
java version "1.6.0_21"
```

`dbwriter` requires access to the database server via JDBC, so install a suitable JDBC driver that corresponds to the installed database server:

- PostgreSQL
  - Download the JDBC4 driver from http://jdbc.postgresql.org/download.html
  - Copy it to `$SGE_ROOT/dbwriter/lib`
- MySQL
  - Download the driver package from http://dev.mysql.com/downloads/connector/j
  - Unpack it in a temporary directory.
  - Copy `mysql-connector-java-<version>.jar` to `$SGE_ROOT/dbwriter/lib`
- Oracle
  - Use the JDBC driver delivered with the Oracle installation.
  - Copy `$ORACLE_HOME/jdbc/lib/ojdbc14.jar` to `$SGE_ROOT/dbwriter/lib`

The disk space / database size required for running ARCo highly depends on the Univa Grid Engine setup and the `dbwriter` configuration. The following parameters influence the required disk space:

- Cluster size

- Job throughput
- Number of monitored hosts or queue specific variables
- Enabled special features (job log, share log)
- Configured `dbwriter` derived values rules
- `dbwriter` deletion rules

`dbwriter` has moderate memory requirements, so tuning via Java's command line arguments are usually not required.

Setting up the Database

`dbwriter` requires a minimum setup for operation:

- A database (default: `arco`)
- A user who has full access to the database (default: `arco_write`). This user will usually be the owner of the database, and must have permission to create/alter/delete tables and views, create/alter/delete records in the database tables, grant access to the database tables and views.
- A user who has read access to the database (default:`arco_read`). This user shall be used for accessing the reporting database by reporting tools. During the `dbwriter` installation, this user will get read access granted for tables and views in the reporting database.

The following sections describe how to create the reporting database and the database users in the various supported database systems.

PostgreSQL

General Setup

For installation of the PostgreSQL database server, use the packages delivered with the operation system, especially with Linux distributions.

To install it from scratch instead, get the software from http://www.postgresql.org/ and follow the instructions in the PostgreSQL documentation.

For running `dbwriter` with PostgreSQL, make sure the PostgreSQL database is running and accessible via internet socket.

The following two configuration files contain the necessary parameters for configuring access to the PostgreSQL database:

- postgresql.conf:

Make sure listen_addresses is set to "*" or contains the IP address of the host running `dbwriter`:

```
listen_addresses = '*'                  # what IP address(es) to listen on;
                                        # comma-separated list of addresses;
                                        # defaults to 'localhost', '*' = all
                                        # (change requires restart)
port = 5432                             # (change requires restart)
```

- pg_hba.conf:

This configuration file contains rules for client authentication.

Allow access to the database server from the required hosts (at least the host running `dbwriter`).

The following line in `pg_hba.conf` will grant all hosts in network 192.168.56.0 access to all databases in the PostgreSQL server where authentication is done via md5 encrypted password:

```
host    all         all         192.168.56.0/24       md5
```

If `dbwriter` is running on the database host, the following line will allow access to the database from localhost only:

```
host    all         all         127.0.0.1/32          md5
```

After changing `postgresql.conf` or `pg_hba.conf`, restart the PostgreSQL server.

```
/etc/init.d/postgresql restart
```

Creating the `arco` Users and the `arco` Database

Before starting the `dbwriter` installation, first create `arco` specific PostgreSQL users and an `arco` database.

Execute the following steps as the postgres user:

1. Create the `arco_write` user.

   - The `arco_write` user is the owner of the `arco` database and has full access to the `arco` database. The `dbwriter` will connect to the `arco` database as user `arco_write`.

     ```
     $ createuser -S -D -R -l -P -E arco_write
     Enter password for new role:
     Enter it again:
     $
     ```

2. Create the `arco` database.

   ```
   $ createdb -O arco_write arco
   ```

3. Create the `arco_read` user.

   - The `arco_read` user has read only access to the `arco` database, and it should be used to run queries on the `arco` database.

```
$ createuser -S -D -R -l -P -E arco_read
Enter password for new role:
Enter it again:
```

MySQL

General Setup

Install MySQL via the operating system's package manager or from scratch following the instructions on http://www.mysql.com.

The main configuration file for MySQL is `my.cnf`. For example, Debian packages on Ubuntu Linux install it in `/etc/mysql/my.cnf`.

If `dbwriter` is running on a host different than the host running the MySQL server, make sure `mysqld` listens on the correct network interface by modifying the `bind-address` parameter:

```
bind-address = 192.168.56.100
```

Or make `mysqld` listen on all network interfaces:

```
bind-address = 0.0.0.0
```

Creating the `arco` Users and the `arco` Database

Assuming user root has MySQL administrative rights, start the `mysql` command line client:

```
mysql -u root -p

Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 36
Server version: 5.1.41-3ubuntu12.10 (Ubuntu)

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Create the `arco_write` user:

```
mysql> CREATE USER arco_write IDENTIFIED BY '<password>'
```

Create the `arco` database:

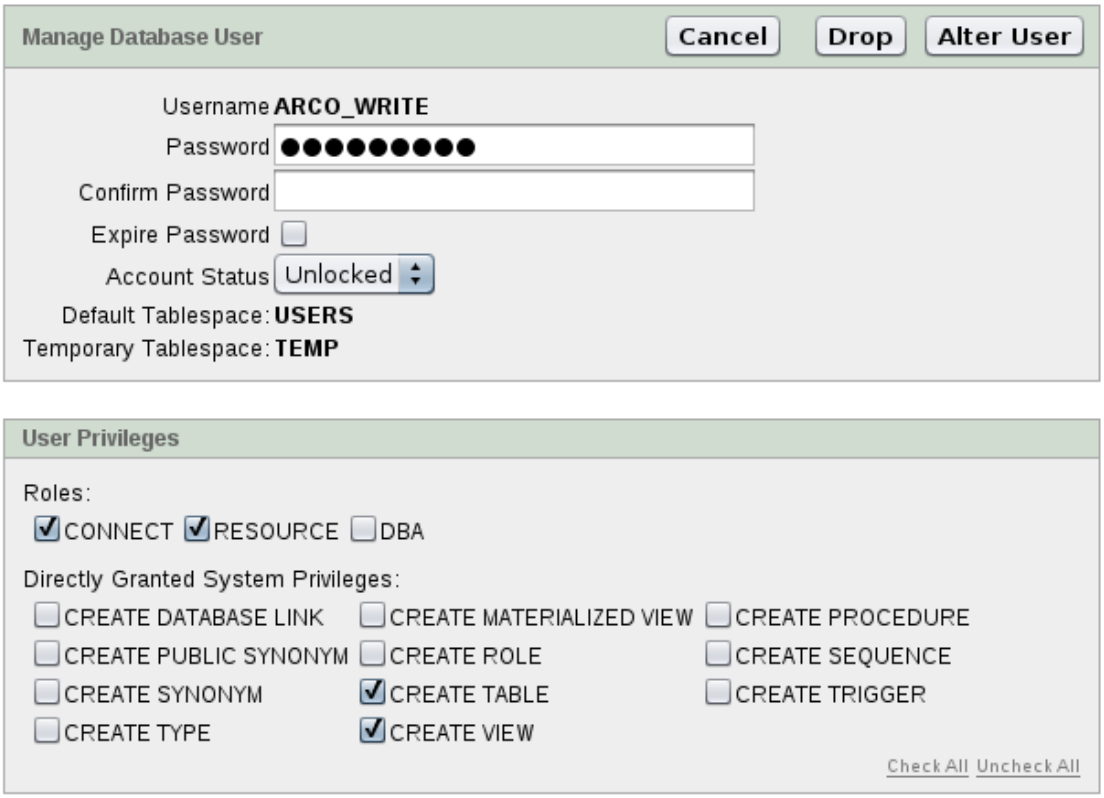
```
mysql> CREATE DATABASE arco;
mysql> GRANT ALL ON arco.* TO arco_write WITH GRANT OPTION;
```

Oracle

Install Oracle, and create a database instance for ARCo. Alternately, ask the database administrator to provide a database instance.

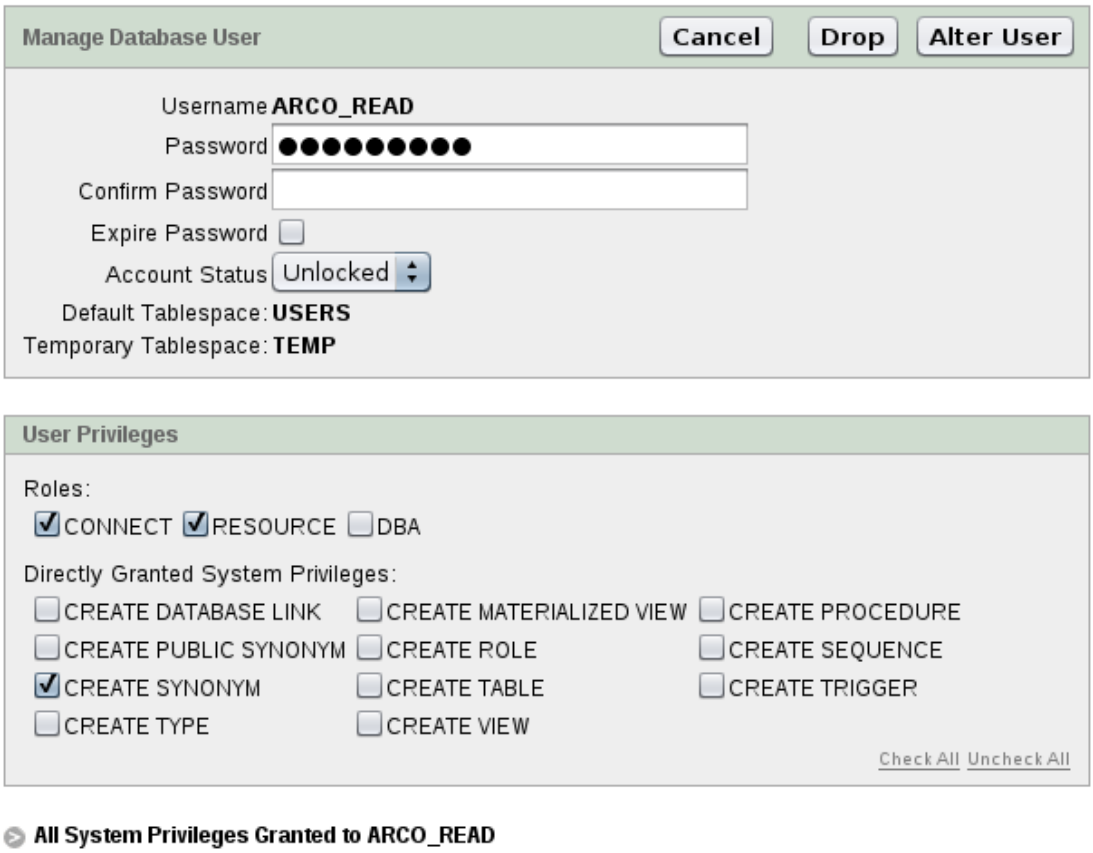
Create the users `arco_write` and `arco_read`.

The `arco_write` user needs to be able to create tables and views:



The `arco_read` user needs to be able to create synonyms.

Access to the tables created during installation in the ARCo database is granted to `arco_read` at installation time.



3.2 Installing the `dbwriter`

Before starting the `dbwriter` installation, make sure the following requirements are met:

- The Univa Grid Engine is installed and running.
- A database server is installed with `arco` database, and `arco_write` and `arco_read` users have been created.

The installation procedure asks for the following parameters, some of which provide suggested defaults that reflect a standard setup for the database and `dbwriter`:

- `SGE_ROOT` - root directory of the Univa Grid Engine installation
- `SGE_CELL` - cell directory of the Univa Grid Engine installation
- database type - PostgreSQL, MySQL, Oracle
- host - name of the host running the database server
- port - socket port used to contact the database server

- database - name of the database (default: `arco`)
- write user - name of the user having write access to the database (default: `arco_write`)
- read user - name of a user having read access to the database (default: `arco_read`)

Install the ARCo Package.

As root, cd to the Univa Grid Engine root directory (`SGE_ROOT`), and unpack the ARCo package:

```
tar xzf <package_directory>/ge-Univa Grid Engine 8.6.x-arco.tar.gz
```

Install the `dbwriter`.

Install `dbwriter` by the installation script `dbwriter/inst_dbwriter`.

Option	Description
-nosmf	use rc scripts instead of SMF on Solaris 10 and higher
-upd	update from versions prior to 6.2
-rmrc	remove 6.2 RC scripts of SMF service
-h	show the inst_dbwriter help

Table 6: `inst_dbwriter` Command Line Options

The following steps describe the `dbwriter` installation, using a PostgreSQL database for the examples:

1. Start the `dbwriter` installation.
   cd source /common/settings.sh cd dbwriter ./inst_dbwriter
2. Accept the license agreement.

   - The license agreement gets displayed in the preferred PAGER. To continue installation, accept the license agreement by entering *y*:

     ```
     ...
     Do you agree with that license? (y/n) [n] >> y
     ```

3. Describe the Univa Grid Engine installation.

   - The following screens ask about the Univa Grid Engine installation - the defaults presented should match the installation.

4. Enter the path to the Java installation.

- If JAVA_HOME is set in the environment, the path to the Java installation will be filled in automatically.

```
Java setup
----------

ARCo needs at least java 1.6.0_04

Enter the path to your java installation [] >> /usr/lib/jvm/java-6-sun
```

5. Select the database type.

```
Setup your database connection parameters
-----------------------------------------

Enter your database type ( o = Oracle, p = PostgreSQL, m = MySQL ) [] >> p
```

6. Enter the host name of the database server.

```
Enter the name of your postgresql database host [] >> hapuna
```

7. Enter the port of the database server.

- Unless some special setup was performed, press RETURN to accept the default.

```
Enter the port of your postgresql database [5432] >>
```

8. Enter the name of the database.

- Press RETURN to use the default.

```
Enter the name of your postgresql database [arco] >>
```

9. Specify the database user with write access.

- Press RETURN to use the default.

```
Enter the name of the database user [arco_write] >>
```

10. Enter the password of the user with write access.

```
Enter the password of the database user >>
Retype the password >>
```

11. Configure a table space to use instead of the default.

- Separate table spaces can be used for data (tables) and indexes. Using separate table spaces for data and indexes (on separate file systems) can significantly increase database performance.
- Press RETURN to accept the default.

```
Enter the name of TABLESPACE for tables [pg_default] >>
Enter the name of TABLESPACE for indexes [pg_default] >>
```

12. Enter the name of the database schema.

- Using different schemas can be used in multi cluster setup running multiple instances of `dbwriter` storing data into the same ARCo database.
- Press RETURN to accept the default.

```
Enter the name of the database schema [public] >>
```

13. Enter the name of the database user with read only access.

- Reporting applications should connect to the database with a user who has restricted access.
- The name of this database user is needed to grant him access to the sge tables and must be different from `arco_write`.

```
Enter the name of this database user [arco_read] >>
```

14. Perform a database connection test.

- At this point, the installation script has enough information to perform a connection test on the database.
- If the JDBC driver has not yet been installed in <ccode>`$SGE_ROOT/dbwriter/lib`, the following screen will be shown. Copy the JDBC driver to `$SGE_ROOT/dbwriter/lib` and press RETURN to restart the connection test.

```
Database connection test
------------------------

Searching for the jdbc driver org.postgresql.Driver
in directory /home/joga/develop/univa/clusters/mt/dbwriter/lib

Error: jdbc driver org.postgresql.Driver
        not found in any jar file of directory
        /home/joga/develop/univa/clusters/mt/dbwriter/lib

Copy a jdbc driver for your database into
this directory!

Press enter to continue >>
```

15. Set the `dbwriter` parameters.

    - The following screen asks for a number of parameters influencing `dbwriter` operation:
      - interval between two `dbwriter` runs
      - path of the `dbwriter` spool directory
      - path to the file containing rules for the calculation of derived values and deletion rules
      - `dbwriter` debug level
    - For standard installations accept the default values by pressing RETURN.

16. Review the parameters.

    - This screen shows the previously entered parameters. Enter *y* to accept the values, or enter *n* to restart the installation process.

```
All parameters are now collected
--------------------------------

        SGE_ROOT=/home/joga/develop/univa/clusters/mt
        SGE_CELL=default
       JAVA_HOME=/usr/lib/jvm/java-6-sun (1.6.0_24)
          DB_URL=jdbc:postgresql://hapuna:5432/arco
         DB_USER=arco_write
       READ_USER=arco_read
      TABLESPACE=pg_default
TABLESPACE_INDEX=pg_default
       DB_SCHEMA=public
        INTERVAL=60
       SPOOL_DIR=/home/joga/develop/univa/clusters/mt/default/spool/dbwriter
    DERIVED_FILE=/home/joga/develop/univa/clusters/mt/dbwriter/database/
                 postgres/dbwriter.xml
     DEBUG_LEVEL=INFO

Are these settings correct? (y/n) [y] >>
```

17. Create the database tables.

    - In an initial installation, the `arco` database will still be empty and no tables will be found.
    - Press RETURN to have the database tables be generated.

```
Database model installation/upgrade
-----------------------------------
Query database version ... no sge tables found
New version of the database model is needed

Should the database model be upgraded to version 10 6.2u1? (y/n) [y] >>
```

18. Review the configuration file information.

    - After the database has been initialized, the startup script and the configuration file are generated, and their paths are output for information.
    - Press RETURN to continue.

```
...
Version 6.2u1 (id=10) successfully installed
OK

Create start script `sgedbwriter` in `/home/joga/develop/univa/clusters/mt/
                                        default/common`

Create configuration file for `dbwriter` in `/home/joga/develop/univa/
                                clusters/mt/default/common`

Hit <RETURN> to continue >>
```

19. Install the startup scripts.

    - Select *y* to have `dbwriter` started at boot time.

```
dbwriter startup script
-----------------------

We can install the startup script that will
start dbwriter at machine boot (y/n) [y] >>
```

20. Start `dbwriter`.

    - If the following screen is shown, the `dbwriter` installation succeeded, and `dbwriter` is running.

```
Creating dbwriter spool directory /home/joga/develop/univa/clusters/
                                     mt/default/spool/dbwriter
starting dbwriter
dbwriter started (pid=19052)
Installation of dbwriter completed
```

3.3 Starting and Stopping the `dbwriter`

Start `dbwriter`:

```
$SGE_ROOT/$SGE_CELL/common/sgedbwriter [start]
```

Stop `dbwriter`:

```
$SGE_ROOT/$SGE_CELL/common/sgedbwriter stop
```

3.4 Configuring Univa Grid Engine Reporting

Once `dbwriter` is installed and running, it is ready to store data produced by `sge_qmaster` in the `arco` database.

Generation of reporting data in `sge_qmaster` has to be switched on. Which data to write to the reporting database can be configured.

Enabling Reporting

Enabling reporting and activating special reporting features like job log or share log is done in the global configuration.

Edit the global configuration by issuing the following command:

```
qconf -mconf
```

The global configuration is loaded into an EDITOR. Go to the line specifying the reporting:

```
reporting_params              accounting=true reporting=false \
                              flush_time=00:00:05 joblog=true sharelog=00:10:00
```

Setting `reporting=true` enables reporting.

See also Common_Tasks -> Understanding and Modifying the Cluster Configuration.

3.5 Configuring which Variables to Report

Besides job information (job log and accounting) and sharetree usage information (share log), values of complex variables can be written to the reporting database:

- load values can be written to the reporting database whenever they are reported by a `sge_execd`
- values of consumables can be written whenever they change

To activate reporting of complex variables, configure them in the `reporting_variables` attribute in these places:

- The global host to have them written for all hosts in the cluster, e.g. slots, global licenses, load values like load_avg, cpu, mem_free.
- a specific host to have them written for that host only, e.g. a special host specific license.

Modifying the reporting variables is done by editing the execution host:

```
qconf -me global
```

Edit the report_variables attribute:

```
report_variables   slots, license, np_load\_avg, cpu, mem_free
```

See also Common_Tasks -> Configuring Hosts.

3.6 Configuring Rules

`dbwriter` contains a rule engine that executes rules in defined intervals.

Rules can be used for the following purposes:

- generation of new data, e.g. values derived from the raw data stored into the reporting database from `sge_qmaster` reporting data or statistical data
- deletion of outdated data to limit the size of the reporting database

Rules are defined in a configuration file in XML format:

```
$SGE_ROOT/dbwriter/database/<database type>/dbwriter.xml
```

where `database_type` is one of the following:

- `mysql`
- `oracle`
- `postgres`

The XML `dbwriter` configuration file can contain 3 types of XML nodes:

- `derive` specifying rules for generation of derived values
- `statistic` specifying rules for generation of statistical information
- `delete` specifying when (after what time interval) data gets deleted from the reporting database

The file format is as follows:

```
<DbWriterConfig>
  <derive ...>
    ...
  </derive>

  <statistic ...>
     ...
  </statistic>

  <delete ...>
    ...
  </delete>
</DbWriterConfig>
```

The `dbwriter.xml` file can contain any number of derive, statistic and delete rules, which are explained in more detail in the following sections.

Derived Values

Derived value rules use the raw data from the reporting file generated by `sge_qmaster`. `dbwriter` takes that raw data, generates new data from it, and writes the new data to the reporting database.

There are two types of derived values rules used for 2 different purposes:

- Automatic derived values rules are used to apply mathematical functions on existing data, such as average, minimum, maximum on certain data over a time period.
- SQL based derived value rules can be used to generate completely new data items by running arbitrary SQL queries on the data in the reporting database.

All derived value rules have the following attributes in common:

- **object**: Specifies on which data in the reporting database the derived value rule will operate. The following values are valid:
  - *department*: Rule operates on data in the tables `sge_department` and `sge_department_values`. Derived values get stored into the table `sge_department_values`.
  - *group*: Rule operates on data in the tables `sge_group` and `sge_group_values`.
  - *host*: Rule operates on data in the tables `sge_host` and `sge_host_values`.
  - *project*: Rule operates on data in the tables `sge_project` and `sge_project_values`.
  - *queue*: Rule operates on data in the tables `sge_queue` and `sge_queue_values`.
  - *user*: Rule operates on data in the tables `sge_user` and `sge_user_values`.
- **interval**: Specifies the time interval used for data generation, such as generating hourly averages, daily minimum etc. The following values are valid:
  - *hour*
  - *day*
  - *month*
  - *year*
- **variable**: The name of the variable that holds the data generated by the derived value rule. For example, a variable *h*cpu_ might contain hourly averages of the raw data in the variable *cpu*.

Automatic Derived Value Rules

Automatic derived value rules are used to apply mathematical functions to data, such as average, minimum, or maximum, on arbitrary values of a specific complex variable over a specific time period.

Example:

```
<derive object="host" interval="hour" variable="h_cpu">
  <auto function="AVG" variable="cpu" />
</derive>
```

The example above reads the values of the complex variable *cpu* from the database table sge__host**values of the last *hour*, calculates the average (*AVG*) of the values, and stores the result in the variable *h*cpu_ in table sge**host__values.

The mathematical functions are the functions available in the respective database system. The following are commonly available functions:

- AVG: average value of all individual values in the analyzed time interval
- MIN: minimum
- MAX: maximum
- COUNT: number of individual values in the analyzed time interval

SQL based Derived Value Rules

SQL based derived value rules allow the generation of data via arbitrary SQL statements.

The SQL statement must return a single row with the following columns:

- time_start
- time_end
- value

*time*start_ and *time*end_ specify the time range for which *value* is valid. The storage location for *value* is defined in the `<derive>` node.

Example:

```
<derive object="user" interval="hour" variable="h_jobs_finished">
```

The example above defines that a variable *h*jobs_finished_ gets stored in the table sge__user__values holding *hour*ly values.

The SQL query can contain special placeholders that are filled in by dbwriter's derived value engine:

- `__key_0__`, `__key_1__`: Primary key of the parent table
- `__time_start__`: Start time of the analyzed time interval
- `__time_end__`: End time of the analyzed time interval

Warning

Less than (<) and greater than (>) signs cannot be directly written into the SQL statement, use XML syntax instead: < for <, > for >, <= for <=, >= for >=

Example of a SQL based derived value rule: The following rule stores how many jobs have finished per user and hour in a variable `h_jobs_finished` in the `user_values` table (written for PostgreSQL).

- The query is called once an hour to generate hourly values.
- It is called once per user in the table `sge_user`.
- The place holder `__key_0__` is replaced by the primary key of the table `sge_user` (the user name).
- The place holders `__time_start__` and `__time_end__` are replaced by the start and end times of the analyzed time intervals.
- The query retrieves accounting records of all jobs for a specific user that finished in the defined time interval and counts them.
- The result is stored in the table `sge_user_values` in the `variable h_jobs_finished`.

```
<derive object="user" interval="hour" variable="h_jobs_finished">
  <sql>
    SELECT DATE_TRUNC('hour', ju_end_time) AS time_start,
           DATE_TRUNC('hour', ju_end_time) + INTERVAL '1 hour' AS time_end,
           COUNT(*) AS value
    FROM sge_job, sge_job_usage
    WHERE j_owner = __key_0__ AND
          j_id = ju_parent AND
          ju_end_time &amp;lt;= '__time_end__' AND
          ju_end_time &amp;gt; '__time_start__' AND
          ju_exit_status != -1 AND
          j_pe_taskid = 'NONE'
    GROUP BY time_start
  </sql>
</derive>
```

Statistical Values

Statistic rules can be used to generate statistical data stored in the tables `sge_statistic` and `sge_statistic_values`. `dbwriter` itself writes statistical data into these tables. Here are some examples of the statistical data that can be captured:

- The speed for storing data from the reporting file into the database in lines per second.
- The time `dbwriter` needs for calculating derived values, or for deleting outdated values, etc.

A rule for generating statistical data is similar to derived value rules and has the following attributes:

- **interval**: time interval in which the rule is executed, one of the following:
  - *hour*
  - *day*
  - *month*
  - *year*
- **variable**: name of a variable holding specific statistic data over time
- **type**: describes the data source, either of the following:
  - *seriesFromColumns*: The query specified for the statistics rule returns one row containing data; the statistic's name is taken from the column header.
  - *seriesFromRows*: The query specified returns multiple rows with two columns; one column contains the statistic's name, and the other one the value.
- **nameColumn** (needed when type=seriesFromRows): name of the column to be used for the statistic's name
- **valueColumn** (needed when type=seriesFromRows): name of the column to be used for the statistic's value

A statistics rule also contains a `<sql>` subnode listing the SQL query used to produce the statistics data.

Examples

The following examples show how different types of statistic rules work. Written for MySQL, both the rule and some sample output of the generated data are shown. Raw data produced by statistic rules can be post processed by derived value rules. Deletion rules are used to delete outdated values.

Number of records in the various ARCo tables

This statistic rule is part of the `dbwriter.xml` file delivered with Univa Grid Engine. It generates statistics for the number of records per ARCo table.

XML Rule in MySQL:

```
<statistic interval="hour" variable="row_count" type="seriesFromColumns">
    <sql>
      SELECT sge_host, sge_queue, sge_user, sge_group, sge_project,
             sge_department,
      sge_host_values, sge_queue_values, sge_user_values, sge_group_values,
      sge_project_values, sge_department_values,
      sge_job, sge_job_log, sge_job_request, sge_job_usage, sge_statistic,
      sge_statistic_values,
      sge_share_log, sge_ar, sge_ar_attribute, sge_ar_usage, sge_ar_log,
```

```
            sge_ar_resource_usage
            FROM (SELECT count(*) AS sge_host FROM sge_host) AS c_host,
            (SELECT count(*) AS sge_queue FROM sge_queue) AS c_queue,
            (SELECT count(*) AS sge_user FROM sge_user) AS c_user,
            (SELECT count(*) AS sge_group FROM sge_group) AS c_group,
            (SELECT count(*) AS sge_project FROM sge_project) AS c_project,
            (SELECT count(*) AS sge_department FROM sge_department) AS c_department,
            (SELECT count(*) AS sge_host_values FROM sge_host_values) AS c_host_values,
            (SELECT count(*) AS sge_queue_values FROM sge_queue_values) AS c_queue_values,
            (SELECT count(*) AS sge_user_values FROM sge_user_values) AS c_user_values,
            (SELECT count(*) AS sge_group_values FROM sge_group_values) AS c_group_values,
            (SELECT count(*) AS sge_project_values FROM sge_project_values) AS c_project_values,
            (SELECT count(*) AS sge_department_values FROM sge_department_values) AS
                                c_department_values,
            (SELECT count(*) AS sge_job FROM sge_job) AS c_job,
            (SELECT count(*) AS sge_job_log FROM sge_job_log) AS c_job_log,
            (SELECT count(*) AS sge_job_request FROM sge_job_request) AS c_job_request,
            (SELECT count(*) AS sge_job_usage FROM sge_job_usage) AS c_job_usage,
            (SELECT count(*) AS sge_share_log FROM sge_share_log) AS c_share_log,
            (SELECT count(*) AS sge_statistic FROM sge_statistic) AS c_sge_statistic,
            (SELECT count(*) AS sge_statistic_values FROM sge_statistic_values) AS
                                c_sge_statistic_values,
            (SELECT count(*) AS sge_ar FROM sge_ar) AS c_sge_ar,
            (SELECT count(*) AS sge_ar_attribute FROM sge_ar_attribute) AS c_sge_ar_attribute,
            (SELECT count(*) AS sge_ar_usage FROM sge_ar_usage) AS c_sge_ar_usage,
            (SELECT count(*) AS sge_ar_log FROM sge_ar_log) AS c_sge_ar_log,
            (SELECT count(*) AS sge_ar_resource_usage FROM sge_ar) AS c_sge_ar_resource_usage
        </sql>
    </statistic>
```

Generated Data:

```
mysql> select * from view_statistic where variable = 'row_count' order by time_start limit 10;
+----------------------+---------------------+---------------------+-----------+-----------+
| name                 | time_start          | time_end            | variable  | num_value |
+----------------------+---------------------+---------------------+-----------+-----------+
| sge_queue            | 2011-04-26 17:35:59 | 2011-04-26 17:35:59 | row_count |         4 |
| sge_ar               | 2011-04-26 17:35:59 | 2011-04-26 17:35:59 | row_count |         0 |
| sge_group            | 2011-04-26 17:35:59 | 2011-04-26 17:35:59 | row_count |         2 |
| sge_ar_usage         | 2011-04-26 17:35:59 | 2011-04-26 17:35:59 | row_count |         0 |
| sge_department       | 2011-04-26 17:35:59 | 2011-04-26 17:35:59 | row_count |         1 |
| sge_ar_resource_usage | 2011-04-26 17:35:59 | 2011-04-26 17:35:59 | row_count |         0 |
| sge_user_values      | 2011-04-26 17:35:59 | 2011-04-26 17:35:59 | row_count |         2 |
| sge_project_values   | 2011-04-26 17:35:59 | 2011-04-26 17:35:59 | row_count |         2 |
| sge_job              | 2011-04-26 17:35:59 | 2011-04-26 17:35:59 | row_count |      5109 |
| sge_job_request      | 2011-04-26 17:35:59 | 2011-04-26 17:35:59 | row_count |         0 |
+----------------------+---------------------+---------------------+-----------+-----------+
10 rows in set (0.00 sec)
```

Querying Data (for one table, the ARCo `sge_job` table):

```
mysql> select * from view_statistic where variable = 'row_count' and name = 'sge_job' order by
time_start limit 10;
+---------+---------------------+---------------------+-----------+-----------+
| name    | time_start          | time_end            | variable  | num_value |
+---------+---------------------+---------------------+-----------+-----------+
| sge_job | 2011-04-26 15:35:58 | 2011-04-26 15:35:58 | row_count |         0 |
| sge_job | 2011-04-26 16:35:58 | 2011-04-26 16:35:58 | row_count |      2393 |
| sge_job | 2011-04-26 17:35:59 | 2011-04-26 17:35:59 | row_count |      5109 |
| sge_job | 2011-04-26 18:35:59 | 2011-04-26 18:35:59 | row_count |      7825 |
| sge_job | 2011-04-26 19:36:00 | 2011-04-26 19:36:00 | row_count |     10542 |
| sge_job | 2011-04-26 20:36:00 | 2011-04-26 20:36:00 | row_count |     13258 |
| sge_job | 2011-04-26 21:36:01 | 2011-04-26 21:36:01 | row_count |     15975 |
| sge_job | 2011-04-26 22:36:01 | 2011-04-26 22:36:01 | row_count |     18693 |
| sge_job | 2011-04-26 23:36:02 | 2011-04-26 23:36:02 | row_count |     21408 |
| sge_job | 2011-04-27 00:36:02 | 2011-04-27 00:36:02 | row_count |     24125 |
+---------+---------------------+---------------------+-----------+-----------+
10 rows in set (0.00 sec)
```

Counting the number of jobs finished

The following rule can be used to retrieve the number of jobs finished in the cluster per hour. The result is exactly one value, allowing the use of the *seriesFromColumns* type.

XML Rule in MySQL:

```
<statistic interval="hour" variable="finished" type="seriesFromColumns">
   <sql>
      SELECT count(*) AS jobs FROM sge_job_usage WHERE ju_end_time &lt; now() AND
       ju_end_time &gt;= subtime(now(), '1:0:0')
   </sql>
</statistic>
```

Generated Data:

```
mysql> select * from view_statistic where variable = 'finished' order by time_start;
+------+---------------------+---------------------+----------+-----------+
| name | time_start          | time_end            | variable | num_value |
+------+---------------------+---------------------+----------+-----------+
| jobs | 2011-04-27 11:15:43 | 2011-04-27 11:15:43 | finished |      2466 |
| jobs | 2011-04-27 11:33:35 | 2011-04-27 11:33:35 | finished |      2458 |
| jobs | 2011-04-27 11:34:31 | 2011-04-27 11:34:31 | finished |      2462 |
| jobs | 2011-04-27 11:35:56 | 2011-04-27 11:35:56 | finished |      2464 |
| jobs | 2011-04-27 11:37:40 | 2011-04-27 11:37:40 | finished |      2462 |
| jobs | 2011-04-27 11:47:19 | 2011-04-27 11:47:19 | finished |      2324 |
| jobs | 2011-04-27 12:47:19 | 2011-04-27 12:47:19 | finished |      1688 |
| jobs | 2011-04-27 13:47:20 | 2011-04-27 13:47:20 | finished |      1689 |
| jobs | 2011-04-27 14:47:20 | 2011-04-27 14:47:20 | finished |      1687 |
```

```
+------+--------------------+--------------------+----------+-----------+
9 rows in set (0.00 sec)
```

Counting the number of jobs finished per account

This query resembles the above query retrieving the number of jobs finished per hour, but this time the goal is to retrieve the number of jobs finished per hour and account. The finished jobs could have run under an arbitrary number of accounts, so use the *seriesFromRows* type to report one value per account string.

XML Rule in MySQL:

```
<statistic interval="hour" variable="finished_account" type="seriesFromRows"
                    nameColumn="account" valueColumn="jobs">
   <sql>
      SELECT account, count(*) AS jobs FROM view_accounting WHERE end_time
         &lt; now() AND end_time &gt;= subtime(now(), '1:0:0') GROUP BY account
   </sql>
</statistic>
```

Generated Data:

The jobs that ran for this example belonged to 3 different accounts, *sge* (default when an account string isn't specified), *test* and *production*.

```
mysql> select * from view_statistic where variable = 'finished_account' order by time_start;
+------------+---------------------+---------------------+------------------+-----------+
| name       | time_start          | time_end            | variable         | num_value |
+------------+---------------------+---------------------+------------------+-----------+
| sge        | 2011-04-27 11:37:40 | 2011-04-27 11:37:40 | finished_account |      1989 |
| sge        | 2011-04-27 11:47:19 | 2011-04-27 11:47:19 | finished_account |      1869 |
| production | 2011-04-27 11:47:19 | 2011-04-27 11:47:19 | finished_account |         1 |
| test       | 2011-04-27 11:47:19 | 2011-04-27 11:47:19 | finished_account |         3 |
| sge        | 2011-04-27 12:47:19 | 2011-04-27 12:47:19 | finished_account |      1401 |
| sge        | 2011-04-27 13:47:20 | 2011-04-27 13:47:20 | finished_account |      1393 |
| sge        | 2011-04-27 14:47:20 | 2011-04-27 14:47:20 | finished_account |      1401 |
+------------+---------------------+---------------------+------------------+-----------+
```

Deletion Rules

As a cluster's ARCo database runs over a long period of time, the database size can get very large. The rate at which it grows is highly dependent on the number of hosts and the number of jobs run per day.

Most of the data in an ARCo database is very detailed raw data, such as the following:

- the `np_load_avg` per host reported every 10 seconds
- detailed accounting information for every job and for every task of a tightly integrated parallel job

- job log listing every state transition a job went through
- every single change in the usage of consumables (slots, licenses etc.)

Although this detailed raw data is very valuable for debugging and close analysis of cluster behavior, it is usually not desirable or even possible to keep all that data due to limitations on the database storage.

For long term archival and analysis, compressed data is easier to manage and consumes less space. The following are sample strategies for data compression:

- Instead of keeping every job accounting record, store daily or monthly accounting information per user or project.
- For analyzing usage patterns, hourly averages / minimum / maximum host load values, such as `np_load_avg`, will usually be sufficient while consuming much less space and being faster to query than keeping the raw `np_load_avg` records (one per host every 10 seconds).

Deletion rules remove data that is no longer required. One rule is represented by one node in the `dbwriter.xml` file. A `<delete>` node has the following attributes:

- **scope**: Defines on which table delete operations are performed. Valid values for scope are:
  - *host*values_: Delete from the sge_host_values table.
  - *queue*values_: Delete from the sge_queue_values table.
  - *user*values_: Delete from the sge_user_values table.
  - *group*values_: Delete from the sge_group_values table.
  - *project*values_: Delete from the sge_project_values table.
  - *department*values_: Delete from the sge_department_values table.
  - *job*: Delete from the sge_job table and all tables referencing jobs (sge_job_request, sge_job_usage, sge_job_log, sge_job_online_usage).
  - *job*log_: Delete from the sge_job_log table. When a job is deleted from sge_job, corresponding records in the sge_job_log table are also deleted.
  - *job*online_usage_: Delete from the sge_job_online_usage table. When a job is deleted from sge_job, corresponding records in the sge_job_online_usage table are also deleted.
  - *share*log_: Delete from the sge_share_log table.
  - *statistic*values_: Delete from the sge_statistic_values table.
  - *ar*values_: Delete advance reservation information from the sge_ar, sge_ar_attribute, sge_ar_log, sge_ar_resource_usage and sge_ar_usage table.
- **time_range**: The unit used for specifying time information:
  - *hour*

  - *day*
  - *month*
  - *year*

- **time_amount**: The number of hours/days/month/years to keep data.

A `<delete>` node can have sub nodes `<sub_scope>`, restricting a deletion rule to specific data, such as deleting only certain variables (the raw data) from a `sge_*_values` table, but keeping the derived data (like averages, sums etc.).

Examples

Host Related Data

This rule keeps host related raw data like `np_load_avg` only 7 days, but keeps the derived values for 2 years:

```
<delete scope="host_values" time_range="day" time_amount="7">
  <sub_scope>np_load_avg</sub_scope>
  <sub_scope>cpu</sub_scope>
  <sub_scope>mem_free</sub_scope>
  <sub_scope>virtual_free</sub_scope>
</delete>

<delete scope="host_values" time_range="year" time_amount="2"/>
```

The first rule deletes records from the `sge_host_values` table older than 7 days, but restricts the rule to the variables *np*load_avg_, *cpu*, *mem*free_ and *virtual*free_.

The second rule makes sure that **all** records in `sge_host_values` older than 2 years are deleted.

Job Related Data

This rule keeps job related data, including general job information like submission time, user, project etc. and detailed information like job requests and job accounting, for one year, while only keeping the job log for one month.

Make sure to actually use a shorter time range for detailed job information like job log than for the general job rule. The general job rule will delete **all** job related information, including the job log.

3.7 Troubleshooting the `dbwriter`

General Problems

Where do I find the `dbwriter` log file?

The `dbwriter` log file is `$SGE_ROOT/$SGE_CELL/spool/dbwriter/dbwriter.log`

How can I set the debug level?

The amount of information written to the `dbwriter` log file defined in the `dbwriter` configuration file.

The default debug level is INFO.

The INFO debug level generates a significant amount of data, so it can make sense to reduce the debug level to WARNING.

In case of problems running `dbwriter`, increasing the debug level to INFO again, or to even higher levels CONFIG, FINE, FINEST can make sense.

To change the debug level:

- Shut down dbwriter.
- Edit `$SGE_ROOT/$SGE_CELL/common/dbwriter.conf` and modify the setting for DBWRITER_DEBUG if needed.
- Start up `dbwriter` again.

4 Upgrading Univa Grid Engine

Warning

- Be sure to source the correct settings file before executing Univa Grid Engine commands. - Backup the existing configuration before starting any upgrade process.

Besides installing a new cluster (I) and configuring it manually, there are two ways to setup a new Univa Grid Engine cluster with using the configuration of an old installation of Grid Engine:

- Cloning (C) a Grid Engine configuration provides a way to transfer configuration objects from an old installation to a new Univa Grid Engine installation in a new location, using different port numbers and a different cluster name.
- In place upgrading (U) keeps the location and all settings while upgrading the binaries and the configuration to Univa Grid Engine 8.6.x, replacing the old cluster. For upgrading to UGE 8.6.x, the cluster must be drained of all jobs!

Which options are available depends on which version of Grid Engine is currently installed. To find the currently installed version of Grid Engine, execute a command-line client; the first line of the output provides the version information.

```
# qstat -help
UGE 8.x.x
```

From Version	Available upgrade methods
8.0 FCS - 8.6.x	C, U

From Version	Available upgrade methods
6.2u5 - 6.2u7	C, U
5.3 - 6.2u4	I

The following table describes the difference between Clone upgrading and In place upgrading a cluster:

Clone upgrade	In place upgrade
Creates a new cluster reusing configuration data from an existing installation.	Upgrades an existing cluster.
Makes it possible to test the new cluster before it is made active. Old cluster remains available meanwhile.	The cluster is not available during the upgrade.
Existing load values will not be transferred to the cloned cluster. Static values will be replicated as soon as they are reported from corresponding execution daemons.	No changes to dynamic or static load values will be applied.
Sharetree usage will be lost.	Sharetree usage will still be available.

Both options use the same concept to upgrade a cluster: First, in the running old cluster, the configuration is saved. Then the new binaries are extracted and the `inst_sge` script is used to setup the new configuration using the saved configuration and by asking some questions to the user. The details are described in the following sections.

Note
It is not possible to upgrade Windows SFU/SUA/Interix execution hosts to the new native Windows execution hosts. Depending on the kind of upgrade, different steps are describe in the respective section.

4.1 Clone Upgrade

The upgrade steps provided below describe how to set up a new cluster using the configuration information of an existing cluster. Steps marked with the tag `REAL-UPGRADE` are optional and should only be applied if the existing cluster will be disabled during the clone process. If they are skipped, the first cluster will not be disabled and remains fully functional. Instead, an additional second cluster will be set up using a copy of the configuration on the same resource pool as the first cluster. This type of installation can be helpful to test the upgrade before a real update is done. It also allows old jobs to finish on the old cluster while new jobs are already started on the new cluster.

The `OPTIONAL` tag is used for all update steps that can be performed only if the corresponding functionality (e.g. IJS, ARCo, ...) was setup in the existing cluster, and/or if that functionality would also be available in the cloned installation.

Step-by-Step Instructions:

1. Prepare the configuration.

- Download the necessary files.
  - Binary packages and the common package are required.
  - If using ARCo or if intending to use ARCo after the upgrade, download the ARCo package.
- The following list of environment variables and configuration settings will conflict with the existing cluster configuration. Decide on new values before beginning the installation process.
  - `$SGE_ROOT`: new installation location
  - `$SGE_CELL`: cell name. Can be the same name as in the existing cluster.
  - `$SGE_CLUSTER_NAME`: new cluster name.
  - `$SGE_QMASTER_PORT`: new qmaster port
  - `$SGE_EXECD_PORT`: new port used for execs
  - `qmaster_spool_dir`: new spooling location for qmaster
  - `execd_spool_dir`: new spooling location for execd
  - `gid_range`: new gid range. Can be the same as the gid range of the existing cluster, if that cluster is drained during the upgrade process.

2. Save the configuration of the old cluster.

- Check the version of the existing Grid Engine installation. The version information is the first line of the help output from the command line utilities.

```
# qstat -help
UGE 8.x.x
```

- Grid Engine installations version 6.2 and above contain the backup script `util/upgrade_modules/save_sge_config.sh`. For existing clusters older than version 6.2, please contact our support.
- Make sure the qmaster of the old cluster is running, then run the save_sge_config script on the qmaster host of the old cluster. The first argument to the script must be an absolute path to a file system location where backup information will be stored.

```
# <path_to_backup_script>/save_sge_config.sh <ge_backup_location>
```

Note

The save_sge_config script saves all configuration objects as well as following files:

- accounting
- act_qmaster
- arseqnum
- bootstrap
- cluster_name
- dbwriter.conf
- host_aliases
- jobseqnum
- jmx
- qtask
- sge_aliases
- sge_ar_request
- sge_request
- sge_qstat
- sge_qquota
- sge_qstat
- shadow_masters

(Beginning with version 8.5.x the save_sge_config.sh script takes also care of any existing sgeCA related files on the master host, so when upgrading from older versions the > 8.5.x save_sge_config.sh file shall be used to create the backup. Otherwise CSP related files cannot be handled correctly during update and must be integrated manually.)

3. `REAL-UPGRADE`: Drain the cluster. (see Common Tasks -> Draining the Cluster and Stopping it Successively)
4. `REAL-UPGRADE`: Shut down the execution daemons, qmaster and shadow daemons:

- Shut down all execution daemons:

```
# qconf -ke all
```

- Shut down the qmaster:

```
# qconf -km
```

- Shut down the shadow daemons:

```
# $SGE_ROOT/$SGE_CELL/common/sgemaster -shadowd stop
```

6. `REAL-UPGRADE` and `OPTIONAL` : Prepare ARCo for the upgrade.

- Only necessary if the existing cluster used ARCo.
- Ensure that the reporting file has been completely processed by dbwriter. Wait until the reporting file does not exist anymore.
- Stop dbwriter

```
# $SGE_ROOT/$SGE_CELL/common/sgedbwriter stop
```

- Backup existing ARCo database

7. Extract packages to the new `$SGE_ROOT` directory.

- Extract the binary packages.
- Extract the common package.
- `OPTIONAL` : Extract the ARCo package only if ARCo will be available in the new cluster.

8. Upgrade the qmaster installation.

Note
Cloning a configuration might change the copied configuration objects, possibly influencing the operations in the cloned cluster. New configuration attributes could be added or removed to align the cloned objects with the new installation's object configuration. Read the Release Notes to find out which configuration objects might be affected, and verify the installation after the upgrade finishes.

- Change to the new `$SGE_ROOT` directory.
- The upgrade process must be started on the host where the original cluster's qmaster process was running. Use additional flags to enable or disable certain features of Univa Grid Engine (like CSP, old IJS, . . .).

```
# ./inst_sge -upd
```

Warning
When cloning a configuration make sure that the environment of the shell in which `inst_sge -upd` is called does **NOT** have the environment setup for the original cluster!

- Read and accept the displayed license.
- Provide the absolute path to the backup directory.
- Verify if the backup (Grid Engine version and date/time) is the correct one, and accept with `y`.

- Specify the new `$SGE_ROOT` directory.
- Accept or change the `$SGE_CELL` directory.
- Enter the new `$SGE_QMASTER_PORT` number.
- Enter the new `$SGE_EXECD_PORT` number.
- Accept or change the admin user.
- Specify the new qmaster spooling directory.
- Accept or select the new `$SGE_CLUSTER_NAME`.
- Specify the number of reader threads. The value 0 will disable the read-only component.
- Choose if the file permissions have to be checked.
- Select the spooling method.
  - The spooling method for the new cluster does not need to match the existing cluster's spooling method.
- Specify if the interactive job configuration.
  - Either use the job configuration contained in the backup in the new cluster, or use the default for the Univa Grid Engine version.
- Specify a group id range.
  - If the existing cluster still contains active jobs, or if the existing cluster will be used in parallel to the new one, then the specified `gid` range is not allowed to be the same or to overlap in any way.
- Specify the new spooling directory to be used on execution hosts.
- Specify none or the administrators mail address to receive problem reports.
- Select the next job number to be used in the new cluster.
- Select the next advance reservation number.
- Select system boot time startup options.
- Now the old configuration is copied to a temporary directory and upgraded there. This upgraded configuration is used to configure the new cluster. In case of any errors, this command can be executed manually to repeat the last step after fixing any problems. More detailed error messages are located in `/tmp/sge_backup_load_<date>-<time>.log`. Now qmaster is running with the same setup as the original cluster. Verify the configuration or adjust certain parameters before execution hosts are started.

9. `OPTIONAL` : Upgrade ARCo.

- OPTIONAL : Migrate PostgreSQL Database to a different Schema.
- Upgrade ARCo.

10. OPTIONAL : Copy the binaries and the `$SGE_ROOT/$SGE_CELL/common` directory to all execution hosts in the cluster if they do not use a shared filesystem.
11. Upgrade the execution environment.

- Upgrading the execution environment will properly initialize local `execd` spooling directories.

Note
It's not possible to upgrade an existing SFU/SUA/Interix Windows execution host to a native Windows execution host. Instead, please run the full execution host installation on each Windows host.

- Set up the shell environment for the new cluster.

```
# . $SGE_ROOT/$SGE_CELL/common/settings.sh
```

- Initialize the spooling directory.

```
# $SGE_ROOT/inst_sge -upd-execd
```

- Update the startup/shutdown scripts.

```
# $SGE_ROOT/inst_sge -upd-rc
```

12. Start the execution daemons.

- To shutdown certain hosts in the initial cluster and restart them in the cloned cluster, then see Common Tasks -> Starting Up and Activating Nodes Selectively.
- If passwordless ssh to all hosts is setup, the following command can be used to activate all execution nodes in the new cluster:

```
# ./inst_sge -start-all
```

Note
Native Windows execution nodes are not started by this command. They have to be started on each Windows host separately.

13. Upgrade advance/standing reservations

- Optional step if you want to keep advance or standing reservations from the initial cluster.
- Requires that the upgrade host is submit host and that users can be resolved on this host.

```
# ./inst_sge -upd-ars
```

14. Post-installation steps.

- Make sure there are no user generated scripts in the old Grid Engine directory that would get lost if this directory was deleted. It makes sense to not move them to the new `$SGE_ROOT`, but to a location that is independent of the Grid Engine installation. If you move scripts or methods, make sure the Grid Engine configuration is changed accordingly. Such scripts and methods are: prolog, epilog, pe_start, pe_stop, starter_method, suspend_method, resume_method, terminate_method, checkpointing scripts, JSV scripts and so on.

4.2 Inplace Upgrade

The upgrade steps below describe how to replace the existing set of binaries and scripts of an existing Univa Grid Engine installation. This type of upgrade is recommended for patch releases, but it might also be used for major upgrades when the $SGE_ROOT directory and the used ports cannot be changed or the sharetree usage must be kept. Consult the Release Notes of the target Grid Engine Installation as well as the Upgrade Matrix to find out if the Inplace Upgrade is applicable to the existing cluster.

1. Prepare the configuration.

- Download the necessary binary packages and the common package.
- If using ARCo or if intending to use ARCo after the upgrade, download the ARCo package, too.

2. `OPTIONAL` : Backup the existing cluster (in case something goes wrong)

- This can be achieved with the `inst_sge` script part of the existing installation.

```
# cd $SGE_ROOT
# ./inst_sge -bup
```

Note

If the upgrade fails, try restoring the existing cluster by unpacking the original packages and restoring the old configuration.

```
# cd $SGE_ROOT
# ./inst_sge -rst
```

3. Disable the cluster.

   - Make sure that no new jobs can be submitted into the cluster by adding a JSV that rejects all jobs.

     ```
     # qconf -mconf
     ...
     jsv_url <sge_root_path>/util/resources/jsv/jsv_reject_all.sh
     ```

4. Wait until all jobs have finished

Review the Univa Grid Engine release notes distribution for additional information.

5. Save the configuration of the old cluster.

   - Check the version of the existing Grid Engine installation. The version information is the first line of the help output from the command line utilities.

     ```
     # qstat -help
     UGE 8.x.x
     ```

   - Grid Engine installations version 6.2 and above contain the backup script `util/upgrade_modules/save_sge_config.sh`. For existing clusters older than version 6.2, please contact our support.
   - Make sure the qmaster of the old cluster is running, then run the save_sge_config script on the qmaster host of the old cluster. The first argument to the script must be an absolute path to a file system location where backup information will be stored.

     ```
     # <path_to_backup_script>/save_sge_config.sh <ge_backup_location>
     ```

Note
The save_sge_config script saves all configuration objects as well as following files:

- accounting
- act_qmaster
- arseqnum
- bootstrap
- cluster_name
- dbwriter.conf
- host_aliases
- jobseqnum
- jmx
- qtask
- sge_aliases
- sge_ar_request
- sge_request
- sge_qstat
- sge_qquota
- sge_qstat
- shadow_masters

Note
It's not possible to upgrade SFU/SUA/Interix Windows execution, submit and administration hosts to the new native Windows execution, submit and administration host functionality. Please uninstall the old Windows execution hosts before proceeding with the upgrade.

6. Shut down the execution daemons, qmaster and shadow daemons.

- Shut down all execution daemons:

```
# qconf -ke all
```

- Shut down the qmaster:

```
# qconf -km
```

- Shut down the shadow daemons:

```
# $SGE_ROOT/$SGE_CELL/common/sgemaster -shadowd stop
```

7. Prepare ARCo to be updated. (only necessary if the existing cluster used ARCo)

- Shut down ARCo.

```
# $SGE_ROOT/$SGE_CELL/common/sgedbwriter stop
```

- Backup the ARCo database.

8. Delete old directories of old Grid Engine versions

- When you are upgrading a completely empty cluster from an SGE version to UGE then please delete the old architecture dependent directories (lx24*). Unpacking the new packages do not overwrite them, because the architecture string from Linux does not contain the 24 kernel string anymore. Removing them reduces the risk that open terminals are accessing the new qmaster with old binaries, which could cause problems.

9. Extract new packages to `$SGE_ROOT`.

- Extract binary packages.
- Extract common package.
- (Optional) If enabling ARCo in new cluster, extract the ARCo package.

10. Upgrade the qmaster installation.

- Change to the new `$SGE_ROOT` directory.
- The upgrade process must be started on the host where the original cluster's qmaster process was running. Use additional flags to enable or disable certain features of Univa Grid Engine (like CSP, old IJS, . . .).

```
# ./inst_sge -upd
```

- Read and accept the displayed license.
- Provide the absolute path to the backup directory.
- Verify if the backup (Grid Engine version and date/time) is the correct one, and accept with `y`.
- Accept the `$SGE_ROOT` directory.
- Accept the `$SGE_CELL` directory.
- Accept the `$SGE_QMASTER_PORT` number.
- Accept the `$SGE_EXECD_PORT` number.
- Accept or change the admin user.
- Accept the qmaster spooling directory.
- Accept or select the new `$SGE_CLUSTER_NAME`.

- Specify the number of reader threads. The value 0 will disable the read-only component.
- Choose if the file permissions have to be checked.
- Select the spooling method.
  - The spooling method for the new cluster does not need to match the existing cluster's spooling method.
- Specify the interactive job configuration.
  - Either use the job configuration contained in the backup in the new cluster, or use the default for the Univa Grid Engine version.
- Specify a group id range.
- Specify the new spooling directory to be used on execution hosts.
- Specify none or the administrators mail address to receive problem reports.
- Select the next job number to be used in the upgraded cluster.
- Select the next advance reservation number.
- Select system boot time startup options.
- Now the old configuration is copied to a temporary directory and upgraded there. This upgraded configuration is used to configure the new cluster. In case of any errors, this command can be executed manually to repeat the last step after fixing any problems. More detailed error messages are located in `/tmp/sge_backup_load_<date>-<time>.log`. Now qmaster is running with the same setup as the original cluster. Verify the configuration or adjust certain parameters before execution hosts are started.

9. `OPTIONAL` : Upgrade ARCo.

- `OPTIONAL` : Migrate PostgreSQL Database to a different Schema.
- Upgrade ARCo.

10. `OPTIONAL` : Copy the binaries and the `$SGE_ROOT/$SGE_CELL/common` directory to all execution hosts in the cluster if they do not use a shared filesystem.
11. Upgrade the execution environment.

- Upgrading the execution environment will properly initialize local `execd` spooling directories.

Note

It's not possible to upgrade an existing SFU/SUA/Interix Windows execution host to a native Windows execution host. Instead, please run the full execution host installation on each Windows host.

- Set up the shell environment for the new cluster.

```
# . $SGE_ROOT/$SGE_CELL/common/settings.sh
```

- Initialize the spooling directory.

```
# $SGE_ROOT/inst_sge -upd-execd
```

- Update the startup/shutdown scripts.

```
# $SGE_ROOT/inst_sge -upd-rc
```

13. Upgrade advance/standing reservations

- Optional step if you want to keep advance or standing reservations from the initial cluster.
- Requires that the upgrade host is submit host and that users can be resolved on this host.

```
# ./inst_sge -upd-ars
```

14. Post-installation steps.

- Enable submission of new jobs by reverting the jsv_url changes from step 3.

```
# qconf -mconf
...
jsv_url ...
```

5 Troubleshooting the Installation

5.1 Prerequisite Steps

Incorrect accounting records and abnormal termination of jobs when NFS shares are shared between execution hosts.

The set of gid ranges on an execution host exporting file-systems via NFS to other execution hosts has to be disjoint to the sets of gid ranges of these hosts. Reason for this is that NFS-server-components will have the same set of group IDs when NFS clients access a network share. As a result of that it might happen that a job on a NFS client has the same group ID as a job running on the NFS server. As the NFS-server-components take over the gid of the client the job on the NFS server is charged with the consumption of resources of the NFS server processes. This can lead to an abnormal termination of processes and also to an incorrect accounting record for this job. Distinct group ID ranges for execution hosts acting either as NFS server or client will avoid this problem.

The same accounting problem as mentioned before for the NFS case may appear if a job gets the same gid as another user running applications on the same host. In this situation the job usage and the usage of the user processes are accumulated and a wrong job accounting is the result. This problem also exists vice versa, if a job is terminated having a gid, which is in use by another user, UGE is using the gid to track all processes belonging to the job. In this case it might happen that foreign processes are killed, when the job finished or when it is deleted.

5.2 Automatic Installation

`qmon` fails due to missing Motif libraries.

Some systems do not automatically install the Motif library `libXm.so.?` by default. This missing library causes `qmon` to abort.

To solve this issue, find the correct software package that contains the Motif or OpenMotif library, and install it. It might also be necessary to adjust the `LD_LIBRARY_PATH` or variable for the corresponding OS architecture. To test if `qmon` found all required libraries, use the `ldd` command.

```
# ldd <path_to_qmon>
...
libXm.so.4 =>    <path_to_the_lib>/libXm.so.?
...
```

Automatic installation terminates to avoid overwriting files.

The automatic installation terminates when the `$SGE_ROOT/$SGE_CELL` already exists. This is intended behavior to avoid having the automatic installation overwrite files of a previous installation.

To solve this issue, check if there was already an installation with the corresponding cell name. Then, choose a different name and restart the automatic installation or rename/remove the directory.

Although the automatic installation of an execution host seams to succeed, the daemon was not started.

Check if user *root* has password-less ssh/rsh access to the remote host. If there is in general no password-less root access, then log in to that host manually, and start the automatic installation on that host with the command:

```
# ./inst_sge -noremote -x -auto <cfg_file>
```

6 Installing Univa Grid Engine on Microsoft Windows

6.1 Planning the installation

6.1.1 Supported roles of Microsoft Windows hosts

Windows hosts can only have the Execution, Submit and Admin host role. Using a Windows host as Master host or Shadow Master host is not supported. Further, Microsoft Windows Domain controllers are not suited for Execution hosts.

6.1.2 Preparing the Microsoft Windows hosts

A Windows host that is to be installed as Univa Grid Engine Submit, Admin or Execution host must be member of a Windows Domain. The Domain controller is needed to resolve the user name and to automatically expand the short user name to the full qualified user name in the format "<domain>\<username>". While something similar is possible with local users, it works internally in a different way, which is not supported by Univa Grid Engine.

Before using a host as Windows Execution host, the "Microsoft Visual C++ 2010 Redistributable Package" must be installed. Specifically the MSVCR100.DLL must be located in the system directory, which usually is `c:\Windows\system32`. The "Microsoft Visual C++ 2010 Redistributable Package" can be downloaded from `http://www.microsoft.com/en-us/download/details.aspx?id=5555`. Please install the 32 bit version, even if you run a 64 bit Windows!

Also, before installing an Univa Grid Engine Execution host on the Windows host, the directory `C:\tmp` must be created and must be made writable for everyone. The easiest way to do this is to add the built-in group "Everyone" to the Access Control List of the `C:\tmp` directory and give it all permissions except the "Special Permissions".

Also consider to configure these options:

Disable sleep mode of Windows
The performance settings of Windows might prevent the Windows execution host from working properly. On Windows 7 and later, the "Power Plan" could set the host to a sleep mode from which Univa Grid Engine cannot awaken it againt. To change this plan, in the Control Panel select "Power Options" and select the "High Performance" power plan there to prevent Windows from setting the host to a sleep mode when it is idle for a certain time. The "High Performance" power plan might be hidden in the "Show additional plans" pull down menu.

Reduce time until a new file appears on a SMB share
On Windows Vista and later, if a new file takes long to appear on a SMB shares, add these registry key to drastically reduce the time until the file appears:

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Lanmanworkstation\Parameters\
DirectoryCacheLifetime[DWORD] = 0
FileInfoCacheLifetime[DWORD] = 0
FileNotFoundCacheLifetime[DWORD] = 0
```

6.1.3 Selecting a file system for spooling operations of Windows hosts

Spooling is done only by Execution hosts, but not by Submit and Administrative hosts. On Windows, it is mandatory to spool on a local file system, spooling on network based file systems is not possible.

6.1.4 Selecting the security mode

It is not possible to operate Windows Execution, Submit or Admin hosts in a cluster that uses the CSP (Certificate Security Protocol) mode of Univa Grid Engine.

6.1.5 The Univa Grid Engine admin user

Windows Execution hosts can only be installed in a cluster that defines an “admin user”. It is not possible to run Windows Execution hosts in a “root user system” or “test user system”. The “admin user” is defined as one of the first steps during the Master Host installation. If you are not sure if your cluster defines and “admin user”, check the “admin_user” value in the `$SGE_ROOT/$SGE_CELL/common/bootstrap` file to see how the cluster was installed.

6.1.6 Understanding the roles of users on the Microsoft Windows hosts

These users are needed to install Univa Grid Engine on Windows hosts:

- an Administrative user that has permissions to change the Operating System - usually the local Administrator, a member of the local Administrators group or a member of the Domain Administrators group, depending on how the Windows host and the Windows Domain are set up. This user will be called “Administrator” from here on. From Windows Vista on, an Administrator can be in restricted or in elevated mode. To install Univa Grid Engine, the permission to elevate the Administrator is needed.
- the Univa Grid Engine “admin user”. This user must not be a member of any Administrators group or be the local or Domain Administrator, but it must be a Windows Domain user. The Windows Execution Host must be a member of the Windows Domain, and the Windows Domain must be the default Domain of the host, i.e. the user name without any Domain prefix must resolve to the full Windows Domain user name. The short name of this user must be exactly the same as specified in the “admin_user” value in the “bootstrap” file.

Further, these users exist in Univa Grid Engine:

- the Univa Grid Engine Managers and Operators. These are users that have special permissions to Univa Grid Engine, but must be normal Windows Domain users for Windows. The short names of the Managers and Operators must be exactly the same as on the Unix cluster hosts. The Univa Grid Engine “admin user” can also be Manager or Operator, but no “Administrator” can be Manager or Operator.

- the Univa Grid Engine job users. These users must be normal Windows Domain users and have permission to submit jobs and check the state of the cluster and jobs. The Univa Grid Engine "Admin User" can also be a job user. The short names of the job users must be exactly the same as on the Unix cluster hosts. The Univa Grid Engine "admin user" can also be a job user, but no "Administrator" can be job user.

Using any local Account for any role except the Administrative user is not supported! For all roles except the Administrative user, a user that is defined in the Windows Active Domain must be used! Using any service account or virtual account or any other special account Windows provides for specific purposes is not supported.

This example shall make clearer how the different users have to be defined and used:
The Windows Execution host is called "winhost1". This host is member of the Windows Domain "ITDOM", which is also the default domain of the "winhost1". On this Windows host, the local Administrator is disabled for security reasons, but there is a user called "HostAdmin" which is member of the local Administrators group. In the Windows Domain "ITDOM", the users "uge_admin", "uge_manager" and "uge_user" are defined, i.e. the real name of e.g. "uge_admin" is "ITDOM\uge_admin", but on the "winhost1", "uge_admin" automatically resolves to "ITDOM\uge_admin", because "ITDOM" is the default domain of "winhost1".

The Master host was installed by user root, but during installation, the Admin user "uge_admin" was defined. After Master host installation, in the `$SGE_ROOT/$SGE_CELL/common/bootstrap` the entry "admin_user uge_admin" can be found.

To install a Submit, Admin or Exec host on the Windows host, log in as the "ITDOM\uge_admin" user, open a cmd.exe window a follow the installation steps described below. During installation, you will be prompted to enter the login credentials of "winhost1\HostAdmin" for installation steps that need to modify Windows.

To configure Univa Grid Engine from an installed Windows Admin host, login as "ITDOM\uge_manager" and use qconf etc. to change the configuration.

To submit jobs from an installed Windows Submit host, login as "ITDOM\uge_user" and use qsub to submit a job.

6.1.7 Understanding the need for certificates

While adding Windows submit, admin and execution hosts to a Univa Grid Engine cluster that is running in CSP mode is not supported, it is still necessary to create certificates in order to Windows execution hosts to the cluster.

If Univa Grid Engine executes a job on a Unix/Linux Execution host, the execution daemon, which runs internally as root but masks itself as the Univa Grid Engine admin user, executes the job as the job user. The job has then all permissions the job user has.

On Windows, for the elevated local Administrator, it is possible to run a process as a local user without really logging on as this user, but it there is no way to run process as any other user with full permissions to the network without logging on as this user. This is why a Windows execution host needs the login credentials (i.e. user name and password) of the job user.

In order to store the login credentials safely, the certificates are needed. If the job users registers the password using the "sgepasswd" command, the certificates are used to encrypt the password. The execution daemon on the Windows execution host has the private certificates to decrypt the password, which allows it to log on as the job user and execute the job.

So the certificates are not needed to install a Windows submit or admin host, but for a Windows execution host. Also, registering the user passwords is needed only if a Windows execution host is installed.

6.1.8 Understanding the difference between mounted shares and UNC paths

Windows provides two ways to access files on a network share. The one is mounting a network share to a drive letter and then using it like a local drive. The path format with drive letter and path is usually called DOS format, e.g.: `x:\home\jdoe`. The other way to access files on a network share is specifying the share and the path using the UNC format: `\\server\share\home\jdoe`. While it is not possible to change to a UNC directory in a console (aka. cmd.exe window), all files can be accessed and executed using the UNC path. With the mounted shares, it is also possible to change to a directory on this share.

However, mounting shares is only per user and per session (with some exceptions), so it cannot be assumed that a share exists for other users and also not that it exists in another session of the same user. If Univa Grid Engine starts a job on the Windows Execution Host, it starts this job always in a different session than the interactive one. So even if it is possible to access a file interactively using a mounted share, that does not mean it is also possible for the job in another session. Even if mounts are persistent or are made in a login script, this does not always mean they are available at host boot time or they are usable in the job session. This also means that it is not possible to use the Univa Grid Engine Starter Service to start an execution daemon at host boot time from a network share if the $SGE_ROOT directory is specified in DOS format!

Because of this unreliable behaviour of mounts, it is recommended to always use the UNC format for paths on network shares - for all steps of the Execution, Submit or Administrative host installation, for all configuration and also as arguments to the jobs.

6.2 Prerequisite steps

6.2.1 Network configuration

TCP Port Setup

On the Windows Execution host, the same TCP port has to be used as on the Unix Execution hosts. If this port is not defined via the environment variable SGE_EXECD_PORT, then the value from the "sge_execd" entry in the "services" file is used. This and the "sge_qmaster" entry are defined by IANA to:

```
sge_qmaster 6444/tcp
sge_execd 6445/tcp
```

On Windows hosts, the "services" file is located in `%WINDIR%\system32\drivers\etc\services`. On Windows,it does not contain the sge_qmaster and sge_execd ports, i.e. these have to be added manually if they are going to be used.

Make sure there is no Firewall configured to block the access to the sge_qmaster port from the Windows host and none to block the access from the Master host to the sge_execd port on the Windows host.

6.2.2 Accessing the `$SGE_ROOT` directory

The `\$SGE_ROOT` directory must be accessible on all Windows hosts that are going to be part of the Univa Grid Engine cluster. This directory can be shared using the various methods (by setting up SMB/Samba/CIFS shares on the file server host, by installing a file system specific client on the Windows, etc.), but in the end all binaries must be accessible and the Univa Grid Engine Admin User must have read and write permissions on this share.

Of course, the Univa Grid Engine binary package for Windows (`ge-*-bin-win-x86*.tar.gz`) must have been extracted to the `\$SGE_ROOT` directory before starting installation.

6.2.3 Enabling existing clusters to add Windows hosts

To add a execution, submit or administrative Windows host to a Univa Grid Engine cluster, the cluster must be prepared for this. This is usually done by selecting to install Windows hosts during the master host installation. If this was not selected during the master host installation, an existing cluster can later be enabled to add Windows hosts to it.

If during the master host installation the option to install Windows execution hosts was not selected, the certificates are missing and must be built manually. To do this, log in to an Unix administrative host as `root` and source the `settings.sh` or `settings.csh` file of the Univa Grid Engine cluster. Run `$SGE_ROOT/util/sgeCA/sge_ca -init` and follow the instructions. This creates private certificates in the `/var/sgeCA/sge_qmaster` resp. `/var/sgeCA/port????` directory, as well as public certificates in the `$SGE_ROOT/$SGE_CELL/common/sgeCA` directory. Make sure the private certificates are available on the Windows execution hosts, the permissions must be as restrictive there as in the `/var/sgeCA/...` directory.

6.3 Installing execution hosts on Windows with the command line installation script

There are two options to install an Execution Host on Windows:

- Interactive installation with installation script
- Automatic installation with installation script

To install an execution host on Windows, in addition to all the prerequisites listed above, the `sge_qmaster` daemon must be running and be accessible.

Prerequisites:

- Make sure the Windows host is registered as Administrative host at the qmaster. Run the following command on any existing Administrative host:

```
# qconf -ah <hostname>
```

- Log in 'at the Windows host', i.e. either directly at the machine or via rdesktop. Log in as the Univa Grid Engine "admin user", see Understanding the roles of users on the Microsoft Windows hosts.
- Open a console (aka. "cmd.exe" window).

6.3.1 Understanding the different components of Univa Grid Engine on the Windows execution host

On a Windows execution host, there are more components than on a Unix/Linux execution host. The components are:

- `sge_execd.exe` - like on Unix, it communicates with the `sge_qmaster`, triggers the start of jobs, waits for the end of jobs and reports load and usage to the `sge_qmaster`. On Windows, the `sge_execd.exe` is started under the Univa Grid Engine admin user account.
- `sge_shepherd.exe` - like on Unix, it is started per job (per task of a job), runs as long as the job. On Windows, the `sge_shepherd.exe` does not start the job itself, but it forwards the job (and task) to start to the Univa Grid Engine Job Starter Service.
- `qloadsensor.exe` - retrieves system load and memory values. On Unix, this component is builtin to the `sge_execd`, but on Windows it has to be a separate process.
- `uge_js_service.exe` - the Univa Grid Engine Job Starter Service which starts the jobs under the job users accounts.
- `SGE_Starter.exe` - is needed on Windows Vista and later to start the job in a different session than the Univa Grid Engine Job Starter Service, which is in turn necessary to start all jobs in different sessions.
- `UGE_Starter_Service.exe` - the Univa Grid Engine Starter Service which starts the `sge_execd.exe` at system boot time, i.e. it is the equivalent to the RC scripts on Unix.

6.3.2 Interactive installation of Windows execution hosts

1. Start the execution host installation

- The installation script is called `install_execd.bat`.
- Start it using the UNC path if `\$SGE_ROOT` is located on a network share (e.g. `\\server\share\sge_root\install_execd.bat`. Use the DOS path only if `\$SGE_ROOT` was copied to a local drive (e.g.: `c:\sge_root\install_execd.bat`).
- Start this script.

```
Welcome to the Grid Engine execution host installation
------------------------------------------------------

...

Hit <RETURN> to continue
```

- Press *return* to continue.

2. Choose the installation location.

```
Checking $SGE_ROOT directory
----------------------------

The Grid Engine root directory is not set
Please enter a correct path for SGE_ROOT in DOS or UNC notation
or hit <RETURN> to use the default.
If the directory is network mounted, using the UNC notation is preferred.

Please note: If a mounted share is used (e.g. "X:\sge_root"), the execution
daemon cannot be started automatically at host boot time! Please use an UNC
path (e.g. "\\server\share\sge_root") instead!

[<installation_path>] >>
```

or, if the `\$SGE_ROOT` environment variable already is set:

```
Checking $SGE_ROOT directory
----------------------------

The Grid Engine root directory in DOS or UNC notation is:

  $SGE_ROOT = <installation_path>

Please note: If this directory is located on a mounted share (e.g. "X:\sge_root"),
the execution daemon cannot be started automatically at host boot time! Please use
the equivalent UNC path (e.g. "\\server\share\sge_root") instead!

If this directory is not correct (e.g. it may contain an automounter
prefix) enter the correct path to this directory or hit <RETURN>
to use default [<installation_path>] >>
```

- If `$SGE_ROOT` is located on a network drive, it is preferred to specify the path in UNC notation instead of the DOS notation (see Understanding the difference between mounted shares and UNC paths). Automatically starting the execution daemon at execution host boot time is not possible if it is located on a mounted share!
- Change the directory if necessary and press *return* to continue.

3. Specify the cell name.

```
Grid Engine cells
-----------------

Please enter cell name which you used for the sge_qmaster
installation or press <RETURN> to use [default]
```

   - Change the cell name if necessary and press *return*.

```
Using cell: >default<

Hit <RETURN> to continue
```

   - Press *return* again to continue.

4. Specify the admin user.

```
Univa Grid Engine admin user
----------------------------

In the bootstrap file, the "admin_user" is defined as
  >>jdoe<<

The current user is
  >>WINAD\jdoe<<

Is this the user you want to use as the Univa Grid Engine admin user
on the current host? (y/n) [y] >>
```

   - The name of the current user is fully qualified; the first part is the Windows domain, the second part is the user name itself. If this is the right user in the right domain, press *return* to accept this user as the `admin_user` on this Windows host.

5. Specify the TCP/IP port number.

```
Grid Engine TCP/IP communication service
----------------------------------------

The port for sge_execd.exe is set to

  SGE_EXECD_PORT = 5001

Hit <RETURN> to continue (enter 'q' to quit) >>
```

   - Press *return* to continue.

6. Define the path mapping.

Note
There is only one path map for all Windows execution hosts, so make sure the paths are the same on all Windows execution hosts!

```
Defining path mappings between UNIX and Windows
-----------------------------------------------

This execution daemon will get sent several paths from the Qmaster.
Because the Qmaster knows only UNIX paths, the execution daemon will
have to translate these paths from UNIX to Windows format.

This can't be done fully automatic, because mount points etc. usually
differ between UNIX and Windows. The execution daemon needs to know
some base paths in order to translate all paths.

These base paths are the users home directory, a temporary directory
and the directory where the certificates can be found.

These paths will be defined in the next steps

Hit <RETURN> to continue (enter 'q' to quit) >>
```

- Press *return* to continue.

```
Defining the mapping of the >sgeCA< directory
---------------------------------------------

During Qmaster installation, the certificates to encrypt and decrypt
the Windows user passwords were created on the Qmaster host in the directory
/var/sgeCA/port5000

This directory must be available on this Windows execution host. It can
be made available either via a network share or by copying it to this host.

Please enter the path of the >sgeCA< directory on the Qmaster host
or press <RETURN> to use [/var/sgeCA] >>
```

- The default path `/var/sgeCA` should always be right and for the mapping it is not necessary to specify the actual path to the certificates in more detail (i.e. the two top levels of the path are already sufficient), so press *return* to continue.

```
Please enter the DOS/UNC path of the >sgeCA< directory on this Windows host
>>
```

- Here specify just the top level of the path. If you copied the certificates e.g. from `/var/sgeCA/port5000` to `c:\certs\port5000` then specify just `c:\certs`. Then press *return* twice to continue.

```
Defining the mapping of the users home directory
------------------------------------------------

Please specify the users home base directory, e.g. "/home" and not
"/home/jdoe", both for the UNIX and the Windows path.

The users home directory paths should point to the same directory
on UNIX and Windows.  If they point to two different directories,
most of the execution host functionality will work, but it can also
cause problems.

Please enter the users home dir on the Qmaster host in UNIX notation
or press <RETURN> to use [/home] >>
```

- If you have several root directories of your homes, you can add them later manually. If so, specify here the home directory root of your UGE admin user.

```
Please enter the users home dir on this Windows host in DOS/UNC notation
>>
```

- The same applies to the Windows counterpart of your home directories. If you have several, you can add them later manually. First enter the one of the UGE admin user. Though, it is not possible to specify several Windows directories for one Unix directory! But it is possible to specify several Unix directories for one Windows directory. Remember to enter just the top level directory, not the user specific directory, i.e. `\\server\homes\` instead of `\\server\homes\jdoe`. Using UNC paths is preferred, if possible.
  Press *return* to continue.

```
Defining the mapping of the temporary directories
-------------------------------------------------

The paths of the temporary directory on UNIX and Windows can point to
two different directories, the execution daemon just needs to know both
paths. Please specify the base directory also here.

!! During the Alpha test phase, it's necessary to create the directory
    C:\tmp and make it writable for all users !!

Please enter the tmp dir on the UNIX hosts in UNIX notation
or press <RETURN> to use [/tmp] >>
```

- This should always be /tmp on UNIX/Linux systems. Just press *return* to continue.

```
Please enter the tmp dir on this Windows host in DOS/UNC notation
>>
```

- This directory has to be `c:\tmp`. Please create this directory and make it writable for everyone, see Preparing the Microsoft Windows hosts! Then enter `c:\tmp` here and press *return* to continue.

7. Specify the local execution daemon spool directory

```
Execd spool directory configuration
-----------------------------------

Please define a spool directory for this execution host.

ATTENTION: On Windows systems, the spool directory MUST be located
on a local disk. If you install an execution daemon on a Windows system
without a local spool directory, the execution host is unusable.

Please enter the spool directory in DOS/UNC notation
>>
```

- Enter a local spool directory here. As it is local, DOS notation should be used. Then press *return* to continue.

8. Confirm the path mappings

```
Confirming path mappings
------------------------

These path  mappings will be saved:

SGE_ROOT (UNIX):     /work/clusters/820
SGE_ROOT (Windows):  \\server\work\clusters\820
sgeCA dir (UNIX):    /var/sgeCA
sgeCA dir (Windows): \\server\var\sgeCA
home dir (UNIX):     /home
home dir (Windows):  \\server\home
tmp dir (UNIX):      /tmp
tmp dir (Windows):   c:\tmp
execd_spool_dir (UNIX):    /execd_spool_dir/win-x86/placeholder
execd_spool_dir (Windows): c:\tmp\spool

(The execd_spool_dir (UNIX) is a fixed value and must not be changed!)

Use these paths (y) or enter them again (n)? (y/n) [y] >>
```

- Check again if the path mappings are OK. The “execd_spool_dir (UNIX)” is a special value, this will be used in the local configuration of this execution daemon so the qmaster doesn’t need to know the mapped execd_spool_dir. If everything is fine, press *return* to continue.

9. Checking host name resolving

```
Checking hostname resolving
---------------------------
```

```
This hostname is known at sge_qmaster as an administrative host.

Hit <RETURN> to continue (enter 'q' to quit) >>
```

- Press *return* to continue.

10. Creating the local configuration

```
Creating local configuration
----------------------------
jdoe@winhost added "winhost" to configuration list
Local configuration for host >winhost< created.


Hit <RETURN> to continue (enter 'q' to quit) >>
```

- This creates the local configuration for this host. You can use `qconf -sconf <hostname>` on the qmaster host to see it. Press *return* to continue.

11. Writing settings.bat and sgeexecd.bat file

```
Writing settings.bat and sgeexecd.bat file
------------------------------------------

Writing settings.bat file to
>\\server\work\clusters\820\default\common\<

Done writing settings.bat file

Writing sgeexecd.bat file to
>\\server\work\clusters\820\default\common\<

Done writing sgeexecd.bat file
Hit <RETURN> to continue (enter 'q' to quit) >>
```

- These two file must be run later to set the necessary environment variables resp. to start the execution daemon manually. Press *return* to continue.

12. Installing UGE services

```
Installing UGE services
-----------------------

Do you want to install a service that automatically starts the execution
daemon at machine boot time? (y/n) [y] >>
```

- It is recommended to install this service, because only the service starts the execution daemon properly at host boot time. Press *return* to continue.

```
The UGE Starter Service will be installed.

Hit <RETURN> to continue (enter 'q' to quit) >>
```

- Press *return* to continue.

13. Adding a queue

```
Adding a queue for this host
----------------------------

We can now add a queue instance for this host:

   - it is added to the >allhosts< hostgroup
   - it is made sure the all.q queue instance on this host
     provides at least one slot

You do not need to add this host now, but before running jobs on this host

Do you want to add a default queue instance for this host (y/n) [y] >>
```

- If you do not plan to use just other queues than the “all.q” on this host, just press *return* twice to continue.

14. Performing tasks as the local Administrator

```
Performing tasks with elevated Administrator permissions
--------------------------------------------------------

Now several tasks  have to be performed with elevated Administrator permissions.
These tasks are:
a) Copying these files to c:\\Windows in order to make them accessible at boot time:
* pthreadVC2.dll - needed by most UGE executables
* SGE_Starter_Service.exe - starts the execution daemon at boot time
* UGE_JS_Service.exe - does the job execution in the Windows environment
* SGE_Starter.exe - starter method needed for GUI jobs on Windows Vista and later
b) Installing the SGE_Starter_Service and the UGE_JS_Service
c) Adding the UGE admin user to the access list of the perflib registry key.
    This registry key provides load values, but is accessible by default only
    by interactive users, not by services.

Please enter the Administrator name and password in the Windows User Account
Control dialog box that will pop up now and then follow the steps in the
new cmd window that is going to open. If the dialog box states that it is
already running as this user, please enter the user name and the password,
```

```
anyway, just clicking OK won't work

Hit <RETURN> to continue (enter 'q' to quit) >>
```

- Press *return* to continue.
- Depending on your Windows version and User Account Control settings, a dialog box will open where you have to enter a user name and a password. Here, the name and the password of a user with local Administrator permissions has to be entered. To avoid confusion: This user might be a domain user, or it might be a local user - it just has to have local Administrator permissions. You have to enter the name and the password, it is **NOT** sufficient to just press the `OK` button!
- A second cmd.exe window will then open, copy some files and ask for the password of the UGE admin user - the user as which you started this installation script. The password is needed to start the `UGE Starter Service` as the UGE admin user at next system boot time.
- In the second window, some more files will be copied. After this, press *return* to close this window and continue in the first window.

15. Starting execution daemon

- This screen will be showed only if no service to start the execution daemon at boot time was installed.

```
Starting execution daemon
-------------------------

Please wait until the execution daemon is started...

Hit <RETURN> to continue (enter 'q' to quit) >>
```

- Press *return* to continue.

16. Summary screen

```
Using Grid Engine
-----------------

You should now enter the command:

   \\server\work\clusters\820\default\common\settings.bat

This will set or expand the following environment variables:

  - SGE_ROOT         (always necessary)
  - SGE_CELL         (if you are using a cell other than >default<)
  - SGE_CLUSTER_NAME (just for compatibility)
```

```
  - SGE_QMASTER_PORT (if you haven't added the service >sge_qmaster<)
  - SGE_EXECD_PORT   (if you haven't added the service >sge_execd<)
  - PATH             (to find the Grid Engine binaries)

Hit <RETURN> to end execution daemon installation >>
```

- Press *return* to end the installation.

```
Your execution daemon installation is now complete.
```

- Now enter the command to set the environment variables:

```
\\server\work\clusters\820\default\common\settings.bat
```

- After this, the following commands should work:

```
> qhost
> qstat -f
```

17. Modifying the path_map file

- The `path_map` file is located in `$SGE_ROOT\$SGE_CELL\common` and contains the mappings that were specified during the execution host installation. If the home directories of the UGE users are distributed over several shares, for each share a line can be added to the `path_map` file now. If there is e.g. a `/home1` and a `/home2` on the UNIX hosts and the mapping from `/home1` to `\\server\home1` was specified during the installation, just add a line `/home2, \\server\home2` to the `path_map` file.

18. Continue to install the next execution host

6.3.3 Interactive uninstallation of Windows execution host

1. Prepare to uninstall.

- Log in on the master host as user *root*.
- Set the necessary environment variables.

```
# . <installation_path>/default/common/settings.sh
```

- Change to the installation directory.

```
# cd $SGE_ROOT
```

- Be sure that jobs are not currently running on that host nor will any be started during the uninstallation. First disable the host:

```
# qmod -d "*@win_host"
```

where “win_host” is the name of the Windows Execution host to uninstall. Then either wait until all jobs finished or delete them using

```
# qdel -j <jobid1>,<jobid2>,...
```

where “jobid1” and “jobid2” are the IDs of the jobs that run on the Windows Execution host.

2. Start the uninstallation.

   - Execute the following command on a execution host as user *root* to uninstall the execution daemon:

     ```
     ./inst_sge -ux

     Grid Engine uninstallation
     --------------------------

     You are going to uninstall a execution host <hostname>!
     If you are not sure what you are doing, than please stop
     this procedure with <CTRL-C>!

     Hit <RETURN> to continue
     ```

   - Press *return* to continue.

     ```
     Grid Engine TCP/IP communication service
     ---------------------------------------

     The port for sge_execd is currently set by the shell environment.

        SGE_EXECD_PORT = 6444

     Hit <RETURN> to continue
     ```

   - Press *return* to continue.

     ```
     Checking hostname resolving
     ---------------------------

     This hostname is known at qmaster as an administrative host.

     Hit <RETURN> to continue
     ```

   - Press *return* to continue.

```
hostname              <hostname>
load_scaling          NONE
complex_values        NONE
load_values           ...

...

Removing execution host <hostname> now!

...

Detected a presence of old RC scripts.
/etc/init.d/sgeexecd.p5000
```

3. Remove startup scripts.

```
Checking for installed rc startup scripts!

Removing execd startup script
-----------------------------

Do you want to remove the startup script
for execd at this machine? (y/n)
```

- Press *y* and *return* to remove the startup script for the execution host.

```
/usr/lib/lsb/remove_initd /etc/init.d/sgeexecd.p5000

Hit <RETURN> to continue
```

- Press *return* to finish the uninstallation.

4. OPTIONAL : Remove admin host privileges.

- If the host is not a shadow host or master host, and if it should not be allowed to execute administrative commands, then the administrator host privileges can be removed with the following command:

```
# qconf -dh <hostname>
```

6.3.4 Automated installation of Windows execution hosts

The `install_execd.bat` script can be used to automate the installation of Univa Grid Engine. Instead of interactively asking for configuration values, this installation method reads all parameters from a configuration template. Unlike on Unix/Linux, the automated installation can be done only on the local host, and it can not run unattended, because there usually is a password dialog from Windows during the installation.

Please see the execution host related sections of Automated Installation for details. It works the same for Unix/Linux and Windows execution hosts.

Please note that the Windows execution host installation does not create certificates nor does it copy existing certificates to the Windows execution host.

6.3.5 Automated uninstallation of Windows execution hosts

Not yet implemented.

6.4 Installing submit and/or administrative hosts on Windows with the command line installation script

There are two options to install an submit and/or administrative host on Windows:

- Interactive installation with installation script
- Automatic installation with installation script

To install an submit and/or administrative host on Windows, in addition to all the prerequisites listed above, the `sge_qmaster` daemon must be running and be accessible.

Prerequisites:

- Make sure the Windows host is registered as Administrative host at the qmaster. Run the following command on any existing Administrative host:

```
# qconf -ah <hostname>
```

- If you want to use the Windows host also as Submit host, run the following command to add the host to the list of submit hosts:

```
# qconf -as <hostname>
```

- Log in 'at the Windows host', i.e. either directly at the machine or via rdesktop. Log in as the Univa Grid Engine "admin user", see Understanding the roles of users on the Microsoft Windows hosts.
- Open a console (aka. "cmd.exe" window).

6.4.1 Interactive installation of Windows submit and/or administrative hosts

1. Start the installation script

- The Execution Host installation script is also used to install an Administrative or Submit Host.

- Start the execution host installation script “install_execd.bat”. Use the UNC path if `\$SGE_ROOT` is located on a network share, i.e. `\\server\share\$SGE_ROOT\install_execd.bat`. Use the DOS path (e.g. `c:\UGE\install_execd.bat`) only if `\$SGE_ROOT` was copied to a local drive.
- Follow the steps 2. to 12. in the Interactive Windows Execution host installation. If the installer asks for paths, if possible provide them in UNC format (i.e. `\\server\share\path`), not in DOS format. Don’t change the path mapping if it already exists. If it doesn’t yet exist, provide all paths and at least an existing dummy directory to map the `/var/sgeCA` directory to, e.g. `c:\tmp`.
- After the installation script wrote the “settings.bat” and “sgeexecd.bat” files (this is step 12.), quit the installation!

2. Run the settings.bat script

```
> $SGE_ROOT\$SGE_CELL\common\settings.bat
```

- Now this command should work:

```
> qhost
```

- If you added the Windows host to the submit host list, it should also be possible to submit a test job:

```
> qsub -l "a=lx*" %SGE_ROOT%\examples\jobs\sleeper.sh
```

3. OPTIONAL : Remove admin host privileges.

- If it should not be allowed to execute administrative commands from this host, then the administrator host privileges can be removed with the following command:

```
> qconf -dh <hostname>
```

6.4.2 Interactive uninstalling of Windows submit and/or administrative hosts

Not yet implemented.

6.4.3 Automated installation of Windows submit and/or administrative hosts

Not yet implemented.

6.4.4 Automated uninstallation of Windows submit and/or administrative hosts

Not yet implemented.

6.5 Post-installation steps

6.5.1 Register job user passwords

In order to run jobs on Windows Execution hosts, all job users must register their passwords using the `sgepasswd` binary. This is possible only on Unix/Linux hosts that are part of the same Univa Grid Engine cluster.

E.g. a user named `jdoe` should use the following command:

```
# sgepasswd
Changing password for jdoe
New password:
Re-enter new password:
Password changed
```

Please see the sgepasswd(1) man page for the proper file permissions of the sgepasswd application and other details.

6.5.2 Running test jobs

After installation of the Windows Execution host, run some tests jobs to check if the installation was successful.

- Submit a simple test job

```
> qsub -l host=winhost1 -o /tmp -j y -b y notepad.exe
```

- Now a "notepad.exe" should show up in the process list of the Task Manager. Check if the job is known as running in the Qmaster (this can take some seconds):

```
> qstat -f
queuename                      qtype resv/used/tot. np_load  arch          states
---------------------------------------------------------------------------------
all.q@masterhost               BIPC  0/0/40         0.13     lx-amd64
---------------------------------------------------------------------------------
all.q@winhost1                 BIPC  0/1/10         0.01     win-x86
20         0.55476 notepad.ex jdoe         r     12/23/2015 19:19:51     1
```

- Job 20 is running on the win-x86 host "winhost1". Delete the job and check again:

```
> qdel 20
> qstat -f
queuename                      qtype resv/used/tot. np_load  arch          states
---------------------------------------------------------------------------------
all.q@masterhost               BIPC  0/0/40         0.13     lx-amd64
---------------------------------------------------------------------------------
all.q@winhost1                 BIPC  0/1/10         0.01     win-x86
```

- The job was deleted. Check in the Task Manager on the Windows Execution host that the notepad.exe process was ended.
- Check if the user home directory is accessible for the output files:

```
> qsub -l host=winhost1 -b y notepad.exe
```

- As soon as the job is scheduled, a file “notepad.exe.o21” should exist in the users home directory. If the job fails, it likely cannot write to the users home directory.
- Check if Windows applications can display their GUI on the visible desktop. This is often helpful to debug Windows applications, some display a message box in case of error even it they run in background. In order to allow this, someone must be logged on to the execution host of the job, either locally or using a RemoteDesktop session. This examples job automatically gets scheduled to a Windows execution host that allows to display the GUI on the visible desktop, which is the default for all Windows execution hosts. It should show a Notepad window there:

```
> qsub -l display_win_gui=true -b y notepad.exe
```

- The third test job is a batch script:

```
> qsub -l host=winhost1 -b y cmd.exe /c %SGE_ROOT%\examples\jobs\sleeper.bat 60
```

- To submit the same job from a Unix host, double all backslashes, because the Unix shell eats up half of them:

```
# qsub -l host=winhost1 -b y cmd.exe /c %SGE_ROOT%\\examples\\jobs\\sleeper.bat 60
```

- While the paths and variables in the arguments to the qsub command get mapped to Windows if possible, this is not possible for the arguments to the job. Univa Grid Engine cannot know what the arguments to the job mean and if there some mapping should be done, so the arguments must be given in the format the execution host understands.