Table of Contents

Table of Contents

1 Installation Guide

1.1 Installing Univa Grid Engine

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.

1.1.1 Planning the Installation

Univa Grid Engine supports the following hardware architectures and operating systems versions:

TABLE: Supported Platforms, Operating Systems and Architectures

Operating System	Version	Architecture
SLES	10,11	x86, x86-64
RHEL	4-5.6, 6	x86, x86-64
CentOS	4-5.6, 6	x86, x86-64
Oracle Linux	4-5.6, 6	x86, x86-64
Ubuntu Server	10.04LTS-10.10	x86, x86-64
Microsoft Windows[1]	Vista, HPC Server 2003, 7, XP SP3	x86, x86-64
Oracle Solaris	9,10	x86_64
HP-UX	11.0 or higher	32 and 64bit
IBM AIX	5.3, 6.1 or later	64 bit

[1] Hosts running the Microsoft Windows operations system cannot be used as master or shadow hosts.

1.1.1.1 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.

TABLE: Memory and CPU Requirements for Different Host Types

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.
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/default/common` directory.

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 `$SGE_ROOT/default/common` directory.
Execution Hosts	Multiple execution hosts can exist in a cluster. Each of these hosts runs the `sge_execd` process. Hosts running this process provide their compute resources to the corresponding cluster. Requirements: • Has to be setup as administrative host before the installation is started. • Should be setup as execution hosts for only one cell, otherwise special cluster setup has to be done so that corresponding resources are not oversubscribed. • Hardware and software requirements are dictated by the types of jobs to be executed on theses hosts. • Univa Grid Engine has no special requirements concerning memory or CPU resources. • Needs access to `$SGE_ROOT/default/common` directory.

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 Oracle Berkeley Database (BDB) 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.
- In contrast to classic spooling, BDB spooling requires local spooling on the master host. Local spooling can also be used to improve cluster throughput. As as result, the 50-200 MB would be needed on the master machine instead of the network disk.
- 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.

1.1.1.2 Selecting a File System for Spooling Operations

Univa Grid Engine supports two different spooling methods on the master host: classic spooling and BDB spooling. With classic spooling, the `sge_qmaster` service creates files containing the configuration objects of a Univa Grid Engine installation in human readable format. When BDB server spooling is enabled, a BDB database will be used to make data persistent. Both methods have different requirements and characteristics.

Classic spooling can be done on shared filesystems, whereas BDB spooling is only possible on filesystems that provide the necessary locking infrastructure. NFS3 cannot be used to do BDB spooling. NFS4 is recommended, but other filesystems like Lustre do work properly. When using Lustre file shares, disable file striping for Univa Grid Engine directories.

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 either requires classic spooling over an NFS3/NFS4 share or BDB spooling over NFS4.

Warning
Installing a shadow master with BDB server spooling is not supported in Univa Grid Engine 8.0.0.

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.

1.1.1.3 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.

1.1.1.4 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.

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.

1.1.1.5 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 or graphical installation.

TABLE: Necessary Installation Parameters

Parameter	Description	Value
Admin User	User Account for executing all Univa Grid Engine components.	*root* or a different user account (recommended). The same user will own the files of the Univa Grid Engine installation.
`$SGE_ROOT`	Base directory of the Univa Grid Engine installation.	
`$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.	
Spooling Mechanism	The *classic* or *BDB* 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.	
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, graphical 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.	

1.1.2 Prerequisite Steps

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

1.1.2.1 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.

1.1.2.1.1 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.

1.1.2.1.2 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_execd's`. 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.

1.1.2.1.3 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 both the automated and graphical installations. 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.

1.1.2.2 Preparing Windows Hosts

Hosts running certain Microsoft Windows operating systems can be integrated into Univa Grid Engine to act as execution, admin and submit hosts. This requires Microsoft Windows Services for UNIX (SFU) or Subsystem for Unix-based Applications (SUA) to be installed on all Windows hosts. This software can be downloaded from Microsoft. After installation, it provides the following features:

- Interix (UNIX) subsystem
- csh/ksh support
- Tools and utilities including development tools and libraries
- Access to NFS3 filesystems
- Access to PCNFS, NIS
- User mapping functionality
- Password synchronisation functionality

Note
Univa Grid Engine currently does not support master and shadow host functionality nor the qmon and qsh command line application on Microsoft Windows hosts.

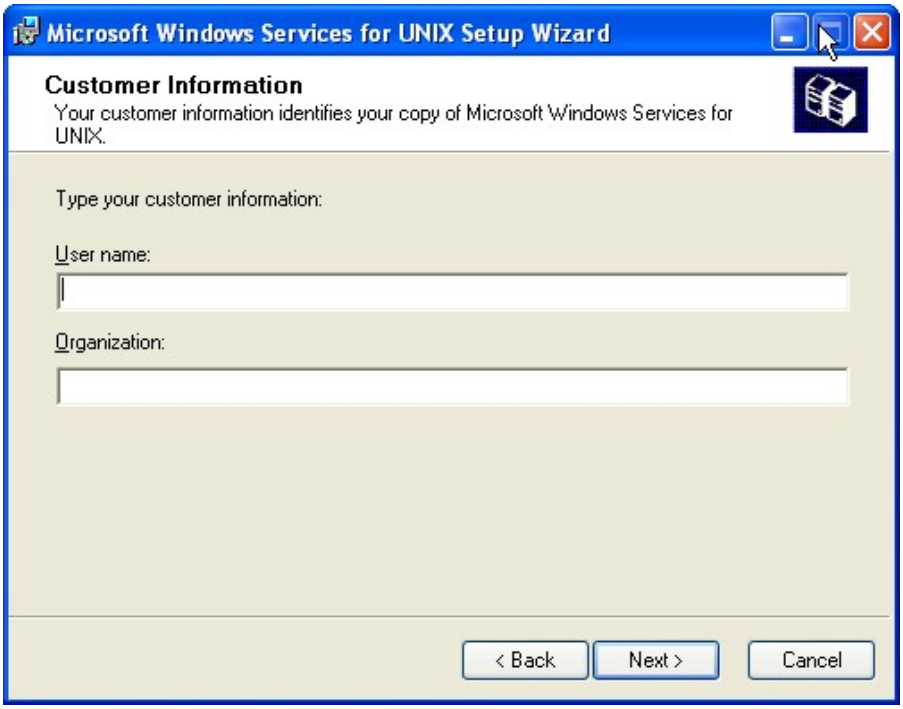
1.1.2.2.1 Install Microsoft Services for UNIX

The following steps show the Microsoft Windows Services for UNIX standard installation process and the setup of user mapping functionality. Some of the steps are marked as OPTIONAL because, depending on the operating system version or depending on the previous selection, they might not appear.

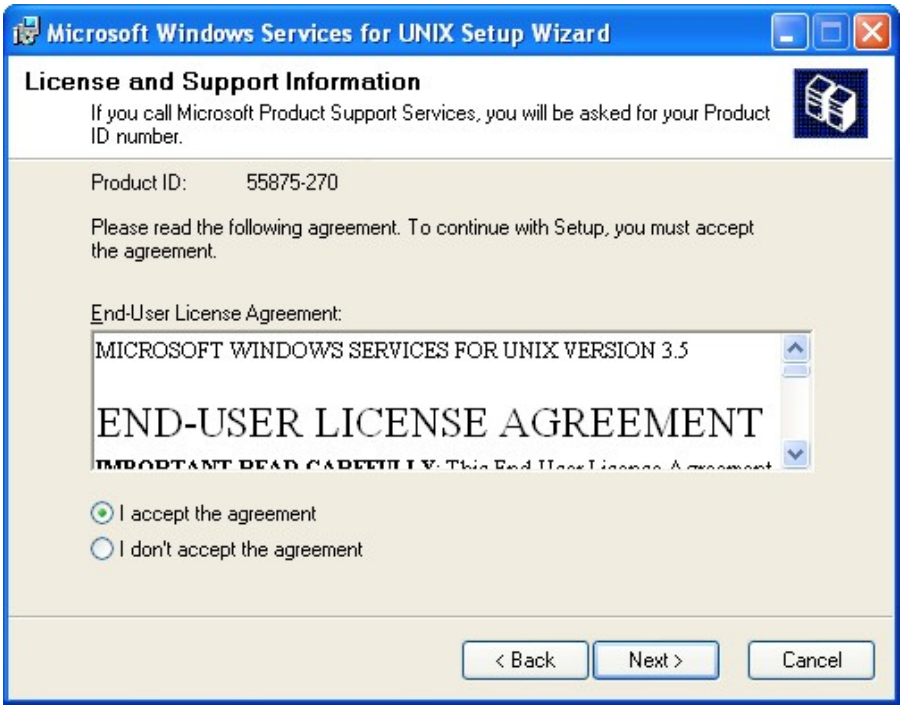
1. Prepare the configuration.
    - Make sure that the administrator accounts on all machines that could later be used as execution hosts for Univa Grid Engine use the same account name. This documentation assumes that this account name is *Administrator*.
    - If there is a Domain Controller available in the Windows environment, then start with the installation of SFU on that host.
    - Download the necessary files.
    - Execute the application to unzip the files into a directory.
    - Log in to the Windows system with the *Administrator* account.

2. Start the `setup.exe` application.

3. Enter the user name and Organization.



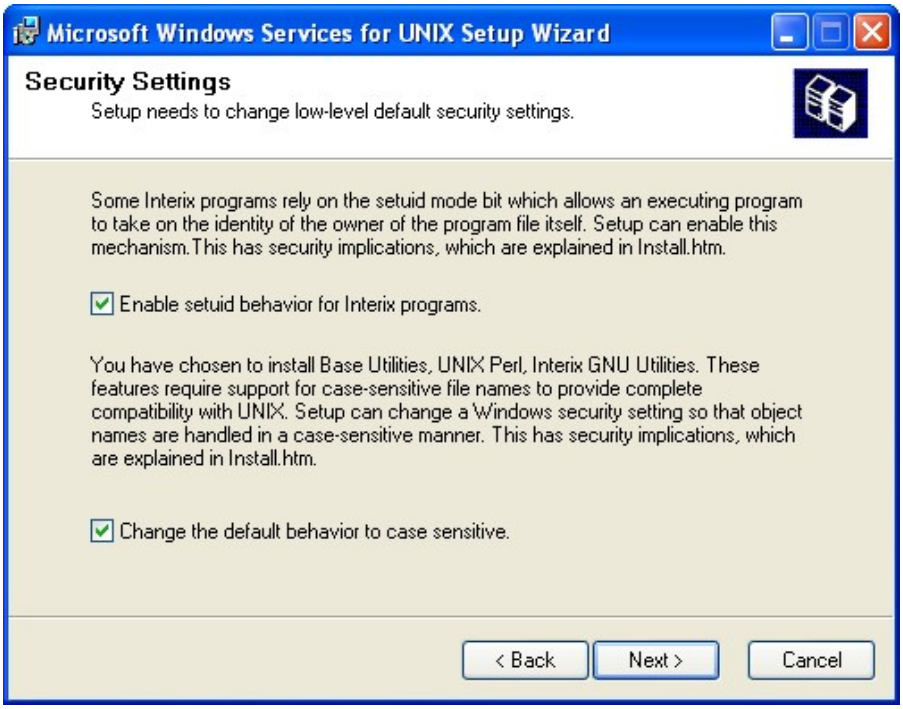
4. Read and accept the license agreement.



5. Choose the standard installation.

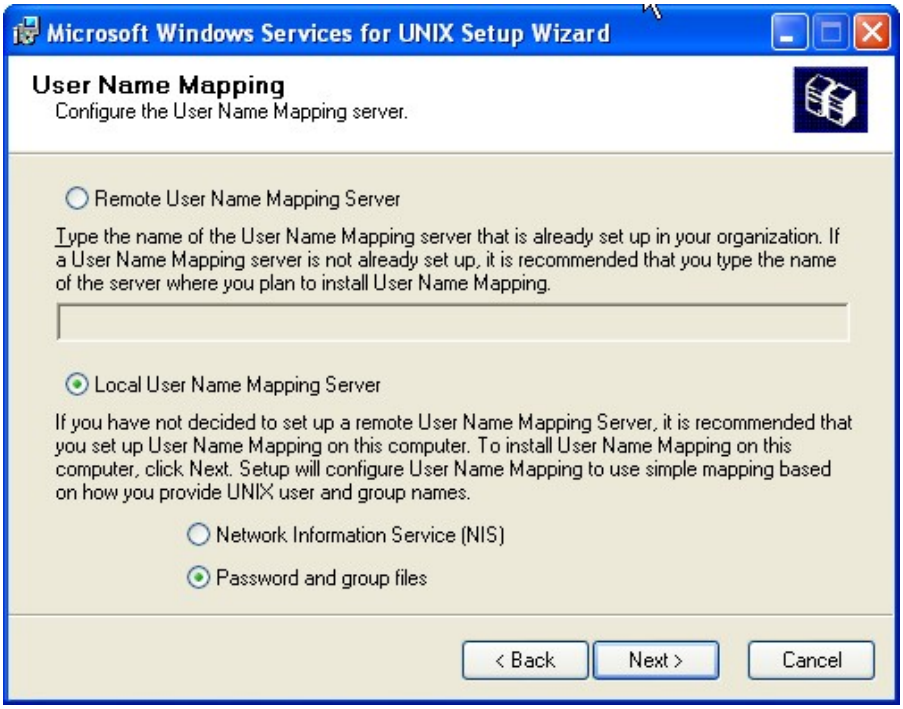
- Although custom installation might be used to save disk space, the following product parts are required:
  - *Utilities -> Base Utilities*
  - *Interix GNU components -> Interix GNU Utilities*
  - *Remote connectivity components -> Telnet Server and Windows Remote Shell*
  - *Authentication tools for NFS -> User Mapping and Server for NFS Authentication*

6. OPTIONAL Choose the security setting.



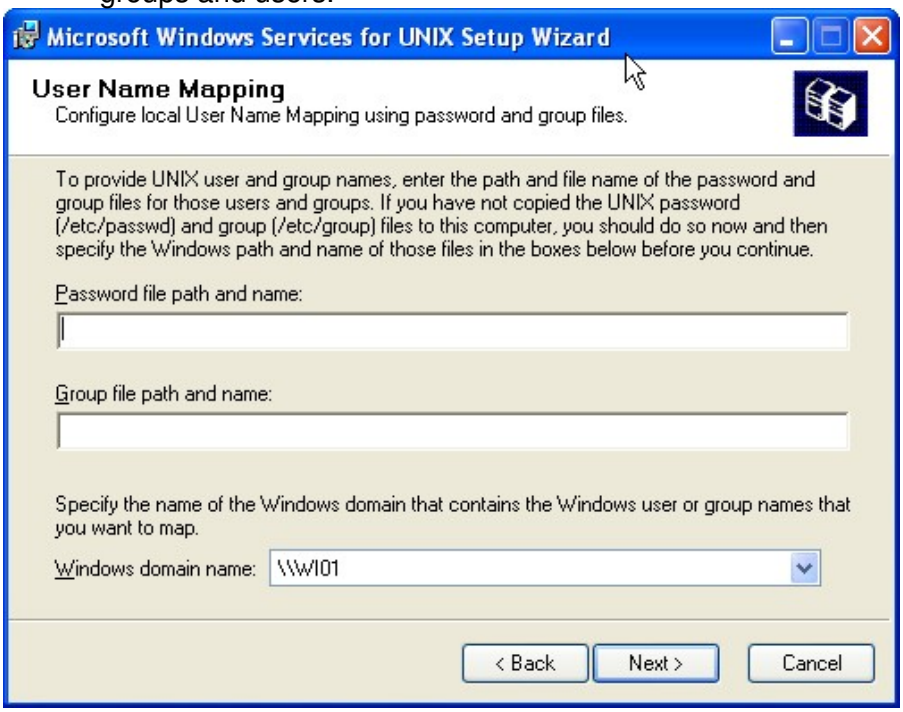
- Depending on the Windows operating system, the install dialog might not be shown.
- Choose *Enable setuid bahavior for Interix programs*.
- Choose *Change the default behavior to case sensitive*.

7. Configure user name mapping.



- Choose *Local User Name Mapping Serve* on your Domain Controller. If there are NIS maps available for user administration, choose *Network Information Service (NIS)*; otherwise choose *Password and group files*.
- On other hosts, choose *Remote User Mapping Server* and specify the name of the Domain Controller.

8. Specify details for user name mapping.
    - Depending on the previous installation step, either enter the *NIS domain* and *NIS server* name, or specify the path of a *Password File* and *Group file* that contains all UNIX groups and UNIX users that could be mapped to Windows groups and users.



    - The passwd file has the following format:

```
#username:x:uid:gid:full user name:home directory:shell path
root:x:0:0:UNIX root user:/root:/bin/sh
user1:x:1001:100:Full name of user1:/home/user1:/bin/tcsh
...
```

    - The group file has the following format:

```
#groupname::gid:
root::0:
group1::100:
```

Note

To use NIS maps when no entry for the root user account exists in the NIS map, use the following workaround to achieve *root<->Administrator* mapping:

- Create a password file containing only the root user account.
- When the SFU installation is finished, use the Services for UNIX Administration application to create a mapping for *root<->Administrator*.
- The created *root<->Administrator* mapping will not be deleted when switching to NIS user mapping now.
- Either choose the simple mapping, or add mappings manually.
- Continue the installation process.

9. Post-installation steps.
    - Check Services.
        - Depending on the Windows version, it might be necessary to restart the machine.
        - Check that the *Interix Subsystem* is started during boot time.
        - OPTIONAL: When intending to use NFS, be sure that the *Client for NFS* and *User Name Mapping* are started.
    - OPTIONAL: Automount NFS shares.
      This is the recommended way to access NFS shares. Create symbolic links to network shares that are all available in the Interix subsystem through the special directory `/net` followed by the server name and share name. The following example makes `/home` a link that directs to the network share that is automatically mounted as soon as a user who has the appropriate access permissions tries to access that directory.

```
# ln -s /net/<fileserver>/<home_share> /home
# ls -la /home/<username>
...
```

    - OPTIONAL: Manually mount NFS shares.
      Network shares can also be linked to drive letters. The following command mounts a network share to the drive letter *Z:*. Drives with drive letters can be accessed through subdirectories located in `/dev/fs` on the Interix subsystem.

```
# /usr/sbin/nfsmount -u: \\\\net\\<fileserver>\\<home_share> Z:
# ls -la /dev/fs/Z
...
```

    - OPTIONAL: Use NFS shares as Windows home directory.
        - Open the *Control Panel*, and follow these links:
          Administrative Tasks -> *Computer Management* -> *Users* -> *Properties* -> *Profile*
        - Select *Connect*.
        - Select a drive letter.
        - Enter the users's home directory path in UTC notation:
          \\<fileserver>\<home_share>\<username>

    - Start an Interix shell and switch to a non-*Administrator* user.

```
# login <username>
...
# id
...
```

    - Try to access a network drive to see if the user has the correct access permissions.

```
# ls -la /net/<server>/<share>
...
# touch /net/<server>/<share>/<new_filename>
...
# ls -la /net/<server>/<share>/<new_filename>
...
# rm /net/<server>/<share>/<new_filename>
```

Note
User Mapping is part of SFU. When encountering any errors, please read the documentation provided from Microsoft and/or contact Microsoft support.

- Register Windows Domain User Passwords.
  Windows Domain Users have to register their Windows password so that the Univa Grid Engine System is able to start jobs under their account. A user named *John* could do this using following command, if we assume that this user is part of the Windows domain named *DESIGN*.

```
# sgepasswd -D DESIGN
Changing password for DESIGN+John
New password:
Re-enter new password:
Password changed
```

- Check other requirements for Univa Grid Engine.
  - Make sure that the Windows *Administrator* has admin privileges in the Univa Grid Engine cluster.
  - Set the EDITOR environment variable correctly for all users who want to use Univa Grid Engine client commands.

1.1.2.3 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:

TABLE: Available Files

Filename	Description
ge-8.0-common.tar.gz	Architecture independent file
ge-8.0-bin-lx-amd64.tar.gz	Linux x86; 64 bit binaries
ge-8.0-bin-lx-x86.tar.gz	Linux x86; 32 bit binaries
ge-8.0-bin-sol-amd64.tar.gz	Solaris x86; 64 bit binaries
ge-8.0-bin-sol-sparc64.tar.gz	Solaris SPARC platform; 64 bit binaries

   - 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-8.0-common.tar.gz | tar xvpf -
# gzip -dc <download_dir>/ge-8.0-bin-lx-amd64.tar.gz | tar xvpf -
# ...
```

4. Correct the file permission.

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

1.1.3 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/asc_depend_<arch>.asc` where `<arch>` is the architecture name.

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

- Installation with a graphical user interface
- Interactive installation with installation scripts
- Auto 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 Automated Installation. The installation with the graphical installer is described in chapter Installing with the Graphical Installer.

1.1.3.1 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.
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.

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.

1.1.3.1.1 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.

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 [p64444]
```

- 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. 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.

10. 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*.

11. 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 unaccessible 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 >dtrace<
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 >start_gui_installer<
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<
Verifying and setting file permissions and owner in >hadoop<

Your file permissions were set

Hit <RETURN> to continue
```

♦ Press *return* to continue.

12. 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.

13. 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.

14. Set up the spooling method.

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

...

Please choose a spooling method (berkeleydb|classic)
```

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

```
Berkeley Database 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]
```

- Initial spooling information will be created then.

```
Dumping bootstrapping information
Initializing spooling database

Hit <RETURN> to continue >>
```

- Press *return* to continue.

15. Specify the group ID range.

```
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.

16. 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*.

17. Set up administrator mail.

```
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 >qtask< qtcsh sample default request 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.

18. 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.

19. 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.

20. WIN-only: Identify the Windows administrator account.

```
Windows Administrator Name
--------------------------

Please, enter the Windows Administrator name [Default: Administrator]
```

- Enter the name, and press *return*.

```
root@master.univa.com added "Administrator" to manager list
Hit <RETURN> to continue >>
```

- Hit *return* to continue.

21. 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 seperated 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.

22. 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.

23. 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.

24. 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*.

```
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)
```

25. 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.

26. 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.

27. CSP-only or WIN-only: Transfer certificate files and private keys (manually).
    - Installing in CSP mode or specifying the use of Windows execution nodes meant skipping the distribution of security information via ssh/rsh. Now, this step must be performed manually to continue with 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_service* 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
```

    - 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 .
```

    - WIN-only: The `tar` program on Windows execution hosts is not able to restore the ownership and permissions. The *Administrator* has to be sure that this is done manually.
    - Check that the permissions are correct.

```
# 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
```

28. 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.
    - Execution nodes can now be set up. Instructions are in the section Execution host installation.

1.1.3.1.2 Shadow Master Host Installation

1. Prepare to start.
    - Complete the master host installation as outlined in section 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
# $SGE_ROOT/util/sgeCA/sge_ca -copy
# 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 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.

1.1.3.1.3 Execution Host Installation

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

```
# SGE_ROOT=<installation_path>
# export SGE_ROOT
# . $SGE_ROOT/$SGE_CELL/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
# $SGE_ROOT/util/sgeCA/sge_ca -copy
# 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

Hit <RETURN> to continue
```

   - ♦ Hit *return* to continue.

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

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

   - ♦ Hit return to continue.

```
Hit <RETURN> to continue
```

10. 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 >>
```

11. 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.
```

12. Review next steps.
    - Continue to install the next execution host.

1.1.3.1.4 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/setting.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 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>
```

1.1.3.1.5 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/setting.sh
```

   - Change to the installation directory.

```
# cd $SGE_ROOT
```

2. Start the uninstallation.
    - Execute 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>
```

1.1.3.1.6 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/setting.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.

1.1.3.2 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 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.

1.1.3.2.1 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 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/u
```

   - 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 # SGE_ROOT Path, this is basic information
008 #(mandatory for qmaster and execd installation)
009 SGE_ROOT="Please enter path"
010
011 # SGE_QMASTER_PORT is used by qmaster for communication
012 # Please enter the port in this way: 1300
013 # Please do not this: 1300/tcp
014 #(mandatory for qmaster installation)
015 SGE_QMASTER_PORT="Please enter port"
016
017 # SGE_EXECD_PORT is used by execd for communication
018 # Please enter the port in this way: 1300
019 # Please do not this: 1300/tcp
020 #(mandatory for qmaster installation)
021 SGE_EXECD_PORT="Please enter port"
022
023 # SGE_ENABLE_SMF
024 # if set to false SMF will not control SGE services
025 SGE_ENABLE_SMF="false"
026
027 # SGE_CLUSTER_NAME
028 # Name of this cluster (used by SMF as an service instance name)
029 SGE_CLUSTER_NAME="Please enter cluster name"
030
031 # SGE_JMX_PORT is used by qmasters JMX MBean server
032 # mandatory if install_qmaster -jmx -auto <cfgfile>
033 # range: 1024-65500
034 SGE_JMX_PORT="Please enter port"
035
036 # SGE_JMX_SSL is used by qmasters JMX MBean server
037 # if SGE_JMX_SSL=true, the mbean server connection uses
```

```
038 # SSL authentication
039 SGE_JMX_SSL="false"
040
041 # SGE_JMX_SSL_CLIENT is used by qmasters JMX MBean server
042 # if SGE_JMX_SSL_CLIENT=true, the mbean server connection uses
043 # SSL authentication of the client in addition
044 SGE_JMX_SSL_CLIENT="false"
045
046 # SGE_JMX_SSL_KEYSTORE is used by qmasters JMX MBean server
047 # if SGE_JMX_SSL=true the server keystore found here is used
048 # e.g. /var/sgeCA/port<sge_qmaster_port>/<sge_cell>/private/keystore
049 SGE_JMX_SSL_KEYSTORE="Please enter absolute path of server keystore file"
050
051 # SGE_JMX_SSL_KEYSTORE_PW is used by qmasters JMX MBean server
052 # password for the SGE_JMX_SSL_KEYSTORE file
053 SGE_JMX_SSL_KEYSTORE_PW="Please enter the server keystore password"
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 # Leaving this blank, the owner of the SGE_ROOT dir will be used as admin user
076 ADMIN_USER=""
077
078 # The dir, where qmaster spools this parts, which are not spooled by DB
079 #(mandatory for qmaster installation)
080 QMASTER_SPOOL_DIR="Please, enter spooldir"
081
082 # The dir, where the execd spools (active jobs)
083 # This entry is needed, even if your are going to use
084 # berkeley db spooling. Only cluster configuration and jobs will
085 # be spooled in the database. The execution daemon still needs a spool
086 # directory
087 #(mandatory for qmaster installation)
088 EXECD_SPOOL_DIR="Please, enter spooldir"
089
090 # For monitoring and accounting of jobs, every job will get
091 # unique GID. So you have to enter a free GID Range, which
092 # is assigned to each job running on a machine.
093 # If you want to run 100 Jobs at the same time on one host you
094 # have to enter a GID-Range like that: 16000-16100
095 #(mandatory for qmaster installation)
096 GID_RANGE="Please, enter GID range"
```

```
097
098 # If SGE is compiled with -spool-dynamic, you have to enter here, which
099 # spooling method should be used. (classic or berkeleydb)
100 #(mandatory for qmaster installation)
101 SPOOLING_METHOD="berkeleydb"
102
103 # Name of the Server, where the Spooling DB is running on
104 # if spooling methode is berkeleydb, it must be "none", when
105 # using no spooling server and it must contain the servername
106 # if a server should be used. In case of "classic" spooling,
107 # can be left out
108 DB_SPOOLING_SERVER="none"
109
110 # The dir, where the DB spools
111 # If berkeley db spooling is used, it must contain the path to
112 # the spooling db. Please enter the full path. (eg. /tmp/data/spooldb)
113 # Remember, this directory must be local on the qmaster host or on the
114 # Berkeley DB Server host. No NFS mount, please
115 DB_SPOOLING_DIR="spooldb"
116
117 # This parameter set the number of parallel installation processes.
118 # The prevent a system overload, or exeeding the number of open file
119 # descriptors the user can limit the number of parallel install processes.
120 # eg. set PAR_EXECD_INST_COUNT="20", maximum 20 parallel execd are installed.
121 PAR_EXECD_INST_COUNT="20"
122
123 # A List of Host which should become admin hosts
124 # If you do not enter any host here, you have to add all of your hosts
125 # by hand, after the installation. The autoinstallation works without
126 # any entry
127 ADMIN_HOST_LIST="host1 host2 host3 host4"
128
129 # A List of Host which should become submit hosts
130 # If you do not enter any host here, you have to add all of your hosts
131 # by hand, after the installation. The autoinstallation works without
132 # any entry
133 SUBMIT_HOST_LIST="host1 host2 host3 host4"
134
135 # A List of Host which should become exec hosts
136 # If you do not enter any host here, you have to add all of your hosts
137 # by hand, after the installation. The autoinstallation works without
138 # any entry
139 # (mandatory for execution host installation)
140 EXEC_HOST_LIST="host1 host2 host3 host4"
141
142 # The dir, where the execd spools (local configuration)
143 # If you want configure your execution daemons to spool in
144 # a local directory, you have to enter this directory here.
145 # If you do not want to configure a local execution host spool directory
146 # please leave this empty
147 EXECD_SPOOL_DIR_LOCAL="Please, enter spooldir"
148
149 # If true, the domainnames will be ignored, during the hostname resolving
150 # if false, the fully qualified domain name will be used for name resolving
151 HOSTNAME_RESOLVING="true"
152
153 # Shell, which should be used for remote installation (rsh/ssh)
154 # This is only supported, if your hosts and rshd/sshd is configured,
155 # not to ask for a password, or promting any message.
```

```
156 SHELL_NAME="ssh"
157
158 # This remote copy command is used for csp installation.
159 # The script needs the remote copy command for distributing
160 # the csp certificates. Using ssl the command scp has to be entered,
161 # using  the not so secure rsh the command rcp has to be entered.
162 # Both need a passwordless ssh/rsh connection to the hosts, which
163 # should be connected to. (mandatory for csp installation mode)
164 COPY_COMMAND="scp"
165
166 # Enter your default domain, if you are using /etc/hosts or NIS configuration
167 DEFAULT_DOMAIN="none"
168
169 # If a job stops, fails, finish, you can send a mail to this adress
170 ADMIN_MAIL="none"
171
172 # If true, the rc scripts (sgemaster, sgeexecd, sgebdb) will be added,
173 # to start automatically during boottime
174 ADD_TO_RC="false"
175
176 #If this is "true" the file permissions of executables will be set to 755
177 #and of ordenary file to 644.
178 SET_FILE_PERMS="true"
179
180 # This option is not implemented, yet.
181 # When a exechost should be uninstalled, the running jobs will be rescheduled
182 RESCHEDULE_JOBS="wait"
183
184 # Enter a one of the three distributed scheduler tuning configuration sets
185 # (1=normal, 2=high, 3=max)
186 SCHEDD_CONF="1"
187
188 # The name of the shadow host. This host must have read/write permission
189 # to the qmaster spool directory
190 # If you want to setup a shadow host, you must enter the servername
191 # (mandatory for shadowhost installation)
192 SHADOW_HOST="hostname"
193
194 # Remove this execution hosts in automatic mode
195 # (mandatory for unistallation of execution hosts)
196 EXEC_HOST_LIST_RM="host1 host2 host3 host4"
197
198 # This option is used for startup script removing.
199 # If true, all rc startup scripts will be removed during
200 # automatic deinstallation. If false, the scripts won't
201 # be touched.
202 # (mandatory for unistallation of execution/qmaster hosts)
203 REMOVE_RC="false"
204
205 # This is a Windows specific part of the auto isntallation template
206 # If you going to install windows executions hosts, you have to enable the
207 # windows support. To do this, please set the WINDOWS_SUPPORT variable
208 # to "true". ("false" is disabled)
209 # (mandatory for qmaster installation, by default WINDOWS_SUPPORT is
210 # disabled)
211 WINDOWS_SUPPORT="false"
212
213 # Enabling the WINDOWS_SUPPORT, recommends the following parameter.
214 # The WIN_ADMIN_NAME will be added to the list of SGE managers.
```

```
# Without adding the WIN_ADMIN_NAME the execution host installation
# won't install correctly.
# WIN_ADMIN_NAME is set to "Administrator" which is default on most
# Windows systems. In some cases the WIN_ADMIN_NAME can be prefixed with
# the windows domain name (eg. DOMAIN+Administrator)
# (mandatory for qmaster installation, if windows hosts should be installed)
WIN_ADMIN_NAME="Administrator"

# This parameter is used to switch between local ADMINUSER and Windows
# Domain Adminuser. Setting the WIN_DOMAIN_ACCESS variable to true, the
# Adminuser will be a Windows Domain User. It is recommended that
# a Windows Domain Server is configured and the Windows Domain User is
# created. Setting this variable to false, the local Adminuser will be
# used as ADMINUSER. The install script tries to create this user account
# but we recommend, because it will be saver, to create this user,
# before running the installation.
# (mandatory for qmaster installation, if windows hosts should be installed)
WIN_DOMAIN_ACCESS="false"

# This section is used for csp installation mode.
# CSP_RECREATE recreates the certs on each installtion, if true.
# In case of false, the certs will be created, if not existing.
# Existing certs won't be overwritten. (mandatory for csp install)
CSP_RECREATE="true"

# The created certs won't be copied, if this option is set to false
# If true, the script tries to copy the generated certs. This
# requires passwordless ssh/rsh access for user root to the
# execution hosts
CSP_COPY_CERTS="false"

# csp information, your country code (only 2 characters)
# (mandatory for csp install)
CSP_COUNTRY_CODE="DE"

# your state (mandatory for csp install)
CSP_STATE="Germany"

# your location, eg. the building (mandatory for csp install)
CSP_LOCATION="Building"

# your arganisation (mandatory for csp install)
CSP_ORGA="Organisation"

# your organisation unit (mandatory for csp install)
CSP_ORGA_UNIT="Organisation_unit"

# your email (mandatory for csp install)
CSP_MAIL_ADDRESS="name@yourdomain.com"
```

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 "".

1.1.3.2.2 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:

TABLE: Command-line Options for inst_sge

Flag	Description
-auto <filename>	Enables the automated installation
-m	Install master host
-x	Install execution host
-sm	Install shadow master host
-s	Install submit host
-csp	Enables enhanced security features (CSP)

   - The different flags can be combined.
2. Start the inst_sge script.

```
# cd $SGE_ROOT
# ./inst_sge -m -x -auto $SGE_ROOT/util/install_modules/uge_configuration.conf
```

   - 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>.log` 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.

1.1.3.2.3 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.

TABLE: Command-line Options for `inst_sge`

Flag	Description
-auto <filename>	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).

    - The different flags can be combined.
2. Start the `inst_sge` script.

```
# cd $SGE_ROOT
# ./inst_sge -ux -auto $SGE_ROOT/util/install_modules/uge_configuration.conf
```

    - 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>.log` 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.

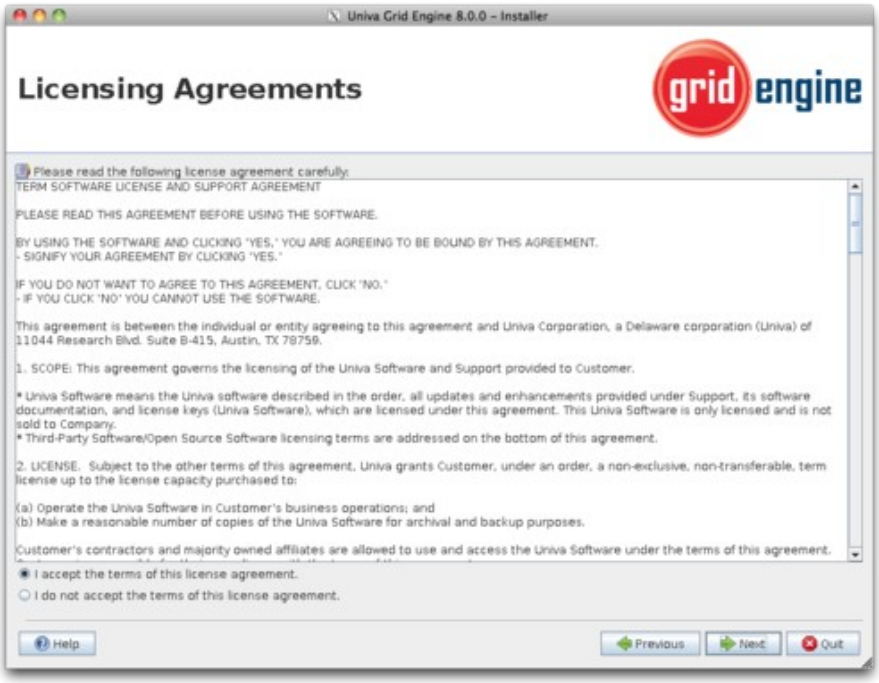
1.1.4 Installing with the Graphical Installer

The step-by-step instructions below show all installation screens that would be shown for an installation in *custom* mode with the CSP security feature enabled. Doing an *express* installation will cause all screens marked with the CUSTOM-MODE not to be shown. For an installation with CSP mode disabled, all parts tagged with CSP-only will not be required and will automatically be skipped by the installer.

1. Requirements
    - The graphical installer has following requirements:
        - ◊ Java JRE >= 5
        - ◊ Screen resolution 1024x768 or larger
        - ◊ OPTIONAL: Password-less root or rsh access to remote hosts that should be installed. If no password-less root access is available, then directly log in to all machines, and start the graphical installer to install a subcomponent.
2. Start the installer.
    - Log in as root.
    - Start the graphical installer.

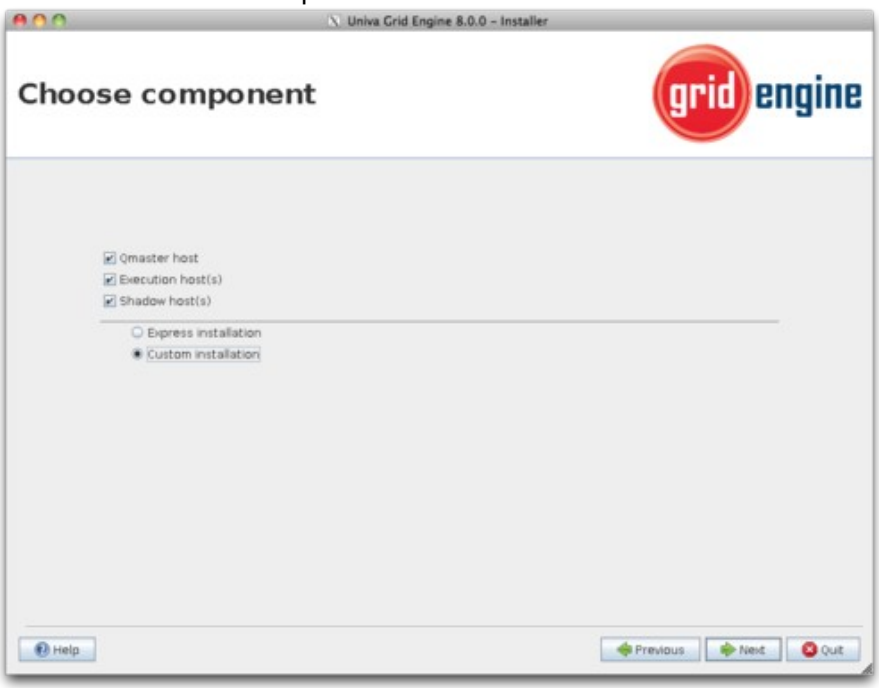
```
# cd $SGE_ROOT
# ./start_gui_installer
Starting Installer ...
```

3. Read and accept the license agreement.



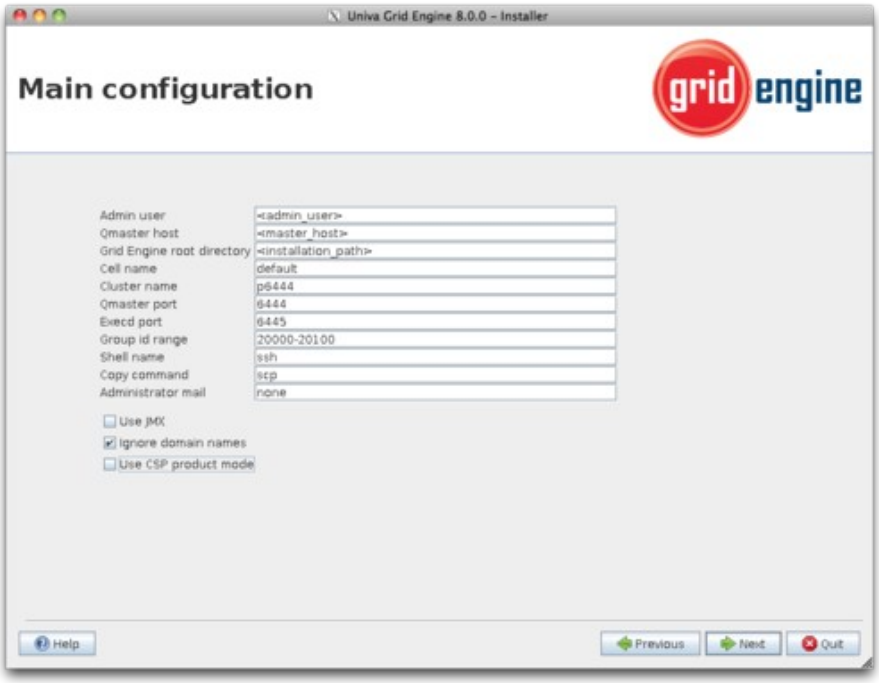
- Read and accept the license to continue.

4. Choose the components.
   - Choose the components that should be installed.

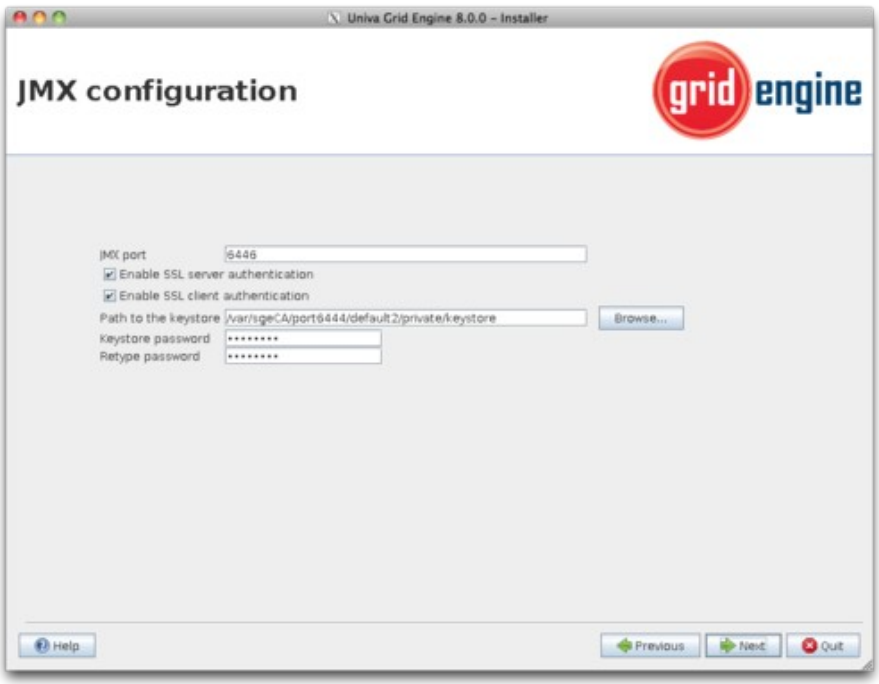


- Select the installation mode.

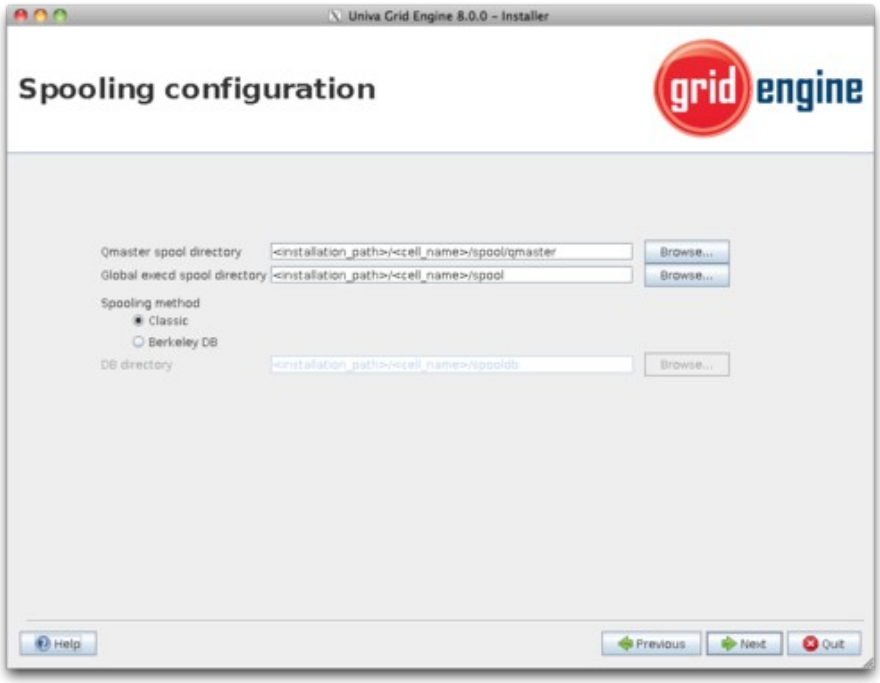
5. Change the configuration.
    - ♦ Change the values for the displayed settings.



6. CUSTOM-MODE: Modify the JMX configuration.

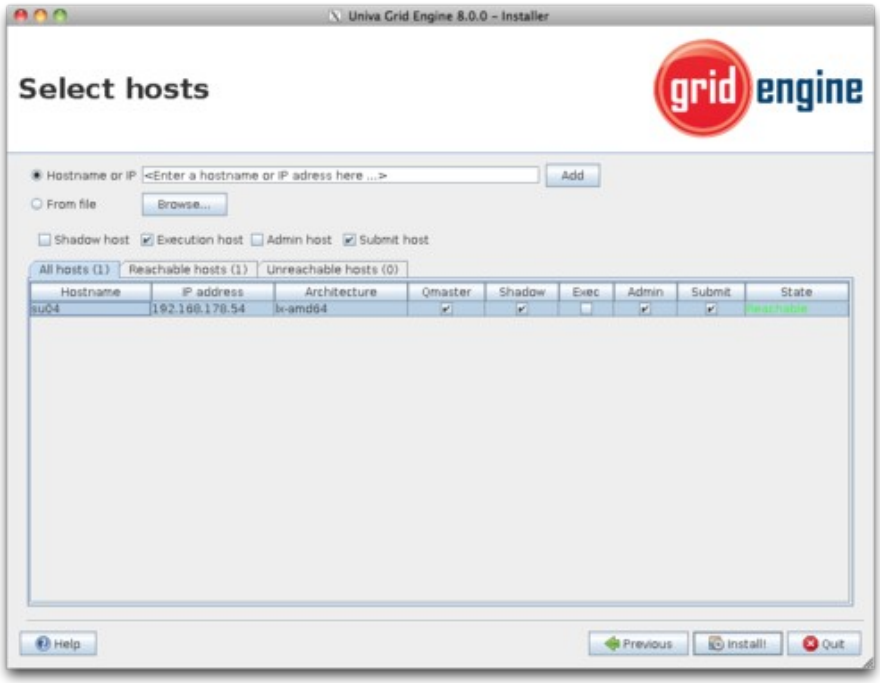


7. CUSTOM-MODE: Modify the spooling configuration.



8. CUSTOM-MODE and CSP-only: Provide the SSL certificate information.

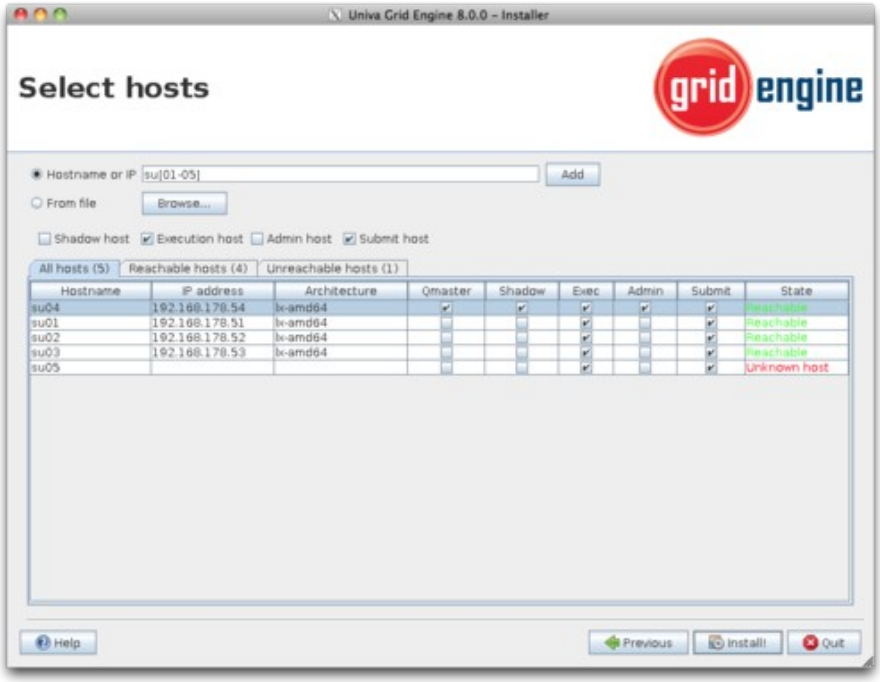
9. Select the hosts.



- Select the hosts and components to be installed. The qmaster host is added by default. Additional hosts can either be added by specifying a host file or by entering the IP addresses, IP address patterns, hostnames or hostname patterns. The table below shows some examples:

TABLE: Hostnames Possibilities in the *Hostname or IP* field

Description	Input	Result
Host name	host00	host00
IP address	192.168.0.1	192.168.0.1
List of hosts	host00 host01 host03	host00 host01 host03
List of IP addresses	192.168.0.1 192.168.0.2 192.168.1.1	192.168.0.1 192.168.0.2 192.168.1.1
Host name pattern	host[0-3]	host00 host01 host02 host03
IP address pattern	192.168.[0-1].[1-3]	192.168.0.1 192.168.0.2 192.168.0.3 192.168.1.1 192.168.1.2 192.168.1.3



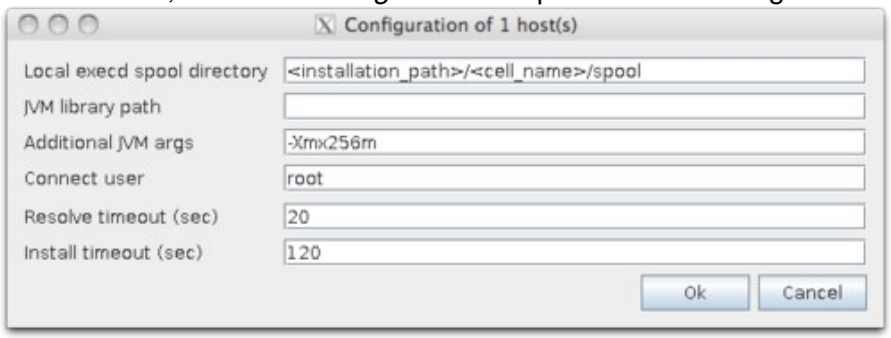
- New hosts are added in the *New unknown host* state. When the installer tries to resolve the host, if this step is successful, the the installer tries to log in via ssh/rsh to identify the host architecture. If this also is successful, then the host will change into the *Reachable* state. Other resolving results can be found in the following table:

TABLE: Host Resolving States

State	Description
New unknown host	Start state when a host was added.
Resolving	The installer is currently resolving the host.
Unknown host	Installer was not able to resolve the host.
Resolvable	Hostname was resolvable. If a host stays in this state, the installer was not able to ssh/rsh to the host to get the host architecture.
Contacting	Installer is currently in process to identify the host architecture.
Missing remote files	The installer was not able to execute `$SGE_ROOT/util/arch` on the host to get the host architecture.
Reachable	Host is resolved and architecture is known by the installer.
Unreachable	ssh/rsh access is not working properly.
Canceled	Host identification was canceled by the user.

- After the host names have been added, select the host roles that should be adopted by the corresponding host.

10. OPTIONAL: Change the host configuration.
    - To change the host configuration, select a host, right click to open the context menu, and click *Configuration* to open the host configuration dialog.



    - Here you can enter the local spooling directory for the execution host if it should be different from the global execd spool directory.
    - Press *Next* to continue.
11. OPTIONAL: Fix problems
    - Hosts that could be resolved and where the host architecture is known are moved to the *Reachable* tab, and those hosts can be used for installation. The installer starts further testing those hosts before the real installation starts. Possible results of the validation process can be found in the table below. There you can also find hints of how to solve the corresponding problem.

TABLE: Host Resolving States

State	Description	Resolution
Copy timeout or Copy failed	Timeout or error occurred when the installer tried to copy a file.	Tooltip will show the name of the file. Press the *Install* Button again. If the copy operation fails again, test if `scp` or `rcp` work correctly. Repeated timeouts might be eliminated by restarting the graphical installer with the command line parameter `-install_timeout=<sec>`. The specified value should be > 120.
Permission denied	The installer was not able to write a file.	Tooltip will show if it was not possible to write spool files during qmaster or execution host installation. This error might happen when the installer was not started as *root*, when the NFS setup defines that *root* account is mapped to *nobody* or when the admin user ID is different on different hosts.

Admin user missing	The admin user name that was entered in a previous step does not exist.	Return to the previous installation screen, and enter the correct name or create the user account.
Directory exists or Wrong filesystem type	Either the directory already exists or the filesystem is not appropriate for BDB spooling method.	Go back to the previous installation step. Check the specified spooling method, and be sure that the directory does not already exist.
Unknown error	An unknown error has occurred.	
Canceled	Installation was interrupted by user intervention.	
Reachable	Validation process did not find any misconfigurations for the remote host.	

If errors are found during these checks, return to the host selection dialog to adjust the hosts that are used for the installation process.

- Hosts that have been resolved successfully and where it was possible to retrieve the host architecture change to the *Reachable* state.

12. Monitor the installation.
    - When the installation starts, the installer prepares some tasks that need to be executed. One or more tasks will be started in parallel, based on installation dependencies.

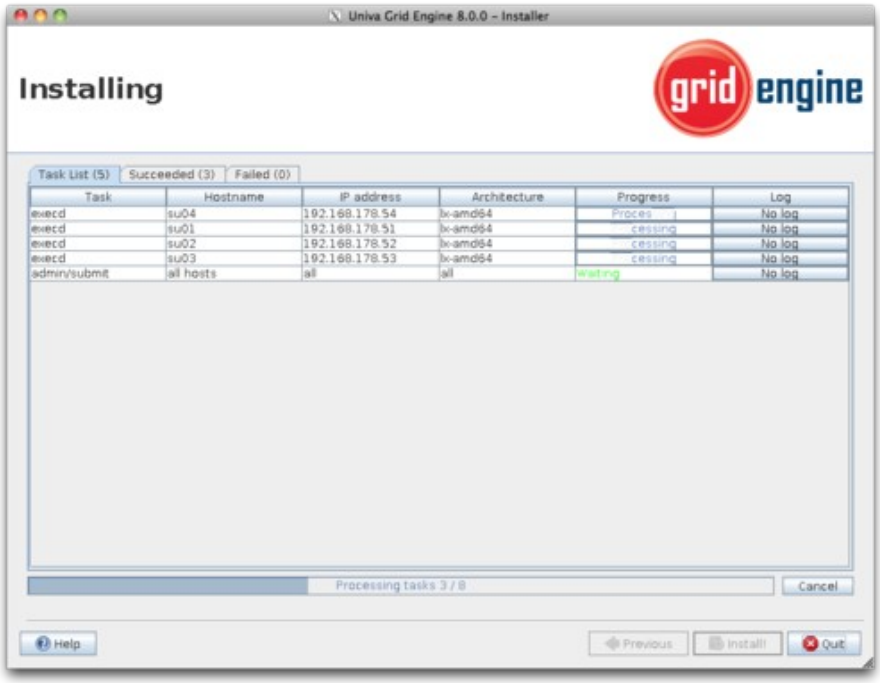
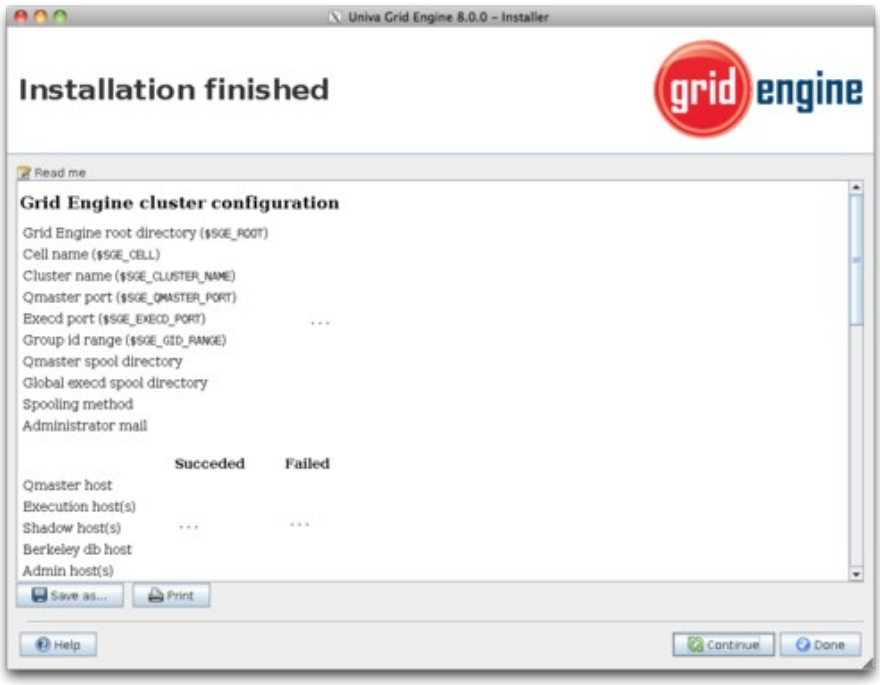


TABLE: Task States

State	Description
Waiting	Task is waiting for execution.
Processing	Task is currently executing.
Success	Task was successfully executed.
Failed	Execution of task failed.
Timeout	Timeout was reached before the task could be completely executed.
Failed due to dependency	Task execution could not be started because dependent tasks were not executed successfully.
Component already exists	Component has already been installed. The Log button will provide more information.
Canceled	Task was interrupted by user intervention.

13. Review the results.



1.1.5 Verifying the Installation

Inbetween 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.

1.1.5.1 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.

1.1.5.2 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)
```

1.1.5.3 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 user's home directory. The names of those files are `<jobname>.e<jobid>` and `<jobname>.o<jobid>`.

1.1.6 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 according to your needs.

1.2 Setting Up the Accounting and Reporting Database

For an introduction to Accounting and Reporting, see The Accounting and Reporting Database.

1.2.1 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 your 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

The attached spreadsheet can be used to roughly calculate the required disk space.

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

1.2.1.1 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.

1.2.1.1.1 PostgreSQL

1.2.1.1.1.1 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
```

1.2.1.1.1.2 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:
```

1.2.1.1.2 MySQL

1.2.1.1.2.1 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
```

1.2.1.1.2.2 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:

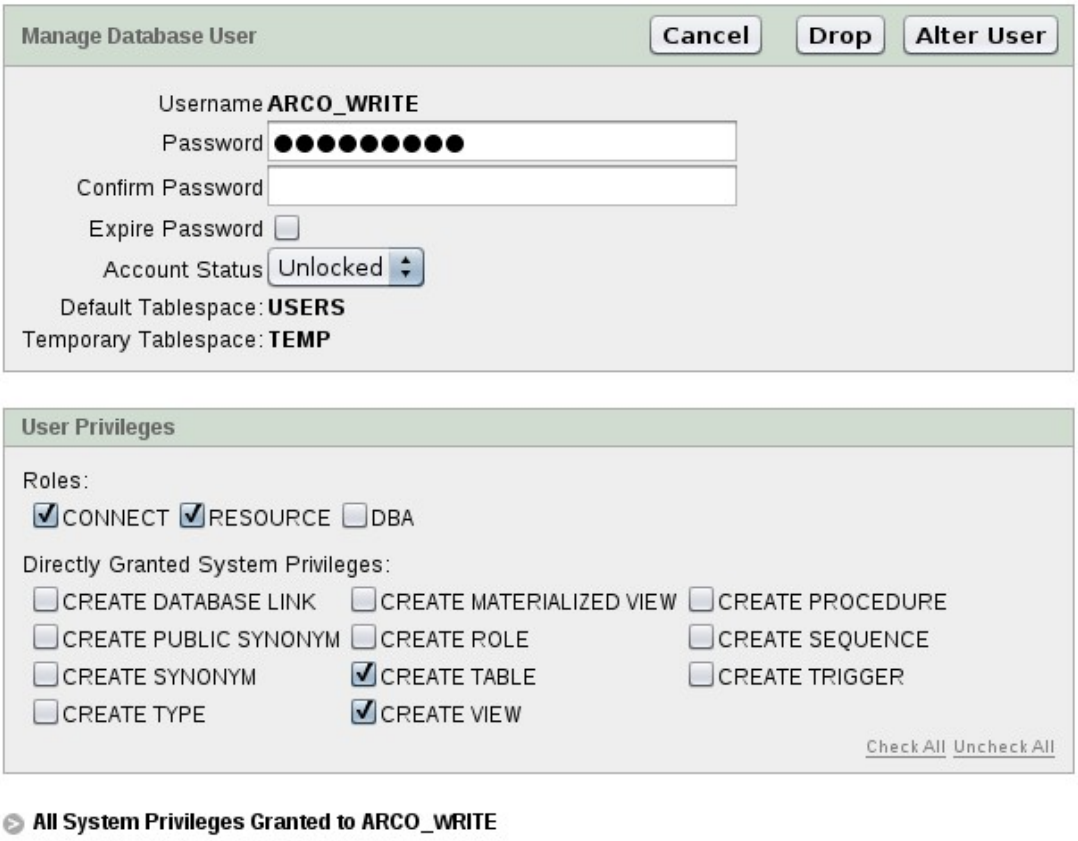
```
mysql> CREATE DATABASE arco;
mysql> GRANT ALL ON arco.* TO arco_write WITH GRANT OPTION;
```

1.2.1.1.3 Oracle

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

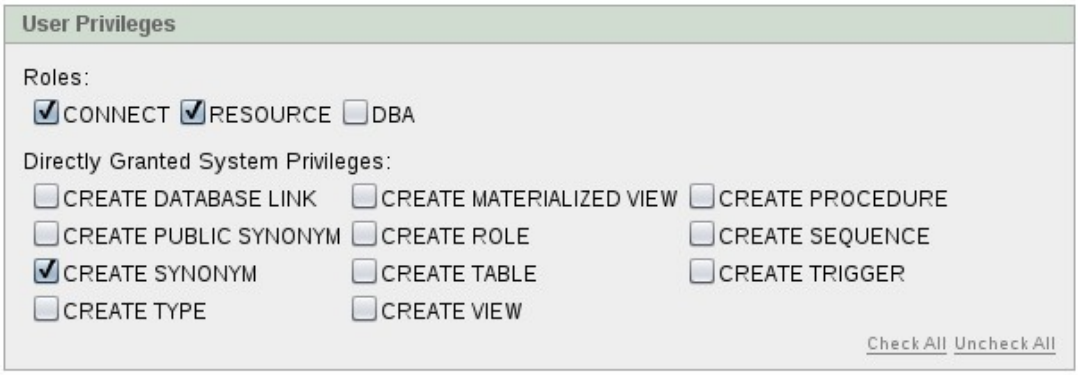
Create the users `arco_write` and `arco_read`.

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.



All System Privileges Granted to ARCO_READ

1.2.2 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`)

1.2.2.1 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-8.0.0alpha-arco.tar.gz
```

1.2.2.2 Install the `dbwriter`.

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

TABLE: `inst_dbwriter` Command Line Options

-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

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

1. Start the `dbwriter` installation.

```
cd <sge_root_directory>
source <sge_cell>/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 you about your Univa Grid Engine installation - the defaults presented should match your 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</code>, 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
     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/co

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/dbwr
starting dbwriter
dbwriter started (pid=19052)
Installation of dbwriter completed
```

1.2.3 Starting and Stopping the `dbwriter`

Start `dbwriter`:

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

Stop `dbwriter`:

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

1.2.4 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.

1.2.4.1 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 Understanding and Modifying the Cluster Configuration.

1.2.4.2 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 Configuring Hosts.

1.2.5 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.

1.2.5.1 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*.

1.2.5.1.1 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

1.2.5.1.2 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 &lt;= '__time_end__' AND
          ju_end_time &gt; '__time_start__' AND
          ju_exit_status != -1 AND
          j_pe_taskid = 'NONE'
    GROUP BY time_start
  </sql>
</derive>
```

1.2.5.2 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.

1.2.5.2.1 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.

1.2.5.2.1.1 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_value
      sge_job, sge_job_log, sge_job_request, sge_job_usage, sge_statistic, sge_statistic_valu
      sge_share_log, sge_ar, sge_ar_attribute, sge_ar_usage, sge_ar_log, sge_ar_resource_usag
      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_v
      (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_
      (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
+---------+---------------------+---------------------+-----------+-----------+
| 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)
```

1.2.5.2.1.2 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
     </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)
```

1.2.5.2.1.3 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="acco
     <sql>
        SELECT account, count(*) AS jobs FROM view_accounting WHERE end_time &lt; now() AND end
     </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 |
+------------+---------------------+---------------------+------------------+-----------+
```

1.2.5.3 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, sge_job_request and sge_job_usage table.
  - *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.
  - *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.).

1.2.5.3.1 Examples

1.2.5.3.1.1 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.

1.2.5.3.1.2 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.

```
<delete scope="job" time_range="year" time_amount="1"/>
<delete scope="job_log" time_range="month" time_amount="1"/>
```

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.

1.2.6 Troubleshooting the `dbwriter`

1.2.6.1 General Problems

1.2.6.1.1 Where do I find the `dbwriter` log file?

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

1.2.6.1.2 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 according to your needs.
- Start up `dbwriter` again.

1.3 Updating 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 reinstalling a new cluster (R), there are two additional ways to get a new Univa Grid Engine cluster when using an old installation of the Open Source Grid Engine, Sun Grid Engine or Oracle 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. The Hot Update (H) makes it possible to migrate an existing cluster including certain running and pending jobs that were already submitted.

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
GE 8.0.0
```

Note that options in parentheses show the recommended way to upgrade the system:

TABLE: Upgrade Matrix

Current Version	Target Version		
	8.0.1	**8.0 FCS**	**6.2u5**
8.0 FCS	C/(H)	-	-
8.0 alpha	-	C/(H)	-
6.2u6, 6.2u7	-	C	-
6.2u5	-	C/(H[1])	-
6.2 FCS, 6.2u1 ... 6.2u4	-	-	C/(H)
6.1 FCS, 6.1u?	-	R	(C)
6.0 FCS, 6.0u?	-	R	(C)
5.3 FCS, 5.3u?	-	(R)	-

[1] not possible if BDB server spooling is used

The following table describes the difference between Hot Update and Cloning a configuration:

TABLE: Differences Between Clone and Hot Update

Clone Configuration	Hot Update
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.
Pending and running jobs are not migrated.	Pending jobs and a certain set of running jobs may remain in a cluster during the upgrade process. What type of jobs are allowed in the cluster depends on the Grid Engine version. See the release notes for more details.
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.

1.3.1 Updating with Two Separate Clusters on the Same Resource Pool (Clone Configuration)

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 should also be applied when deactivating the old cluster step-by-step in order to disable certain resources in the first cluster and to provide them in the second one.

The OPTIONAL tag is used for all update steps that can only be performed if the corresponding functionality (e.g. BDB server, IJS, ARCo, ...) were 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. Back up the existing cluster settings.
    - 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
GE 8.0.0 alpha
```

    - 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, download the backup script `save_sge_config.sh`.
    - OPTIONAL: If downloading a backup script, verify that the script is executable.
    - Run the backup script on the same host where the qmaster process is running. The first argument 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 backup script saves all configuration objects as well as following files:

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

3. REAL-UPGRADE: Drain the cluster. (see Draining the Cluster and Stopping it Sucessively)
4. REAL-UPGRADE: Shut down the existing cluster.
   - OPTIONAL Only for Grid Engine prior to 6.2: also shutdown the scheduler:

```
# qconf -ks
```

   - Shut down the execution daemons and qmaster:

```
# qconf -ke all
# qconf -km
```

5. REAL-UPGRADE and OPTIONAL: Stop the BDB server.
   - Only necessary if the existing cluster used spooling with BDB server.
   - Shut down the BDB server with following command:

```
# $SGE_ROOT/$SGE_CELL/common/sgebdb 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.

- 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`.
- Select the spooling method.
  - The spooling method for the new cluster does not need to match the existing cluster's spooling method.
  - Note that BDB sever spooling is no longer available as of Univa Grid Engine version 8.
- 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 automatic startup options.
- Load the old configuration. Copy the displayed command. 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 PostgressSQL 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. For Windows hosts, create new startup and shutdown scripts for the host or update the Windows helper service. All of these steps can be applied more easily if using passwordless root or rsh access to the execution hosts; ssh is used by default. Also specify the `-rsh` flag when using rsh.
    - 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. OPTIONAL: Upgrade the Windows helper service.

    ⚠ **Warning**
    Only one Windows helper service may run on a Windows host. As a result, Windows hosts that are prepared for Grid Engine 6.2 or Univa Grid Engine 8.0 will not work properly with previous versions of Grid Engine. In this case, either disable the Windows hosts in the original cluster, or skip this upgrade step and remove the Windows hosts from the cloned cluster.

    To perform the upgrade step, do the following:

    - If the Windows administrator user is the same for all windows hosts, then set the environment variable `SGE_WIN_ADMIN` to the name of that user. This will avoid being asked for that name for each host in the next upgrade step.

```
# export SGE_WIN_ADMIN=Administrator
```

    - Perform the Windows helper service upgrade.

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

13. Start the execution daemons.
    - To shutdown certain hosts in the initial cluster and restart them in the cloned cluster, then see Activating Nodes Selectively.
    - To activate all execution nodes in the new cluster execute the following command:

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

1.3.2 Updating Manually by Replacing Parts of an Old Installation (Hot Update)

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 pending and some running jobs should survive the upgrade process. Please consult the Release Notes of the target Grid Engine Installation as well as the Upgrade Matrix to find out if the Hot Update 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. Backup the existing cluster.
   - 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
```

   - Disable all queues to make sure that no pending jobs are started.

```
# qmod -d "*"
```

4. Remove jobs that are not allowed during the upgrade.
    - Depending on the targeted Univa Grid Engine version, it might be necessary to remove certain jobs from the cluster. Not doing so could cause Univa Grid Engine to fail after the upgrade process when new daemons are started.

TABLE: Jobs That Need to be Removed for the Upgrade

Update from -> to	List of NOT allowed jobs
6.2u5 -> 8.0.0	tightly integrated parallel jobs in running state (qrsh -inherit) qmake jobs in running state qlogin/qsh jobs in running state batch jobs using the -sync switch

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

5. Shut down the cluster.
    - Note the biggest job number of the running jobs.
    - Shut down the scheduler. (only for Grid Engine prior 6.2)

```
# qconf -ks
```

    - Shut down execution daemons and qmaster.

```
# qconf -ke all
# qconf -km
```

6. 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.
7. Move applications/directories that contain running applications.
    - Moving some directories out of the way is recommended. They should be moved and not be deleted to ensure that still running jobs can continue.

```
# mv bin bin.old
# mv utilbin utilbin.old
# mv lib lib.old
```

8. Extract new packages to `$SGE_ROOT`.
    - Extract binary packages.
    - Extract common package.
    - (Optional) If enabling ARCo in new cluster, extract the ARCo package.
9. Start up the new components.
    - Start the new qmaster process as user root on the corresponding host.

```
# $SGE_ROOT/default/common/sgemaster
```

    - Next, start the execution nodes by invoking the startup script of the corresponding execution host.

```
# $SGE_ROOT/default/common/sgeexecd
```

- Alternatively, all execution nodes can be started from the qmaster host when password-less ssh or rsh access for the root user is available. To activate all execution nodes in the new cluster, execute following command. Also specify the `-rsh` flag if using rsh.

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

10. Post-installation steps.
    - Enable submission of new jobs by reverting the jsv_url changes from step 3.

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

    - Depending on the initial state setting of the queues, it might be necessary to enable queues again.

```
# qmod -e "*"
```

    - As soon as the job with the id noted in step 5 and all jobs that were previously submitted have finished, the directories moved during step 6 can be removed.

```
# rm -rf bin.old
# rm -rf utilbin.old
# rm -rf lib.old
```

1.4 Troubleshooting the Installation

1.4.1 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` (or in case of BDB spooling, the qmaster spool directory) 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 or BDB spooling path. 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>
```

Go back to the Univa Grid Engine Documentation main page.