Installing Guide (Printable)

The information presented here is also available in individual pages in the Installing section. This page is designed to enable you to easily export the information to a PDF or Word document.

To print a version of this document, log in to wikis.sun.com, click Tools, then select Export to PDF or Export to Word.

Contents

Installing Sun Grid Engine

Installing Guide (Printable)

To effectively install Sun Grid Engine, perform the following tasks in the order that they are listed:

Topic	Description
Release Notes	Learn about product enhancements that have been added since the release.
Planning the Installation	Strategically plan your installation to achieve results that fit your environment.
Loading the Distribution Files on a Workstation	Unpack and load the distribution files onto a workstation.
Installing the Software With the GUI Installer	Learn how to run the new GUI installer and install whole cluster.
Installing the Software From the Command Line	Learn how to run an installation script on the master host and on every execution host in the Grid Engine system and to register information about administration hosts and submit hosts.
Installing Security Features	Set up your system more securely.
Installing the Accounting and Reporting Console	Install the Accounting and Reporting Console, an optional feature that enables you to gather live reporting data from the Grid Engine system.
SDM Installation Overview	Install the Service Domain Manager module, an optional feature that distributes resources between different services according to configurable Service Level Agreements (SLAs).
Verifying the Installation	Verify that the daemon is running on the master host and on the Execution Hosts and how to run simple commands and submit test jobs.

In addition, you might need to perform one or more related tasks:

Topic	Description
Automating the Installation Process	Learn how to automate the Grid Engine installation process.
Installing SMF Services	Learn how to install the Service Management Facility (SMF) services.
Installing a JMX-Enabled System	Learn how to install a JMX-enabled system.
Removing the Software	Learn how to remove the Sun Grid Engine software.
Additional Software for the Microsoft Operating System	Learn how to install Sun Grid Engine on Microsoft Windows operating system.
User Management on Windows Hosts	Learn how to manage user accounts on Windows hosts.
Other Installation Issues	Learn how to identify additional considerations for installing Sun Grid Engine software.

To print this section, see the Installing Guide (Printable).

Loading the Distribution Files on a Workstation

The Sun Grid Engine 6.2 software is distributed on CD-ROM and through electronic download. The CD-ROM distribution contains a directory named `Sun_Grid_Engine_6_2`. The product distribution is in this directory, in both `tar.gz` format and the `pkgadd` format. The `pkgadd` format is provided for the Solaris Operating System (Solaris OS). For all supported operating systems, the software is distributed in `tar.gz` format. For more on how to obtain the distribution files, see Getting the Software.

How to Load the Distribution Files on a Workstation

Ensure that the file systems and directories that are to contain the Grid Engine software distribution and the spool and configuration files are set up properly by setting the access permissions as defined in File Access Permissions.

Steps

1. Provide access to the distribution media.
   If you downloaded the software, rather than getting it on CD-ROM, just unzip the files into a directory. This directory must be located on a file system that has at least 350 MBytes free space.

2. Log in to a system.
   Log in preferably on a system that has a direct connection to a file server.

3. Create the installation directory.
   Create an installation directory as described in $SGE_ROOT Installation Directory.

   ```
   # mkdir /opt/sge6-2
   ```

   In these instructions, the installation directory is abbreviated as `sge-root`.

4. Install the binaries for all binary architectures that are to be used by any of your master, execution, and submit hosts in your Grid Engine system cluster.
   You can use either the pkgadd method or the tar method.

pkgadd Method

The `pkgadd` format is provided for the Solaris Operating System. To facilitate remote installation, the `pkgadd` directories are also provided in zip files.

You can install the following packages:

Package	Description
SUNWsgeec	Architecture independent files
SUNWsgeex	Solaris (SPARC platform) 64-bit binaries for Solaris 8, Solaris 9, and Solaris 10 Operating Systems
SUNWsgeei	Solaris (x86 platform) binaries for Solaris 8, Solaris 9, and Solaris 10 Operating Systems
SUNWsgeeax	Solaris (x64 platform) binaries for Solaris 10 Operating System
SUNWsgeea	Accounting and Reporting Console (ARCo) packages for the Solaris and Linux Operating systems.

As you type the following commands, you must be prepared to respond to script questions about your base directory, `sge-root`, and the administrative user. The script requests the choices that you made during the planning steps of this installation. See Decisions That You Must Make for further details.

At the command prompt, type the following commands, responding to the script questions.

```
# cd cdrom_mount_point/Sun_Grid_Engine_6_2
# pkgadd -d ./Common/Packages SUNWsgeec
```

Depending on the Solaris binary that you need, type one of the following commands:

```
# pkgadd -d ./Solaris_sparc/Packages SUNWsgee
# pkgadd -d ./Solaris_sparc/Packages SUNWsgeex
# pkgadd -d ./Solaris_x86/Packages SUNWsgeei
# pkgadd -d ./Solaris_x64/Packages SUNWsgeeax
```

tar Method

For all supported operating systems, the software is distributed in tar.gz format.

The following table contains files that you need to install, regardless of platform.

File	Description
Common/tar/sge-6_2-common.tar.gz	Architecture independent files

The tar files that contain platform-specific binaries use the naming convention of sge-6_2-bin-architecture.tar.gz.

The following table lists the platform-specific binaries. Install the file for each platform that you need to support. Note that each platform has its own directory under Sun_Grid_Engine_6_2.

Platform-Specific File	Platform
Solaris_sparc/tar/sge-6_2-bin-solaris-sparcv9.tar.gz	Solaris (SPARC platform) 64-bit binaries for Solaris 8, Solaris 9, and Solaris 10 Operating Systems
Solaris_x86/tar/sge-6_2-bin-solaris-i586.tar.gz	Solaris (x86 platform) binaries for Solaris 8, Solaris 9, and Solaris 10 Operating Systems
Solaris_x64/tar/sge-6_2-bin-solaris-x64.tar.gz	Solaris (x64 platform) 64-bit binaries for Solaris 10
Windows/tar/sge-6_2-bin-windows-x86.tar.gz	Microsoft Windows (x86 platform) 32-bit binaries for Windows 2000, XP and Windows Server 2003
Linux24_i586/tar/sge-6_2-bin-linux24-i586.tar.gz	Linux (x86 platform) binaries for the 2.4 and 2.6 kernel
Linux24_amd64/tar/sge-6_2-bin-linux24-ia64.tar.gz	Linux (Itanium platform) binaries for the 2.4 and 2.6 kernel
Linux24_amd64/tar/sge-6_2-bin-linux24-x64.tar.gz	Linux binaries for the 2.4 and 2.6 kernel
MacOSX/tar/sge-6_2-bin-darwin-ppc.tar.gz	Apple Mac OS/X (PowerPC platform)
MacOSX/tar/sge-6_2-bin-darwin-x64.tar.gz	Apple Mac OS/X (Intel-based platform)
HPUX11/tar/sge-6_2-bin-hp11.tar.gz	Hewlett-Packard HP-UX 11 or higher
HPUX11/tar/sge-6_2-bin-hp11-64.tar.gz	64-bit binaries for Hewlett-Packard HP-UX 11 or higher
Aix43/tar/n1ge-6_1-bin-aix51.tar.gz	IBM AIX 5.1 and 5.3

Type the following commands at the command prompt. In the example, <basedir> is the abbreviation for the full directory, cdrom-mount-point /Sun_Grid_Engine_6_2.

```
% su
# cd <sge-root>
# gzip -dc <basedir>/Common/tar/sge-6_2-common.tar.gz | tar xvpf -
# gzip -dc <basedir>/Solaris_sparc/tar/sge-6_2-bin-solsparc32.tar.gz | tar xvpf -
# gzip -dc <basedir>/Solaris_sparc/tar/sge-6_2-bin-solsparc64.tar.gz | tar xvpf -
# SGE_ROOT=<sge-root>; export SGE_ROOT
# util/setfileperm.sh $SGE_ROOT
```

Installing the Software With the GUI Installer

A new GUI installer to simplify the installation process is available since Sun Grid Engine 6.2u2. The GUI installer enables you to easily install a whole cluster interactively. To install a cluster, you need to set up the environment in a similar way to an automatic installation.

Requirements

- The GUI installer requires at least Version 5 of the Java™ platform.
- Screen resolution of 1024x768 or larger.
- (Optional) Password-less `ssh` or `rsh` access as `root` user to all remote hosts that you want to install. If this requirement is not met you can only install Grid Engine components on a local host. For more information, see How to Configure Password-less Access for the root User. You can still use the GUI installer by starting it locally from each remote host.

✓ Recommendations

1. Start the installer as `root` user.
2. Ensure that you start the installation from the qmaster host when password-less `root` access is available.

For information on installation modes supported by the GUI installer, see these topics:

Topic	Description
Express Installation	Enables first-time users to install the software easily. Provides a significantly reduced set of parameters that need to be configured. Requires password-less `ssh` access as `root` user to all remote hosts that you want to install.
Custom Installation	Enables you to configure almost all existing options that are available during the command-line installation. Offers more advanced features for the cluster host selection. Requires password-less `ssh` or `rsh` access as `root` user to all remote hosts that you want to install.

For additional reference information, see these topics:

Topic	Description
How to Configure Password-less Access for the root User	Procedure for configuring a password-less `ssh` or `rsh` access for the root user to install a whole Sun Grid Engine (SGE) cluster by using the GUI Installer.
Understanding Host and Installation States	Describes the different installation states that you might encounter while using the GUI installer.
Tweaking start_gui_installer	Describes the command-line options of the `start_gui_installer` command and how to use them to fine tune the performance of the installer.
Troubleshooting the GUI Installer	Contains known issues and their workarounds.

Express Installation

The express installation mode is targeted at first-time users and provides a significantly reduced set of parameters to configure. This mode also provides reasonable default values for most of the parameters. You must have a password-less `ssh` or `scp` access if you are planning to install Sun Grid Engine on remote hosts. The following steps describe a complete cluster installation and assume that the password-less access is configured. (Click any of the screen captures in the following steps to view more details.)

Using the Express Installation Mode

The express installation steps are as follows.

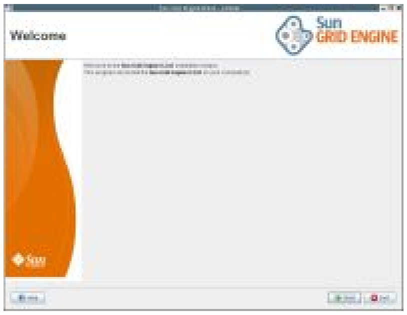
Steps

1. Start the GUI installer. On the welcome screen, click Next.

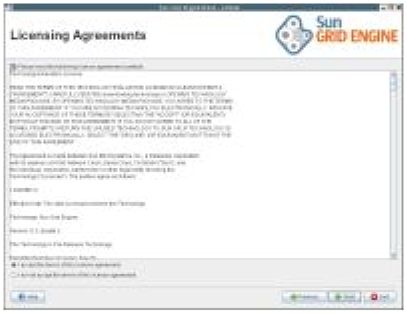
   > Note
   >
   > Ensure that you start the GUI installer on the qmaster host.

   As `root`, run the `start_gui_installer` command in your `sge-root` directory. For example:

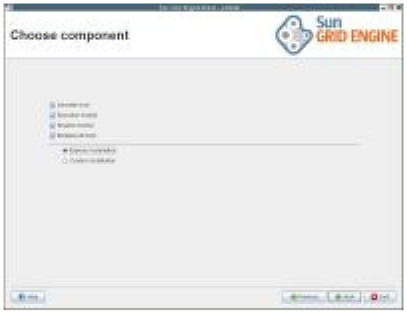
   ```
   qmaster:/sge# ./start_gui_installer
   Starting Installer ...
   ```

2. Agree to the terms of the license. Click Next.

3. Choose components to install. Click Next.

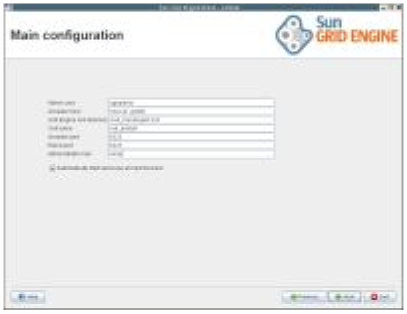
   

   See the following table for a brief explanation of options displayed on this screen.

Host type	Description
Qmaster host	Main component in Sun Grid Engine software. You must install exactly one qmaster component per Sun Grid Engine cluster installation.
Execution host(s)	Hosts that execute the tasks (jobs).
Shadow host(s)	Hosts that provide a high availability feature to the cluster. In case the qmaster fails (for example, due to a crash or network issue), one of the shadow hosts takes over the qmaster responsibility.
Berkeley db host	Host that implies a Berkeley db host spooling option. Sun Grid Engine then spools data to a remote server. Not recommended as the default option.

If you are not sure what you want to install, keep the components selected by default.

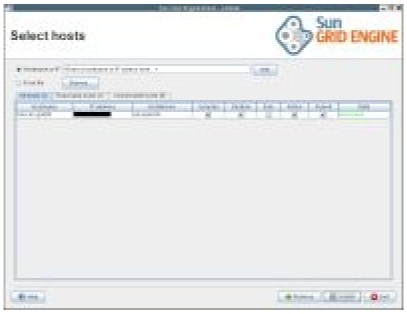
4. Modify the main configuration details. Click Next.



Option	Description
Admin user	Sun Grid Engine processes will be executed under this user name, and certain directories will be owned by this user.
Qmaster host	Host that will run qmaster daemon (main component). It can be changed later in the host selection.
Grid Engine root directory	Directory where you unpacked Sun Grid Engine `tar.gz` archive or installed a package (for example, `rpm`, `pkg`). It must not contain an automounter prefix.
Cell name	Name of this Sun Grid Engine cell, a value that identifies an instance of Sun Grid Engine when several instances run simultaneously.
Cluster name	Name of this Sun Grid Engine instance used by SMF on Solaris machines. In express installation mode, this instance is hidden and has a default value of p6444. The following naming restrictions apply to this field: The cluster name must start with a letter ([A-Za-z]), followed by letters, digits ([0-9]), dashes ("-"), or underscores ("_").
Qmaster port	Port that will be used by the qmaster daemon. Default value is 6444.
Execd port	Port that will be used by the execution daemon. Default value is 6445.
Administrator mail	Email address used by Sun Grid Engine to report issues to the grid administrator. Default value is none (no emails will be sent).
Automatically start service(s) at machine boot	Component (service) will be automatically started at machine boot. By default, this is selected.

Typically, one would provide a valid administrator email and click next.

5. Select hosts to be installed and fix reported problems. Click Install to start the installation on the reachable hosts.



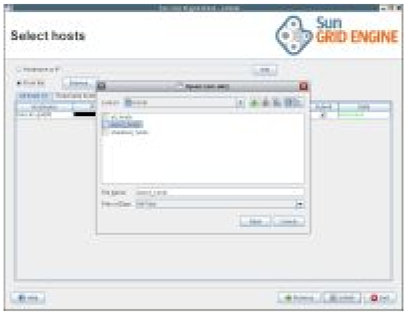
This screen allows you to select the hosts and components that you would like to install. Express installation mode has a slightly simplified selection model. Custom installation mode enables you to change the components that will be selected once new hosts are added. The qmaster host is added based on the qmaster host value from the main configuration screen by default. You can select the hosts in one of two different ways:

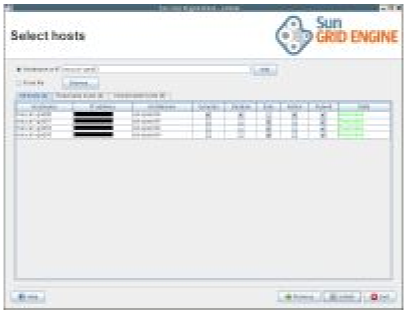
1. By a host name, host name pattern, or by an IP address or IP address pattern
2. From a file that you create using the installer's save action

The patterns do not support regular expressions. The supported expressions are lists and numeric ranges. For more information, see the following table:

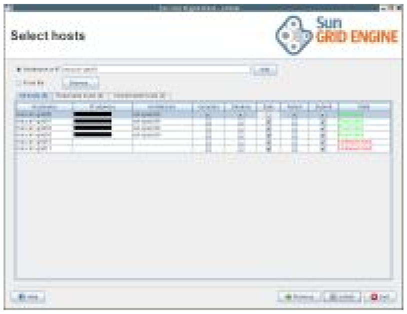
Description	Input	Resolved Value
Host name	`grid00`	`grid00`
IP address	`192.168.0.1`	`192.168.0.1`
List of hosts	`grid00 grid01 grid03`	`grid00 grid01 grid03`
List of IP addresses	`192.168.0.1 192.168.0.2 192.168.0.5`	`192.168.0.1 192.168.0.2 192.168.0.5`
Host ranges	`grid[00-03]`	`grid00, grid01, ..., grid03`
Range of IP addresses	`192.[168-169].0.[50-60]`	`192.168.0.50 ... 192.168.0.60, 192.169.0.50 ... 192.169.0.60`

In the following screen sequence, hosts `grid01` to `grid03` are added as execution and hosts `grid00` to `grid03` submit hosts. However, two hosts `grid10` to `grid11` have errors. See Understanding Host and Installation States for a complete list of errors and possible solutions. Note that each state has a tooltip that displays a better error message. Once the errors are resolved on the problematic hosts, select hosts that you want to verify and right-click. A pop-up menu enables you to refresh selected hosts. Optionally, invalid hosts can be removed. Once the states have been refreshed, a different error state or reachable state will be displayed.



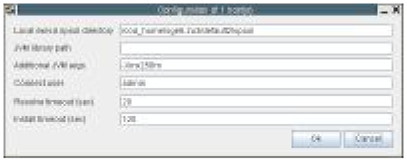


[redacted]



[redacted]

6. (Optional) Modify the host configuration. Click OK.
   Select a host in the Select hosts screen, right-click on the host and click Configure to modify the host configuration.

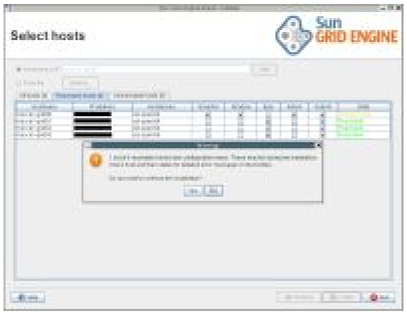


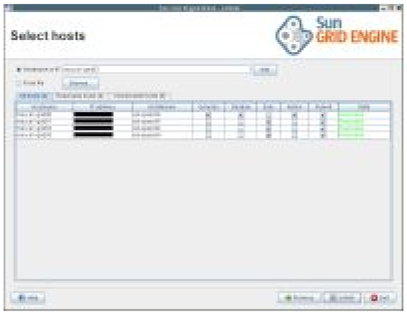
Option	Description
Local execd spool directory	Directory for local execd spooling data.
JVM library path	Path to the JVM library on the qmaster and/or shadow hosts.
Additional JVM args	Additional arguments to be used when starting the JVM in qmaster.
Connect user	The user which will be used to connect to the remote host using ssh or scp.
Resolve timeout(sec)	Timeout value for any resolving task.
Install timeout (sec)	Timeout value for any installation task.

7. (Optional) Fix problems reported during pre-install validation, then click Install.
   When you click the Install button as described in Step 5, the installation does not start immediately. First, the installer executes a series of advanced checks for each host to verify that there is no misconfiguration. If the validation fails, host states are updated and you are presented with an option to return to the host selection or to continue with the installation.

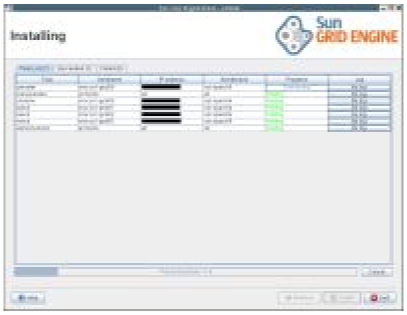
> Note
>
> Continuing the installation after the installer reports errors will likely result in a failed installation. Before restarting the installation, you should return to the host selection and either resolve the reported problems or remove the hosts that have configuration errors.

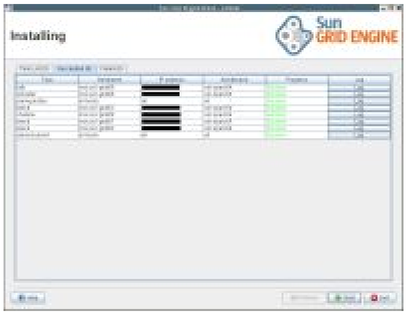
In the following screen sequence, one host has a configuration error. See Understanding Host and Installation States for a complete list of errors and possible solutions. Notice that each state includes a tooltip that displays an error message.



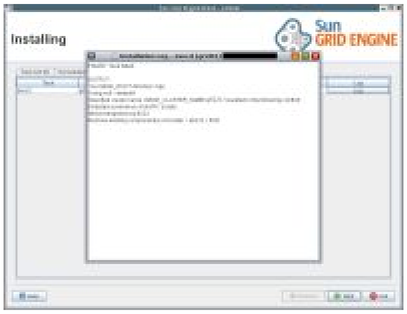


8. Monitor the progress of the installation, then click Next.



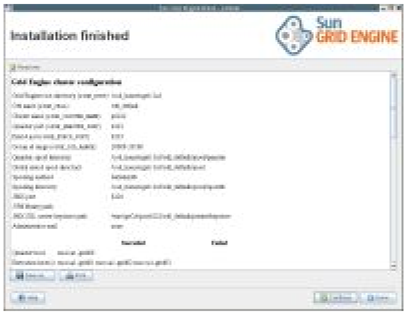


If there were any failures during the installation, the Failed tab is selected. See Understanding Host and Installation States for a complete list of installation states. Click the Log button for each failed installation for more information.



This error is displayed because the cluster name p6222 already exists on this host (installation was not attempted).

9. Review the overview information, then click Done.



Optionally, print or save the information about the Sun Grid Engine configuration for future reference. The page is also automatically saved to the `$SGE_ROOT/$SGE_CELL/Readme_TIMESTAMP.html` file. If the page could not be saved there, due to `root` being mapped to `nobody` on NFS shared file system, it is saved to `/tmp/Readme_TIMESTAMP.html`.
To verify the installation, go to Verifying the Installation.

Custom Installation

The custom installation mode is targeted at the experienced users. It offers more advanced customization of Sun Grid Engine installation than the express installation. It provides default values for most of the parameters. You must have a password-less `ssh` or `rsh` access if planning to install Sun Grid Engine on remote hosts. The following steps assume that the password-less access is configured and describe a cluster installation consisting of:

- Qmaster host with JMX feature enabled
- Three execution hosts on various architectures
- One shadow host
- One administrative host
- Four submit hosts

Using the Custom Installation Mode

The custom installation steps are as follows.

Steps

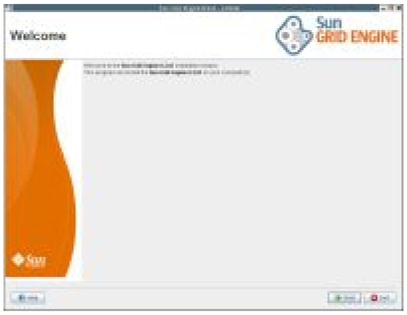
1. Start the GUI Installer. On the welcome screen, click Next.

Note

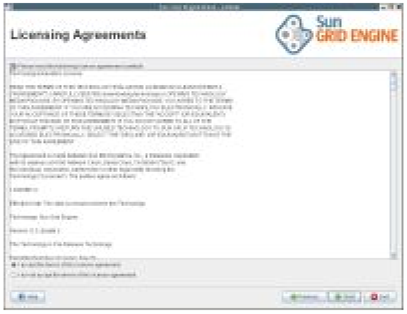
Ensure that you start the GUI installer on the qmaster host.

As root, run the start_gui_installer command in your sge-root directory. For example:

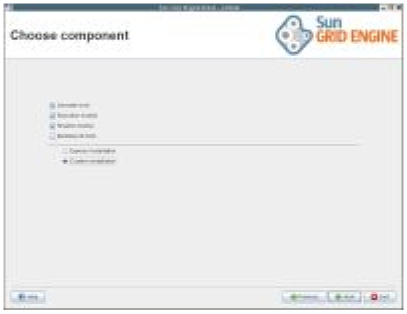
```
qmaster:/sge# ./start_gui_installer
Starting Installer ...
```



2. Agree to the terms of the license. Click Next.



3. Choose components to install, including a shadow host and the custom installation option, and click Next.

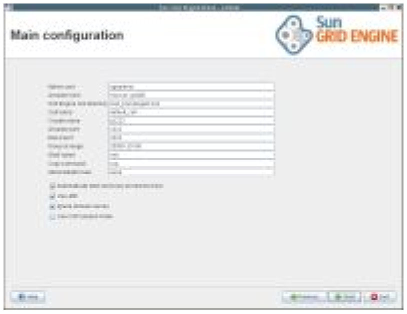


See the following table for a brief explanation of options displayed on this screen.

Host type	Description
Qmaster host	Main component in Sun Grid Engine software. Exactly one qmaster component must be installed per Sun Grid Engine cluster installation.
Execution host(s)	Hosts that execute the tasks (jobs).
Shadow host(s)	Shadow hosts provide a high availability feature to the cluster. In case that the qmaster fails (crash, network issue), one of the shadow hosts will take over the qmaster responsibility.

Berkeley db host	Selecting it implies a Berkeley db host spooling option. The Sun Grid Engine then spools data to a remote server. Not recommended as default option.

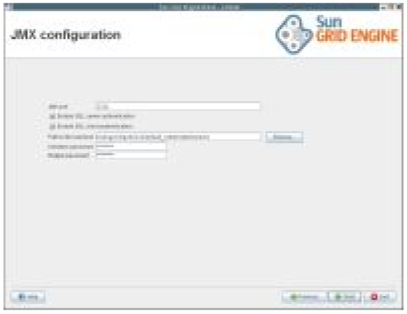
4. Modify the main configuration details. Click Next.



Option	Description
Admin user	Sun Grid Engine processes will be executed under this user name, and certain directories will be owned by this user.
Qmaster host	Host that will run qmaster daemon (main component). It can be changed later in the host selection.
Grid Engine root directory	Directory where you unpacked the Sun Grid Engine `tar.gz` archive or installed a package (for example, `rpm`, `pkg`). It must not contain an automounter prefix.
Cell name	Name of this Sun Grid Engine cell, a value that identifies an instance of a Sun Grid Engine when several instances run simultaneously.
Cluster name	Name of this Sun Grid Engine instance used by SMF on Solaris machines. The following naming restrictions apply to this field: The cluster name must start with a letter ([A-Za-z]), followed by letters, digits ([0-9]), dashes ("-"), or underscores ("_").
Qmaster port	Port that will be used by the qmaster daemon. Default value is 6444.
Execd port	Port that will be used by the execution daemon. Default value is 6445.
Group id range	Range of additional group IDs. The group IDs in this range must not be used anywhere else. The size of the range determines how many concurrent jobs can run in Sun Grid Engine. Choose a large value.
Shell name	Shell to be used while connecting to remote hosts (with `ssh` or `rsh` syntax). Expected values for this field are `ssh` or `rsh`.
Copy command	Command to be used while copying files to remote hosts (with `scp` or `rcp` syntax). Expected values for this field are `scp` or `rcp`.
Administrator mail	Email address used by the Sun Grid Engine to report issues to the grid administrator. Default value is none (no emails will be sent).
Automatically start service(s) at machine boot	Component (service) will be automatically started at machine boot. By default, this is selected.
Use JMX	Triggers installation of a JVM thread in qmaster. Currently only needed when you plan to install Service Domain Manager or SGE Inspect. By default, this is selected.
Ignore domain names	Sun Grid Engine will ignore domain names when comparing host names. By default, this is selected.
Use CSP product mode	Sun Grid Engine will be installed with certificate security protocol (CSP). Communication between Sun Grid Engine daemons will be protected by an SSL certificate. Has impact on cluster throughput. By default, this is not selected.

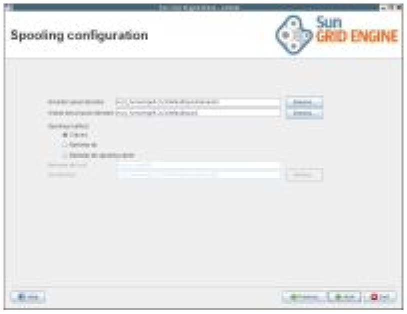
Typically, one would customize the default values and click Next.

5. Modify the JMX configuration details. Click Next.



Option	Description
JMX port	Port number to be used by JVM thread in qmaster process.
Enable SSL server authentication	Once enabled, SSL certificate configuration will be presented later. The server certificate will be used for authentication and encryption.
Enable SSL client authentication	Client authentication will be used.
Path to the keystore	Path to Java keystore file that will be created during the qmaster installation.
Keystore password	Keystore password. Default value is `changeit`.
Retype password	Password to retype. Default value is `changeit`.

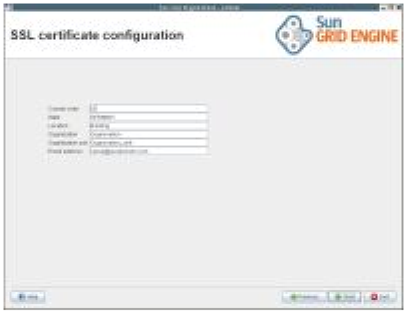
6. Modify the spooling configuration. Click Next.



Option	Description
Qmaster spool directory	Directory for qmaster spooling data.
Global execd spool directory	Directory for execution daemon spooling directory used by default for all execution hosts. Unless overridden in the host selection screen, each execution host creates a subdirectory in the global `execd` spool directory.
Classic spooling method	Spooling is done in human readable format.
Berkeley db spooling method	Spooling is done to local Berkley db.
Berkeley db spooling server spooling method	Spooling is done to Berkley db server.

Berkeley db host	Host for Berkeley db server, enabled only when Berkeley db spooling server method is selected.
Db directory	Berkeley db spooling directory, either on local host or Berkeley db host, if Berkeley db spooling server method is selected.

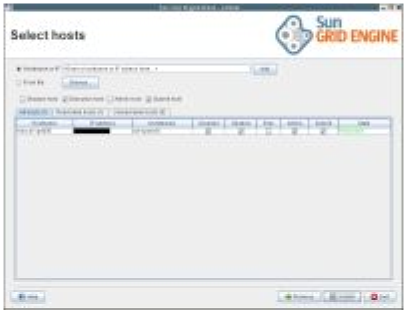
7. (Optional) Provide SSL certificate information. Click Next.



This screen is displayed only when you have previously selected the JMX or CSP features. An SSL certificate will be generated as part of qmaster installation. This certificate will then be used throughout the Sun Grid Engine.

Option	Description
Country code	Two-character country code. Default value is DE.
State	State. Default value is GERMANY.
Location	Location. Default value is `Building`.
Organization	Organization. Default value is `Organisation`.
Organization unit	Organization unit. Default value is `Organisation_unit`.
Email address	Email address. Default value is `name@yourdomain.com`.

8. Select hosts to be installed and fix reported problems. Click Install to start the installation on the reachable hosts.

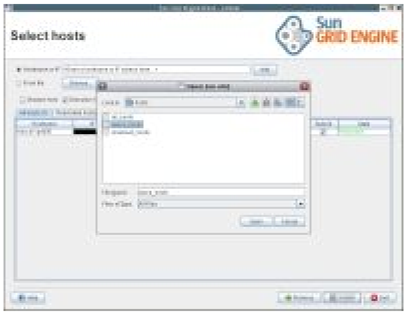


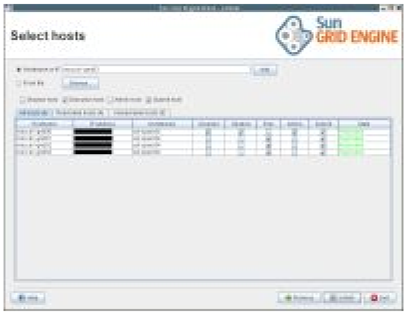
This screen allows you to select the hosts and components that you would like to install. The qmaster host is added based on the qmaster host value from the main configuration screen by default. You can select the hosts in one of two different ways:
1. By a host name, host name pattern, or by an IP address or IP address pattern
2. From a file that you create using the installer's save action
The patterns do not support regular expressions. The supported expressions are lists and numeric ranges. For more information, see the following table:

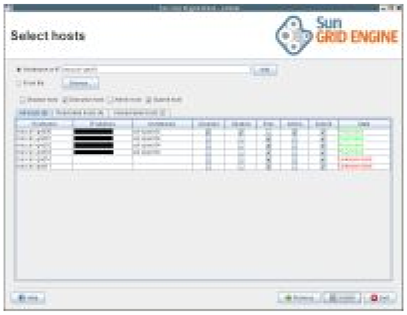
Description	Input	Resolved Value
Host name	`grid00`	`grid00`
IP address	`192.168.0.1`	`192.168.0.1`

List of hosts	`grid00 grid01 grid05`	`grid00 grid01 grid05`
List of IP addresses	`192.168.0.1 192.168.0.2 192.168.0.5`	`192.168.0.1 192.168.0.2 192.168.0.5`
Host ranges	`grid[00-10]`	`grid00, grid01, ..., grid10`
Range of IP addresses	`192.[168-169].0.[50-60]`	`192.168.0.50 ... 192.168.0.60, 192.169.0.50 ... 192.169.0.60`

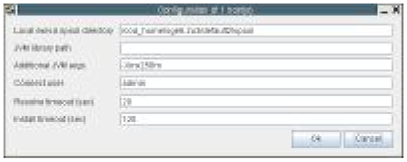
In the following screen sequence, five execution and six submit hosts are added from a file. Two hosts have errors; they are unreachable. See Understanding Host and Installation States for a complete list of errors and possible solutions. Note that each state has a tooltip that displays a better error message. Hosts can be refreshed or removed using a context menu. In addition, two hosts are added as shadow and administrative hosts. Before actually adding the hosts by clicking the Add button, the default component selection must be changed from execution and submit host to shadow and admin host.







9. (Optional) Modify the host configuration. Click OK.
   Select a host in the Select hosts screen, right-click on the host and click Configure to modify the host configuration.

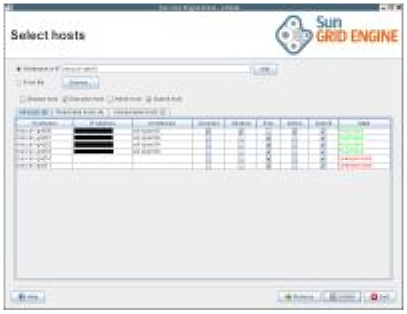


Configuration of 1 host(s)

Local execd spool directory	[illegible]
JVM library path	
Additional JVM args	[illegible]
Connect user	admin
Resolve timeout (sec)	20
Install timeout (sec)	120

Option	Description
Local execd spool directory	Directory for local execd spooling data.
JVM library path	Path to the JVM library on the qmaster and/or shadow hosts.
Additional JVM args	Additional arguments to be used when starting the JVM in qmaster.
Connect user	The user which will be used to connect to the remote host using ssh or scp.
Resolve timeout(sec)	Timeout value for any resolving task.
Install timeout (sec)	Timeout value for any installation task.

10. (Optional) Fix problems reported during pre-install validation. Click Install.



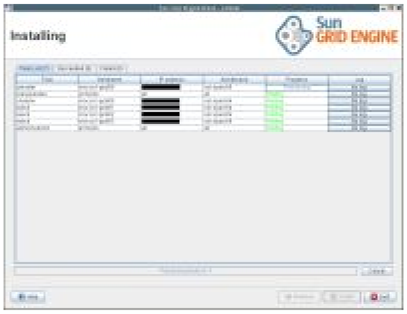
[redacted]

When you click the Install button as described in Step 8, the installation does not started immediately. First, the installer executes a series of advanced checks for each host to verify that there is no misconfiguration. If the validation fails, host states are updated and you may return to the host selection or continue with the installation.

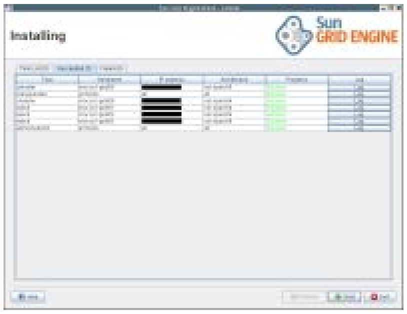
> Note
>
> Continuing the installation after the installer reports errors will likely result in a failed installation. Before restarting the installation, you should return to the host selection and either resolve the reported problems or remove the hosts that have configuration errors.

See Understanding Host and Installation States for a complete list of errors and possible solutions. An example of a pre-install validation with errors can be found in Express Installation.

11. Monitor the progress of the installation, then click Next.

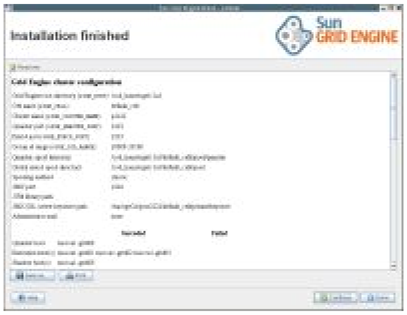


[redacted]



If there were any failures during the installation, see Understanding Host and Installation States for a complete list of installation states. Click the Log for each failed installation for more information as shown in this example.

12. Review the overview information, then click Done.



Optionally, print or save the information about the Sun Grid Engine configuration for future reference. The page is also automatically saved to the `$SGE_ROOT/$SGE_CELL/Readme_TIMESTAMP.html` file. If the page could not be saved there, due to root being mapped to nobody on NFS shared file system, it is saved to `/tmp/Readme_TIMESTAMP.html`.
To repeat the installation or to install more hosts, click Continue.

To verify the installation, go to Verifying the Installation.

How to Configure Password-less Access for the `root` User

This section describes how to set up a password-less `ssh` or `rsh` access for the `root` user to install a whole Sun Grid Engine cluster at once by using the GUI Installer. The Sun Grid Engine installation must be started on the qmaster host, so you need to first decide which host is going to be the qmaster host. The following instructions use qmaster as the qmaster host name. You must replace qmaster with your qmaster host name.

✓ Recommendation
You can skip this procedure if you plan to install Sun Grid Engine only on a local host.

⚠ Warning
Enabling `root` login without a password can be a security risk!
The commands and configuration files used in the following procedure are applicable only to the Solaris 10 operating system. You can substitute these with commands and configuration files that are appropriate for your operating system.

ℹ Note
Installing Grid Engine cluster with CSP option may additionally require password-less access to the localhost (qmaster host to the qmaster host).

Configuring Password-less **ssh** Access for the **root** User

1. Enable **root** login.
   For security reasons, using ssh as root is disabled on many platforms by default. Perform the following for each host on which you will log in using password-less ssh as the root user:
   a. As root, open the /etc/ssh/sshd_config file.
   b. Modify PermitRootLogin no to PermitRootLogin yes.
2. Restart **ssh** service on all remote hosts.
   As root type the following command.

```
svcadm disable -st ssh ; svcadm enable ssh
```

3. Generate a certificate on the qmaster host.
   As root, type the following command to generate the RSA key on the qmaster host. You should leave the passphrase empty.

```
# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
ec:fa:48:55:c4:3d:59:40:a6:27:10:a2:90:11:de:dc root@qmaster
```

4. Copy the certificate to all remote hosts.
   Copy the generated public key contained in a id_rsa.pub file to every remote host that should accept root login without a password from this host.
   The following example enables root access to host grid05 from host qmaster.

```
qmaster# cat /root/.ssh/id_rsa.pub
ssh-rsa
ACCCB3NzaC1yc2EBBBBBIwAAAIEA1xfRiZMV7xt8EMDollLQH5RTAVz3lIXkr/FTfcbwjuMa0t/PdO9gBnJY03e1mIIpjDPiqT
root@qmaster
qmaster# ssh grid05
he authenticity of host 'grid05 (192.168.1.5)' can't be established.
RSA key fingerprint is ec:fa:48:55:c4:3d:59:40:a6:27:10:a2:90:11:de:dc.
Are you sure you want to continue connecting (yes/no)? yes
Password:
grid05# mkdir -p ~/.ssh
grid05# echo "ssh-rsa
ACCCB3NzaC1yc2EBBBBBIwAAAIEA1xfRiZMV7xt8EMDollLQH5RTAVz3lIXkr/FTfcbwjuMa0t/PdO9gBnJY03e1mIIpjDPiqT
root@qmaster" >> ~/.ssh/authorized_keys
```

5. Verify if you are able to connect to the hosts as **root** without a password.
   As root, type the following command.

```
ssh <remote_password-less_host>
```

   If you are able to connect to the hosts without being prompted, password-less access to the hosts has been set up. Now, you can invoke the GUI installer using the start_gui_installer command from your sge-root directory.

Configuring Password-less **rsh** Access for the **root** User

1. Enable **root** login.
   Normally, the root user can only log in to the console /dev/console. You can remove this restriction by performing the following.
   a. Open the /etc/default/login file.
   b. Comment out the CONSOLE=/dev/console line by inserting a # character at the beginning of the line.
      You need to perform this for each remote host you would like to log in to.
2. Set up access without a password.
   a. Create a .rhosts file.
   b. Add a single line that contains the qmaster's host name optionally followed by a + sign.
      For example, if foo is the qmaster's host name, add the line foo + or simply foo to the .rhosts file.
   c. Copy this file to the root user's home directory on each of the remote hosts where you wish to install Sun Grid Engine.
      This will allow root to log in from the qmaster host without a password to any machine that will be part of the cluster.
3. Restart **rlogin** service on all remote hosts.
   As root, type the following command.

```
svcadm disable -st rlogin ; svcadm enable rlogin
```

4. Verify if you are able to connect to the hosts as **root** without a password.
   As root, type the following command.

```
rlogin <remote_password-less_host>
```

   If you are able to connect to the hosts without being prompted, password-less access to the hosts has been set up. Now, you can invoke the GUI installer using the start_gui_installer command from your sge-root directory. Choose the custom installation mode and replace ssh with rsh and scp with rcp in the Main configuration panel.

Understanding Host and Installation States

This section lists the different installation states that you might encounter while using the GUI installer. The installation states can be divided into the following three categories.

Host Resolving

When a new host is added in the Select hosts screen, the host name State field is immediately set to New unknown host and host name resolving process is initiated. The host name is marked as Reachable only if the architecture of the host can be retrieved. All the other states specify an error. The GUI installer cannot perform any installation on such a host. The following table lists all possible states.

State	Description
New unknown host	Initial state. When the host name is added, the GUI installer immediately starts resolving the host name or IP address of the host, if there are available threads in the resolve pool.
Resolving	Temporary state. The host is being resolved based on the host name or IP address by using the default name service.
Unknown host	Final state. The host cannot be resolved by the name service.
Resolvable	Temporary state. After the host is resolved, the GUI installer immediately tries to retrieve the host's architecture, if there are available threads in the resolve pool.

Contacting	Temporary state. The host has been resolved and the host's architecture is being retrieved.
Missing remote file	Final state. Missing file '$SGE_ROOT/util/arch' on remote host. Is the sge-root path the same for the remote host and the local host? If not, fix the path or refer Using Path Aliasing.
Reachable	Final state. The host architecture cannot be retrieved. Password-less `ssh` or `rsh` access to remote hosts is working properly.
Unreachable	Final state. The host architecture cannot be retrieved. Password-less `ssh` or `rsh` access to remote hosts is not working properly. See How to Configure Password-less Access for more information.
Canceled	Final state. The user has canceled the host resolving process.

Host Validating

After the hosts have been resolved and their architecture has been retrieved, they are moved to the `Reachable` tab in the Select hosts screen. You can install Sun Grid Engine on a host that is in the Reachable state. While clicking the `Install` button, the GUI installer first invokes additional remote host validation. If the installer discovers any configuration errors (see RED and ORANGE states in the list below), the installation is not initiated and the appropriate error message is displayed. You can return to the Select hosts screen and proceed with the installation if you wish.

State	Description	Problem Resolution
Copy timeout	Timeout occurred when copying `check_host` or `install_component` files. See tooltip for the exact file name.	Try again (press `Install` button one more time). If timeout reoccurs, save your host list to a file, stop the installer and restart it with increased timeout values. See tweaking start_gui_installer.
Copy failed	Copying files `check_host` or `install_component` to the remote host failed. See tooltip for the exact file name.	Try again (press `Install` button one more time). If problems reoccurs try to copy a any file with `scp` or `rcp` to verify these commands work properly. If not make sure they do before new installation attempt.
Permission denied	Either of Berkeley DB, qmaster, execution daemon spool directory or JMX keystore file is not writable. See tooltip for the exact message. Installation will most likely fail, if you proceed anyway.	Did you start the installation as root? What permissions are for the first existing directory? Are you on a NFS file system with `root` mapped to nobody? Is the UID for the admin user the same on the local and remote machine?
Admin user missing	The admin user entered in the main configuration screen does not exist on the remote machine.	Setup the host properly so that name service provides the name properly to the remote machine (or create the user locally).
Directory exists	Berkeley DB spool directory already exists!	Check the remote host for existing Berkeley DB installations. Remove the existing directory.
Wrong FS type	Specified Berkeley DB spool directory is on a local file system.	Go back to the spooling configuration screen and choose a proper local directory.
Unknown error	Unknown error has occurred.	Try again (press `Install` button one more time). If reoccurring, ignore and try to install anyway.
Reachable	Validation did not discover any issues for this remote host.	
Canceled	User canceled further host validation.	

Installation States

When the installation is started the host list with the chosen components is transformed to a task list. The task list is better suited to handle dependencies. These are the states one may encounter during the installation.

State	Description
Waiting	Task is waiting to be executed.
Processing	Temporary state. Task is being processed.
Timeout	Task did not finish before timeout value has been reached.
Success	Task finished successfully.
Failed	Task finished unsuccessfully. Click the `Log` button to get more information.
Failed due to dependency	Task was not started, because it depended on a task that failed. Click the `Log` button to get more information.
Component already exists	Task was not started. The installation detected a previous conflicting component installation. Click the `Log` button to get more information. Remove any remains of the old installation, before trying again.
Canceled	User canceled the installation process.

Tweaking start_gui_installer

The `start_gui_installer` command will start the Java™ GUI installer. This section describes the command-line options of `start_gui_installer`, that you might use to affect the performance of the installer in your environment or possibly use as a workaround for yet unknown issues.

The Help text can be invoked by calling the `-help` option.

```
master:/sge62u2 # ./start_gui_installer -help
Usage: start_gui_installer [-help] [-resolve_pool=<num>] [-resolve_timeout=<sec>]
       [-install_pool=<num>] [-install_timeout=<sec>] [-connect_user=<usr>]
       [-connect_mode=windows]

   <num> ... decimal number greater than zero
   <sec> ... number of seconds, must be greater then zero
   <usr> ... user id
```

If no parameter is specified, the `start_gui_installer` command is started as if the following command was called:

```
master:/sge62u2 # ./start_gui_installer -resolve_pool=12 -resolve_timeout=20 -install_pool=8
-install_timeout=120
```

Every installation generates installation logs in the `sge_root/sge_cell`/install_logs directory. In addition, a GUI log file is created in a $TEMP directory (usually /var/tmp or /tmp) named `SGE_Gui-Installer_Log_<date>.txt`.

Description of **`start_gui_installer`** Options

Option	Description
-help	displays help for `start_gui_installer`
-resolve_pool=<num>	Defaults to 12. Defines how many hosts can be resolved in parallel when adding new hosts, refreshing their states or when validating hosts. The higher the value the higher load will be generated when resolving hosts, refreshing host states, copying an installation script to remote hosts or validating hosts.

-resolve_timeout=<sec>	Defaults to 20 seconds. A timeout value for any operation in a resolve_pool (resolving hosts, refreshing host states, copying an installation script to remote host). Host validation has a timeout which is always equal to 2*resolve_timeout value. Increase the default value if you see hosts with Unreachable state and you are sure that password-less access is working correctly for the connect_user.
-install_pool=<num>	Defaults to 8. Defines how many execution daemons can be installed in parallel. The higher the value the higher load will be generated when performing installation tasks.
-install_timeout=<sec>	Defaults to 120 seconds. A timeout value for any installation task. Increase the default value if you see that the installation tasks are failing with a Timeout state.
-connect_user=<user>	Defaults to current user. User name that will be used when connecting to remote hosts.
-connect_mode=windows	When set, each connect_user is prefixed by a host domain (see examples below). This is useful when installing multiple windowd execution hosts that require a different connect_user.
-debug	Starts the installer in a debug mode. Prints a lot of output to the terminal. Intended for developer purposes, but may provide additional information when unexpected circumstances occur.

Using **`start_gui_installer`** Options

Installing as a different connect_user
Suppose that you cannot log in as the `root` user, but can log in as another privileged user with uid=0, called `admin`. In this case an attempt for a remote connection would be done as current user, but due to uid=0 we would connect as `root` if root is the primary user with uid=0 on the remote host. Users `admin` and `root` would have different home directories and we assume that the password-less access was setup only for the user `admin` , so the connection without a password as currect user would fail. Invoking the following command will enforce that every remote connection is established as the `admin` user.

```
master:/sge62u2 # ./start_gui_installer -connect_user=admin
```

Installing single Windows execution host
Suppose you want to use the installer to add a single windows execution hosts to the existing cluster. The host is called `win-01` and belongs to the `WIN-01` domain. Also, the privileged user in this case is `Admin` (part of Administrators group).

The windows hosts can only be installed remotely from a UNIX/LINUX system and you cannot become an `Admin` user there. So you might use `-connect_user=WIN01+Admin` to connect as the correct user directly.

```
master:/sge62u2 # ./start_gui_installer -connect_user=WIN01+Admin
```

Installing multiple Windows execution hosts
Suppose you have additional hosts `win-02` belonging to the `WIN-02` domain and `win_vista-01` belonging to the `WIN_VISTA-01` domain. All hosts have `Administrator` user privileges. In this case, you can use the following command to start the GUI installer that will allow you to install all the three Windows execution hosts simultaneously.

```
master:/sge62u2 # ./start_gui_installer -connect_user=Administrator -connect_mode=windows
```

Every remote connection to host `win-01` would be done as `WIN-01+Administrator` user.
Every remote connection to host `win-02` would be done as `WIN-02+Administrator` user.
Every remote connection to host `win_vista-01` would be done as `WIN_VISTA-01+Administrator` user.

Troubleshooting the GUI Installer

You will find the known issues and their workarounds in this section as well as additional answers to some frequently asked questions.

FAQs

1. I cannot start the installer. It throws an exception!

Most likely a general problem with any GUI application in your current environment. You are probably starting the installer on a remote host and either did not export the DISPLAY variable properly or did not allow displaying remote GUI applications on the target system (where the GUI should pop-up).

a. Display variable is not set.

If your DISPLAY variable is not set and you are not locally on the system you will see a similar message:

```
hostA# ./start_gui_installer
Starting Installer ...
java.awt.HeadlessException:
No X11 DISPLAY variable was set, but this program performed an operation which requires
it.
        at java.awt.GraphicsEnvironment.checkHeadless(GraphicsEnvironment.java:159)
        at java.awt.Window.<init>(Window.java:317)
        at java.awt.Frame.<init>(Frame.java:419)
        at java.awt.Frame.<init>(Frame.java:384)
        at javax.swing.JFrame.<init>(JFrame.java:150)
        at com.izforge.izpack.installer.GUIInstaller.loadLangPack(Unknown Source)
        at com.izforge.izpack.installer.GUIInstaller.access$000(Unknown Source)
        at com.izforge.izpack.installer.GUIInstaller$1.run(Unknown Source)
        at java.awt.event.InvocationEvent.dispatch(InvocationEvent.java:199)
        at java.awt.EventQueue.dispatchEvent(EventQueue.java:461)
        at
java.awt.EventDispatchThread.pumpOneEventForHierarchy(EventDispatchThread.java:242)
        at
java.awt.EventDispatchThread.pumpEventsForHierarchy(EventDispatchThread.java:163)
        at java.awt.EventDispatchThread.pumpEvents(EventDispatchThread.java:157)
        at java.awt.EventDispatchThread.pumpEvents(EventDispatchThread.java:149)
        at java.awt.EventDispatchThread.run(EventDispatchThread.java:110)
java.lang.NullPointerException
        at com.izforge.izpack.installer.GUIInstaller.loadGUI(Unknown Source)
        at com.izforge.izpack.installer.GUIInstaller.access$100(Unknown Source)
        at com.izforge.izpack.installer.GUIInstaller$2.run(Unknown Source)
        at java.awt.event.InvocationEvent.dispatch(InvocationEvent.java:209)
        at java.awt.EventQueue.dispatchEvent(EventQueue.java:461)
        at
java.awt.EventDispatchThread.pumpOneEventForHierarchy(EventDispatchThread.java:242)
        at
java.awt.EventDispatchThread.pumpEventsForHierarchy(EventDispatchThread.java:163)
        at java.awt.EventDispatchThread.pumpEvents(EventDispatchThread.java:157)
        at java.awt.EventDispatchThread.pumpEvents(EventDispatchThread.java:149)
        at java.awt.EventDispatchThread.run(EventDispatchThread.java:110)
```

If you start the installer on hostA, but want to display it on hostB, you need to set a proper DISPLAY variable. If hostB has your graphical session on port 22, type the following command as user that will start the installer:

```
hostA# DISPLAY=hostB:22 ; export DISPLAY
```

See next step to finish the setup.

b. Remote host does not allow remote GUI applications.
   In this case you will see a similar message:

```
hostA# ./start_gui_installer
Starting Installer ...
Xlib: connection to "hostB:22" refused by server
Xlib: No protocol specified

Exception in thread "main" java.lang.InternalError: Can't connect to X11 window server
using 'hostB:22' as the value of the DISPLAY variable.
        at sun.awt.X11GraphicsEnvironment.initDisplay(Native Method)
        at sun.awt.X11GraphicsEnvironment.access$000(X11GraphicsEnvironment.java:53)
        at sun.awt.X11GraphicsEnvironment$1.run(X11GraphicsEnvironment.java:142)
        at java.security.AccessController.doPrivileged(Native Method)
        at sun.awt.X11GraphicsEnvironment.<clinit>(X11GraphicsEnvironment.java:131)
        at java.lang.Class.forName0(Native Method)
        at java.lang.Class.forName(Class.java:164)
        at
java.awt.GraphicsEnvironment.getLocalGraphicsEnvironment(GraphicsEnvironment.java:68)
        at sun.awt.motif.MToolkit.<clinit>(MToolkit.java:93)
        at java.lang.Class.forName0(Native Method)
        at java.lang.Class.forName(Class.java:164)
        at java.awt.Toolkit$2.run(Toolkit.java:821)
        at java.security.AccessController.doPrivileged(Native Method)
        at java.awt.Toolkit.getDefaultToolkit(Toolkit.java:804)
        at javax.swing.UIManager.initialize(UIManager.java:1262)
        at javax.swing.UIManager.maybeInitialize(UIManager.java:1245)
        at javax.swing.UIManager.getDefaults(UIManager.java:556)
        at javax.swing.UIManager.put(UIManager.java:841)
        at com.izforge.izpack.installer.GUIInstaller.loadLookAndFeel(Unknown Source)
        at com.izforge.izpack.installer.GUIInstaller.<init>(Unknown Source)
        at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
        at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:39
at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl
at java.lang.reflect.Constructor.newInstance(Constructor.java:494)
        at java.lang.Class.newInstance0(Class.java:350)
        at java.lang.Class.newInstance(Class.java:303)
        at com.izforge.izpack.installer.Installer.main(Unknown Source)
```

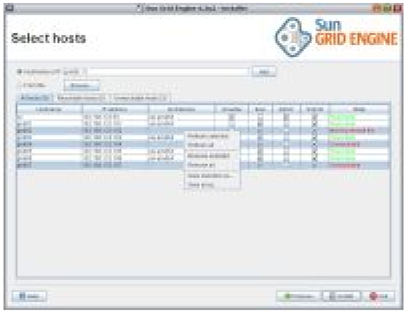
You have to explicitly allow remote GUI connections from hostA. Type the following command as the user running the graphical session on hostB:

```
hostB# xhost +hostA
```

Now you may start the `start_gui_installer` and the Welcome screen should get displayed on the remote host.

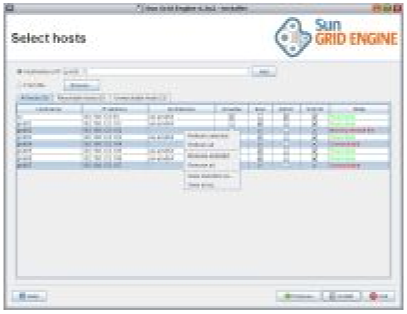
2. How can I remove a host from the host selection that I previously added?

Right_click the host and select `Remove selected` action from the pop-up menu.



3. Can I save hosts that I selected in the host selection to a file?

Yes, you can. Select multiple hosts using CTRL + left-click and do a right-click. A pop-up menu appears allowing you to save all hosts in the current tab or just the selected hosts.



4. Qmaster JMX thread does not appear to be running.

The qmaster messages file shows message `could not load libjvm ld.so.1: sge_qmaster: fatal: jvm_missing: open failed: No such file or directory`. Message means that the installer could not auto-detect a suitable JVM library. Possible reasons include being on a 64-bit platform and not having the 64-bit Java installed at all on the target hosts. Once you install correct Java you may change the `libjvm_path` attribute from `jvm_missing` to the correct path to the JVM library by calling `qconf -mconf` command.

Known issues and workarounds

1. Installing BDB server always fails with a timeout state.

Unfortunately you can't currently use the GUI installer to install a BDB server to any other platform, but Solaris OS. You may use the CLI installation (`inst_sge -db`) to do the job locally. You may then use the GUI installer to install qmaster and any number of shadow and executions hosts if the password-less access is configured. See issue 2941 for more information.

2. Cannot install additional execd hosts from a remove host (different from qmaster) when qmaster was installed with CSP or JMX SSL and a custom connect user is used.

It's recommended to always start the installation in the qmaster host. The reason is that in the subsequent standalone execd installation there is no way to specify a connect user for qmaster host. The remote connection to a qmaster host will be attempted as a user who started the GUI installer.

Installing the Software From the Command Line

Note

- The instructions in this section assume that you are installing the software on a computer running the Solaris ^{TM} Operating System. Any difference in functionality created by other operating system architecture that the Grid Engine software runs on is documented in files starting with the string `arc_depend` in the `$SGE_ROOT/doc` directory. The remainder of the file name indicates the operating system architectures to which the comments in the files apply, as in the `arc_depend_irix.asc` file.
- Also note that there are several prerequisites that you must satisfy for Windows systems before you can install Grid Engine. See Microsoft Services For UNIX and Microsoft Subsystem for UNIX-based Applications for further details.
- This section does not cover the upgrade process or the installation of the Accounting and Reporting Module, ARCo. For information about upgrading, see Upgrading From a Previous Release of the Software. For information about installing ARCo, see Installing the Accounting and Reporting Console.

Installation Overview

Note
The instructions in this section are for a new Grid Engine system only. For instructions on how to install a new system with additional security protection, see Installing the Increased Security Features. For instructions on how to upgrade an existing installation of an earlier version of the Grid Engine software, see Upgrading From a Previous Release of the Software.

Full installation includes the following tasks:

- Running an installation script on the master host and on every execution host in the Grid Engine system
- Registering information about administration hosts and submit hosts

Performing an Installation

The following sections describe how to install all the components of the Grid Engine system, including the master, execution, administration, and submit hosts. If you need to install the system with enhanced security, see Installing the Increased Security Features before you continue installation. For more information about installing Grid Engine SMF services see Installing the SMF Services before you start the installation.

Topic	Description
How to Install the Master Host (Example Master Host Installation)	Procedure for installing the master host.
How to Install the Shadow Master Host (Example Shadow Master Host Installation)	Procedure for installing the shadow master hosts.
How to Install Execution Hosts (Example Execution Host Installation)	Procedure for installing the execution host.
How to Register Administration Hosts	Procedure for registering an administration host.
How to Register Submit Hosts	Procedure for registering a submit host.
How to Install the Berkeley DB Spooling Server (Example Berkeley DB Spooling Server Installation)	Procedure for installing the necessary software for Berkeley DB spooling.

How to Install the Master Host

The master host installation procedure creates the appropriate directory hierarchy that the master daemon requires and starts the Grid Engine master daemon `sge_qmaster` on the master host. The master host is also registered as a host with administrative and submit permission. The installation procedure creates a default configuration for the system on which it is run. The installation script queries the system for the type of operating system. The script then makes meaningful settings based on this information.

If, at any time during the installation, you think something went wrong, you can quit the installation procedure and restart it.

Before You Begin

- Extract the Grid Engine software, as described in Loading the Distribution Files on a Workstation.
- If you have decided to use an administrative user, as described in User Account Considerations, you should create that user before installing the master host.

  Note
  Windows hosts cannot act as master hosts.

Steps

1. Log in to the master host as **`root`**.

2. If the **$SGE_ROOT** environment variable is not set, set it by typing:

   ```
   # SGE_ROOT=<path_to_installation_directory (the directory MUST contain all SGE files such as SGE
   binaries)>; export SGE_ROOT
   ```

   To confirm that you have set the $SGE_ROOT environment variable, type:

   ```
   # echo $SGE_ROOT
   ```

3. Go to the installation directory.
   - If the directory where the installation files reside is visible from the master host, change directory (`cd`) to the installation directory `sge-root`, and then proceed to the next step.
   - If the directory is not visible and cannot be made visible, do the following:
     - Create a local installation directory, `sge-root`, on the master host.
     - Copy the installation files to the local installation directory `sge-root` across the network, for example, by using `ftp` or `rcp`.
     - Change directory (`cd`) to the local `sge-root` directory.

4. Type the **`inst_sge -m`** command, adding the **`-csp`** flag if you are installing using the Certificate Security Protocol method described in Installing the Increased Security Features.
   This command starts the master host installation procedure. You are asked several questions, and you might be required to run some administrative actions.
   For a complete installation example, see Example Master Host Installation.

   ```
   # ./inst_sge -m
   Welcome to the Grid Engine installation
   ---------------------------------------

   Grid Engine qmaster host installation
   -------------------------------------

   .
   .
   .

   The qmaster installation procedure will take approximately 5-10 minutes.

   Hit <RETURN> to continue >>
   ```

5. Choose an administrative account owner.
   See Step 5 in the Example Master Host Installation.

6. Verify the **$SGE_ROOT** directory setting.
   In the example shown Step 6 of the Example Master Host Installation, the value of $SGE_ROOT is `/opt/sge62`.

7. Set up the TCP/IP services for the Grid Engine software.
   See Step 7 in the Example Master Host Installation.
   If TCP/IP services have not been configured, you will be notified. To configure TCP/IP services:
   a. Start a new terminal session or window to add the information **`/etc/services`** file or your NIS maps.
   b. Add the correct ports to the **`/etc/services`** file or your NIS services map, as described in Network Services.
      The following example adds entries for both `sge_qmaster` and `sge_execd` to your `/etc/services` file.

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

c. Save your changes and return to the window where the installation script is running.

8. Type the name of your cell or accept the default cell name.
   See Step 8 in the Example Master Host Installation.
   The use of Grid Engine system cells is described in Cells.
   - If you have decided to use cells, type the cell name now.
   - If you have decided not to use cells, press the Return key.

9. Set up a unique cluster name.
   See Step 9 in the Example Master Host Installation.
   For more information, see Cluster Name.
   - To accept the default cluster name, press the Return key.
   - To enter a new cluster name, type the cluster name and press the Return key.

10. Specify a spool directory.
    See Step 10 in the Example Master Host Installation.
    For guidelines on disk space requirements for the spool directory, see Disk Space Requirements. For information on where spool directory is installed, see Spool Directories Under the Root Directory.
    - To accept the default spool directory, press the Return key.
    - If you want to use a different spool directory, then answer y to the prompt and provide a complete path name to the directory.

11. Specify whether you plan to use Windows-based execution hosts.
    See Step 11 in the Example Master Host Installation.
    - If you do not plan to use Windows support, answer No.
    - If you want Windows support, answer Yes. You will be asked some Windows-specific questions later in the installation process. These questions will be marked as WINDOWS-ONLY.

12. Verify or set the correct file permissions.
    See Step 12 in the Example Master Host Installation.
    - If you used `pkgadd` or you know that the file permissions are correct, answer y to accept the current permissions.
    - Answer n if you need to verify or change the file permissions.
    - WINDOWS ONLY – If you specified that you wanted Windows Execution Host support in the previous question, you should let the script set the file permissions for you.

13. Specify whether all Grid Engine hosts for this cluster are located in a single DNS domain.
    See Step 13 in the Example Master Host Installation.
    - If all of your Grid Engine system hosts are located in a single DNS domain, then answer y. Grid Engine will not care if domain information is supplied. `hostA` and `hostA.foo.com` are equivalent.
    - If all of your Grid Engine system hosts are not located in a single DNS domain, then answer n. You will be asked to configure a default domain to use in case a host is specified without domain information.

14. Watch while Grid Engine creates directories according to the information that you provided so far.
    See Step 14 in the Example Master Host Installation.

15. Specify whether you want to enable the JMX MBean Server to use the SGE Inspect or the SDM SGE Adapter.
    See Step 15 in the Example Master Host Installation.
    - If you enable the JMX MBean Server, you are asked to enter the following information:

      > ⚠ Caution
      >
      > If you are on a 64-bit system, you need to provide `JAVA_HOME` for a 64-bit Java (usually installed as an addition to the 32-bit Java).

      - `JAVA_HOME` path
      - Additional JVM arguments
      - JMX MBean Server port number
      - JMX SSL server authentication
      - JMX SSL client authentication
      - JMX SSL server keystore path

    - JMX SSL server keystore password

16. Specify whether you want to use classic spooling or Berkeley DB.
    See Step 16 in the Example Master Host Installation. By default, Grid Engine uses Berkeley Database spooling.
    For more information on how to determine the type of spooling mechanism you want, please see Choosing Between Classic Spooling and Database Spooling.
    - If you choose Berkeley DB spooling, you are asked to choose whether to use a local directory or a Berkeley DB Spooling Server.

      > Tip
      > To use a shadow master host for increased availability of the database, use the Berkeley DB Spooling Server.

        - To use a Berkeley DB spooling server, enter y. To install the Berkeley DB Spooling Server:
          a. Start a new terminal session or window and install the software, as described in How to Register Submit Hosts.
          b. After you have installed the software on the spooling server, return to the master installation window, and press the Return key.
          c. Type the name of the spooling server.
             In Step 16 of the Example Master Host Installation, vector is the host name of the spooling server.
          d. Type the name of the spooling directory.
             In Step 16 of the Example Master Host Installation, `/opt/sge62/default/spool/spooldb` is the spooling directory.
        - If you do not want to use a Berkeley DB spooling server, type n. You are asked to provide the complete path to the database directory. If the directory does not exist, it is created.
    - To specify classic spooling, type `classic`.

17. Type a range of IDs that will be assigned dynamically for jobs.
    See Step 17 in the Example Master Host Installation.
    For more information, see Group IDs.

18. Verify the spooling directory for the execution daemon.
    See Step 18 in the Example Master Host Installation.
    The Grid Engine administrator must have access to create and write into this directory. For information on spooling, see Spool Directories Under the Root Directory.

19. Type the email address of the user who should receive problem reports.
    See Step 19 in the Example Master Host Installation. In the example, the user who will receive problem reports is `me@my.domain`.

20. Verify the configuration parameters.
    See Step 20 in the Example Master Host Installation.
    - If configuration parameters are correct, Grid Engine proceeds to create the local configuration.
    - If configuration parameters are not correct, type y to change them.

21. Specify whether you want the daemons to start when the system is booted.
    See Step 21 in the Example Master Host Installation.

22. WINDOWS-ONLY – If you specified that you want Windows support, you are asked to create Certificate Security Protocol (CSP) certificates. Even if the system is not running in CSP mode, it is necessary to create certain CSP certificates for Windows support. These certificates are automatically generated during the master host installation. For instructions on how to transfer these certificates to the Windows execution hosts, see Step 6 of How to Install a CSP-Secured System.

23. WINDOWS-ONLY – Add the Windows Administrator name to the Grid Engine manager list.

24. Identify the hosts that you will later install as execution hosts.
    See Step 24 in the Example Master Host Installation.

    > Tip
    > You can list hosts individually, separated by a blank space, or you can supply a file that contains host names.

> Note
> You can use the master host for executing jobs. To do so, you must carry out the execution host installation for the master machine. However, if you use a very slow machine as master host, or if your cluster is significantly large, do not use the master host as an execution host.

25. Select a scheduler profile.
    See Step 25 in the Example Master Host Installation.
    For information on how to determine which profile you should use, see Scheduler Profiles.
    Once you answer this question, the installation process is complete. Several screens of information will be displayed before the script exits.

26. WINDOWS-ONLY – Copy the certificate files to the Windows execution hosts.
    You can use a script to perform this function.

    > Tip
    > To use this functionality without being asked for a password, the root user should use `rsh` or `ssh` to access the execution hosts.

27. Create the environment variables (**`$SGE_ROOT`** and **`$SGE_CELL`**) for use with the Grid Engine software.
    See Step 27 in the Example Master Host Installation.

    > Note
    > If no cell name was specified during installation, the value of cell is `default`.

    - If you are using a C shell, type the following command:

      ```
      % source $SGE_ROOT/$SGE_CELL/common/settings.csh
      ```

    - If you are using a Bourne shell or Korn shell, type the following command:

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

> See Also
> For details about how you can verify that the execution host has been set up correctly, see [How to Verify That the Daemons Are Running on the Master Host].

Example Master Host Installation

The following example shows a complete Sun Grid Engine master host installation. Remember that this is only one step in the entire Sun Grid Engine installation process. The steps in this example coordinate with the master host installation description at How to Install the Master Host.

Steps 1-4

```
% su -
# cd sge-install-dir
# ./inst_sge -m
Grid Engine License is displayed.

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

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

Grid Engine qmaster host installation
-------------------------------------

Before you continue with the installation please read these hints:

   - Your terminal window should have a size of at least
     80x24 characters

   - The INTR character is often bound to the key Ctrl-C.
     The term >Ctrl-C< is used during the installation if you
     have the possibility to abort the installation

The qmaster installation procedure will take approximately 5-10 minutes.

Hit <RETURN> to continue >>

```

Step 5

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

The current directory

/opt/sge62

is owned by user

myusername

If user >root< does not have write permissions in this directory on *all*
of the machines where Grid Engine will be installed (NFS partitions not
exported for user >root< with read/write permissions) it is recommended to
install Grid Engine that all spool files will be created under the user id
of user >myusername<.

IMPORTANT NOTE: The daemons still have to be started by user >root<.

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

Installing Grid Engine as admin user >myusername<

Hit <RETURN> to continue >>
Choosing Grid Engine admin user account
---------------------------------------

You may install Grid Engine that all files are created with the user id of an
unprivileged user.

This will make it possible to install and run Grid Engine in directories
where user >root< has no permissions to create and write files and directories.

   - Grid Engine still has to be started by user >root<

   - This directory should be owned by the Grid Engine administrator

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

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

Please enter a valid user name >> sgeadmin

Installing Grid Engine as admin user >sgeadmin<

Hit <RETURN> to continue >>

```

Step 6

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

The Grid Engine root directory is:

   $SGE_ROOT = /opt/sge62

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 [/opt/sge62] >>

Your $SGE_ROOT directory: /opt/sge62

Hit <RETURN> to continue >>

```

Step 7

Two actions – one for **`qmaster`**, one for **`execd`**

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

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

   SGE_QMASTER_PORT = 10500

Now you have the possibility to set/change the communication ports by using the
>shell environment< or you may configure it via a network service, configured
in local >/etc/services<, >NIS< or >NIS+<, adding an entry in the form

   sge_qmaster <port_number>/tcp

to your services database and make sure to use an unused port number.

How do you want to configure the Grid Engine communication ports?

 Using the >shell environment<:                           [1]

 Using a network service like >/etc/services<, >NIS/NIS+<: [2]

(default: 1) >> 1

 Using the environment variable

   $SGE_QMASTER_PORT=10500

as port for communication.

Hit <RETURN> to continue >>

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

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

   SGE_EXECD_PORT = 10501

Now you have the possibility to set/change the communication ports by using the
>shell environment< or you may configure it via a network service, configured
in local >/etc/services<, >NIS< or >NIS+<, adding an entry in the form

   sge_execd <port_number>/tcp

to your services database and make sure to use an unused port number.

How do you want to configure the Grid Engine communication ports?

Using the >shell environment<:                           [1]

Using a network service like >/etc/services<, >NIS/NIS+<: [2]

(default: 1) >> 1

Using the environment variable

   $SGE_EXECD_PORT=10501

as port for communication.

Hit <RETURN> to continue >>
```

Step 8

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

Grid Engine supports multiple cells.

If you are not planning to run multiple Grid Engine clusters or if you don't
know yet what is a Grid Engine cell it is safe to keep the default cell name

   default

If you want to install multiple cells you can enter a cell name now.

The environment variable

   $SGE_CELL=<your_cell_name>

will be set for all further Grid Engine commands.

Enter cell name [default] >>

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

```

Step 9

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

The cluster name uniquely identifies a specific Sun Grid Engine cluster.
The cluster name must be unique throughout your organization. The name
is not related to the SGE cell.

The cluster name must start with a letter ([A-Za-z]), followed by letters,
digits ([0-9]), dashes (-) or underscores (_).

Enter new cluster name or hit <RETURN>
to use default [p10500] >>

Your $SGE_CLUSTER_NAME: p10500

Hit <RETURN> to continue >>
```

Step 10

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

The qmaster spool directory is the place where the qmaster daemon stores
the configuration and the state of the queuing system.

The admin user >myusername< must have read/write access
to the qmaster spool directory.

If you will install shadow master hosts or if you want to be able to start
the qmaster daemon on other hosts (see the corresponding section in the
Grid Engine Installation and Administration Manual for details) the account
on the shadow master hosts also needs read/write access to this directory.

The following directory

[/opt/sge62/default/spool/qmaster]

will be used as qmaster spool directory by default!

Do you want to select another qmaster spool directory (y/n) [n] >>

```

Step 11

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

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

```

Step 12

```
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 (y/n) [y] >>

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) [y] >>

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

Your file permissions were set

Hit <RETURN> to continue >>

```

Step 13

```

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

Are all hosts of your cluster in one DNS domain? If this is
the case the hostnames

   >hostA< and >hostA.foo.com<

would be treated as equal, because the DNS domain name >foo.com<
is ignored when comparing hostnames.

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

Ignoring domainname when comparing hostnames.

Hit <RETURN> to continue >>

```

Step 14

```

Making directories
------------------

creating directory: /opt/sge62/default/spool/qmaster
creating directory: /opt/sge62/default/spool/qmaster/job_scripts
Hit <RETURN> to continue >>

```

Step 15

```
Grid Engine JMX MBean server
----------------------------

In order to use the SGE Inspect or the Service Domain Manager (SDM)
SGE adapter you need to configure a JMX server in qmaster. Qmaster
will then load a Java Virtual Machine through a shared library.

Do you want to enable the JMX MBean server (y/n) [y] >> y

Please give some basic parameters for JMX MBean server
We will ask for
  - JAVA_HOME
  - additional JVM arguments (optional)
  - JMX MBean server port
  - JMX ssl authentication
  - JMX ssl client authentication
  - JMX ssl server keystore path
  - JMX ssl server keystore password

Detecting suitable JAVA ...
Please enter JAVA_HOME or press enter [/usr] >> /usr
Please enter additional JVM arguments (optional, default is [-Xmx256m]) >> -Xmx256m
Please enter an unused port number for the JMX MBean server [6444] >> 6444
Enable JMX SSL server authentication (y/n) [y] >> y

Enable JMX SSL client authentication (y/n) [y] >> y

Enter JMX SSL server keystore path [/var/sgeCA/port6442/def2/private/keystore] >> /var/sgeCA/port6442/def2/private
/keystore
Enter JMX SSL server keystore pw (at least 6 characters) >> ********

Using the following JMX MBean server settings.
libjvm_path              >/usr/jdk/instances/jdk1.5.0/jre/lib/sparcv9/server/libjvm.so<
Additional JVM arguments >-Xmx256m<
JMX port                 >6444<
JMX ssl                  >true<
JMX client ssl           >true<
JMX server keystore      >/var/sgeCA/port6442/def2/private/keystore<
JMX server keystore pw   >****************<

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

Hit <RETURN> to continue >>

Making directories
------------------

creating directory: /cod_home/sge6.2u3/def2/spool/qmaster
creating directory: /cod_home/sge6.2u3/def2/spool/qmaster/job_scripts
Hit <RETURN> to continue >>

```

Step 16

```
Setup spooling
--------------
Your SGE binaries are compiled to link the spooling libraries
during runtime (dynamically). So you can choose between Berkeley DB
spooling and Classic spooling method.
Please choose a spooling method (berkeleydb|classic) [berkeleydb] >>

The Berkeley DB spooling method provides two configurations!

1) Local spooling:
The Berkeley DB spools into a local directory on this host (qmaster host)
This setup is faster, but you can't setup a shadow master host

2) Berkeley DB Spooling Server:
If you want to setup a shadow master host, you need to use
Berkeley DB Spooling Server!
In this case you have to choose a host with a configured RPC service.
The qmaster host connects via RPC to the Berkeley DB. This setup is more
failsafe, but results in a clear potential security hole. RPC communication
(as used by Berkeley DB) can be easily compromised. Please only use this
alternative if your site is secure or if you are not concerned about
security. Check the installation guide for further advice on how to achieve
failsafety without compromising security.

Do you want to use a Berkeley DB Spooling Server? (y/n) [n] >> y

Berkeley DB Setup

-----------------
Please, log in to your Berkeley DB spooling host and execute "inst_sge -db"
Please do not continue, before the Berkeley DB installation with
"inst_sge -db" is completed, continue with <RETURN>

Berkeley Database spooling parameters
-------------------------------------

Please enter the name of your Berkeley DB Spooling Server! >> vector


Do you want to use a Berkeley DB Spooling Server? (y/n) [n] >>

Hit <RETURN> to continue >>

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: [/opt/sge62/default/spool/spooldb] >> /tmp/dom/spooldb

Dumping bootstrapping information
Initializing spooling database

Hit <RETURN> to continue >>
```

Step 17

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

When jobs are started under the control of Grid Engine an additional group id
is set on platforms which do not support jobs. This is done to provide maximum
control for Grid Engine jobs.

This additional UNIX group id range must be unused group id's in your system.
Each job will be assigned a unique id during the time it is running.
Therefore you need to provide a range of id's which will be assigned
dynamically for jobs.

The range must be big enough to provide enough numbers for the maximum number
of Grid Engine jobs running at a single moment on a single host. E.g. a range
like >20000-20100< means, that Grid Engine will use the group ids from
20000-20100 and provides a range for 100 Grid Engine jobs at the same time
on a single host.

You can change at any time the group id range in your cluster configuration.

Please enter a range >> 20000-20100

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

```

Step 18

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

Please give the basic configuration parameters of your Grid Engine
installation:

   <execd_spool_dir>

The pathname of the spool directory of the execution hosts. User >myusername<
must have the right to create this directory and to write into it.

Default: [/opt/sge62/default/spool] >>

```

Step 19

```
Grid Engine cluster configuration (continued)
---------------------------------------------
<administator_mail>

The email address of the administrator to whom problem reports are sent.

It is recommended to configure this parameter. You may use >none<
if you do not wish to receive administrator mail.

Please enter an email address in the form >user@foo.com<.

Default: [none] >> me@my.domain

```

Step 20

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

   execd_spool_dir        /opt/sge62/default/spool
   administrator_mail     me@my.domain

Do you want to change the configuration parameters (y/n) [n] >> n

Creating local configuration
----------------------------
Creating >act_qmaster< file
Adding default complex attributes
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 >>

```

Step 21

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

Do you want to start qmaster automatically at machine boot?
NOTE: If you select "n" SMF will be not used at all! (y/n) [y] >>


Hit <RETURN> to continue >>

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

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

Step 24

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

Please now add the list of hosts, where you will later install your execution
daemons. These hosts will be also added as valid submit hosts.

Please enter a blank separated list of your execution hosts. You may
press <RETURN> if the line is getting too long. Once you are finished
simply press <RETURN> without entering a name.

You also may prepare a file with the hostnames of the machines where you plan
to install Grid Engine. This may be convenient if you are installing Grid
Engine on many hosts.

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

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

```

```
Please enter a blank seperated list of hosts.

Stop by entering <RETURN>. You may repeat this step until you are
entering an empty list. You will see messages from Grid Engine
when the hosts are added.

Host(s): host1 host2 host3 host4

host1 added to administrative host list
host1 added to submit host list
host2 added to administrative host list
host2 added to submit host list
host3 added to administrative host list
host3 added to submit host list
host4 added to administrative host list
host4 added to submit host list
Hit <RETURN> to continue >>

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

Please enter a blank seperated list of hosts.

Stop by entering <RETURN>. You may repeat this step until you are
entering an empty list. You will see messages from Grid Engine
when the hosts are added.

Host(s):
Finished adding hosts. Hit <RETURN> to continue >>

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

If you are not sure, it is also possible to add or remove hosts after the
installation with <qconf -ah hostname> for adding and <qconf -dh hostname>
for removing this host

Attention: This is not the shadow host installation
procedure.
You still have to install the shadow host separately

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

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

Please now add the list of hosts, where you will later install your shadow
daemon.

Please enter a blank separated list of your execution hosts. You may
press <RETURN> if the line is getting too long. Once you are finished
simply press <RETURN> without entering a name.

You also may prepare a file with the hostnames of the machines where you plan
to install Grid Engine. This may be convenient if you are installing Grid
Engine on many hosts.

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

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

Please enter a blank seperated list of hosts.

Stop by entering <RETURN>. You may repeat this step until you are
entering an empty list. You will see messages from Grid Engine
when the hosts are added.

Host(s): es-ergb01-01
adminhost "es-ergb01-01" already exists
```

```
Hit <RETURN> to continue >>

Please enter a blank seperated list of hosts.

Stop by entering <RETURN>. You may repeat this step until you are
entering an empty list. You will see messages from Grid Engine
when the hosts are added.

Host(s):
Finished adding hosts. Hit <RETURN> to continue >>

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

root@myhost added "@allhosts" to host group list
root@myhost added "all.q" to cluster queue list

Hit <RETURN> to continue >>

```

```
No CSP system installed!
No CSP system installed!
```

Step 25

```
Scheduler Tuning
----------------
The details on the different options are described in the manual.

Configurations
 --------------
1) Normal
    Fixed interval scheduling, report scheduling information,
    actual + assumed load

2) High
    Fixed interval scheduling, report limited scheduling information,
    actual load

3) Max
    Immediate Scheduling, report no scheduling information,
    actual load

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


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

changed scheduler configuration
```

Step 27

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

You should now enter the command:

   source /scratch2/myusername/sge62/default/common/settings.csh

if you are a csh/tcsh user or

   # . /scratch2/myusername/sge62/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 >>

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

```

Grid Engine messages can be found at:

   Startup messages can be found in SMF service log files.
   You can get the name of the log file by calling svcs -l <SERVICE_NAME>
   E.g.: svcs -l svc:/application/sge/qmaster:p10500

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

   Qmaster:     /scratch2/myusername/sge62/default/spool/qmaster/messages
   Exec daemon: <execd_spool_dir>/<hostname>/messages


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

Grid Engine startup scripts can be found at:

   /scratch2/myusername/sge62/default/common/sgemaster (qmaster)
   /scratch2/myusername/sge62/default/common/sgeexecd (execd)

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

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> >>
```

```

sge_qmaster successfully installed!
```

How to Install Execution Hosts

The execution host installation procedure creates the appropriate directory hierarchy required by sge_execd, and starts the sge_execd daemon on the execution host. This section describes how to install execution hosts interactively from the command line. You can automate the installation of execution of multiple hosts by using the procedure described in Automating the Installation Process.

Before You Begin

Before installing an execution host, you first need to install the master server as described in How to Install the Master Host and share the common directory.

> Caution
>
> If you the fail to share the $SGE_ROOT/$SGE_CELL/common directory, you will not able to install execution hosts on nodes other than the qmaster host.

> Windows-Only
>
> You must satisfy several prerequisites before you can install Grid Engine execution hosts with Windows operating systems.
>
> - You might have to install additional software on your computer. See Microsoft Services for UNIX and Microsoft Subsystem for UNIX-based Applications.
> - See the steps described in How to Install a CSP-Secured System – Steps 6a, 6b and 6c.
> - After the installation, each user has to register their Windows password with Grid Engine using the sgepasswd client application. See User Management on Windows Hosts for more information.

Steps

1. Log in to the execution host as **root**.

2. As you did for the master installation, either copy the installation files to a local installation directory **sge-root** or use a network installation directory.

3. If the **$SGE_ROOT** environment variable is not set, set it by typing:

   ```
   # SGE_ROOT=<path_to_install/unpacked_directory>; export SGE_ROOT
   ```

   To confirm that you have set the $SGE_ROOT environment variable, type:

   ```
   # echo $SGE_ROOT
   ```

4. Change directory (**cd**) to the installation directory, **sge-root**.

5. Verify that the execution host has been declared on the administration host.
   - If you do not see the name of this execution host in the output of the `qconf -sh` command, you will need to declare it as an administration host.
     - Start a new terminal session or window.
     - In that window, log into the master host.
     - Declare the execution host as an administration host, using the `qconf` command.

       ```
       # qconf -ah quark
       quark added to administrative host list
       ```

     - Log back out of the master host, and continue with the installation of the execution host.

6. Type the **`install_execd`** command, adding the **`-csp`** flag if you are installing using the Certificate Security Protocol method described in Installing the Increased Security Features.
   This command starts the execution host installation procedure.
   For a complete installation example, see Example Execution Host Installation.

   ```
   # ./inst_sge -x
   Welcome to the Grid Engine execution host installation
   ------------------------------------------------------

   .
   .
   .

   The execution host installation will take approximately 5 minutes.

   Hit <RETURN> to continue >>
   ```

7. Verify the **$SGE_ROOT** directory setting.
   In the example shown in lines 27 through 41 of the Example Execution Host Installation, the value of $SGE_ROOT is `/scratch2/myusername/sge62`.

8. Type the name of your cell or accept the default cell name.
   See lines 042 through 076 of the Example Execution Host Installation.
   The use of Grid Engine system cells is described in Cells.
   - If you have decided to use cells, then type the cell names now.
   - If you have decided not to use cells, then press the Return key.

9. The install script checks to see what ports have been defined for the execution daemon.
   See lines 077 through 085 of the Example Execution Host Installation.
   If no ports have been defined, you will be asked to define them.

10. The install script checks to see whether the admin user already exists.
    If the admin user already exists, the script continues uninterrupted. If the admin user does not exist, the script shows the following screen where you must supply a password for the admin user. After the admin user is created, press the Return key.

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

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

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

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

11. Verify the execution host has been declared as an administration host.
    See lines 086 through 092 of the Example Execution Host Installation.

12. Specify whether you want to use a local spool directory.
    See lines 093 through 122 of the Example Execution Host Installation.
    For information on spooling, see Spool Directories Under the Root Directory.
    - If you do not want a local spool directory, answer n.
    - If you do want a local spool directory, answer y.
      In the example, `/tmp/dom/execs` is used as the local spool directory on `domain.com`. Choose any directory that meets the disk space requirements described in Disk Space Requirements.

13. Specify whether you want **execd** to start automatically at boot time.
    See lines 123 through 131 of the Example Execution Host Installation.
    You might not want to install the startup script if you are installing a test cluster or you would rather start the daemon manually on reboot.

14. WINDOWS ONLY – Choose whether to display the GUI for Windows jobs.
    See lines 132 through 163 of the Example Execution Host Installation.
    A Grid Engine Helper Service is included with the Sun Grid Engine distribution. This service enables Windows jobs to display a GUI on the visible desktop of the execution host. The visible desktop is either the desktop of the user currently logged in on the execution host or the desktop of the next user who will log in. It is not the log in screen.
    The Helper Service is a independent component loosely coupled with the execution daemon. The startup of the Helper Service is plugged in the Services dialog box in the Windows control panel. You can install only one Helper Service per host. There can be only one execution daemon installed per Helper Server.
    The installation script asks during the installation of a execution host whether you want to see the GUI of Windows jobs.

15. Specify a queue for this host.
    See lines 164 through 183 of the Example Execution Host Installation.
    Once you answer this question, the installation process is complete. Several screens of information will be displayed before the script exits.

16. Create the environment variables (**`$SGE_ROOT`** and **`$SGE_CELL`**) for use with the Grid Engine Software.
    See lines 184 through 234 of the Example Execution Host Installation.

    > ⓘ Note
    > If no cell name was specified during installation, the value of cell is default.

    - If you are using a C shell, type the following command:

      ```
      % source $SGE_ROOT/$SGE_CELL/common/settings.csh
      ```

    - If you are using a Bourne shell or Korn shell, type the following command:

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

See Also
For details about how you can verify that the execution host has been set up correctly, see How to Verify That the Daemons Are Running on the Execution Hosts.

Example Execution Host Installation

The following example shows a complete Sun Grid Engine execution host installation. Before you install the execution host, you need to first install the master server as described in How to Install the Master Host. The line numbers in this example are referred to from the execution host installation description at How to Install Execution Hosts.

Steps 1-6

```
001   % su -
002   # qstat -f
003   # ./ins_sge -x
004
005   Welcome to the Grid Engine execution host installation
006   ------------------------------------------------------
007
008   If you haven't installed the Grid Engine qmaster host yet, you must execute
009   this step (with >install_qmaster<) prior the execution host installation.
010
011   For a sucessful installation you need a running Grid Engine qmaster. It is
012   also necessary that this host is an administrative host.
013
014   You can verify your current list of administrative hosts with
015   the command:
016
017      # qconf -sh
018
019   You can add an administrative host with the command:
020
021      # qconf -ah <hostname>
022
023   The execution host installation will take approximately 5 minutes.
024
025   Hit <RETURN> to continue >>
026
```

Step 7

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

The Grid Engine root directory is:

   $SGE_ROOT = /scratch2/myusername/sge62

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 [/scratch2/myusername/sge62] >>

Your $SGE_ROOT directory: /scratch2/myusername/sge62

Hit <RETURN> to continue >>

```

Step 8

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

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

Using cell: >default<

Hit <RETURN> to continue >>

... set owner of /var/sgeCA/port10500 to bofur+myusername

... copy /var/sgeCA/port10500/default/userkeys/root to
/var/sgeCA/port10500/default/userkeys/bofur+Administrator
cp: /var/sgeCA/port10500/default/userkeys/root: No such file or directory

... copy /var/sgeCA/port10500/default/userkeys/root to
/var/sgeCA/port10500/default/userkeys/Administrator
cp: /var/sgeCA/port10500/default/userkeys/root: No such file or directory

... copy /var/sgeCA/port10500/default/userkeys/myusername to
/var/sgeCA/port10500/default/userkeys/bofur+myusername

... set owner of /var/sgeCA/port10500/default/userkeys/Administrator to Administrator

... set owner of /var/sgeCA/port10500/default/userkeys/bofur+Administrator to bofur+Administrator

... set owner of /var/sgeCA/port10500/default/userkeys/myusername to myusername

... set owner of /var/sgeCA/port10500/default/userkeys/bofur+myusername to bofur+myusername

... remove old /var/sgeCA/port10500/default/userkeys/root certificates

WINDOWS certificates are copied and permissions are set!

```

Step 9

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

The port for sge_execd is currently set BOTH as service and by the
shell environment

   SGE_EXECD_PORT = 10501
   sge_execd service set to port 725

```

Step 10

If the admin user already exists, the script automatically skips this step. See How to Install Execution Hosts for more information.

Step 11

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

This hostname is known at qmaster as an administrative host.

Hit <RETURN> to continue >>

```

Step 12

```
Local execd spool directory configuration
-----------------------------------------

During the qmaster installation you've already entered a global
execd spool directory. This is used, if no local spool directory is configured.

Now you can configure a local spool directory for this host.
ATTENTION: The local spool directory doesn't have to be located on a local
drive. It is specific to the <local> host and can be located on network drives,
too. But for performance reasons, spooling to a local drive is recommended.

FOR WINDOWS USER: On Windows systems the local spool directory MUST be set
to a local harddisk directory.
Installing an execd without local spool directory makes the host unuseable.
Local spooling on local harddisk is mandatory for Windows systems.

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

Please enter the local spool directory now! >> /tmp/dom/execs
Using local execd spool directory [/tmp/dom/execs]
Hit <RETURN> to continue >>

Creating local configuration
----------------------------
myusername@domain.com modified "domain.com" in configuration list
Local configuration for host >domain.com< created.

Hit <RETURN> to continue >>

```

Step 13

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

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


Hit <RETURN> to continue >>

```

Step 14

```
Windows Helper Service Installation
---------------------------------------

If you're going to run Windows job's using GUI support, you have
to install the Windows Helper Service
Do you want to install the Windows Helper Service? (y/n) [n] >> y

Testing, if a service is already installed!

   ... a service is already installed!
   ... stopping service!
   ... uninstalling old service!
Service successfully uninstalled.


   ... moving new service binary!
   ... installing new service!
Service successfully installed.


   ... starting new service!

Hit <RETURN> to continue >>

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

Starting execution daemon. Please wait ...
   starting sge_execd

Hit <RETURN> to continue >>

```

Step 15

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

We can now add a queue instance for this host:

   - it is added to the >allhosts< hostgroup
   - the queue provides 1 slot(s) for jobs in all queues
     referencing the >allhosts< hostgroup

You do not need to add this host now, but before running jobs on this host
it must be added to at least one queue.

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

No modification because "bofur" already exists in "hostlist" of "hostgroup"
root@domain.com modified "@allhosts" in host group list
root@domain.com modified "all.q" in cluster queue list

Hit <RETURN> to continue >>

```

Step 16

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

You should now enter the command:

   source /scratch2/myusername/sge62/default/common/settings.csh

if you are a csh/tcsh user or

   # . /scratch2/myusername/sge62/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 >>

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:     /scratch2/myusername/sge62/default/spool/qmaster/messages
   Exec daemon: <execd_spool_dir>/<hostname>/messages


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

Grid Engine startup scripts can be found at:

   /scratch2/myusername/sge62/default/common/sgemaster (qmaster)
   /scratch2/my/sge62/default/common/sgeexecd (execd)

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

Your execution daemon installation is now completed.

```

How to Register Administration Hosts

The master host is implicitly allowed to run administrative tasks and to submit, monitor, and delete jobs. The master host does not require any additional installation or configuration to perform administration functions. By contrast, pure administration hosts do require registration.

Note
You can also install administration hosts by using the QMON graphical user interface. See How to Configure Administration Hosts With QMON.

To register an administration host from the command line:

1. On the master host, log in to the Grid Engine system administrative account, for example, the **sgeadmin** account.

2. Type the following command:

```
% qconf -ah <admin-host-name>[,...]
```

How to Register Submit Hosts

Note
You can also install submit hosts by using the QMON graphical user interface. See How to Configure Submit Hosts With QMON.

To register a submit host from the command line:

1. On the master host, log in to the Grid Engine system administrative account, for example, the **sgeadmin** account.

2. Type the following command:

```
% qconf -as <submit-host-name>[,...]
```

Refer to About Hosts and Daemons for more details and other means to configure the different host types.

How to Install the Berkeley DB Spooling Server

The installation procedure installs the Grid Engine software necessary for Berkeley DB spooling.

1. Load the Grid Engine software onto a local file system.
   For details on how to extract the files, see How to Load the Distribution Files On a Workstation.

2. Log in to the spooling server host as root.

3. If the **$SGE_ROOT** environment variable is not set, set it by typing:

   ```
   # SGE_ROOT=sge-root; export SGE_ROOT
   ```

   To confirm that you have set the $SGE_ROOT environment variable, type:

```
# echo $SGE_ROOT
```

4. Change to the installation directory.

```
# cd $SGE_ROOT
```

5. Type the **inst_sge** command with the **-db** option.

```
# sge-root/inst_sge -db
```

   This command starts the spooling server installation procedure. You are asked several questions. If you think something went wrong, you can quit the installation procedure and restart it at any time.

6. Choose an administrative account owner.

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

You may install Grid Engine that all files are created with the user id of an
unprivileged user.

This will make it possible to install and run Grid Engine in directories
where user >root< has no permissions to create and write files and directories.

   - Grid Engine still has to be started by user >root<

   - this directory should be owned by the Grid Engine administrator

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

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

Please enter a valid user name >> sgeadmin
Installing Grid Engine as admin user >sgeadmin<

Hit <RETURN> to continue >>
```

7. Verify the $SGE_ROOT directory setting.
   In the following example, the value of $SGE_ROOT is `/opt/sge62`.

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

The Grid Engine root directory is:

   $SGE_ROOT = /opt/sge62

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 [/opt/n1ge6] >>

Your $SGE_ROOT directory: /opt/sge62

Hit <RETURN> to continue >>
```

8. Type the name of your cell.
   The use of Grid Engine system cells is described in Cells.

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

Grid Engine supports multiple cells.

If you are not planning to run multiple Grid Engine clusters or if you don't
know yet what is a Grid Engine cell it is safe to keep the default cell name

   default

If you want to install multiple cells you can enter a cell name now.

The environment variable

   $SGE_CELL=<your_cell_name>

will be set for all further Grid Engine commands.

Enter cell name [default] >>
```

9. Select Berkeley DB spooling.

```
Setup spooling
--------------
Your SGE binaries are compiled to link the spooling libraries
during runtime (dynamically). So you can choose between Berkeley DB
spooling and Classic spooling method.
Please choose a spooling method (berkeleydb|classic) [berkeleydb] >>
```

10. Verify your host name.
    In this example, the installation script is being run on `host2`.

```
Berkeley Database spooling parameters
-------------------------------------

You are going to install an RPC Client/Server mechanism!
In this case, qmaster will
contact an RPC server running on a separate server machine.
If you want to use the SGE shadowd, you have to use the
RPC Client/Server mechanism.

Enter database server name or
hit <RETURN> to use default [host2] >>
```

11. Type the directory path of your spooling directory.
    You might need to change this path if this directory is NFS mounted, or if you do not have write permissions to this directory.

```
Enter the database directory
or hit <RETURN> to use default [/opt/sge62/default//spooldb] >>

creating directory: /opt/sge62/default//spooldb
```

12. Start the RPC server.

```
Now we have to startup the rc script
>/opt/sge62/default/common/sgebdb<
on the RPC server machine

If you already have a configured Berkeley DB Spooling Server,
you have to restart the Database with the rc script now and continue with >NO<

Shall the installation script try to start the RPC server? (y/n) [y] >> y
Starting rpc server on host host2!
The Berkeley DB has been started with these parameters:

Spooling Server Name: host2
DB Spooling Directory: /opt/sge62/default//spooldb

Please remember these values, during Qmaster installation
you will be asked for them! Hit <RETURN> to continue!
```

13. Specify whether you want Berkeley DB service to start automatically at boot time.

    ```
    Berkeley DB startup script
    --------------------------

    We can install the startup script that
    Grid Engine is started at machine boot (y/n) [y] >> y
    ```

    Once you answer this question, the installation process is complete.

14. Create the environment variables for use with the Grid Engine software.

    > Note
    > If no cell name was specified during installation, the value of $SGE_CELL is default.

    - If you are using a C shell, type the following command:

      ```
      % source $SGE_ROOT/$SGE_CELL/common/settings.csh
      ```

    - If you are using a Bourne shell or Korn shell, type the following command:

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

How to Install the Berkeley DB Spooling Server

The installation procedure installs the Grid Engine software necessary for Berkeley DB spooling.

1. Load the Grid Engine software onto a local file system.
   For details on how to extract the files, see How to Load the Distribution Files On a Workstation.

2. Log in to the spooling server host as root.

3. If the **$SGE_ROOT** environment variable is not set, set it by typing:

```
# SGE_ROOT=sge-root; export SGE_ROOT
```

   To confirm that you have set the $SGE_ROOT environment variable, type:

```
# echo $SGE_ROOT
```

4. Change to the installation directory.

```
# cd $SGE_ROOT
```

5. Type the **`inst_sge`** command with the **`-db`** option.

```
# sge-root/inst_sge -db
```

   This command starts the spooling server installation procedure. You are asked several questions. If you think something went wrong, you can quit the installation procedure and restart it at any time.

6. Choose an administrative account owner.

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

You may install Grid Engine that all files are created with the user id of an
unprivileged user.

This will make it possible to install and run Grid Engine in directories
where user >root< has no permissions to create and write files and directories.

   - Grid Engine still has to be started by user >root<

   - this directory should be owned by the Grid Engine administrator

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

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

Please enter a valid user name >> sgeadmin
Installing Grid Engine as admin user >sgeadmin<

Hit <RETURN> to continue >>
```

7. Verify the $SGE_ROOT directory setting.
   In the following example, the value of $SGE_ROOT is `/opt/sge62`.

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

The Grid Engine root directory is:

   $SGE_ROOT = /opt/sge62

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 [/opt/n1ge6] >>

Your $SGE_ROOT directory: /opt/sge62

Hit <RETURN> to continue >>
```

8. Type the name of your cell.
   The use of Grid Engine system cells is described in Cells.

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

Grid Engine supports multiple cells.

If you are not planning to run multiple Grid Engine clusters or if you don't
know yet what is a Grid Engine cell it is safe to keep the default cell name

   default

If you want to install multiple cells you can enter a cell name now.

The environment variable

   $SGE_CELL=<your_cell_name>

will be set for all further Grid Engine commands.

Enter cell name [default] >>
```

9. Select Berkeley DB spooling.

```
Setup spooling
--------------
Your SGE binaries are compiled to link the spooling libraries
during runtime (dynamically). So you can choose between Berkeley DB
spooling and Classic spooling method.
Please choose a spooling method (berkeleydb|classic) [berkeleydb] >>
```

10. Verify your host name.
    In this example, the installation script is being run on `host2`.

```
Berkeley Database spooling parameters
-------------------------------------

You are going to install an RPC Client/Server mechanism!
In this case, qmaster will
contact an RPC server running on a separate server machine.
If you want to use the SGE shadowd, you have to use the
RPC Client/Server mechanism.

Enter database server name or
hit <RETURN> to use default [host2] >>
```

11. Type the directory path of your spooling directory.
    You might need to change this path if this directory is NFS mounted, or if you do not have write permissions to this directory.

```
Enter the database directory
or hit <RETURN> to use default [/opt/sge62/default//spooldb] >>

creating directory: /opt/sge62/default//spooldb
```

12. Start the RPC server.

```
Now we have to startup the rc script
>/opt/sge62/default/common/sgebdb<
on the RPC server machine

If you already have a configured Berkeley DB Spooling Server,
you have to restart the Database with the rc script now and continue with >NO<

Shall the installation script try to start the RPC server? (y/n) [y] >> y
Starting rpc server on host host2!
The Berkeley DB has been started with these parameters:

Spooling Server Name: host2
DB Spooling Directory: /opt/sge62/default//spooldb

Please remember these values, during Qmaster installation
you will be asked for them! Hit <RETURN> to continue!
```

13. Specify whether you want Berkeley DB service to start automatically at boot time.

```
Berkeley DB startup script
--------------------------

We can install the startup script that
Grid Engine is started at machine boot (y/n) [y] >> y
```

    Once you answer this question, the installation process is complete.

14. Create the environment variables for use with the Grid Engine software.

> Note
> If no cell name was specified during installation, the value of $SGE_CELL is default.

- If you are using a C shell, type the following command:

```
% source $SGE_ROOT/$SGE_CELL/common/settings.csh
```

- If you are using a Bourne shell or Korn shell, type the following command:

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

Installing the Increased Security Features

Use the instructions in this section to set up your system more securely. These instructions will help you set up your system with Certificate Security Protocol (CSP)-based encryption.

Why Install the Increased Security Features?

Instead of transferring messages in clear text, the messages in this secure system are encrypted with a secret key. The secret key is exchanged using a public/private key protocol. Users present their certificates through the Grid Engine system to prove identity. Users receive the certificate to ensure that they are communicating with the correct systems. After this initial announcement phase, communication continues transparently in encrypted form. The session is valid only for a certain period, after which the session must be re-announced.

Additional Setup Required

The steps required to set up the Certificate Security Protocol enhanced version of the Grid Engine system are similar to the standard setup. You generally follow the instructions in Planning the Installation, Loading the Distribution Files on a Workstation, How to Install the Master Host, How to Install Execution Hosts and How to Register Administration Hosts.

However, the following additional tasks are required:

- Generating the Certificate Authority (CA) system keys and certificates on the master host by calling the installation script with the `-csp` flag
- Distributing the system keys and certificates to the execution and submit hosts using a secure method such as `ssh`
- Generating user keys and certificates automatically, after master installation
- Adding new users

Topic	Description
How to Install a CSP-Secured System	Procedure for installing a CSP-secured system.
How to Generate Certificates and Private Keys for Users	Procedure for generating user-specific certificates and private keys.
How to Renew Certificates	Procedure for renewing user-specific certificates.
How to Check Certificates	Procedure for checking user-specific certificates.

How to Install a CSP-Secured System

Install the Grid Engine software as outlined in Performing an Installation, with the following exception: use the additional flag `-csp` when invoking the various installation scripts. To install a CSP-secured system do the following:

1. Change the master host installation procedure.

Type the following command and respond to the prompts from the installation script.

```
# ./install_qmaster -csp
```

2. Supply the following information to generate the CSP certificates and keys:
    - Two-letter country code, for example, US for the United States
    - State
    - Location, such as a city
    - Organization
    - Organizational unit
    - CA email address

      As the installation proceeds, the Certificate Authority is created. A CA specific to the Grid Engine system is created on the master host. The directories that contain information relevant to security are as follows:
    - The publicly accessible CA and daemon certificate are stored in

      ```
      $SGE_ROOT/$SGE_CELL/common/sgeCA
      ```

    - The corresponding private keys are stored in

      ```
      /var/sgeCA/{sge_service| portSGE_QMASTER_PORT}/cell/private
      ```

    - User keys and certificates are stored in

      ```
      /var/sgeCA/{sge_service| portSGE_QMASTER_PORT}/cell/userkeys/$USER
      ```

3. The script prompts you for site information.

4. Confirm whether the information you supplied is correct.

5. Continue the installation.
   After the security-related setup of the master host `sge_qmaster` is finished, the script prompts you to continue with the rest of the installation procedure, as in the following example:

   ```
   SGE startup script
   --------------------

   Your system wide SGE startup script is installed as:

        "/scratch2/eddy/sge_sec/default/common/sgemaster"

   Hit Return to continue >>
   ```

6. Transfer the directory that contains the private key and the random file to each execution host.
    a. As root on the master host, type the following commands to prepare to copy the private keys to the machines you set up as execution hosts:

```
# umask 077
# cd /
# tar cvpf /var/sgeCA/port536.tar /var/sgeCA/port536/default
```

b. As root on each execution host, use the following commands to securely copy the files:

```
# umask 077
# cd /
# scp masterhost:/var/sgeCA/port536.tar .
# umask 022
# tar xvpf /port536.tar
# rm /port536.tar
```

> Note
> On a Windows execution host, the `tar` utility cannot restore the ownerships and permissions. In this case, the Administrator must set the ownerships and permissions manually.

c. Type the following command to verify the file permissions:

```
# ls -lR /var/sgeCA/port536/
```

The output should look like the following example:

```
/var/sgeCA/port536/:
total 2
drwxr-xr-x   4 eddy     other        512 Mar  6 10:52 default
/var/sgeCA/port536/default:
total 4
drwx------   2 eddy     staff        512 Mar  6 10:53 private
drwxr-xr-x   4 eddy     staff        512 Mar  6 10:54 userkeys
/var/sgeCA/port536/default/private:
total 8
-rw-------   1 eddy     staff        887 Mar  6 10:53 cakey.pem
-rw-------   1 eddy     staff        887 Mar  6 10:53 key.pem
-rw-------   1 eddy     staff       1024 Mar  6 10:54 rand.seed
-rw-------   1 eddy     staff        761 Mar  6 10:53 req.pem
/var/sgeCA/port536/default/userkeys:
total 4
dr-x------   2 eddy     staff        512 Mar  6 10:54 eddy
dr-x------   2 root     staff        512 Mar  6 10:54 root
/var/sgeCA/port536/default/userkeys/eddy:
total 16
-r--------   1 eddy     staff       3811 Mar  6 10:54 cert.pem
-r--------   1 eddy     staff        887 Mar  6 10:54 key.pem
-r--------   1 eddy     staff       2048 Mar  6 10:54 rand.seed
-r--------   1 eddy     staff        769 Mar  6 10:54 req.pem
/var/sgeCA/port536/default/userkeys/root:
total 16
-r--------   1 root     staff       3805 Mar  6 10:54 cert.pem
-r--------   1 root     staff        887 Mar  6 10:54 key.pem
-r--------   1 root     staff       2048 Mar  6 10:53 rand.seed
-r--------   1 root     staff        769 Mar  6 10:54 req.pem
```

7. Install the Grid Engine software on each execution host.

   ```
   # cd $SGE_ROOT
   # ./install_execd -csp
   ```

8. Respond to the prompts from the installation script.
   The execution host installation procedure creates the appropriate directory hierarchy required by `sge_execd`, and starts the `sge_execd` daemon on the execution host.
   If the root user does not have write permissions in the $SGE_ROOT directory on all of the machines where Grid Engine software will be installed, you are asked whether to install the software as the user to whom the directory belongs. If you answer yes, you must install the security-related files into that user's $HOME/.sge directory, as shown in the following example.

   ```
   % su - sgeadmin
   % source $SGE_ROOT/default/common/settings.csh
   % $SGE_ROOT/util/sgeCA/sge_ca -copy
   % logout
   ```

   In the above example, sgeadmin is the name of the user who owns the installation directory.

9. After completing all remaining installation steps, refer to the instructions below in How to Generate Certificates and Private Keys for Users.

How to Generate Certificates and Private Keys for Users

To use the CSP-secured system, the user must have access to a user-specific certificate and private key. The most convenient method of gaining access is to create a text file identifying the users.

1. On the master host, create and save a text file that identifies users.
   Use the format of the file `myusers.txt` shown in the following example. The fields of the file are `UNIX_username:Gecos_field:email_address`.

   ```
   eddy:Eddy Smith:eddy@my.org
   sarah:Sarah Miller:sarah@my.org
   leo:Leo Lion:leo@my.org
   ```

2. As root on the master host, type the following command:

   ```
   # $SGE_ROOT/util/sgeCA/sge_ca -usercert myusers.txt
   ```

3. Confirm by typing the following command:

   ```
   # ls -l /var/sgeCA/port536/default/userkeys
   ```

   This directory listing produces output similar to the following example.

```
dr-x------  2 eddy  staff        512 Mar  5 16:13 eddy
dr-x------  2 sarah staff        512 Mar  5 16:13 sarah
dr-x------  2 leo   staff        512 Mar 5 16:13 leo
```

4. Tell each user to install security related files in their directories.
   Tell each user listed in the file (`myusers.txt` in the example) to install the security-related files in their `$HOME/.sge` directories by typing the following commands.

   ```
   % source $SGE_ROOT/default/common/settings.csh
   % $SGE_ROOT/util/sgeCA/sge_ca -copy
   ```

   Users should see the following confirmation (user `eddy` in the example).

   ```
   Certificate and private key for user
   eddy have been installed
   ```

   For every Grid Engine software installation, a subdirectory for the corresponding SGE_QMASTER_PORT number is installed. The following example, based on the `myusers.txt` file, is a result of issuing the command preceding the output.

   ```
   % ls -lR $HOME/.sge

   /home/eddy/.sge:
   total 2
   drwxr-xr-x  3 eddy staff        512 Mar  5 16:20 port536

   /home/eddy/.sge/port536:
   total 2
   drwxr-xr-x  4 eddy staff        512 Mar  5 16:20 default

   /home/eddy/.sge/port536/default:
   total 4
   drwxr-xr-x  2 eddy staff        512 Mar  5 16:20 certs
   drwx------  2 eddy staff        512 Mar 5 16:20 private

   /home/eddy/.sge/port536/default/certs:
   total 8
   -r--r--r--  1 eddy staff       3859 Mar  5 16:20 cert.pem

   /home/eddy/.sge/port536/default/private:
   total 6
   -r--------  1 eddy staff        887 Mar  5 16:20 key.pem
   -r--------  1 eddy staff       2048 Mar 5 16:20 rand.seed
   ```

How to Renew Certificates

1. Change to **`$SGE_ROOT`** and become root on the master host.

```
# tcsh
# source $SGE_ROOT/default/settings.csh
```

> Note
> This assumes that `$SGE_CELL` is the default.

2. Edit **`$SGE_ROOT/util/sgeCA/renew_all_certs.csh`**, and change the number of days that the certificates are valid:

```
# extend the validity of the CA certificate by
 set CADAYS = 365
 # extend the validity of the daemon certificate by
 set DAEMONDAYS = 365
 # extend the validity of the user certificate by
 set USERDAYS = 365
```

3. Run the changed script.

```
# util/sgeCA/renew_all_certs.csh
```

> Note
> The default for all extension times is 365 days from the day the script is run.

4. Replace the old certificates against the new ones on all hosts that installed them locally.
   That is, under `/var/sgeCA/...`, see the execution daemon installation.

5. If users have copied certificates and keys to **`$HOME/.sge`**, they have to repeat **`$SGE_ROOT/util/sgeCA/sge_ca -copy`** to have access to the renewed certificates.

Verifying the Installation

The verification phase includes the following tasks:

- Ensuring that the master daemon is running on the master host
- Ensuring that the daemons are running on all execution hosts
- Ensuring that you can run simple commands
- Submitting test jobs

To ensure that the Grid Engine system daemons are running, look for the `sge_qmaster` daemon on the master host and the `sge_execd` daemon on the execution hosts. Once you have verified that the daemons are running, you can try to use commands and prepare to submit jobs.

Note
If no cell name was specified during installation, the value of $SGE_CELL is default.

Topic	Description
How to Verify That the Daemon Is Running on the Master Host	Procedure for verifying that the Daemon is running on the master host.
How to Verify That the Daemons Are Running on the Execution Hosts	Procedure for verifying that the Daemons are running on the execution hosts.
How to Run Simple Commands	Procedure for verifying that the Sun Grid Engine software is operational by running some trial commands.
How to Submit Test Jobs	Procedure for submitting test jobs.

How to Verify That the Daemon is Running on the Master Host

1. Log in to the master host.
   Look in the file `$SGE_ROOT/$SGE_CELL/common/act_qmaster` to see if you really are on the master host.

2. Verify that the master daemon is running.
   - On BSD-based UNIX systems, type the following command:

     ```
     % ps -ax | grep sge
     ```

     You should see output similar to the following example.

     ```
     14676 p1 S <  4:47 /gridware/sge/bin/solaris/sge_qmaster
     ```

   - On systems running a UNIX System 5-based operating system (such as the Solaris Operating System), type the following command:

     ```
     % ps -ef | grep sge
     ```

     You should see output similar to the following example.

     ```
     root 439 1 0 Jun 2 ? 3:37 /gridware/sge/bin/solaris/sge_qmaster
     ```

3. If you do not see the appropriate string, restart the daemon.
   To start the master host daemon, `sge_qmaster`:

   ```
   # $SGE_ROOT/$SGE_CELL/common/sgemaster  start
   ```

4. Continue the verification process.
   After you have verified that the master host and the execution host daemons are running, continue the verification process. See How to Run Simple Commands.

How to Verify That the Daemons Are Running on the Execution Hosts

1. Log in to the execution hosts on which you ran the execution host installation procedure.

2. Verify that the daemons are running.
   - On BSD-based UNIX systems, type the following command:

     ```
     % ps -ax | grep sge
     ```

     You should see output similar to the following example.

     ```
     14688 p1 S <    4:27  /gridware/sge/bin/solaris/sge_execd
     ```

   - On systems running a UNIX System 5-based operating system (such as the Solaris Operating System), type the following command:

     ```
     % ps -ef | grep sge
     ```

     You should see output similar to the following example.

     ```
     root 171 1 0 Jun 22 ? 7:11 /gridware/sge/bin/solaris/sge_execd
     ```

3. If you do not see similar output, restart the daemon.

   ```
   # $SGE_ROOT/$SGE_CELL/common/sgeexecd  start
   ```

4. Continue the verification process.
   After you have verified that the master host and the execution host daemons are running, continue the verification process. See How to Run Simple Commands below for details.

How to Run Simple Commands

If both the necessary daemons are running on the master and execution hosts, the Grid Engine software should be operational. Check by issuing a trial command.

1. Log in to either the master host or another administrative host.
   In your standard search path, make sure to include `$SGE_ROOT/bin`.

2. From the command line, type the following command:

   ```
   % qconf -sconf
   ```

   This `qconf` command displays the current global cluster configuration Configuring Clusters.
   If this command fails, your $SGE_ROOT environment variable is not set correctly.

   a. Check whether the environment variables **SGE_EXECD_PORT** and **SGE_QMASTER_PORT** are set in the script files, **$SGE_ROOT/$SGE_CELL/common/settings.csh** or **$SGE_ROOT/$SGE_CELL/common/settings.sh**.

      > Note
      > If no cell name was specified during installation, the value of $SGE_CELL is default.

      - If so, make sure that the environment variables SGE_EXECD_PORT and {{SGE_QMASTER_PORT} are set to the correct value before you try the command again.
      - If not, verify whether your NIS services map contains entries for `sge_qmaster` and `sge_execd`.
        If the SGE_EXECD_PORT and SGE_QMASTER_PORT variables are not used in these files, then the services database (`/etc/services` or the NIS services map for example) on the machine from which you run the command must provide entries for both `sge_qmaster` and `sge_execd`. If these entries do not exist, add an entry to the machine's services database, giving it the same value as is configured on the master host.

   b. Retry the **qconf** command.

3. Try to submit test jobs.

How to Submit Test Jobs

Before you start submitting batch scripts to the Grid Engine system, check to see whether your site's standard shell resource files (`.cshrc`, `.profile`, or `.kshrc`) as well as your personal resource files contain commands such as `stty`. Batch jobs do not have a terminal connection by default, and therefore calls to `stty` result in an error.

1. Log in to the master host.

2. Type the following command.

   ```
   % rsh <exec-host-name> date
   ```

   The exec-host-name refers to one of the already installed execution hosts. You should try this test on all execution hosts if your login or home directories differ from host to host. The `rsh` command should give you output similar to the date command run locally on the master host. If any additional lines contain error messages, you must fix the cause of the errors before you can run a batch job successfully.

   For all command interpreters, you can check on an actual terminal connection before you run a command such as `stty`.
   The following is an example of a Bourne shell script to test the terminal connection.

   ```
   tty -s
   if [ $? = 0 ]; then
      stty erase ^H
   fi
   ```

   The following example shows C shell syntax.

```
tty -s
if ( $status = 0 ) then
   stty erase ^H
endif
```

3. Submit one of the sample scripts contained in the **$SGE_ROOT/examples/jobs** directory.

```
% qsub $SGE_ROOT/examples/jobs/simple.sh
```

4. Use the **qstat** command to monitor the job's behavior.
   For more information about submitting and monitoring batch jobs, see Submitting Batch Jobs.

5. After the job finishes executing, check your home directory for the redirected stdout/stderr files **script-name.e**job-id and **script-name.o**job-id.
   The job-id is a consecutive unique integer number assigned to each job.

In case of problems, see Fine-Tuning Your Environment and Using DTrace for Performance Tuning.

Automating the Installation Process

This section describes how you can automate the software installation process for the following reasons:

- To install the Grid Engine software on many hosts
- To install the Grid Engine software without user interaction

You can use the `$SGE_ROOT/inst_sge` utility to install and uninstall Sun Grid Engine master hosts, execution hosts, shadow host and Berkeley DB spooling server hosts. You can also use this utility to backup automatically the Sun Grid Engine configuration and accounting data.

> Note
> Using the Berkeley DB Spooling Server host does not provide high availability, and it has no authentication mechanism. It should only be used on a closed network with fully trusted users.

You can use the `inst_sge` utility in interactive mode to supplant any of the commands that were described in Installing the Software From the Command Line.

To simplify automatic installation and backup processes, use the configuration templates that are located in the `$SGE_ROOT/util/install_modules` directory.

The automatic installation requires no user interaction. No messages are displayed on the terminal during the installation.

When the installation finishes, a message indicates where the installation log file resides. The name of the installation log file format is `install_hostname_timestamp.log`. Normally, you can find information about errors during installation in this file. In case of serious errors though, the installation script might not be able to move the log file into the spool directory. In this situation, the log file is placed in the `/tmp` directory.

Topic	Description
Automatic Installation	Perform an automatic installation by setting up a configuration file.

Automatic Uninstallation	Learn how to uninstall hosts automatically.
How to Start the Automatic Backup	Procedure for backing up configuration and accounting data by using the interactive backup procedure.
Troubleshooting Automatic Installation and Uninstallation	Troubleshoot errors that might be encountered during automatic installation.

Automatic Installation

Special Considerations

The first step in performing an automatic installation is to set up a configuration file. You can find configuration file templates in the $SGE_ROOT/util/install_modules directory. Consider the following as you plan your automatic installation:

- To use automatic installation on remote hosts, the root user must be able to access those hosts through rsh or ssh without supplying a password.
- For local spooling, that is, spooling on the master host, no special configuration is needed. However, the directory where the spooling occurs must not be on an NFS version 3 volume. You may use an NFS version 4 volume for local spooling.
- To run the Berkeley DB spooling server on a host other than the master host, you must install and configure RPC services on this separate host.

To perform this step manually before you start the automatic installation, use the following command:

```
./inst_sge -db
```

You can also use the following command to install automatically the Berkeley DB Spooling Server:

```
% ./inst_sge -db -m -x -auto <full-path-to-configuration-file>
```

This command checks the SPOOLING_SERVER entry within the configuration file and starts the Berkeley DB installation on the server host.

> Note
> If you start the automatic installation on the master host, the entire cluster can be installed with one command. The automatic installation script accesses the remote hosts through rsh or ssh and starts the installation remotely. This process requires a well-configured configuration file, which each host must be able to read. That file should be installed on each host or shared through NFS.

Using the inst_sge Utility and a Configuration Template

To automate system installation, use the inst_sge utility in combination with a configuration file. See How to Automate Other Installations Through a Configuration File.

> Note
> You cannot use the auto installation procedure to install remotely a Windows execution host. You must run the auto installation procedure directly on the Windows execution host.

Topic	Description
How to Automate the Master Host Installation	Procedure for automating the master host installation.
How to Automate Other Installations Through a Configuration File	Procedure for performing a variety of other automatic installations using the configuration file.
How to Automate Installation With Increased Security (CSP)	Procedure for automating installation with Certificate Security Protocol (CSP) mode.

How to Automate Installation With Increased Security (CSP)

The automatic installation also supports the Certificate Security Protocol (CSP) mode described in Installing the Increased Security Features. To use the CSP security mode, you must fill out the CSP parameters of the template files. The parameters are as follows:

```
# This section is used for csp installation mode.
# CSP_RECREATE recreates the certs on each installation, 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 organisation (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"
```

To start the installation, type the following command:

```
inst_sge -m -csp -auto template-file-name
```

Note
Certificates are created during the installation process. These certificates have to be copied to each host of the installed cluster. The installation process can do this for you; however, you need to perform the following steps to allow the installation process appropriate permissions to copy the certificates:

1. Use **rsh**/**rcp** or **ssh**/**scp** on each host.
2. Provide the root user with access to each host over **ssh** or **rsh**, without entering a password.

How to Automate Other Installations Through a Configuration File

In addition to installing the master host, you can perform a variety of other automatic installations using a similar process. The actual form of the `inst_sge` command differs slightly, and different sections of the configuration file apply. This section provides some examples.

- To install a shadow host, use the following form of the command:

  ```
  inst_sge -sm -auto <full-path-to-configuration-file>
  ```

  Tip
  To install more than one shadow host, enter the host names in the <SHADOW_HOST> parameter section within the configuration file.

- You can install a separate execution host installation if the master host was installed without identified compute hosts or if you need to add additional compute hosts. For the execution host installation, you also need to have a configuration file.

  To install all configured execution hosts, use the following form of the command:

  ```
  inst_sge -x -auto <full-path-to-configuration-file>
  ```

- To install the Berkeley database server, use the following form of the command:

  ```
  inst_sge -db -auto <full-path-to-configuration-file>
  ```

See Configuration File Templates.

How to Automate the Master Host Installation

Before You Begin

You need to complete the planning process as outlined in Planning the Installation.

In addition, you need to be able to connect to each of the remote hosts using the `rsh` or `ssh` commands, without supplying a password. If this type of access is not allowed on your network, you cannot use this method of installation.

Steps

1. Create a copy of the configuration template, **`$SGE_ROOT/util/install_modules/inst_template.conf`**.

   ```
   # cd $SGE_ROOT/util/install_modules
   # cp inst_template.conf my_configuration.conf
   ```

2. Edit your configuration template, using the values from the worksheet you completed in Planning the Installation.
   The configuration file template includes liberal comments to help you decide where appropriate information belongs. See Configuration File Templates.

3. Log in as root on the system that you want to be the Sun Grid Engine master host.

4. Create the **`$SGE_ROOT`** directory.
   The $SGE_ROOT directory is the root directory of the Sun Grid Engine software hierarchy, for example `/opt/sge62`.

5. Go to the **`$SGE_ROOT`** directory and start the installation.

   ```
   # cd $SGE_ROOT
   # ./inst_sge -m -auto <full-path-to-configuration-file>
   ```

The `-m` option starts the master host installation and installs the master daemon on the local machine. In addition, the `-auto` option sets up any remote hosts, as specified in the configuration file.

> Note
> You cannot install remotely a master host. You must always install a master host locally.

To prevent data loss or destroying an already installed cluster, the automatic installation terminates if the configured `$SGE_CELL` directory or the configured Berkeley DB spooling directory already exists. If the installation terminates, the script displays the reason for the termination on the screen.

A log file of the master installation is created in the `$SGE_ROOT/default/spool/qmaster` directory. The file name is created using the format `install_hostname_date_time.log`.

> Tip
> You can also combine options if you want to perform multiple installations with one command. For example, the following command installs the master daemon on the local machine and installs all execution hosts that are configured in the configuration file:

```
./inst_sge -m -x -auto <full-path-to-configuration-file>
```

a. Wait for notification that the installation has completed.

b. When the automatic installation exits successfully, it displays a message similar to the following:

```
The Install log can be found in the
{{/opt/sge62/spool/install_myhost_30mar2007_090152.log}} file.
```

The installation log file includes any script or error messages that were generated during installation. If the `qmaster_spooling_dir` directory exists, the log files will be in that directory. If the directory does not exist, the log files will be in the `/tmp` directory.

Troubleshooting
If you do not want your execution hosts to spool locally, be sure to set EXECD_SPOOL_DIR_LOCAL="", with no space between the double quotes ("").

Automating Other Installations Through a Configuration File

In addition to installing the master host, you can perform a variety of other automatic installations using a similar process. The actual form of the `inst_sge` command differs slightly, and different sections of the configuration file apply. This section provides some examples.

- To install a shadow host, use the following form of the command:

  ```
  inst_sge -sm -auto <full-path-to-configuration-file>
  ```

  Tip
  To install more than one shadow host, enter the host names in the <SHADOW_HOST> parameter section within the configuration file.

- You can install a separate execution host installation if the master host was installed without identified compute hosts or if you need to add additional compute hosts. For the execution host installation, you also need to have a configuration file.

  To install all configured execution hosts, use the following form of the command:

  ```
  inst_sge -x -auto <full-path-to-configuration-file>
  ```

- To install the Berkeley database server, use the following form of the command:

  ```
  inst_sge -db -auto <full-path-to-configuration-file>
  ```

See Configuration File Templates.

Automatic Uninstallation

You can also uninstall hosts automatically.

Note
Uninstall all compute hosts before you uninstall the master host. If you uninstall the master host first, you have to uninstall all execution hosts manually.

To ensure that you have a clean environment, always source the `$SGE_ROOT/$SGE_CELL/common/settings.csh` file before proceeding.

Topic	Description
How to Uninstall the Master Host Automatically	Procedure for uninstalling the master host automatically.
How to Uninstall Execution Hosts Automatically	Procedure for uninstalling the execution hosts automatically.

How to Uninstall the Shadow Master Host	Procedure for uninstalling the shadow host.

How to Uninstall the Master Host Automatically

The master host uninstallation removes all of the Sun Grid Engine configuration files. After the uninstallation procedure completes, only the binary files remain. If you think that you will need the configuration information after the uninstallation, perform a backup of the master host. The master host uninstallation supports both interactive and automatic mode.

To start the automatic uninstallation of the master host, type the following command:

```
% ./inst_sge -um -auto <full-path-to-configuration-file>
```

This command performs the same procedure as in interactive mode, except the user is not prompted for confirmation of any steps and all terminal output is suppressed. Once the uninstall process is started, it cannot be stopped.

How to Uninstall Execution Hosts Automatically

During the execution host uninstallation, all configuration information for the targeted hosts is deleted. The uninstallation attempts to stop the exec hosts in a graceful manner.

First, the queue instances associated with the target host of the uninstallation will be disabled, so that new jobs will not be started. Then, in sequence, the following actions are done on each of the running jobs: checkpoint the job; reschedule the job; do forced rescheduling of the job.

At this point, the queue instance will be empty, and the execution daemon will be shut down, then the configuration, global spool directory or local spool directory will be removed.

The configuration file template has a section for identifying hosts that can be uninstalled automatically. Look for this section:

```
# Remove this execution hosts in automatic mode
EXEC_HOST_LIST_RM="host1 host2 host3 host4"
```

Every host in the EXEC_HOST_LIST_RM list will be automatically removed from the cluster.

To start the automatic uninstallation of execution hosts, type the following command:

```
% ./inst_sge -ux -auto <full-path-to-configuration-file>
```

How to Uninstall the Shadow Master Host

To start the automatic uninstallation of the shadow host, type the following command:

```
% ./inst_sge -usm -auto <full-path-to-configuration-file>
```

Troubleshooting Automatic Installation and Uninstallation

The following errors might be encountered during auto-installation:

Problem	Solution
If the `$SGE_CELL` directory exists, the installation terminates to avoid overwriting a previous installation.	Remove or rename the directory.
If the Berkeley database spooling directory exists, the installation terminates to avoid overwriting a previous installation.	This directory must be removed or renamed in order to proceed. Make sure that the ADMINUSER has permissions to write into the location where the Berkeley database spooling directory is located. The ADMINUSER will be the owner of the Berkeley database spooling directory.
The execution host installation appears to succeed, but the execution daemon is not started, or no load values are shown.	Verify that user root is allowed to `rsh` or `ssh` to the other host, without entering a password.
JMX thread does not appear to be running. The qmaster messages file shows message `could not load libjvm ld.so.1: sge_qmaster: fatal: jvm_missing: open failed: No such file or directory`	Either an incorrect value for SGE_JVM_LIB_PATH is specified in the installation template file or if left empty installer could not autodetect a suitable JVM library. Possible reasons might include being on a 64-bit platform and providing a path to a 32-bit JVM library or not having the 64-bit Java installed at all. Once you install correct Java you may change the `libjvm_path` attribute from `jvm_missing` to the correct path to the JVM library by calling `qconf -mconf` command.

If your network does not allow user root to have permissions to connect to other hosts through `rsh` or `ssh` without asking for a password, the automatic installation will not work remotely. In this case, log in to the host and use the following command to start the automatic installation locally on each host:

```
% ./inst_sge -x -noremote -auto /tmp/install_config_file.conf
```

Installing SMF Services

The Service Management Facility (SMF) is a new feature in Solaris 10. It provides a unified model for controlling services, replaces RC scripts, handles service dependencies, provides better service availability, and speeds up boot process. If you do not use at least Version 10 of the Solaris OS in your cluster, or you do not plan to use SMF, continue with Installing the Software From the Command Line.

> Note
> SMF is now the default for the hosts that run at least Version 10 of the Solaris OS. If you want to use the old behavior (RC files) for the Solaris hosts, you need to start the installation with the `-nosmf` option. Use the following command: `./inst_sge -x -nosmf`

Installing SMF services includes the following topics:

Why Install SMF Services?

SMF provides a unified administrative model of the persistent services. It solves many challenges of the previous approaches. All services have a common place for log files. Persistent services are automatically restarted on failure. For more information, see SMF documentation.

Additional Setup Required

If you want unprivileged users to use SMF services, you should create a role `sge_admin`. Assign this role to the users who should be able to manipulate the Grid Engine SMF services as described here.

Then, you can simply answer `y` when prompted to use SMF during the installation.

How Do SMF Services Compare to the Normal Services?

The biggest difference between SMF and normal services is that SMF does not consider `kill -9` to be a correct service shutdown. SMF interprets `kill -9` to restart the service.

Within the SMF framework, a service is uniquely identified by its fault resource management identifier (FMRI).

qmaster Daemon

Service name (FMRI) is `svc:/application/sge/qmaster:$SGE_CLUSTER_NAME`.

SGE version	sgemaster stop	qconf -km	kill -15	kill -9	reboot
6.1	stop	stop	stop	stop	restart [1]
6.2	stop	stop	stop	restart	restart

[1] - Restart the daemon if RC scripts were installed

shadowd Daemon

Service name (FMRI) is `svc:/application/sge/shadowd:$SGE_CLUSTER_NAME`.

SGE version	sgemaster -shadow stop	kill -15	kill -9	reboot
6.1	stop	stop	stop	restart [1]
6.2	stop	stop	restart	restart

[1] - Restart the daemon if RC scripts were installed

execd Daemon

Service name (FMRI) is `svc:/application/sge/execd:$SGE_CLUSTER_NAME`.

SGE version	sgeexecd stop	qconf -ke	kill -15	kill -9	reboot
6.1	stop	stop	stop	stop	restart [1]
6.2	stop	stop	stop	restart	restart

[1] - Restart the daemon if RC scripts were installed

Berkeley RPC Server

Service name (FMRI) is `svc:/application/sge/bdb:$SGE_CLUSTER_NAME`.

SGE version	berkeley_svc stop	kill -15	kill -9	reboot
6.1	stop	stop	stop	restart [1]
6.2	stop	restart	restart	restart

[1] - Restart the server if RC scripts were installed

dbwriter Software

Service name (FMRI) is `svc:/application/sge/dbwriter:$SGE_CLUSTER_NAME`.

SGE version	sgedbwriter stop	kill -15	kill -9	reboot
6.1	stop	stop	stop	restart [1]
6.2	stop	restart	restart	restart

[1] - Restart the `dbwriter` if RC scripts were installed

Additionally you may use new SMF interfaces to interact with services. For more information, see the `svcadm(1M)` man page. New actions:

Action	Command
Start service temporary	`svcadm enable -t` FMRI
Start service permanently (across reboots)	`svcadm enable` FMRI
Stop service temporary	`svcadm disable -t` FMRI
Stop service permanently (across reboots)	`svcadm disable` FMRI
Restart service	`svcadm reboot` FMRI

Installing a JMX-Enabled System

The JMX agent functionality enables access to a subset of `sge_qmaster` functionality via the JMX protocol. For Sun Grid Engine 6.2, the main purpose of the JMX agent is to provide an interface between the SDM Grid Engine adapter and the Sun Grid Engine system.

Additional Setup Required

The steps required to set up the JMX agent feature of Grid Engine are similar to the standard setup. You generally follow the instructions in Planning the Installation, Loading the Distribution Files on a Workstation, How to Install the Master Host, How to Install Execution Hosts and How to Register Administration Hosts.
However, you have to perform a few additional tasks:

- Generating necessary configuration files on the master host by calling the installation script with the -jmx flag and depending on the JMX specific installation settings the optional generation of certificates, keys and keystore files.
- Optional distribution of security relevant files to the execution and submit hosts using a secure method such as ssh
- Generating user keys, certificates and keys automatically, after master installation
- Adding new users
- Tweaking of JMX-specific files

Topic	Description
How to Install a JMX Agent-Enabled System	Procedure for installing Sun Grid Engine using the jmx flag when invoking the qmaster installation scripts.
How to Generate Certificates, Private Keys and Keystores for Users	Procedure for generating user-specific certificates, private keys, and keystores.
How to Check Certificates, Private Keys and Keystores for Users	Procedure for checking certificates, private keys, and keystores.
JMX Configuration Files	Describes the JMX configuration files in detail.
Testing and Troubleshooting	Testing and troubleshooting a JMX-enabled system.

How to Install a JMX Agent-Enabled System

Install the Grid Engine software as outlined in Installing the Software From the Command Line, with the following exception: use the additional flag `-jmx` when invoking the `qmaster` installation scripts.

To install a JMX agent enabled system do the following:

1. Change the master host installation procedure.
   Type the following command and respond to the prompts from the installation script.

```
# ./install_qmaster -jmx [-csp]
```

2. Supply the following information to generate necessary configuration files and optionally the certificates, keys and keystores:
   - JMX agent options

> ⚠ Caution
>
> If you are on a 64-bit system, you need to provide JAVA_HOME for a 64-bit Java (usually installed as an addition to the 32-bit Java).

   - JAVA_HOME (mandatory)
   - Additional JVM arguments (optional)
   - JMX MBean server port >= 1024 (mandatory)

    - JMX ssl authentication (default: true)
    - JMX ssl client authentication (default: true)
    - JMX ssl server keystore path
      (`/var/sgeCA/{sge_qmaster| port$SGE_QMASTER_PORT}/$SGE_CELL/private/keystore`)
    - JMX ssl server keystore password
  - Optional certificate specific options, if there is no CA available
    - Two-letter country code, for example, US for the United States
    - State
    - Location, such as a city
    - Organization
    - Organizational unit
    - CA email address
      As the installation proceeds, several JMX specific configuration files are installed:
      **jvm_threads is set to 1 instead of 0 if JMX is enabled in `$SGE_ROOT/$SGE_CELL/common/bootstrap`:

      ```
      ...
      jvm_threads             1
      ...
      ```

  - Several JMX agent specific configuration files are generated as:

    ```
    $SGE_ROOT/$SGE_CELL/common/jmx/jaas.config
    $SGE_ROOT/$SGE_CELL/common/jmx/java.policy
    $SGE_ROOT/$SGE_CELL/common/jmx/jmxremote.access
    $SGE_ROOT/$SGE_CELL/common/jmx/jmxremote.password
    $SGE_ROOT/$SGE_CELL/common/jmx/logging.properties
    $SGE_ROOT/$SGE_CELL/common/jmx/management.properties
    ```

    For a detailed description, see the comments in the files and the description below.

    Optionally the Certificate Authority is created. The directories that contain information relevant to security are as follows:

    - The publicly accessible CA and daemon certificate are stored in `$SGE_ROOT/$SGE_CELL/common/sgeCA`
    - The publicly accessible user certificates are stored in `$SGE_ROOT/$SGE_CELL/common/sgeCA/usercerts`
    - The corresponding private keys and keystore are stored in `/var/sgeCA/{sge_qmaster| port$SGE_QMASTER_PORT}/$SGE_CELL/private`
    - User keys, certificates and keystore are stored in `/var/sgeCA/{sge_qmaster| port$SGE_QMASTER_PORT}/$SGE_CELL/userkeys/$USER`

3. The script prompts you for site information.

4. Confirm whether the information you supplied is correct.

5. Continue the installation.
   After the security-related setup of the master host is finished, the script prompts you to continue with the rest of the installation procedure, as in the following example:

   ```
   SGE startup script
   --------------------

   Your system wide SGE startup script is installed as:

        "/scratch2/eddy/sge_sec/$SGE_CELL/common/sgemaster"

   Hit Return to continue >>
   ```

6. Proceed to the next task.

For more information, see How to Generate Certificates and Private Keys for Users.

How to Generate Certificates, Private Keys and Keystores for Users

To use the CSP-secured system, the user must have access to a user-specific certificate, private key and keystore. Usually the steps outlined in How to Generate Certificates and Private Keys for Users are performed. After that the following procedure can be done to generate the corresponding keystore files for the users.

1. As root on the master host run the following command:

```
# $SGE_ROOT/util/sgeCA/sge_ca -userks [-kspwf <kspwf-file>]
```

2. Confirm that the creation has been successful.

```
# ls -lR /var/sgeCA/port$SGE_QMASTER_PORT/$SGE_CELL/userkeys
/var/sgeCA/port$SGE_QMASTER_PORT/$SGE_CELL/userkeys/:
total 8
drwx------   2 eddy  staff        512 Mar 13 11:33 eddy
drwx------   2 sarah  staff        512 Mar 13 11:33 sarah
drwx------   2 leo  staff        512 Mar 13 11:33 leo

/var/sgeCA/port$SGE_QMASTER_PORT/$SGE_CELL/userkeys/eddy:
total 16
-rw-------   1 eddy staff       1586 Mar 13 11:32 cert.pem
-rw-------   1 eddy staff        891 Mar 13 11:32 key.pem
-rw-------   1 eddy staff       3031 Mar 13 11:33 keystore
-rw-------   1 eddy staff       1024 Mar 13 11:32 rand.seed
-rw-------   1 eddy staff        818 Mar 13 11:32 req.pem
...
```

The page XHow to Check Certificates, Private Keys and Keystores for Users does not exist.

JMX Configuration Files

The following configuration files are installed into `$SGE_ROOT/$SGE_CELL/common/jmx` and are explained in detail here. Manual modification is usually not necessary and the preinstalled configurations should be sufficient.

jaas.config

Before using the JMX interface, you must run a special authentication against `sge_qmaster`. This process adds the correct principle that gives you the necessary role to access the JMX interfaces in read-only or read-write mode. The responsible section in the `jaas.config` file is named GridwareConfig or TestConfig (for testing only).
In general, the `jaas.config` file defines which login modules are used for which application case. The choice of the login module is defined either in a configuration file like `management.properties` or programmatically.
The `jaas.config` file contains different sections and allows the replacement of the authentication mechanism, e.g. authentication via unix pam or via LDAP (see the GridwareConfig section and TestConfig section below). The different modules can be stacked. For a general overview of Jaas, see http://java.sun.com/developer/technicalArticles/Security/jaasv2/. Here the procedure consists of two steps in the GridwareConfig section:

- Authenticate the user (for example with keystore or Unix login or with LDAP).

- In the JGDILogin module, add the JMXPrincipal that gives the defined role to the user. This role is used later in the `jmx.access` file to check if the user has read-only or read-write access.

```
/*
 * Default login configuration for qmaster's jmx server
 */
GridwareConfig {

    /**
     *  Accepts all clients which have a certificate which is signed with
     *  the CA certificate.
     */
    com.sun.grid.security.login.GECATrustManagerLoginModule requisite
         caTop="${com.sun.grid.jgdi.caTop}";

    /*
     *  Accepts all clients which has a valid username/password.
     *
     *  The username/password validation is done with the authuser binary (included
     *  in the Grid Engine distribution, $SGE_ROOT/utilbin/$ARCH/authuser).
     *
     *  ATTENTION: The authuser binary needs the suid bit. It does not work if grid
     *  engine is installed on a nosuid file system.
     *
     */
    com.sun.grid.security.login.UnixLoginModule requisite
        sge_root="${com.sun.grid.jgdi.sgeRoot}"
        auth_method="system";

    /*
     * Username password authentication against LDAP.
     *
     * Alternative username/password authentication if
     * com.sun.grid.security.login.UnixLoginModule is not working.
     *
     * The LDAP specific parameters have to be adjusted to the local requirements
     * For details please have a look at the LdapLoginModule javadocs.
     *
     * ATTENTION: The LdapLoginModule is only available in java 6. The
     * parameter libjvm_path must point to a java 6 jvm
     * (qconf -sconf | grep libjvm_path)
     */
    /*
    com.sun.security.auth.module.LdapLoginModule requisite
        userProvider="ldap://sun-ds/ou=people,dc=sun,dc=com"
userFilter="(&(uid={USERNAME})(objectClass=inetOrgPerson))"
        useSSL=false;
    */

    /*
     *  The JGDILoginModule adds a JGDIPrincipal to the subject. The username of
     *  the JGDIPrincipal is the name of the first trusted principal. This name
     *  treated as username for gdi communication.
     *  For each login a new jgdi session id is created.
     *
     *  In the jmxremote.access file users who can access the system are defined
     *  Any principal matching these entries is given the corresponding role.
     *  Usually a jmxPrincipal is defined to give a user access to the system.
     *  (e.g. com.sun.grid.security.login.UserPrincipal = xyz &
     *        jmxPrincipal="controlRole" gives user xyz access under controlRole
     *  )
     */
    com.sun.grid.jgdi.security.JGDILoginModule optional
        trustedPrincipal="com.sun.grid.security.login.UserPrincipal"
        trustedPrincipal1="com.sun.security.auth.UserPrincipal"
        jmxPrincipal="controlRole";
};
```

```
/*
 *  TestConfig accepts any user. Only for testing
 */
TestConfig {

    com.sun.grid.security.login.TestLoginModule requisite;

    com.sun.grid.jgdi.security.JGDILoginModule optional
        trustedPrincipal="com.sun.grid.security.login.UserPrincipal"
        jmxPrincipal="controlRole";
};

/*
 *  Mandatory if native jgdi is used with a csp system
 *  (e.g. jgdish in csp mode)
 */
jgdi {
   com.sun.security.auth.module.KeyStoreLoginModule required
                                                    keyStoreURL="file:./keystore"
                                                    debug=false;
};
```

java.policy

The java.policy file that is used by the JGDIAgent restricts the possibilities of code that can be run in `sge_qmaster`'s JVM.

Usually changes here are only necessary to change the access to a subset of the overall functionality. To tweak the policy settings to your needs it is useful to run the JMX server with security debugging enabled and to consult the generated logging files. (`qconf -mconf`, additional_jvm_args = -Djavax.net.debug=ssl -Djava.security.debug=access,failure)

```
/*
**
** with LdapLoginModule
** grant principal com.sun.security.auth.UserPrincipal "controlRole"
**
** with jmxremote.password
** grant principal javax.management.remote.JMXPrincipal "controlRole"
**
*/
grant codeBase "file:${com.sun.grid.jgdi.sgeRoot}/lib/jgdi.jar"  {
   permission java.net.SocketPermission   "*:1024-", "accept,connect";
   permission java.net.SocketPermission   "localhost:1024-", "listen,resolve";
   permission java.lang.RuntimePermission "loadLibrary.jgdi";
   permission java.lang.RuntimePermission "shutdownHooks";
   permission java.lang.RuntimePermission "setContextClassLoader";
   permission javax.security.auth.AuthPermission "createLoginContext.jgdi";
   permission javax.security.auth.AuthPermission "doAs";
   permission javax.security.auth.AuthPermission "getSubject";
   permission java.util.PropertyPermission "*", "read";
   permission java.util.logging.LoggingPermission "control";

   permission java.lang.FilePermission
"${com.sun.grid.jgdi.sgeRoot}/${com.sun.grid.jgdi.sgeCell}/common/jmx/-", "read";
   permission java.io.FilePermission "${com.sun.grid.jgdi.sgeRoot}/util/-", "execute";
   permission java.io.FilePermission "${com.sun.grid.jgdi.sgeRoot}/utilbin/-", "execute";
   permission javax.management.MBeanServerPermission "createMBeanServer";
   permission javax.management.MBeanPermission "*", "*";
   permission javax.management.MBeanTrustPermission "register";
   permission java.lang.management.ManagementPermission "monitor";
   permission java.lang.management.ManagementPermission "control";

   permission java.lang.RuntimePermission "setIO";
   permission java.io.FilePermission      "jgdi.stdout", "write";
   permission java.io.FilePermission      "jgdi.stderr", "write";
   permission java.io.FilePermission      "jgdi0.log.lck", "delete";
   permission java.io.FilePermission
"${com.sun.grid.jgdi.sgeRoot}/${com.sun.grid.jgdi.sgeCell}/common/jmx/*", "read";
   permission java.io.FilePermission      "${com.sun.grid.jgdi.sgeRoot}/lib/-", "read";
   permission java.lang.RuntimePermission "accessClassInPackage.sun.management.jmxremote";
   permission java.lang.RuntimePermission "accessClassInPackage.sun.management.resources";
   permission java.lang.RuntimePermission "accessClassInPackage.sun.management";
   permission java.lang.RuntimePermission "accessClassInPackage.sun.rmi.server";
   permission java.lang.RuntimePermission "accessClassInPackage.sun.management.snmp.util";
   permission java.lang.RuntimePermission "accessClassInPackage.sun.rmi.registry";

   permission java.util.PropertyPermission "java.rmi.server.randomIDs", "write";

   permission javax.security.auth.AuthPermission "modifyPrincipals";
   permission javax.security.auth.AuthPermission "createLoginContext.*";
   permission javax.security.auth.AuthPermission "createLoginContext.JMXPluggableAuthenticator";
   permission java.security.SecurityPermission "createAccessControlContext";

   permission javax.management.remote.SubjectDelegationPermission
```

```
"javax.management.remote.JMXPrincipal.controlRole";
};

grant principal javax.management.remote.JMXPrincipal "controlRole" {
   permission javax.management.MBeanPermission "com.sun.grid.jgdi.management.mbeans.JGDIJMX#*", "*";
   permission javax.management.MBeanPermission "sun.management.*#*", "*";
   permission javax.security.auth.AuthPermission "createLoginContext.jgdi";
   permission javax.security.auth.AuthPermission "doAs";
   permission javax.security.auth.AuthPermission "getSubject";
   permission java.util.PropertyPermission "*", "read";
   permission java.util.PropertyPermission "user.timezone", "read,write";
   permission java.util.logging.LoggingPermission "control";
   permission java.io.FilePermission      "${com.sun.grid.jgdi.sgeRoot}/lib/-", "read";
   permission java.lang.management.ManagementPermission "monitor";
   permission java.net.SocketPermission "*", "resolve";

   permission javax.management.MBeanPermission
"com.sun.management.UnixOperatingSystem#-[java.lang:type=OperatingSystem]", "isInstanceOf";
   permission javax.management.MBeanPermission
"com.sun.management.UnixOperatingSystem#-[java.lang:type=OperatingSystem]", "getAttribute";
   permission javax.management.MBeanPermission
"com.sun.management.UnixOperatingSystem#ProcessCpuTime[java.lang:type=OperatingSystem]",
"getAttribute";
   permission javax.management.MBeanPermission
"com.sun.management.UnixOperatingSystem#Name[java.lang:type=OperatingSystem]", "getAttribute";
   permission javax.management.MBeanPermission
"com.sun.management.UnixOperatingSystem#Version[java.lang:type=OperatingSystem]", "getAttribute";
   permission javax.management.MBeanPermission
"com.sun.management.UnixOperatingSystem#Arch[java.lang:type=OperatingSystem]", "getAttribute";
   permission javax.management.MBeanPermission
"com.sun.management.UnixOperatingSystem#AvailableProcessors[java.lang:type=OperatingSystem]",
"getAttribute";
   permission javax.management.MBeanPermission
"com.sun.management.UnixOperatingSystem#CommittedVirtualMemorySize[java.lang:type=OperatingSystem]",
"getAttribute";
   permission javax.management.MBeanPermission
"com.sun.management.UnixOperatingSystem#TotalPhysicalMemorySize[java.lang:type=OperatingSystem]",
"getAttribute";
   permission javax.management.MBeanPermission
"com.sun.management.UnixOperatingSystem#FreePhysicalMemorySize[java.lang:type=OperatingSystem]",
"getAttribute";   permission javax.management.MBeanPermission
"com.sun.management.UnixOperatingSystem#TotalSwapSpaceSize[java.lang:type=OperatingSystem]",
"getAttribute";
   permission javax.management.MBeanPermission
"com.sun.management.UnixOperatingSystem#FreeSwapSpaceSize[java.lang:type=OperatingSystem]",
"getAttribute";
   permission javax.management.MBeanPermission
"javax.management.MBeanServerDelegate#-[JMImplementation:type=MBeanServerDelegate]",
"addNotificationListener";
   permission javax.management.MBeanPermission
"javax.management.MBeanServerDelegate#-[JMImplementation:type=MBeanServerDelegate]", "isInstanceOf";
   permission javax.management.MBeanPermission
"javax.management.MBeanServerDelegate#-[JMImplementation:type=MBeanServerDelegate]", "getMBeanInfo";
   permission javax.management.MBeanPermission
"com.sun.management.UnixOperatingSystem#-[java.lang:type=OperatingSystem]", "queryNames";
   permission javax.management.MBeanPermission
"java.util.logging.Logging#-[java.util.logging:type=Logging]", "queryNames";
   permission javax.management.MBeanPermission
"javax.management.MBeanServerDelegate#-[JMImplementation:type=MBeanServerDelegate]", "queryNames";
   permission javax.management.MBeanPermission
"java.util.logging.Logging#-[java.util.logging:type=Logging]", "isInstanceOf";
   permission javax.management.MBeanPermission
"java.util.logging.Logging#-[java.util.logging:type=Logging]", "getMBeanInfo";
   permission javax.management.MBeanPermission
"com.sun.management.UnixOperatingSystem#-[java.lang:type=OperatingSystem]", "getMBeanInfo";

};

grant {
   permission java.util.logging.LoggingPermission "control";
   permission java.util.PropertyPermission "*", "read";
   permission java.util.PropertyPermission "user.timezone", "write";
   permission java.lang.RuntimePermission "setIO";
   permission java.lang.RuntimePermission "loadLibrary.jgdi";
   permission java.io.FilePermission      "jgdi.stdout", "write";
   permission java.io.FilePermission      "jgdi.stderr", "write";
   permission java.io.FilePermission      "${com.sun.grid.jgdi.sgeRoot}/lib/-", "read";
   permission java.io.FilePermission      "${com.sun.grid.jgdi.sgeRoot}/util/arch", "execute";
   permission java.io.FilePermission      "${com.sun.grid.jgdi.sgeRoot}/utilbin/-", "execute";
   permission javax.security.auth.AuthPermission "modifyPrincipals";
   permission java.io.FilePermission "${com.sun.grid.jgdi.caTop}", "read";
   permission java.io.FilePermission "${com.sun.grid.jgdi.caTop}/cacert.pem", "read";
   permission java.io.FilePermission "${com.sun.grid.jgdi.caTop}/ca-crl.pem", "read";
   permission java.io.FilePermission "${com.sun.grid.jgdi.caTop}/usercerts/-", "read";
   permission java.io.FilePermission "${com.sun.grid.jgdi.serverKeystore}", "read";
};

/*
grant {
   permission java.security.AllPermission;
```

```
};
*/
```

management.properties

This file describes the general JMX server configuration and the default template looks similar to this example and is usually adapted automatically during the installation process replacing the @@SGE_*@@ variables by concrete values.
The meaning of the @@SGE_*@@ variables is:

- @@SGE_JMX_PORT@@ is the configured JMX port
- @@SGE_JMX_SSL@@ is true or false if SSL shall be enabled for JMX or not
- @@SGE_JMX_SSL_CLIENT@@ is true or false if client authentication is required
- @@SGE_JMX_SSL_KEYSTORE@@ the keystore used for enabled SSL
- @@SGE_JMX_SSL_KEYSTORE_PW@@ the corresponding keystore password
- @@SGE_ROOT@@ the $SGE_ROOT root directory
- @@SGE_CELL@@ the $SGE_CELL name usually 'default'

```
######################################################################
#  Default Configuration File for JGDI JMX
######################################################################
#
# The Management Configuration file (in java.util.Properties format)
# will be read if one of the following system properties is set:
#    -Dcom.sun.grid.jgdi.management.jmxremote.port=<port-number>
# or -Dcom.sun.grid.jgdi.management.config.file=<this-file>
#
# The default Management Configuration file is:
#
#       $SGE_ROOT/{$SGE_CELL|default}/common/jmx/management.properties
#
# ################ Management Agent Port #########################
#
# For setting the JMX RMI agent port use the following line
# com.sun.grid.jgdi.management.jmxremote.port=<port-number>
com.sun.grid.jgdi.management.jmxremote.port=@@SGE_JMX_PORT@@

######################################################################
#        RMI Management Properties
######################################################################
#
# If system property -Dcom.sun.grid.jgdi.management.jmxremote.port=<port-number>
# is set then
#     - A MBean server is started
#     - JRE Platform MBeans are registered in the MBean server
#     - RMI connector is published  in a private readonly registry at
#       specified port using a well known name, "jmxrmi"
#     - the following properties are read for JMX remote management.
#
# The configuration can be specified only at startup time.
# Later changes to above system property (e.g. via setProperty method),
# this config file, the password file, or the access file have no effect to the
# running MBean server, the connector, or the registry.
#

#
# ###################### RMI SSL #############################
#
# com.sun.grid.jgdi.management.jmxremote.ssl=true|false
#      Default for this property is true. (Case for true/false ignored)
#      If this property is specified as false then SSL is not used.
#

#For RMI monitoring without SSL use the following line
```

```
# com.sun.grid.jgdi.management.jmxremote.ssl=false
com.sun.grid.jgdi.management.jmxremote.ssl=@@SGE_JMX_SSL@@

# com.sun.grid.jgdi.management.jmxremote.ssl.enabled.cipher.suites=<cipher-suites>
#      The value of this property is a string that is a comma-separated list
#      of SSL/TLS cipher suites to enable. This property can be specified in
#      conjunction with the previous property "com.sun.management.jmxremote.ssl"
#      in order to control which particular SSL/TLS cipher suites are enabled
#      for use by accepted connections. If this property is not specified then
#      the SSL RMI Server Socket Factory uses the SSL/TLS cipher suites that
#      are enabled by default.
#

# com.sun.grid.jgdi.management.jmxremote.ssl.enabled.protocols=<protocol-versions>
#      The value of this property is a string that is a comma-separated list
#      of SSL/TLS protocol versions to enable. This property can be specified in
#      conjunction with the previous property "com.sun.management.jmxremote.ssl"
#      in order to control which particular SSL/TLS protocol versions are
#      enabled for use by accepted connections. If this property is not
#      specified then the SSL RMI Server Socket Factory uses the SSL/TLS
#      protocol versions that are enabled by default.
#

# com.sun.grid.jgdi.management.jmxremote.ssl.need.client.auth=true|false
#      Default for this property is false. (Case for true/false ignored)
#      If this property is specified as true in conjunction with the previous
#      property "com.sun.management.jmxremote.ssl" then the SSL RMI Server
#      Socket Factory will require client authentication.
#

#For RMI monitoring with SSL client authentication use the following line
#com.sun.grid.jgdi.management.jmxremote.ssl.need.client.auth=true
com.sun.grid.jgdi.management.jmxremote.ssl.need.client.auth=@@SGE_JMX_SSL_CLIENT@@


#
# ################ RMI User authentication ################
#
# com.sun.grid.jgdi.management.jmxremote.authenticate=true|false
#      Default for this property is true. (Case for true/false ignored)
#      If this property is specified as false then no authentication is
#      performed and all users are allowed all access.
#

# For RMI monitoring without any checking use the following line
# com.sun.grid.jgdi.management.jmxremote.authenticate=false
com.sun.grid.jgdi.management.jmxremote.authenticate=true

#
# ################ RMI Login configuration ###################
#
# com.sun.grid.jgdi.management.jmxremote.login.config=<config-name>
#      Specifies the name of a JAAS login configuration entry to use when
#      authenticating users of RMI monitoring.
#
#      Setting this property is optional - the default login configuration
#      specifies a file-based authentication that uses the password file.
#
#      When using this property to override the default login configuration
#      then the named configuration entry must be in a file that gets loaded
#      by JAAS. In addition, the login module(s) specified in the configuration
#      should use the name and/or password callbacks to acquire the user's
#      credentials. See the NameCallback and PasswordCallback classes in the
#      javax.security.auth.callback package for more details.
#
#      If the property "com.sun.management.jmxremote.authenticate" is set to
#      false, then this property and the password & access files are ignored.
#

# For a non-default login configuration use the following line
# com.sun.grid.jgdi.management.jmxremote.login.config=<config-name>
com.sun.grid.jgdi.management.jmxremote.login.config=GridwareConfig

#
# ################ RMI Password file location ##################
#
# com.sun.grid.jgdi.management.jmxremote.password.file=filepath
#      Specifies location for password file
#      This is optional - default location is
#      $JRE/lib/management/jmxremote.password
#
#      If the property "com.sun.grid.jgdi.management.jmxremote.authenticate" is set to
#      false, then this property and the password & access files are ignored.

# For a non-default password file location use the following line
# com.sun.grid.jgdi.management.jmxremote.password.file=filepath
com.sun.grid.jgdi.management.jmxremote.password.file=@@SGE_ROOT@@/@@SGE_CELL@@/common/jmx/jmxremote.passw
################ RMI Access file location #####################
#
# com.sun.grid.jgdi.management.jmxremote.access.file=filepath
#      Specifies location for access  file
#      This is optional - default location is
#      $JRE/lib/management/jmxremote.access
#
#      If the property "com.sun.management.jmxremote.authenticate" is set to
#      false, then this property and the password & access files are ignored.
#      Otherwise, the access file must exist and be in the valid format.
#      If the access file is empty or non-existent then no access is allowed.
#

# For a non-default access file location use the following line
# com.sun.grid.jgdi.management.jmxremote.access.file=filepath
com.sun.grid.jgdi.management.jmxremote.access.file=@@SGE_ROOT@@/@@SGE_CELL@@/common/jmx/jmxremote.access
For the JGDI keystore module use this settings for the server keystore and keystore password
```

```
com.sun.grid.jgdi.management.jmxremote.ssl.serverKeystore=@@SGE_JMX_SSL_KEYSTORE@@
com.sun.grid.jgdi.management.jmxremote.ssl.serverKeystorePassword=@@SGE_JMX_SSL_KEYSTORE_PW@@
```

jmx.access

The jmx access file defines which principals are mapped to a special role.

```
######################################################################
#     Default Access Control File for Remote JMX(TM) Monitoring
######################################################################
#
# Access control file for Remote JMX API access to monitoring.
# This file defines the allowed access for different roles.  The
# password file (jmxremote.password by default) defines the roles and their
# passwords.  To be functional, a role must have an entry in
# both the password and the access files.
#
# Default location of this file is $JRE/lib/management/jmxremote.access
# You can specify an alternate location by specifying a property in
# the management config file $JRE/lib/management/management.properties
# (See that file for details)
#
# The file format for password and access files is syntactically the same
# as the Properties file format.  The syntax is described in the Javadoc
# for java.util.Properties.load.
# Typical access file has multiple  lines, where each line is blank,
# a comment (like this one), or an access control entry.
#
# An access control entry consists of a role name, and an
# associated access level.  The role name is any string that does not
# itself contain spaces or tabs.  It corresponds to an entry in the
# password file (jmxremote.password).  The access level is one of the
# following:
#       "readonly" grants access to read attributes of MBeans.
#                   For monitoring, this means that a remote client in this
#                   role can read measurements but cannot perform any action
#                   that changes the environment of the running program.
#       "readwrite" grants access to read and write attributes of MBeans,
#                   to invoke operations on them, and to create or remove them.
#         This access should be granted to only trusted clients,
#                   since they can potentially interfere with the smooth
#         operation of a running program
#
# A given role should have at most one entry in this file.  If a role
# has no entry, it has no access.
# If multiple entries are found for the same role name, then the last
# access entry is used.
#
#
# Default access control entries:
# o The "monitorRole" role has readonly access.
# o The "controlRole" role has readwrite access.

monitorRole   readonly
controlRole   readwrite
```

jmx.password

This is also a possible simple authentication mechanism though not recommended. Usually the jaas login module is preferred since it is much more flexible. You can specify a password for the different roles there. If a simple login mechanism is required it is recommended to change management.properties to use TestConfig instead of GridwareConfig, which allows any valid Unix user to connect to JGDI JMX server without a password.

logging.properties

To enable JGDI and JMX logging the delivered logging file has to be adjusted and `sge_qmaster` or at least the JMX server has to be restarted. The generated logging files default to jgdi0.log, jgdi.stderr and jgdi.stdout in the master spooling directory. The logging can also be influenced by changing the additional_jvm_args configuration to enable additional debugging messages for example.

```
#
#  Java Logging Configuration for JMX MBean server
#

# Specify the handlers to create in the root logger
# (all loggers are children of the root logger)
# The following creates two handlers

# Per default we log to the console
#handlers = java.util.logging.ConsoleHandler

# Use FileHandler
handlers = java.util.logging.FileHandler

# -------------------------------------------------------------------------------
#  Definition of log levels
# -------------------------------------------------------------------------------
# Set the default logging level for the root logger
.level = INFO
#com.sun.grid.jgdi.JGDI.level = FINE
#com.sun.grid.jgdi.rmi.level = FINE
#com.sun.grid.jgdi.configuration.xml.XMLUtil.level = FINE
#com.sun.grid.jgdi.configuration.ClusterQueueTestCase.level = FINE
#com.sun.grid.jgdi.management.level = FINER
#com.sun.grid.jgdi.event.level = FINER
# For authuser login module debugging
#com.sun.grid.security.login.level = FINER
#com.sun.grid.util.expect.level = FINER

# -------------------------------------------------------------------------------
#  Settings for ConsoleHandler
# -------------------------------------------------------------------------------
# Set the default logging level for new ConsoleHandler instances
java.util.logging.ConsoleHandler.level = INFO

# Set the default formatter for new ConsoleHandler instances
java.util.logging.ConsoleHandler.formatter = com.sun.grid.jgdi.util.SGEFormatter

# -------------------------------------------------------------------------------
#  Settings for FileHandler
# -------------------------------------------------------------------------------
# Set the default logging level for new FileHandler instances
java.util.logging.FileHandler.level = ALL
# qmaster runs in qmaster spool dir, so the file is created there
java.util.logging.FileHandler.pattern=jgdi%u.log
java.util.logging.FileHandler.formatter=com.sun.grid.jgdi.util.SGEFormatter

#
# Possible columns:
#
#   time    timestamp of the log message
#   host    hostname of the log message
#   name    name of the logger
#   thread  id of the thread
#   level   log level (short form)
#   source  class and method name
#   level_long log_level long form
#
com.sun.grid.jgdi.util.SGEFormatter.columns = time thread source level message
```

```
#
#  Print the stacktrace of the log record
#
com.sun.grid.jgdi.util.SGEFormatter.withStacktrace=true

#
#  Delimiter between columns
```

```
#
com.sun.grid.jgdi.util.SGEFormatter.delimiter = |
```

Testing and Troubleshooting

To connect to the JMX server jconsole can be used for testing. It is the responsibility of the administrator to allow/disallow access to the system via JMX. To force also client authentication of jconsole the management.properties file must be configured with:

- com.sun.grid.jgdi.management.jmxremote.ssl=true
- com.sun.grid.jgdi.management.jmxremote.ssl.need.client.auth=true

```
% jconsole -J-Djava.security.manager=java.rmi.RMISecurityManager \
 -J-Djava.security.policy=$SGE_ROOT/util/rmiconsole.policy \
 -J-Djavax.net.ssl.trustStore=<server truststore> \
 [-J-Djavax.net.ssl.keyStore=/<safe>/mykeystore \
  -J-Djavax.net.ssl.keyStorePassword=<mykeystore_pw> \
  -J-Djavax.net.ssl.keyPassword=<mykeystore_pw> ] \
 [-J-Djavax.net.debug=ssl]
```

where <server truststore> usually is either:
/var/sgeCA/port5322/$SGE_CELL/private/keystore
(only the server certificate is accessible without password)
or a special truststore is made available by the administrator:

```
keytool -export -alias "root" \
        -keystore /var/sgeCA/port$SGE_QMASTER_PORT/$SGE_CELL/private/keystore -rfc -file
/tmp/jmxserver.cer

keytool -import -file /tmp/jmxserver.cer -keystore /tmp/truststore
Enter keystore password:  <pwd>
...
Trust this certificate? [no]:  yes
Certificate was added to keystore
```

The optional arguments are required if client authentication is set to true or for debugging.

The following simple example can be used to connect via JMX and monitor events

```
% java [-Dcom.sun.grid.jgdi.keyStore=\          /var/sgeCA/port$SGE_QMASTER_PORT/$SGE_CELL/private
/keystore \
-Dcom.sun.grid.jgdi.caTop="$SGE_ROOT/$SGE_CELL/common/sgeCA" \
-Djava.util.logging.config.file=util/shell_logging.properties ] \
-cp $SGE_ROOT/lib/juti.jar:$SGE_ROOT/lib/jgdi.jar \
com.sun.grid.jgdi.examples.jmxeventmonitor.Main
```

The optional arguments can be skipped and serve only to preset the login dialog with useful values. If a connection has been established once a preferences file is written, that is reused afterwards.
To have the correct environment variables set the SGE settings.(c)sh file has to be sourced. To get access to the keystore the command must be run by the admin user in the example above.

For troubleshooting the following settings and files might give some additional insights:

- Messages file in the master spool directory if the JMX server can't be started

- $SGE_ROOT/$SGE_CELL/common/bootstrap to check if jvm_threads is enabled at all
- jgdi* log files in the master spool directory are the main source for finding out the reason for failure analysis
- $SGE_ROOT/$SGE_CELL/common/jmx/logging.properties to enable more detailed logging
- qconf -mconf with an additional_jvm_args parameter
  For example, add these two arguments -Djava.security.debug=all -Djavax.net.debug=ssl to trace any permission and authentication problems.

Removing the Software

This section consists of the following topics.

Topic	Description
How to Remove the Software Interactively	Procedure for removing the Sun Grid Engine software interactively.
How to Remove the Software Using the inst_sge Utility and a Configuration Template	Procedure for removing the Sun Grid Engine software using the inst_sge utility and a configuration template.

How to Remove the Software Interactively

To remove the software interactively, follow the steps below.

> Note
> Remove the software from the execution hosts before removing it from the master host. If you remove the software from the master host first, you cannot automate the removal of the software from the execution hosts.

1. Ensure that your environment variables are set up properly.

   > Note
   > If no cell name was specified during installation, the value of $SGE_CELL is default.

   - If you are using a C shell, type the following command:

     ```
     # source $SGE_ROOT/$SGE_CELL/common/settings.csh
     ```

   - If you are using a Bourne or Korn shell, type the following command:

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

2. On the master host, issue the **$SGE_ROOT/inst_sge -ux** command.
   This example uninstalls the execution hosts: host1, host2 and host3.

   ```
   # $SGE_ROOT/inst_sge -ux -host "host1 host2 host3"
   ```

> Note
> You are not prompted for any information during this process. However, the output from this process will be displayed to the terminal window where you run the command.

3. (Optional) If you have any shadow master hosts, uninstall them:

```
# $SGE_ROOT/inst_sge -usm -host "host4"
```

4. Uninstall the master host.

```
# $SGE_ROOT/inst_sge -um
```

How to Remove the Software Using the inst_sge Utility and a Configuration Template

Unlike the interactive uninstallation method, the automated uninstallation method suppresses output during the process. Also, the automated method requires a properly formatted configuration file.

To remove the software using the `inst_sge` utility and a configuration template, follow these steps:

> Note
> Remove the software from the execution hosts before removing it from the master host. If you remove the software from the master host first, you cannot automate the removal of the software from the execution hosts.

1. Ensure that your environment variables are set up properly.

> Note
> If no cell name was specified during installation, the value of $SGE_CELL is default.

- If you are using a C shell, type the following command:

```
# source $SGE_ROOT/$SGE_CELL/common/settings.csh
```

- If you are using a Bourne or Korn shell, type the following command:

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

2. Create a copy of the configuration template, **`$SGE_ROOT/util/inst_sge_modules/inst_sge_template.conf`**.

```
# cd $SGE_ROOT/util/inst_sge_modules/
# cp inst_sge_template.conf  my_configuration.conf
```

3. Edit your configuration template.
   Every host that is in the EXEC_HOST_LIST_RM list will be removed.

   ```
   # Remove these execution hosts in automatic mode
   EXEC_HOST_LIST_RM="host1 host2 host3 host4"
   ```

4. On the master host type the **$SGE_ROOT/inst_sge -ux -auto** command.
   This example uninstalls the execution hosts: host1, host2 and host3.
   Type the following command as one string, with a space between the -auto and the $SGE_ROOT/util/inst_sge_modules/my_configuration.conf components.

   ```
   # $SGE_ROOT/inst_sge -ux -auto $SGE_ROOT/util/inst_sge_modules/my_configuration.conf
   ```

   Note
   You are not prompted for any information during this process. However, the output from this process will be displayed to the terminal window where you run the command.

5. (Optional) If you have any shadow master hosts, uninstall them.
   Type the following command as one string, with a space between the -auto and the $SGE_ROOT/util/inst_sge_modules/my_configuration.conf components.

   ```
   # $SGE_ROOT/inst_sge -usm -auto $SGE_ROOT/util/inst_sge_modules/my_configuration.conf
   ```

6. Uninstall the master host.
   Type the following command as one string, with a space between the -auto and the $SGE_ROOT/util/inst_sge_modules/my_configuration.conf components.

   ```
   # $SGE_ROOT/inst_sge -um -auto $SGE_ROOT/util/inst_sge_modules/my_configuration.conf
   ```

Additional Software for the Microsoft Operating System

Microsoft Windows Services for UNIX (SFU) and Microsoft Subsystem for UNIX-based Applications (SUA) make it possible to integrate some Windows operating systems into existing UNIX environments. SFU and SUA provide components that simplify network administration and user management across the UNIX and Windows platforms.

Additional Software

The following sections describe the Microsoft Windows Services for UNIX (SFU) and Microsoft Subsystem for UNIX-based Applications (SUA) in detail.

Topic	Description
Microsoft Services for UNIX	Learn how Microsoft Windows Services for UNIX (SFU) makes it possible to integrate some Windows operating systems into existing UNIX environments.
Microsoft Subsystem for UNIX-based Applications	Learn how Microsoft Subsystem for UNIX-based Applications (SUA) makes it possible to integrate some Windows operating systems into existing UNIX environments.
Changing Default Behavior to Case Sensitivity	Choose between default behavior and case sensitivity for object names.
Disabling DEP	Learn how to enable DEP for different Windows platforms.
Enabling suid Behavior for Interix Programs	Learn how to enable suid Behavior for Interix Programs.

Microsoft Services for UNIX

Microsoft Windows Services for UNIX (SFU) makes it possible to integrate some Windows operating systems into existing UNIX environments. SFU provides components that simplify network administration and user management across the UNIX and Windows platforms. You can use SFU to do the following:

- Integrate Windows hosts into Grid Engine clusters. This means that the execution and client environment of Grid Engine can be used on Microsoft Windows hosts. You must use Grid Engine in combination with SFU for this to occur.
- Access the network file system (NFS). This makes it possible for you to share files between the UNIX and Windows environments.
- Possibly access account and password services on UNIX and Windows systems (PCNFS, NIS) using the user mapping service.
- Synchronize passwords and map authentication credentials between the UNIX and Windows operating systems. You can use the "single sign-on" capability for Windows and UNIX environments.
- Execute UNIX shell scripts and applications to run on Windows platform-based computers in full-featured UNIX environments.

Interix, SFU's UNIX environment subsystem, offers the following features:

- A complete, high-performance UNIX environment. You can use the `csh` shell or the `ksh` shell.
- Several hundred tools and utilities.
- A complete set of development tools and libraries that make it possible to port your UNIX-based applications to the Interix sub-system.

SFU is an essential prerequisite to install Grid Engine on Microsoft Windows Server 2003, Windows XP Professional with at least Service Pack 1, Windows 2000 Server with at least Service Pack 3, or Windows 2000 Professional with at least Service Pack 3.
For Microsoft Windows Server 2003 Release 2, Windows Server 2008, Windows Vista Enterprise, Windows Vista Ultimate, please see Microsoft Subsystem for UNIX-based Applications.

Unsupported Grid Engine Functionality

The following Grid Engine components are not supported in a Microsoft Windows environment and cannot be used on Windows Hosts even though they are standard to a Grid Engine installation:

- Master and Scheduler (`sge_qmaster` and `sge_shadowd`)
- Graphical User Interface (`qmon`)
- DRMAA
- `qsh` client command

Topic	Description
How to Install Services for Unix	Learn how to install Microsoft Services for Unix.
Troubleshooting SFU	Learn how to troubleshoot Microsoft Services for Unix.
Configuring User Name Mapping	Learn how to configure user name mapping.

How to Install Microsoft Services for Unix

System Requirements

The following system requirements apply to the SFU installation:

- You must install at least Version 5.0 of Internet Explorer, before running the SFU setup.
- You cannot install SFU on a system running Microsoft Services for Network File System. For example, Microsoft Services for NFS is a component of Windows Storage Server 2003.
- You must install the latest Windows service pack before installing SFU and Grid Engine. Then, you can install additional Windows service packs as they become available.
- The hard disk requirements for an SFU installation depend on which components you need to install. The following installation parameters apply:
  - The minimum disk space required is 20 MB.
  - The maximum disk space requirement is 360 MB.
  - SFU must be installed on a partition that is formatted with the NTFS file system.
- You must disable Data Execution Prevention (DEP). DEP is not compatible with some parts of SFU and might cause segmentation faults. See http://support.microsoft.com/kb/875352 for more information about DEP. To disable DEP, see Disabling DEP.

You can find more details concerning SFU requirements at http://www.microsoft.com/windows/sfu/.

Services for UNIX Installation

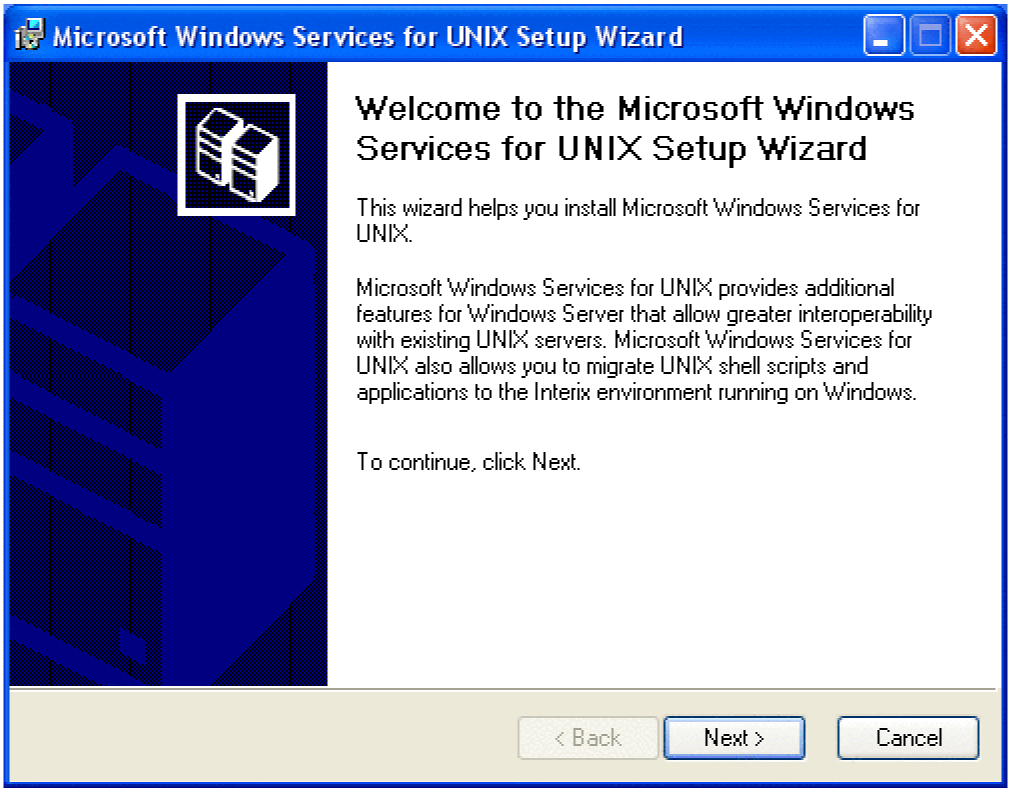
Microsoft's SFU is required to install Grid Engine successfully. You can download SFU from Microsoft. Search the site for "Windows Services for Unix" to find the current download information.

1. Get the SFU distribution media.

2. Execute the application to unzip the files into a directory.
   This directory must be located on a file system that has at least 480 MBytes free space.

3. Log in to the Windows system with the **`Administrator`** account.

4. Start the **`setup.exe`** application that you unpacked previously.



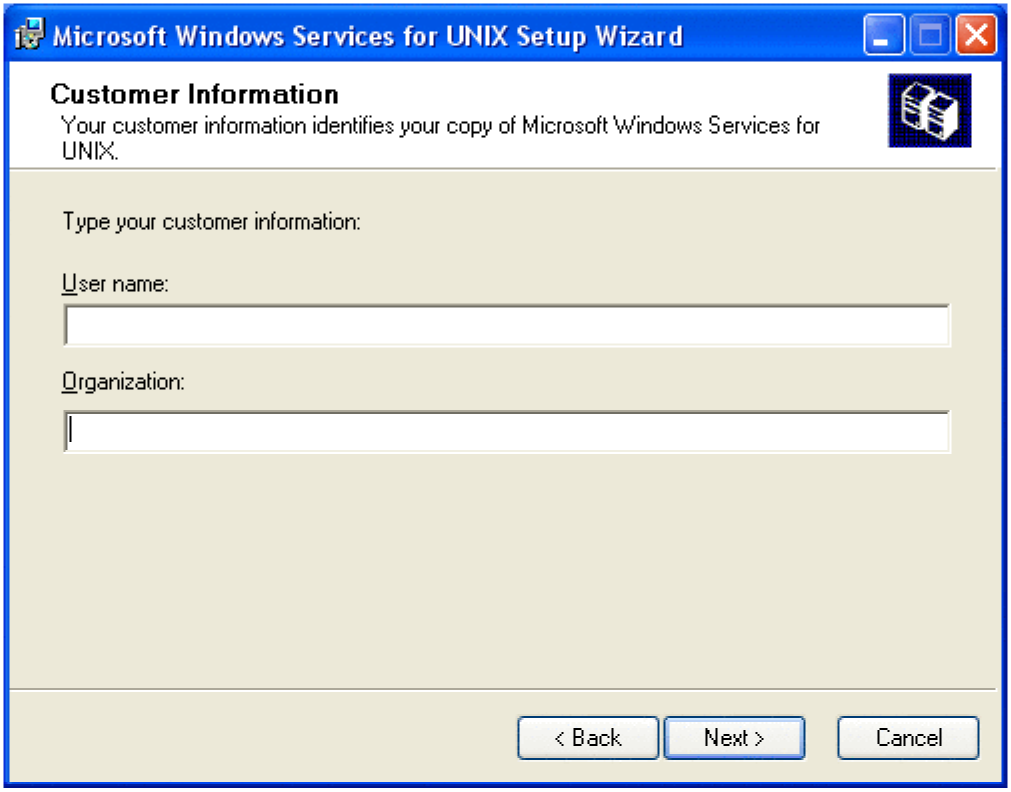
Welcome to the Microsoft Windows Services for UNIX Setup Wizard

This wizard helps you install Microsoft Windows Services for UNIX.

Microsoft Windows Services for UNIX provides additional features for Windows Server that allow greater interoperability with existing UNIX servers. Microsoft Windows Services for UNIX also allows you to migrate UNIX shell scripts and applications to the Interix environment running on Windows.

To continue, click Next.

5. Enter your User name and Organization.



Customer Information

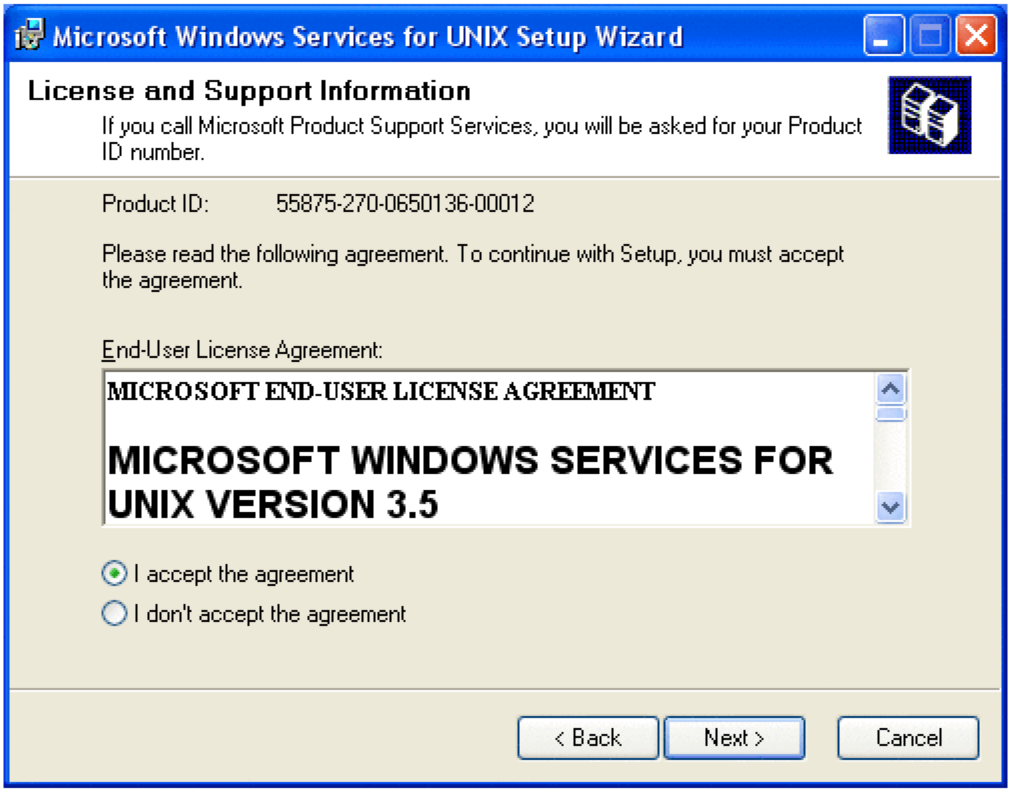
Your customer information identifies your copy of Microsoft Windows Services for UNIX.

Type your customer information:

User name:

Organization:

6. Accept the license agreement for SFU.



Microsoft Windows Services for UNIX Setup Wizard

License and Support Information

If you call Microsoft Product Support Services, you will be asked for your Product ID number.

Product ID: 55875-270-0650136-00012

Please read the following agreement. To continue with Setup, you must accept the agreement.

End-User License Agreement:

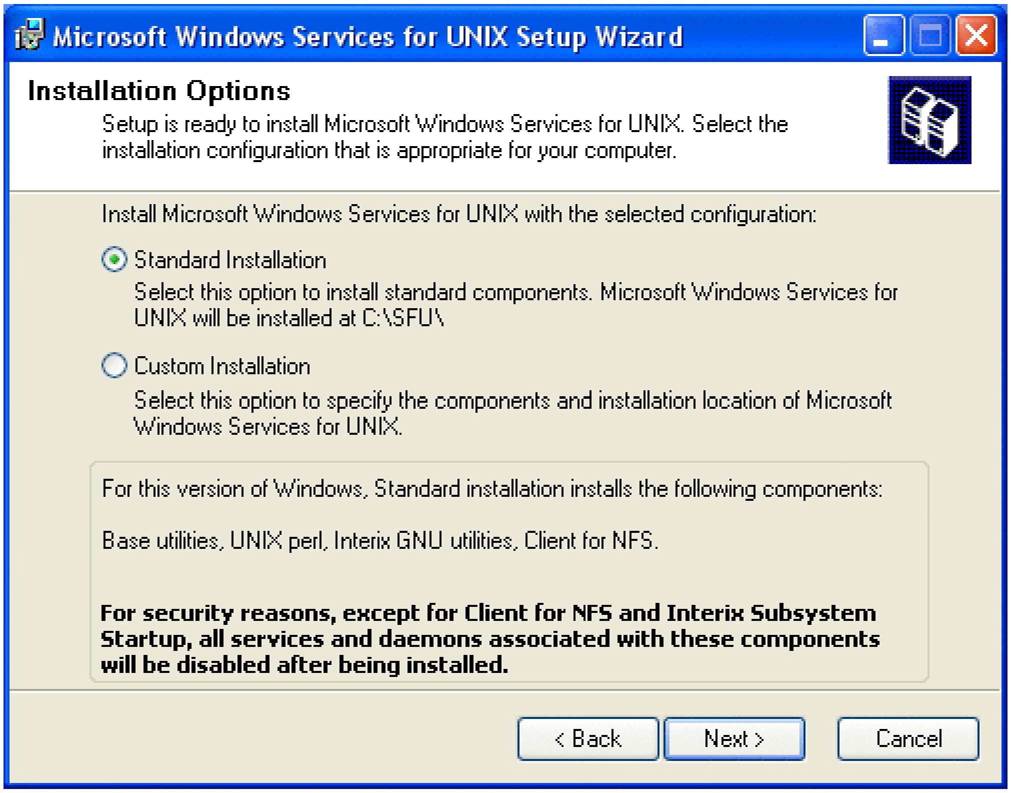
MICROSOFT END-USER LICENSE AGREEMENT

MICROSOFT WINDOWS SERVICES FOR UNIX VERSION 3.5

● I accept the agreement

○ I don't accept the agreement

7. Choose the standard installation (recommended) or the custom installation.



Microsoft Windows Services for UNIX Setup Wizard

Installation Options

Setup is ready to install Microsoft Windows Services for UNIX. Select the installation configuration that is appropriate for your computer.

Install Microsoft Windows Services for UNIX with the selected configuration:

● Standard Installation

Select this option to install standard components. Microsoft Windows Services for UNIX will be installed at C:\SFU\

○ Custom Installation

Select this option to specify the components and installation location of Microsoft Windows Services for UNIX.

For this version of Windows, Standard installation installs the following components:

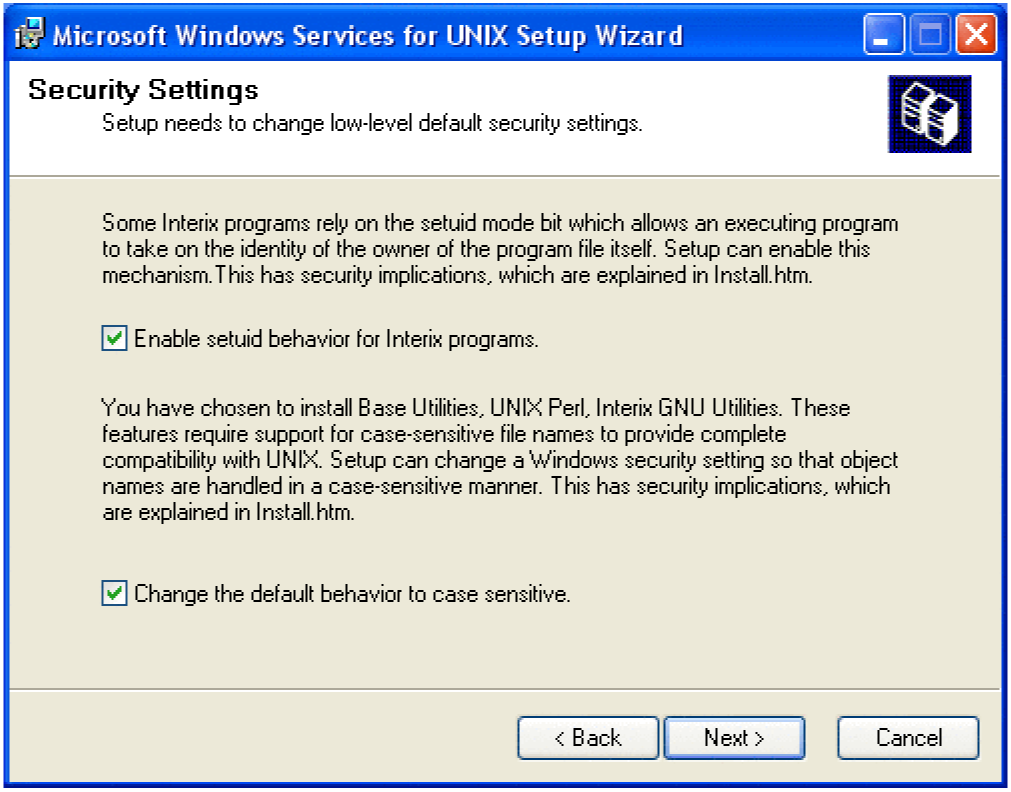
Base utilities, UNIX perl, Interix GNU utilities, Client for NFS.

For security reasons, except for Client for NFS and Interix Subsystem Startup, all services and daemons associated with these components will be disabled after being installed.

If disk space is limited, you might want to choose the custom installation. Make sure that you install at least the following components:

- Utilities -> Base Utilities
- Interix GNU components -> Interix GNU utilities
- Remote connectivity components -> Telnet Server and Windows Remote Shell
- If you intend to use NFS shared file systems, you also need Authentication tools for NFS -> User Mapping and Server for NFS Authentication.

8. Depending on the Windows operating system, you might be presented with the following two options concerning SFU security settings, shown in the dialog box below:



Microsoft Windows Services for UNIX Setup Wizard

Security Settings

Setup needs to change low-level default security settings.

Some Interix programs rely on the setuid mode bit which allows an executing program to take on the identity of the owner of the program file itself. Setup can enable this mechanism.This has security implications, which are explained in Install.htm.

☑ Enable setuid behavior for Interix programs.

You have chosen to install Base Utilities, UNIX Perl, Interix GNU Utilities. These features require support for case-sensitive file names to provide complete compatibility with UNIX. Setup can change a Windows security setting so that object names are handled in a case-sensitive manner. This has security implications, which are explained in Install.htm.

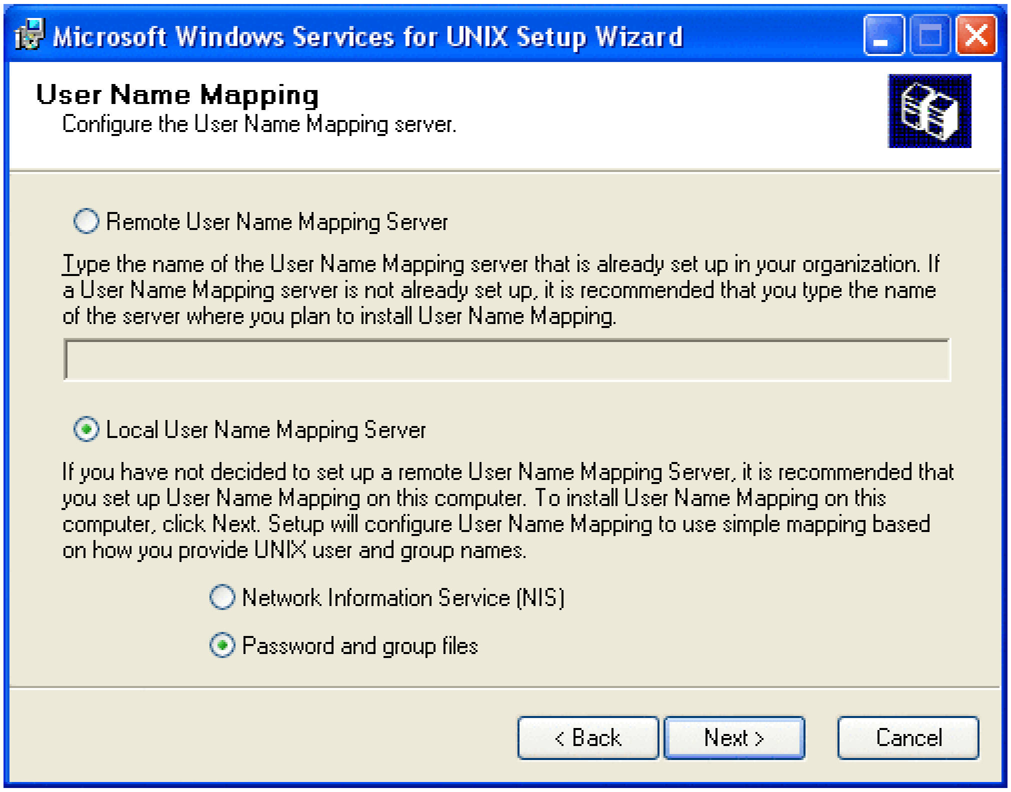
☑ Change the default behavior to case sensitive.

If you need further information, consult Microsoft's SFU documentation.

9. Configure User Name Mapping.

> Note
> User Name Mapping is part of SFU and not part of Sun Grid Engine. Consult Microsoft documentation and support to set up user mapping correctly.

Your selection in the dialog box, shown below, depends on the hosts and services that are currently provided in your Windows and UNIX environments. If there is no Remote User Mapping server in your environment, then you should select Local User Name Mapping Server.



User Name Mapping

Configure the User Name Mapping server.

○ Remote User Name Mapping Server

Type the name of the User Name Mapping server that is already set up in your organization. If a User Name Mapping server is not already set up, it is recommended that you type the name of the server where you plan to install User Name Mapping.

● Local User Name Mapping Server

If you have not decided to set up a remote User Name Mapping Server, it is recommended that you set up User Name Mapping on this computer. To install User Name Mapping on this computer, click Next. Setup will configure User Name Mapping to use simple mapping based on how you provide UNIX user and group names.

○ Network Information Service (NIS)

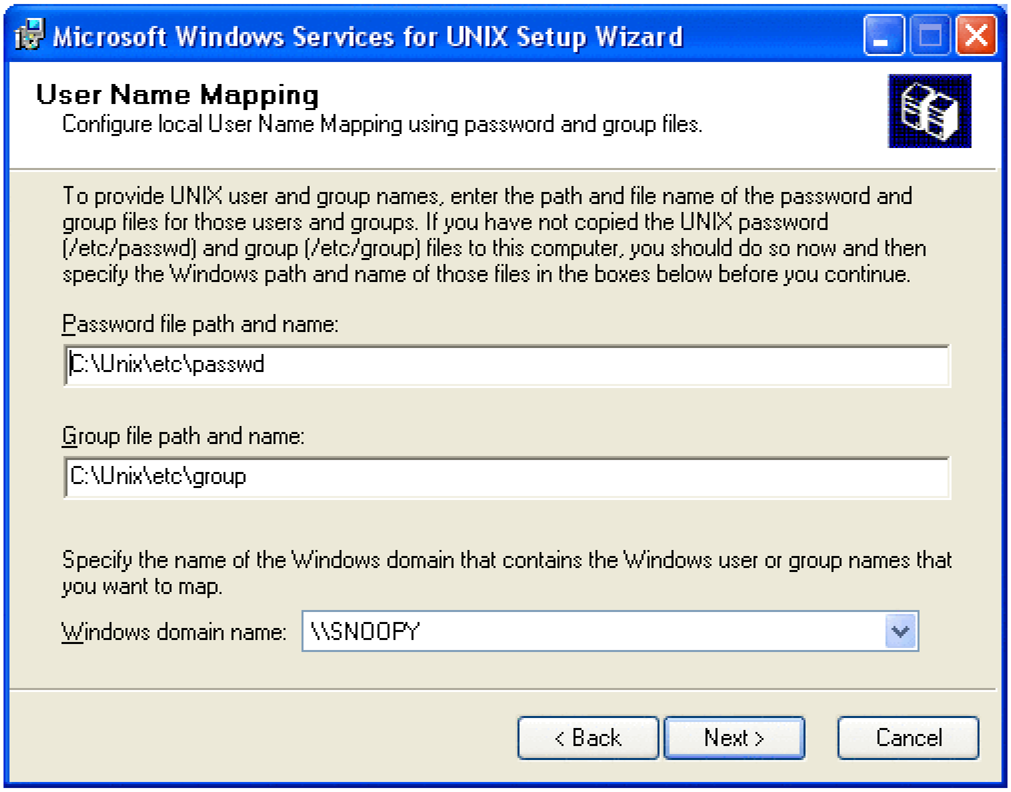
● Password and group files

Note

You should install SFU and enable the User Name Mapping service on your host that acts as a Domain Controller for your windows environment. All other hosts should contact that Remote User Name Mapping Server. If you choose Local User Name Mapping Server, then you might either select Network Information Services (NIS) to access your `passwd` and `group` NIS-maps. Otherwise, select l if you can provide the files yourself.

See Configuring User Name Mapping for further details.

10. Depending on your previous selections, you can either enter the NIS Domain name and NIS Server name or the path of the **`passwd`** and **`group`** files.



Microsoft Windows Services for UNIX Setup Wizard

User Name Mapping

Configure local User Name Mapping using password and group files.

To provide UNIX user and group names, enter the path and file name of the password and group files for those users and groups. If you have not copied the UNIX password (/etc/passwd) and group (/etc/group) files to this computer, you should do so now and then specify the Windows path and name of those files in the boxes below before you continue.

Password file path and name: C:\Unix\etc\passwd

Group file path and name: C:\Unix\etc\group

Specify the name of the Windows domain that contains the Windows user or group names that you want to map.

Windows domain name: \\SNOOPY

Below is an example of the files that have the standard UNIX format. This means that you can also use your `/etc/passwd` and `/etc/group` files from your UNIX environment.

```
C:\Unix\etc\passwd
root:x:0:0:UNIX root user:/home/root:/bin/tcsh
user1:x:1002:100:Full name of user1:/home/user1:/bin/tcsh
C:\Unix\etc\group
root::0:
```

Note

Some NIS maps do not contain an entry for the root user. If this is the case, follow these steps to map Administrator to root:

a. First create a password file containing the root entry.
b. If the SFU installation is finished, start the Services for UNIX Administration application and create the mapping: `Administrator <-> root`.
c. Switch to NIS mapping.
d. Use simple mapping or add manual mappings.
   At this point the installation starts installing components. Wait until all components are installed.

11. When the installation process finishes, you might need to reboot the machine, depending on the version of Windows that you are using.

12. Make sure that the Interix Subsystem Startup starts during boot time.
    If you intend to use NFS shares and user mapping, then also start Client for NFS and User Name Mapping.
    Depending on the installation options and your version of the Windows operating system, one or more of these services are disabled by default.

Post SFU Installation Tasks

There are several steps you should follow after you install the SFU software.

1. Before you start using SFU and install Grid Engine, check that the user mapping is working correctly by following these steps:
   a. Open an Interix shell locally on the Interix host.

b. Use the `login` command to switch to a known user that is not the Administrator.
c. Verify the access permissions for NFS shares that should be accessible to that user.
d. Try to access these network resources. If the user cannot access a network drive and it is a NFS shared drive, most likely the User Name Mapping is not working correctly.

2. Check users' home directories.
   To enable the automounting of the users' home directories, use the following series of menus:

   ```
   Control Panel -> Administrative Tools -> Computer Management -> Users -> Properties -> Profile
   ```

   Click connect to, select a drive letter, and enter the path of the user's home directory in UNC notation: `\\<server>\<share>\<user home>`.
   Within the Interix subsystem, you might access all network shares through the special directory: `/net/`server/share.
   You might also create links to these directories to access the shares directly, for example, `ln -s /net/myserver/export/share00/home /`home. See also 5. Mount network shares below.

3. Enable Administrator names on your machines.
   Make sure that the administrator accounts on all machines that are enabled as execution hosts for Grid Engine use the same account name, such as Administrator.
   Also make sure that this user has manager privileges in your Sun Grid Engine cluster. If this is not the case, add the privileges using `qconf -am` administrator before the installation of the execution daemon.

4. Set the CLI commands.
   This starts an editor. Make sure to set the EDITOR environment variable to `vi`, or your preferred UNIX editor, within the Interix subsystem before you start using UNIX commands.

5. Mount network shares.
   There are two ways to mount network shares to the Interix host:
   - Interix provides a directory in which it makes available all network shares it finds by browsing the network. This works similar to the "Network" folder in the "Windows Explorer", which also searches the network for available shares. The directory where these network shares are provided is `/net`. In this `/net` directory all automatically discovered hosts can be found as subdirectories. Each of this subdirectories will list all network shares of this host as subdirectories, again. The syntax is `/net/`server/share, e.g. `/net/myserver/`home. Eventhough a `ls /net` may list no content but perhaps some errors, a `ls /net/`server/share will list the content of the share. The errors or missing host names seem to be bug in displaying the names, but this bug doesn't affect the functionality of the automatically discovered shares.
     To make these shares available under the same path as on a UNIX host, it's recommended to create links to these shares. For example `ln -s /net/myserver/home /`home makes the users' UNIX home directories accessible through `/`home`/`username on the Windows host.
     The automatically discovered shares are available at boot time for all users who have the permissions to access the shares. Interix will discover the same kinds of network shares (SMB, NFS, CIFS and so on) the "Windows Explorer" can discover. For this, the proper network client must be installed and the permissions must be sufficient.
   - Network shares can also be mapped to drive letters by using the `net` command of Windows. The syntax is `/dev/fs/C/Windows/System32/net.exe <drive letter>: \\<computername>\<sharename> <devicename>`.
     For example:

     ```
     /dev/fs/C/Windows/System32/net.exe Z: \\\\myserver\\home
     ```

     This drive is now accessible through `/def/fs/Z`. A link can be created to this drive to use the same path as on a UNIX host.

     > Note
     > As shown in the example above, all backslashes must be written twice because the shell interprets a single backslash as an escape character.

Troubleshooting SFU

The following section describes some common problems that users may encounter when installing and using Grid Engine in a Services for UNIX environment on a Windows system.

- Impossible to connect to the Interix subsystem through **`telnet`** or **`rsh`**.
  Make sure that the correct services are started. The corresponding Windows services must be disabled. The Interix versions of `telnetd` and `rshd` must be started. You can do this task by removing the pound sign (#) from the following lines in `/etc/inetd.conf`:

  ```
  #telnet stream tcp nowait NULL /usr/sbin/in.telnetd in.telnetd -i
  #shell stream tcp nowait NULL /usr/sbin/in.rshd in.rshd -a
  ```

  If you still cannot connect to the machine, check your firewall configuration. Do not block connections to corresponding ports:

  ```
  Service  |  Ports
  ---------+-----------
  ftp      |  20, 21
  ssh      |  22
  telnet   |  23
  rsh      |  514
  ```

- The wrong default login shell is started. Why?
  Both the `.rhost` and `host.equiv` authentications fail if new user accounts are created and if the passwords of existing users are changed. In this case, the command `regpwd` needs to be called. After that, follow the steps to register passwords correctly.
- Why is the access to NFS mounted home directories slow?
  User Name Mapping might be the cause. For a large number of user maps, installing User Name Mapping on a Domain Controller improves performance by reducing network traffic. You can create a User Name Mapping server pool. This method means that you use DNS round-robin to create a pool of computers running User Name Mapping. This provides improved performance on wide area networks and provides failover when one of the servers is no longer available.
- How can I map user **`root`** if it does not exist in the NIS maps?
  First create a `passwd` file which contains an entry for the user `root`. Then, explicitly map the `root` account (no basic mapping) using the created `passwd` file. Finally, change the mapping to use the NIS maps. Note that the previous root mapping will persist.
- NIS Server cannot be contacted during the SFU installation.
  Interrupt the SFU installation and make sure that there is no other service or application running which already configures or uses the NIS server. If this is the case, then disable this service for the duration of the SFU installation.
- The Interix Subsystem of SFU or the User Mapping is not enabled after reboot.
  Make sure that Interix Subsystem Startup and User Name Mapping are automatically started after machine reboot. Also if you use NFS mounted directories, enable the service by default: Client for NFS.
- Queues stick in unknown state for a very long time.
  After the installation or restart of an execution host, the corresponding queues have attached the unknown (u) state for a very long time. This is normal behavior for Windows machines. After a full load report interval, the u state should be gone. If this is not the case, then check that the `sge_execd` has been started on the corresponding machine.

Configuring User Name Mapping

User Name Mapping acts as a single clearinghouse that provides centralized user mapping services for the NFS client of Interix. User Name Mapping provides a map between the Windows users and groups on the NFS client, and the corresponding UNIX users and groups on the NFS server. In principle, these user and group names might not be identical. However, for users who intend to use Sun Grid Engine, these names must be identical.

User Name Mapping lets you maintain a single mapping database for the entire enterprise. This feature makes it easy to configure authentication for multiple computers running Windows Services for UNIX.

User Name Mapping also permits one-to-many mapping. This lets you associate multiple Windows accounts with a single UNIX account. To do this, you can use simple maps, which map Windows and UNIX accounts with identical names. You can also create advanced maps to associate Windows and UNIX accounts with different names, which you can use with simple maps. This feature can be useful, for example, when you do not need to maintain separate UNIX accounts for individuals and would rather use a few accounts to provide different classes of access permission.

> Note
> For information about simple and advanced maps, see "Simple and Advanced Maps" in Help for Services for UNIX. After the installation has finished, you can find Help for Services for UNIX in Start -> Programs -> Services for UNIX -> Help for Services for UNIX.

Microsoft Subsystem for UNIX-based Applications

Microsoft Subsystem for UNIX-based Applications allows you to integrate Windows operating systems with the existing UNIX environments. This subsystem provides components that simplify network administration and user management across UNIX and Windows platforms. You can use this subsystem to perform the following:

- Integrate Windows hosts with Sun Grid Engine clusters - Enables you to use the execution and client environment of Sun Grid Engine on Microsoft Windows hosts. You must use Sun Grid Engine in combination with Microsoft Subsystem for UNIX-based Applications for this to happen.
- Access the network file system (NFS) - Enables you to share files between the UNIX and Windows environments.
- Synchronize passwords and map authentication credentials between the UNIX and Windows operating systems - Enables you to use the 'single sign-on' capability for Windows and UNIX environments.
- Execute UNIX shell scripts and applications - Enables you to run shell scripts and applications on Windows platform-based computers in full-featured UNIX environments.

Microsoft Subsystem for UNIX-based Application's UNIX environment subsystem, Interix, offers the following features:

- A complete, high-performance UNIX environment. You can use the `csh` or `ksh` shell.
- Several hundred tools and utilities.
- A complete set of development tools and libraries that make it possible to port your UNIX-based applications to the Interix sub-system.

Microsoft Subsystem for UNIX-based Applications is an essential prerequisite to install Sun Grid Engine on Microsoft Windows Server 2003 Release 2, Windows Server 2008, Windows Vista Enterprise, and Windows Vista Ultimate. For Microsoft Windows Server 2003, Windows XP Professional with at least Service Pack 1, Windows 2000 Server with at least Service Pack 3, or Windows 2000 Professional with at least Service Pack 3, see Microsoft Services for UNIX.

Unsupported Sun Grid Engine Functionality

The following Grid Engine components are not supported in the Microsoft Windows environment and cannot be used on Windows hosts even though they are standard to a Sun Grid Engine installation:

- Master and Scheduler (`sge_qmaster` and `sge_shadowd`)
- Graphical user interface (qmon)
- DRMAA
- `qsh` client command

Topic	Description
How to Install a Subsystem for UNIX-based Applications	Learn how to install a subsystem for UNIX-based applications.
Troubleshooting Microsoft Subsystem for UNIX-based Applications	Learn how to troubleshoot a subsystem for UNIX-based applications.

How to Install a Microsoft Subsystem for UNIX-based Applications

This section describes how to install a Microsoft Subsystem for UNIX-based Applications.

System Requirements

The system requirements for a Subsystem for UNIX-based Applications installation are:

- Microsoft Windows Server 2003 Release 2, Windows Server 2008, Windows Vista Enterprise, or Windows Vista Ultimate. Windows Vista Business and all lower Vista versions are not supported by this subsystem.
- You must install the latest Windows service pack before installing Subsystem for UNIX-based Applications and Sun Grid Engine. You can install additional Windows service packs as they become available.
- The hard disk requirement for a Subsystem for UNIX-based Applications installation depends on the components that you are planning to install. The following installation parameters apply.
  - The minimum disk space required is 182 MBytes.
  - The maximum disk space required is approximately 350 MBytes.
  - Subsystem for UNIX-based Applications must be installed on a partition that is formatted with the NTFS file system.
- You must disable the Data Execution Prevention (DEP) feature. The DEP feature is not compatible with some parts of Subsystem for UNIX-based Applications and might cause segmentation faults. For more information about DEP, see http://support.microsoft.com/kb/875352. For information on how to disable DEP, see Disabling DEP.

You can find additional information about Subsystem for UNIX-based Applications requirements at http://technet.microsoft.com/en-us/library/cc779522.aspx.

Installing Subsystem for UNIX-based Applications

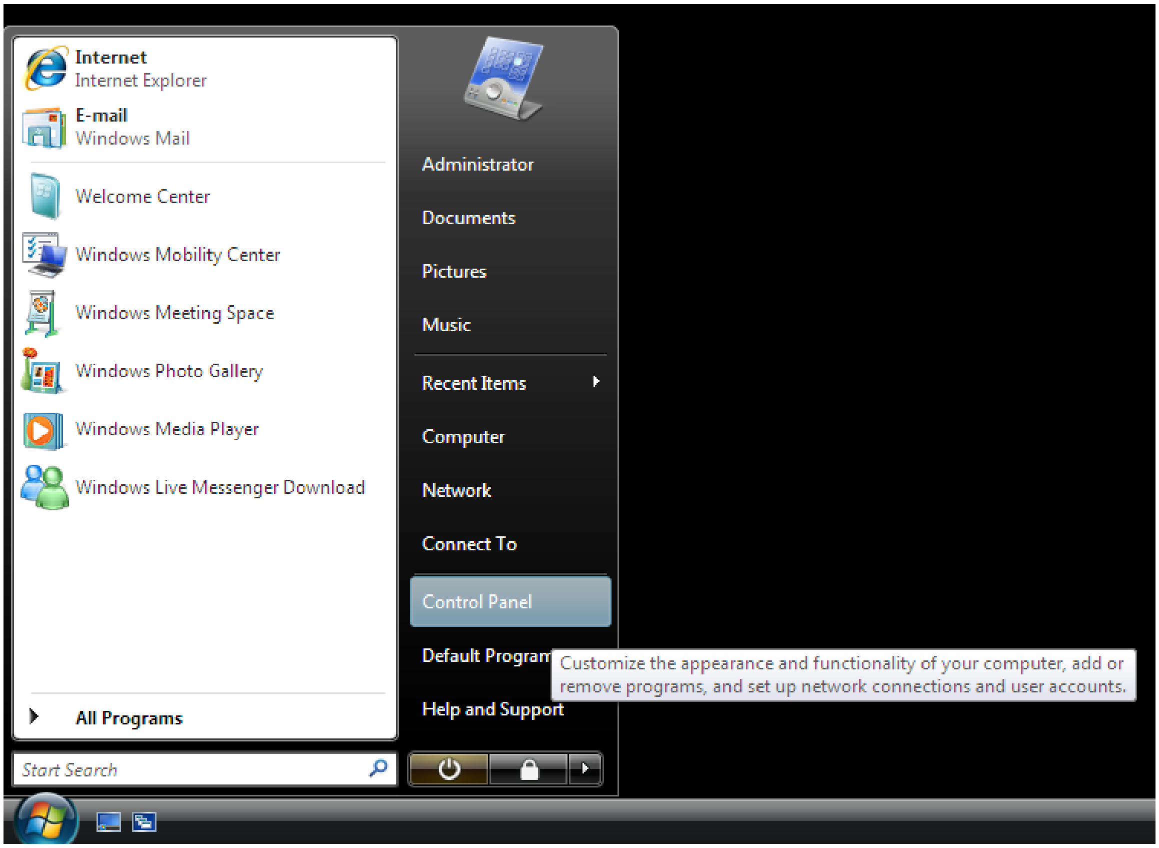
Microsoft Subsystem for UNIX-based Applications is required for installing Sun Grid Engine on Windows Vista, Windows Server 2008, and Windows 2003 R2. Subsystem for UNIX-based Applications is partially delivered with these versions of Windows, but you also need to download some components from the Microsoft web site.

Steps

1. Install the components of Subsystem for UNIX-based Applications that are delivered with Windows.

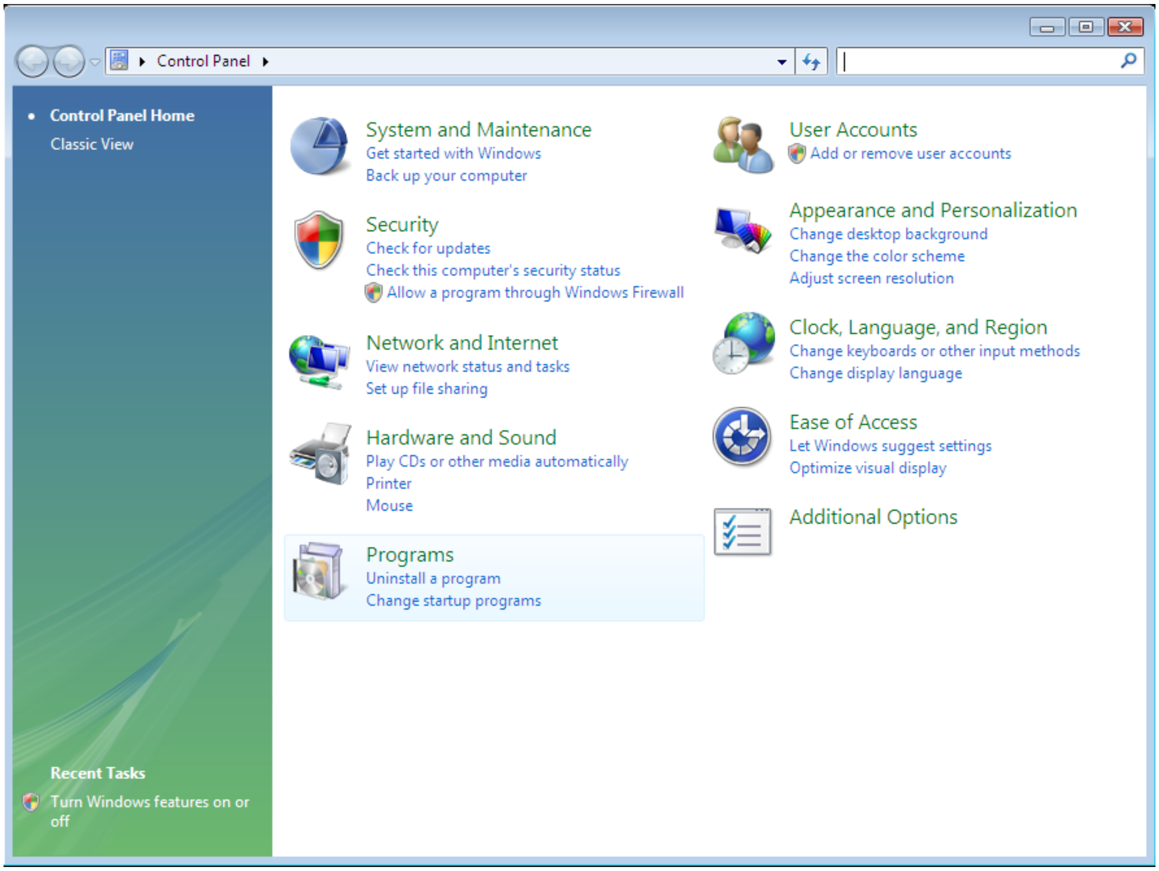
   > ⚠ Note
   >
   > In this procedure Windows Vista is used as an example. Other supported Windows Versions function similarly. You must have the right administrative privileges to perform the installation.

   a. Click Start.

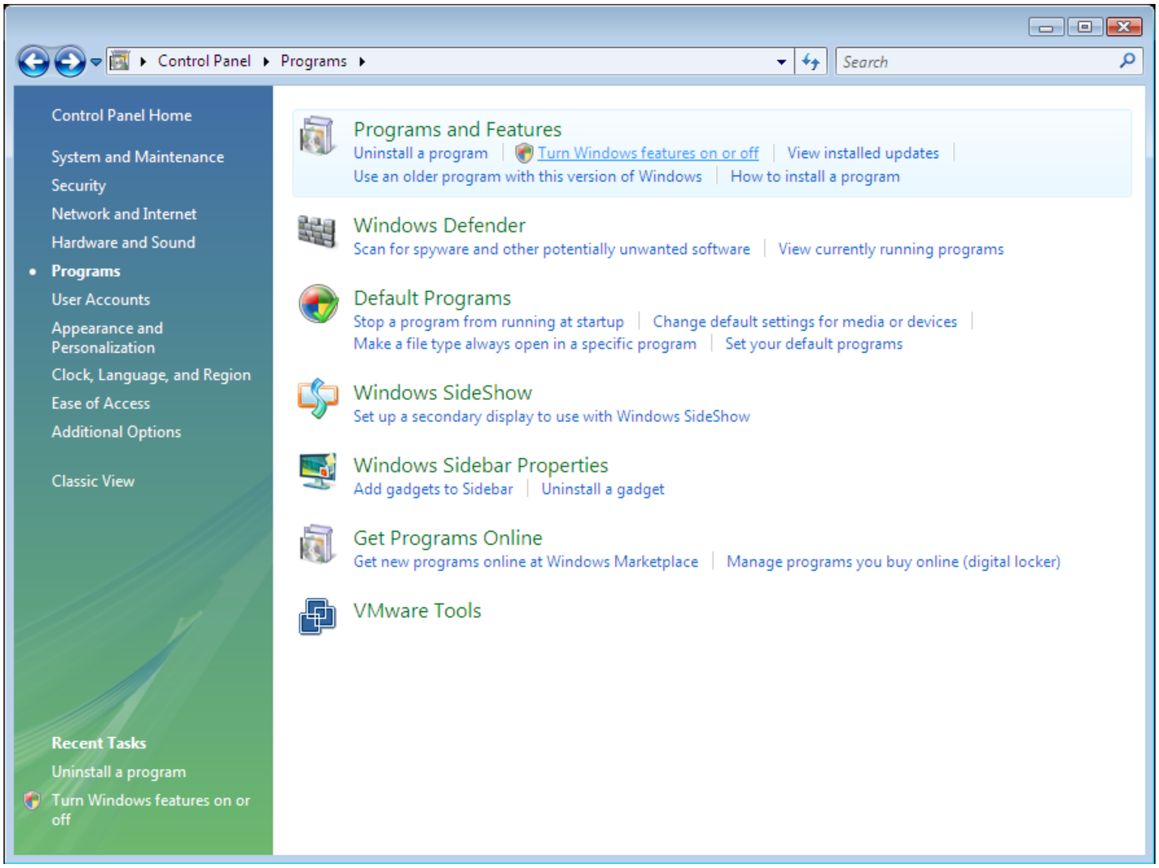


Customize the appearance and functionality of your computer, add or remove programs, and set up network connections and user accounts.

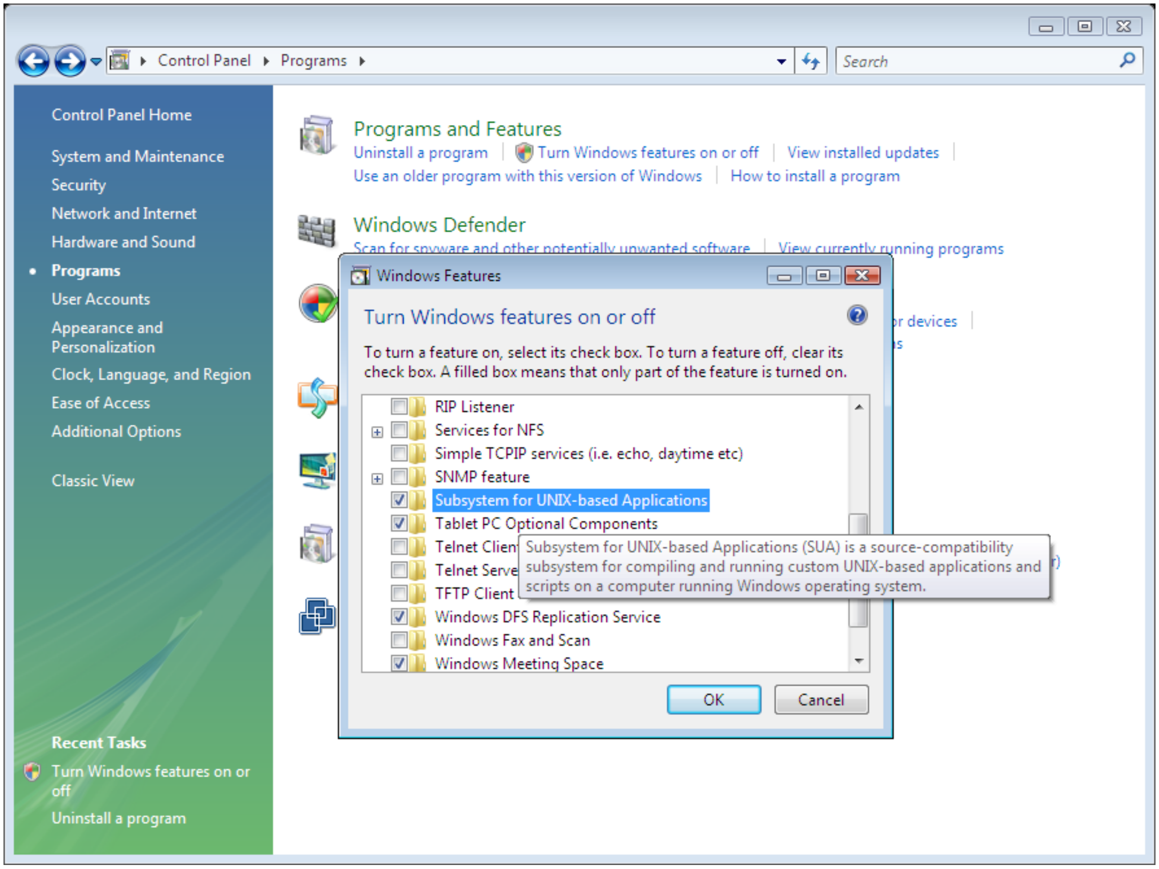
b. Click Control Panel.



c. Click Programs.


Control Panel ▸ Programs ▸
Search
Control Panel Home
System and Maintenance
Security
Network and Internet
Hardware and Sound
Programs
User Accounts
Appearance and Personalization
Clock, Language, and Region
Ease of Access
Additional Options
Classic View
Recent Tasks
Uninstall a program
Turn Windows features on or off
Programs and Features
Uninstall a program | Turn Windows features on or off | View installed updates | Use an older program with this version of Windows | How to install a program
Windows Defender
Scan for spyware and other potentially unwanted software | View currently running programs
Default Programs
Stop a program from running at startup | Change default settings for media or devices | Make a file type always open in a specific program | Set your default programs
Windows SideShow
Set up a secondary display to use with Windows SideShow
Windows Sidebar Properties
Add gadgets to Sidebar | Uninstall a gadget
Get Programs Online
Get new programs online at Windows Marketplace | Manage programs you buy online (digital locker)
VMware Tools

d. Click the Turn Windows features on or off option from the Programs and Features panel. The Windows Features screen appears.


Control Panel ▸ Programs ▸
Search
Control Panel Home
System and Maintenance
Security
Network and Internet
Hardware and Sound
Programs
User Accounts
Appearance and Personalization
Clock, Language, and Region
Ease of Access
Additional Options
Classic View
Recent Tasks
Turn Windows features on or off
Uninstall a program
Programs and Features
Uninstall a program | Turn Windows features on or off | View installed updates | Use an older program with this version of Windows | How to install a program
Windows Defender
Scan for spyware and other potentially unwanted software | View currently running programs
Windows Features
OK
Cancel

Turn Windows features on or off

To turn a feature on, select its check box. To turn a feature off, clear its check box. A filled box means that only part of the feature is turned on.

☐ RIP Listener
⊞ ☐ Services for NFS
☐ Simple TCPIP services (i.e. echo, daytime etc)
⊞ ☐ SNMP feature
☑ Subsystem for UNIX-based Applications
☑ Tablet PC Optional Components
☐ Telnet Clien
☐ Telnet Serve
☐ TFTP Client
☑ Windows DFS Replication Service
☐ Windows Fax and Scan
☑ Windows Meeting Space

Subsystem for UNIX-based Applications (SUA) is a source-compatibility subsystem for compiling and running custom UNIX-based applications and scripts on a computer running Windows operating system.

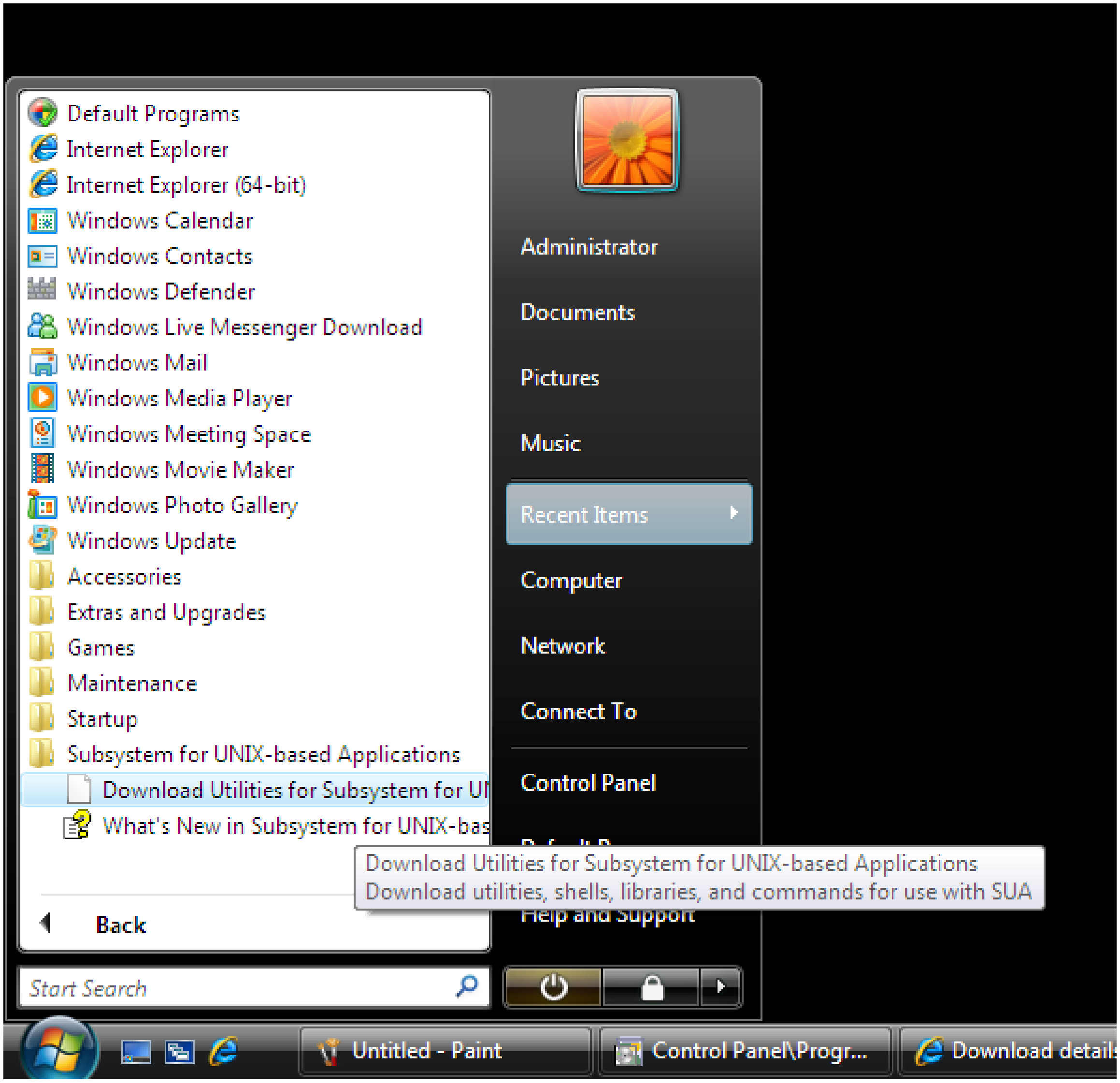
e. Select the Subsystem for UNIX-based Applications option.

f. You can also open the Services for NFS tree and select the appropriate option, if you prefer to use NFS shares.

> Note
>
> Ensure that you use SAMBA for networking shares, as you might have trouble setting up an environment that functions correctly with both Subsystem for UNIX-based Applications NFS and Subsystem for Unix NFS clients.

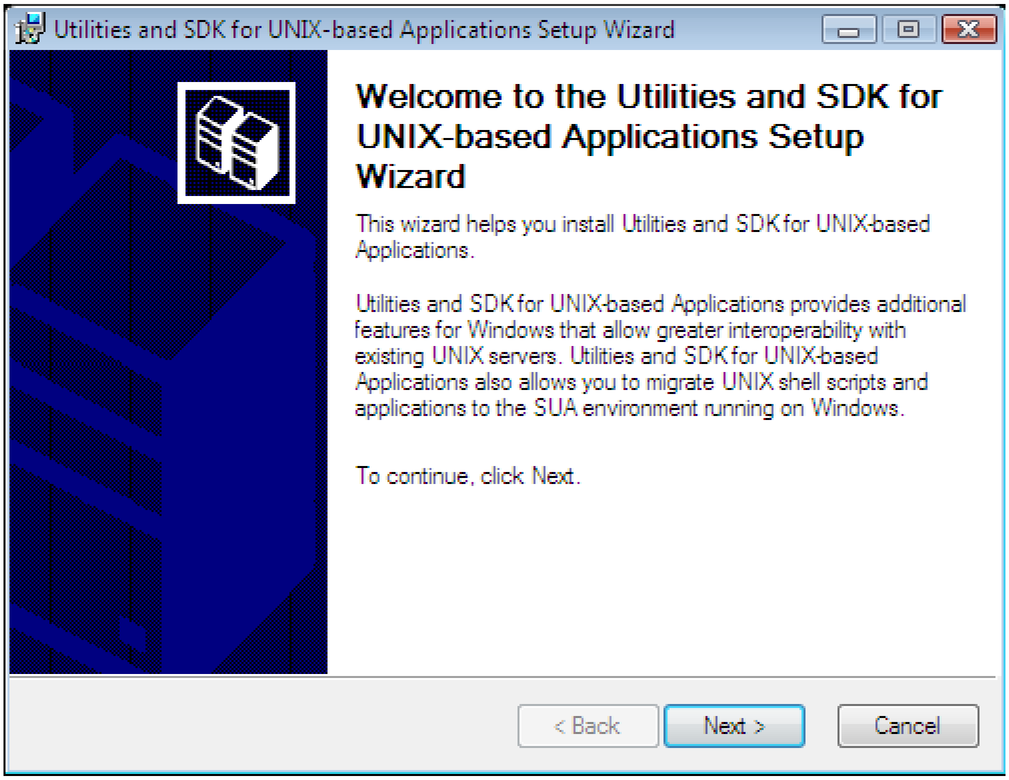
g. Click OK.
Windows installs the new features and might prompt you to insert the Windows installation DVD.

2. Download and install the remaining components of Subsystem for UNIX-based Applications.
   a. Click Start > All Programs.
   You will notice a new folder named Subsystem for UNIX-based Applications in the Windows Start menu. This folder contains the link to the web page where the remaining components of Subsystem for UNIX-based Applications can be downloaded.



Download Utilities for Subsystem for UNIX-based Applications
Download utilities, shells, libraries, and commands for use with SUA

   b. Download the remaining components of Subsystem for UNIX-based Applications and double-click to open the file.
   The file will open in a WinZip Self-Extractor dialog box.
   c. Click Unzip.
   The utility unzips the files.
   d. Click OK.
   The Installer is started and the Subsystem for UNIX-based Applications setup wizard appears.



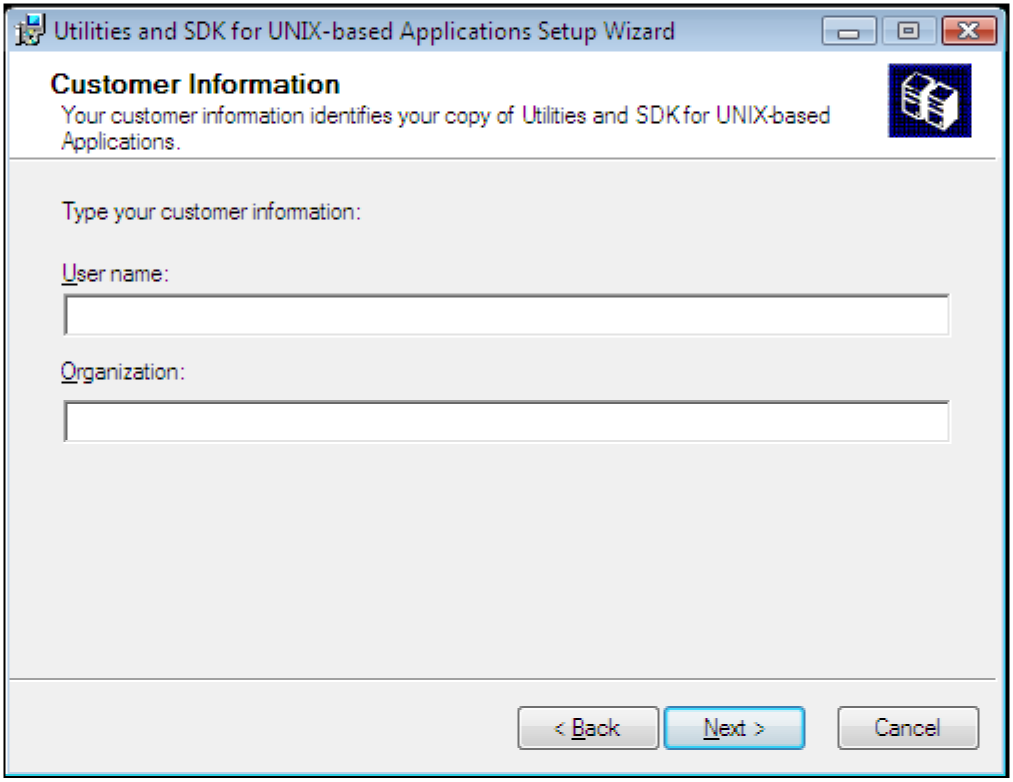
Welcome to the Utilities and SDK for UNIX-based Applications Setup Wizard

This wizard helps you install Utilities and SDK for UNIX-based Applications.

Utilities and SDK for UNIX-based Applications provides additional features for Windows that allow greater interoperability with existing UNIX servers. Utilities and SDK for UNIX-based Applications also allows you to migrate UNIX shell scripts and applications to the SUA environment running on Windows.

To continue, click Next.

e. Click Next.
   The Customer Information screen appears.



Customer Information

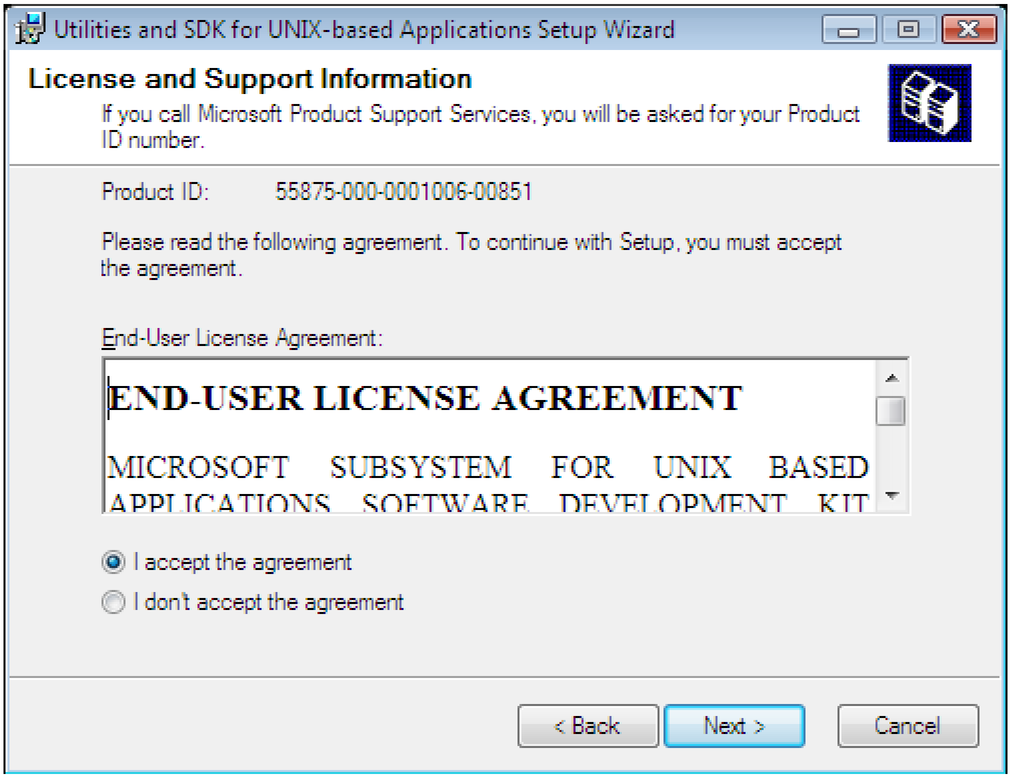
Your customer information identifies your copy of Utilities and SDK for UNIX-based Applications.

Type your customer information:

User name:

Organization:

f. Enter user name and organization name and click Next.
   The License and Support Information screen appears.



License and Support Information

If you call Microsoft Product Support Services, you will be asked for your Product ID number.

Product ID: 55875-000-0001006-00851

Please read the following agreement. To continue with Setup, you must accept the agreement.

End-User License Agreement:

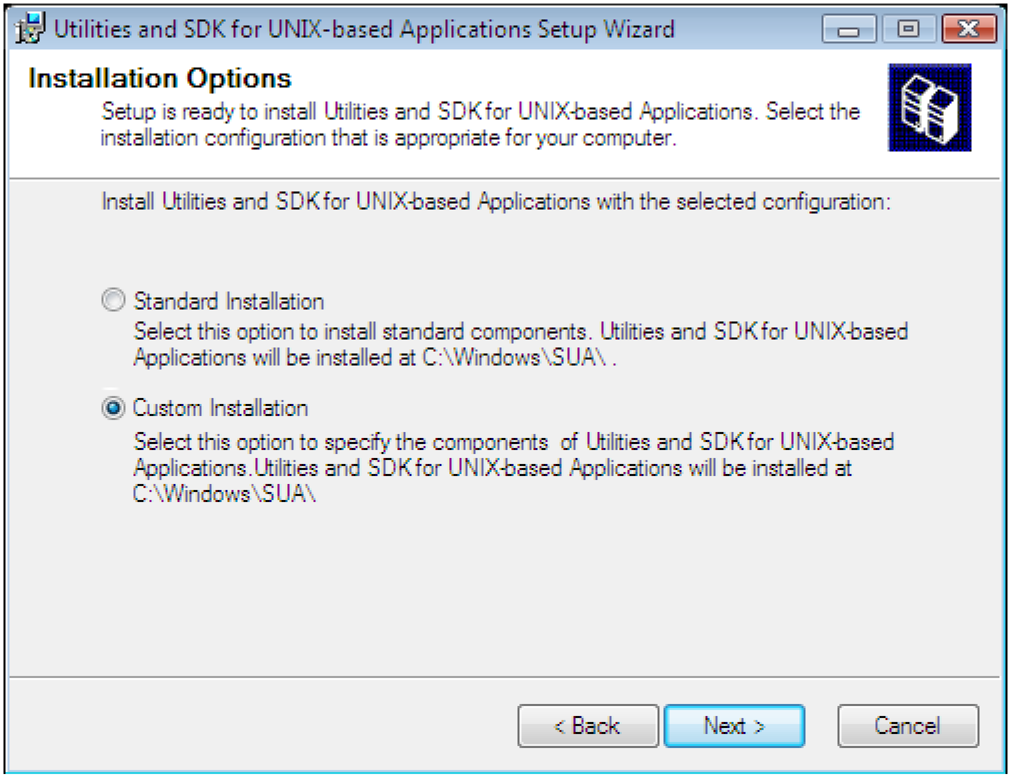
END-USER LICENSE AGREEMENT

MICROSOFT SUBSYSTEM FOR UNIX BASED APPLICATIONS SOFTWARE DEVELOPMENT KIT

● I accept the agreement

○ I don't accept the agreement

g. Accept the terms of the license and click Next.
   The Installation Options screen appears.



Installation Options

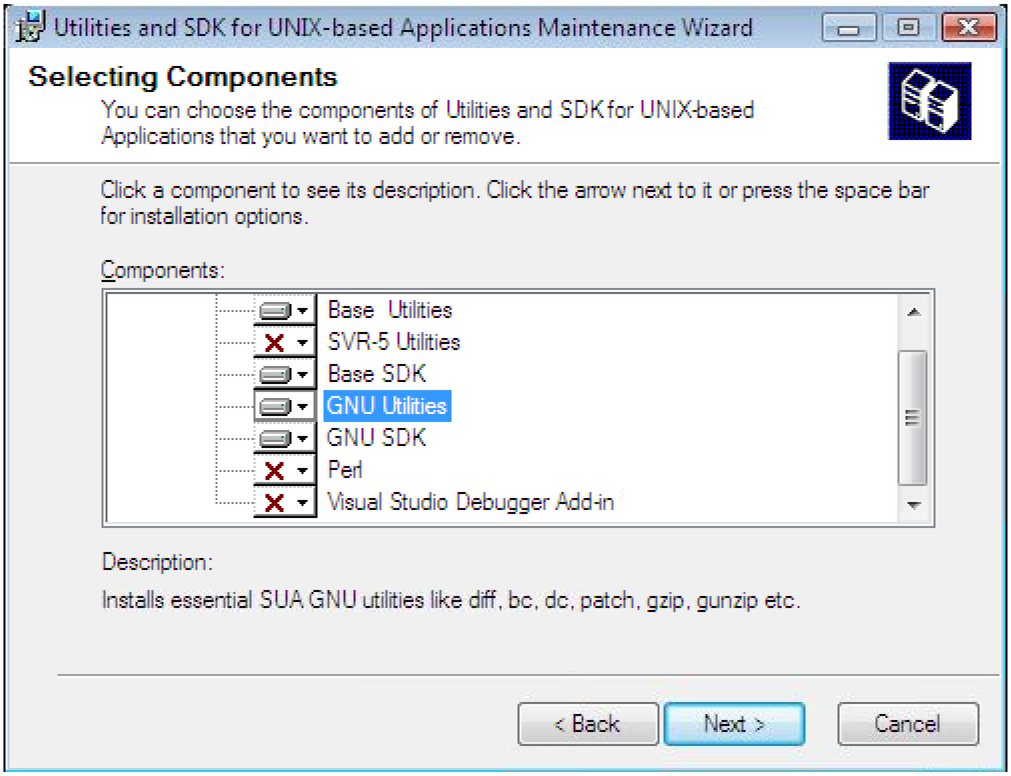
Setup is ready to install Utilities and SDK for UNIX-based Applications. Select the installation configuration that is appropriate for your computer.

Install Utilities and SDK for UNIX-based Applications with the selected configuration:

○ Standard Installation
Select this option to install standard components. Utilities and SDK for UNIX-based Applications will be installed at C:\Windows\SUA\ .

● Custom Installation
Select this option to specify the components of Utilities and SDK for UNIX-based Applications.Utilities and SDK for UNIX-based Applications will be installed at C:\Windows\SUA\

h. Select the Custom Installation option.
   The Selecting Components screen appears.


Utilities and SDK for UNIX-based Applications Maintenance Wizard

Selecting Components

You can choose the components of Utilities and SDK for UNIX-based Applications that you want to add or remove.

Click a component to see its description. Click the arrow next to it or press the space bar for installation options.

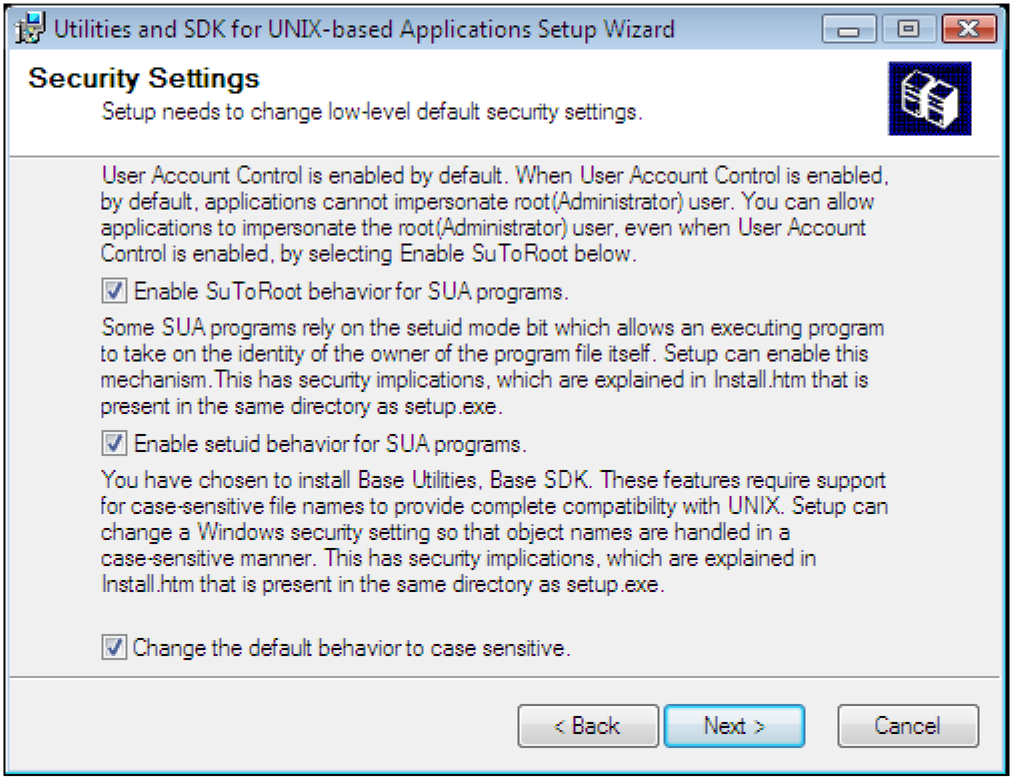
Components:

- Base Utilities
- ✕ SVR-5 Utilities
- Base SDK
- GNU Utilities
- GNU SDK
- ✕ Perl
- ✕ Visual Studio Debugger Add-in

Description:

Installs essential SUA GNU utilities like diff, bc, dc, patch, gzip, gunzip etc.

i. Use the preset selections and select GNU Utilities. Ensure that you also select GNU SDK.
The Security Settings screen appears.



Security Settings

Setup needs to change low-level default security settings.

User Account Control is enabled by default. When User Account Control is enabled, by default, applications cannot impersonate root(Administrator) user. You can allow applications to impersonate the root(Administrator) user, even when User Account Control is enabled, by selecting Enable SuToRoot below.

☑ Enable SuToRoot behavior for SUA programs.

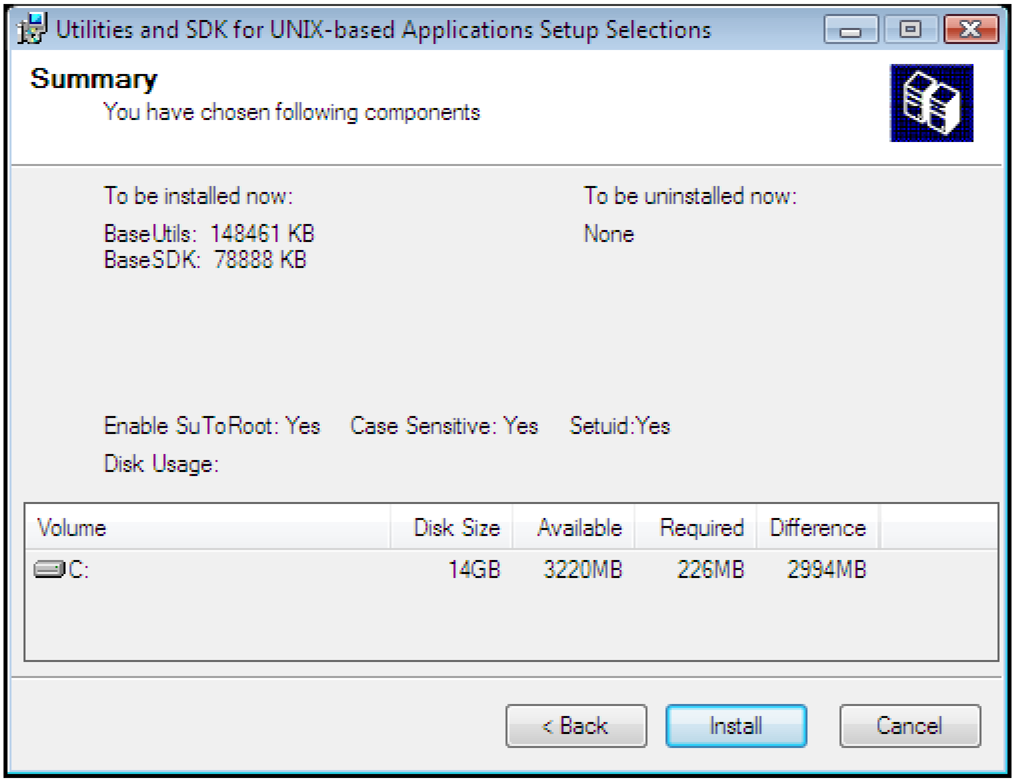
Some SUA programs rely on the setuid mode bit which allows an executing program to take on the identity of the owner of the program file itself. Setup can enable this mechanism.This has security implications, which are explained in Install.htm that is present in the same directory as setup.exe.

☑ Enable setuid behavior for SUA programs.

You have chosen to install Base Utilities, Base SDK. These features require support for case-sensitive file names to provide complete compatibility with UNIX. Setup can change a Windows security setting so that object names are handled in a case-sensitive manner. This has security implications, which are explained in Install.htm that is present in the same directory as setup.exe.

☑ Change the default behavior to case sensitive.

j. Depending on the Windows operating system that you are using, you might be presented with the above Subsystem for UNIX-based Applications security settings. Click Next.
The Summary screen appears.



Utilities and SDK for UNIX-based Applications Setup Selections

Summary

You have chosen following components

To be installed now:

BaseUtils: 148461 KB
BaseSDK: 78888 KB

To be uninstalled now:

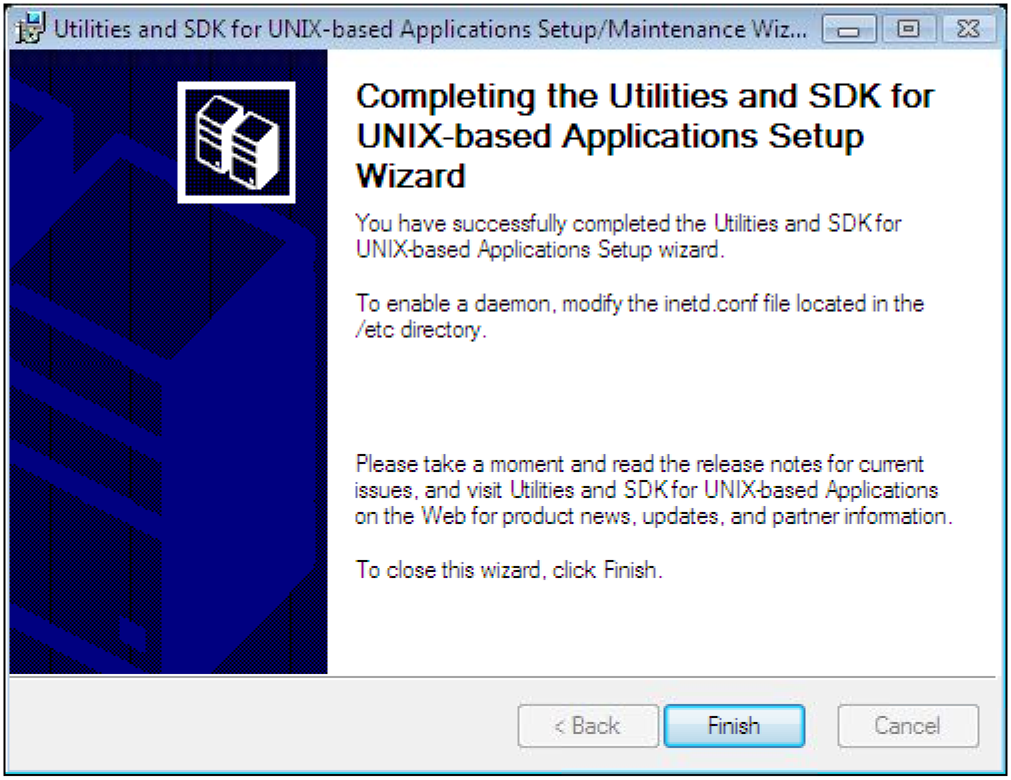
None

Enable SuToRoot: Yes   Case Sensitive: Yes   Setuid:Yes

Disk Usage:

Volume	Disk Size	Available	Required	Difference
C:	14GB	3220MB	226MB	2994MB

k. Select the required disk volume and click Install.
The installation wizard appears.



Utilities and SDK for UNIX-based Applications Setup/Maintenance Wiz...

Completing the Utilities and SDK for UNIX-based Applications Setup Wizard

You have successfully completed the Utilities and SDK for UNIX-based Applications Setup wizard.

To enable a daemon, modify the inetd.conf file located in the /etc directory.

Please take a moment and read the release notes for current issues, and visit Utilities and SDK for UNIX-based Applications on the Web for product news, updates, and partner information.

To close this wizard, click Finish.

l. Click Finish to exit the installation wizard.

m. Reboot the host.
After rebooting, you will notice a C Shell, a Korn Shell, and some additional links and documentation in the folder named Subsystem for UNIX-based Applications in the Windows Start menu.

⚠ Note

You must set the proper firewall rules to access the host. To allow programs running under Interix to access the network, you must add "psxss.exe" to the list of exceptions. "psxss.exe" is the main part of the Interix "kernel". This executable is typically located in "C:\Windows\system32\psxss.exe".

3. Ensure that the Interix Subsystem starts up during system booting.
   If you intend to use NFS shares, start the client for NFS. The mapping between UNIX and Windows user IDs is done by the Windows Active Domain Server; consult the Subsystem for UNIX-based Applications documentation for more information. Depending on the installation options and your version of the Windows operating system, one or more of these services are disabled by default.

Post Installation Tasks

You need to perform the following steps after installing Subsystem for UNIX-based Applications.

1. Before you start using Subsystem for UNIX-based Applications and install Grid Engine, you need to check that the user mapping is working correctly.
   a. Open an Interix shell locally on the Interix host.
   b. Use the `login` command to switch to a known user which is not an Administrator.
   c. Verify the access permissions for network shares that should be accessible to that user.
   d. Try to access these network resources. If the user cannot access a network drive and it is a NFS shared drive, most likely the User Name Mapping is not working correctly.

2. Check the users' home directories.
   To enable the automounting of the users' home directories, click Start > Control Panel > Administrative Tools > Computer Management > Users > Properties > Profile.
   Click Connect to and select the required drive letter. Enter the path of the user's home directory in UNC notation, `\\<server>\<share>\<user home>`.
   Within the Interix subsystem, you can access all network shares through the special directory, `/net/`server`/`share.
   You can also create links to these directories to access the shares directly, for example, `ln -s /net/myserver/export/share00/home /home`. See also 5. Mount network shares below.

3. Enable Administrator names on your machines.
   Ensure that the administrator accounts on all machines that are enabled as execution hosts for Sun Grid Engine use the same account name, such as Administrator.
   Ensure that this user has manager privileges in your Sun Grid Engine cluster. If this is not the case, add the privileges using `qconf -am` administrator before the installation of the execution daemon.

4. Set the CLI commands.
   This opens an editor. Ensure that you set the EDITOR environment variable to `vi`, or your preferred UNIX editor, within the Interix subsystem before you start using UNIX commands.

5. Mount network shares.
   There are two ways to mount network shares to the Interix host:
   - Interix provides a directory in which it makes available all network shares it finds by browsing the network. This works similar to the "Network" folder in the "Windows Explorer", which also searches the network for available shares. The directory where these network shares are provided is `/net`. In this `/net` directory all automatically discovered hosts can be found as subdirectories. Each of this subdirectories will list all network shares of this host as subdirectories, again. The syntax is `/net/`server`/`share, e.g. `/net/myserver/home`. Eventhough a `ls /net` may list no content but perhaps some errors, a `ls /net/`server`/`share will list the content of the share. The errors or missing host names seem to be bug in displaying the names, but this bug doesn't affect the functionality of the automatically discovered shares.
     To make these shares available under the same path as on a UNIX host, it's recommended to create links to these shares. For example `ln -s /net/myserver/home /home` makes the users' UNIX home directories accessible through `/home/`username on the Windows host.

The automatically discovered shares are available at boot time for all users who have the permissions to access the shares. Interix will discover the same kinds of network shares (SMB, NFS, CIFS and so on) the "Windows Explorer" can discover. For this, the proper network client must be installed and the permissions must be sufficient.

- Network shares can also be mapped to drive letters by using the `net` command of Windows. The syntax is `/dev/fs/C/Windows/System32/net.exe <drive letter>: \\<computername>\<sharename> <devicename>`. For example:

```
/dev/fs/C/Windows/System32/net.exe Z: \\\\myserver\\home
```

This drive is now accessible through `/def/fs/Z`. A link can be created to this drive to use the same path as on a UNIX host.

> Note
> As shown in the example above, all backslashes must be written twice because the shell interprets a single backslash as an escape character.

The page Troubleshooting SUA does not exist.

Changing Default Behavior to Case Sensitivity

You might have to choose between default behavior and case sensitivity for object names, such as file names. Your choice will affect system security as well as how Microsoft Services for UNIX (SFU) and Microsoft Subsystem for UNIX-based Applications (SUA) function.

With Microsoft Windows, the names of most objects are case preserving, but case insensitive. So, you cannot have two files in the same directory named `sample.txt` and `Sample.txt` because Windows regards the names as identical.

However, the UNIX operating system is fully case sensitive. So, UNIX systems distinguish between object names even when the only difference between those names is the case of the object name characters. Therefore, `sample.txt` and `Sample.txt` could appear in the same directory and the UNIX system would distinguish between them when performing operations on the files. For example, the command `rm S*.txt` would delete `Sample.txt` but not `sample.txt`. To implement typical UNIX behavior, the server for NFS and the Interix subsystem are normally case sensitive when working with file names.

This behavior can present security issues, particularly for users who are accustomed to the case insensitive conventions of Windows. For example, a Trojan horse version of `edit.exe`, named `EDIT.EXE`, could be stored in the same directory as the original. If a user were to type `edit` at a Windows command prompt, the Trojan horse version (`EDIT.EXE`) could be executed instead of the standard version.

> Caution
> If case sensitivity is enabled, Windows users should be made aware of the security issues.

For Windows XP (Professional) and the Windows Server 2003 family, the default behavior of subsystems (other than the Win32 subsystem) is to preserve case but be case insensitive. In previous versions of Windows, such subsystems were fully case sensitive by default. To support standard UNIX behavior, the SFU and SUA setups allow you to change the default Windows XP and Windows Server 2003 family behavior for non-Win32 subsystems when installing the base utilities (the Interix subsystem) or Server for NFS. If you enable case sensitivity and then subsequently uninstall the base ut

Disabling DEP

How to Disable DEP for Windows XP Professional, Windows Server 2000 and Window Server 2003

1. Right-click the My Computer icon on your desktop.
2. Click Properties.
3. In the Properties dialog box, click the Advanced tab.
4. Click Settings in the Startup and Recovery section.
5. In the next dialog box, click the Edit button to edit the boot command line of your Windows installation.
6. Add `/noexecute=alwaysoff` or modify an existing `/noexecute` option.

How to Disable DEP for Windows Vista (Enterprise & Ultimate) and Windows Server 2008

1. Click Start > All Programs > Accessories.
2. Right-click Command Prompt.
3. Left-click Run as Administrator.
4. Click Allow, if the system prompts you for permission.
5. Type the following text in the command prompt window.

```
bcdedit.exe /set {current} nx AlwaysOff
```

Enabling suid Behavior for Interix Programs

According to the POSIX standard, a file has permissions that include bits to set both a UID (`setuid`) and a GID (`setgid`) when the file is executed. If either or both bits are set on a file, and a process executes that file, the process gains the UID or GID of the file.

When used carefully, this mechanism allows a non-privileged user to execute programs that run with the higher privileges of the file's owner or group.

When used incorrectly, however, this behavior can present security risks by allowing non-privileged users to perform actions that should only be performed by an administrator. For this reason, Windows Services for UNIX and Windows Subsystems for UNIX-based Applications setup does not enable support for this mechanism by default.

You should enable support for `setuid` behavior because Grid Engine runs programs that require this support. If you do not enable support for `setuid` behavior when installing Windows Services for UNIX, you can enable it later.

User Management on Windows Hosts

Every user of the Grid Engine execution environment of a Windows machine must have a user account that has the same name as on the UNIX hosts. User accounts contain information about the user, including name, password, various optional entries that determine when and how users log on and how their desktop settings are stored.

The following sections describe how you would use Windows user management to support Grid Engine.
Windows machines are referred to here using three different terms. The following table lists the terms and the operating systems which might run on each corresponding host:

Windows Host	Microsoft Windows 2000, Microsoft Windows XP, Microsoft Windows 2000 Server, Microsoft Windows Server 2003
Windows Server	Microsoft Windows 2000 Server, Microsoft Windows Server 2003
Windows Workstation	Microsoft Windows 2000, Microsoft Windows XP

Topic	Description
Managing Users on Windows Hosts	Learn how to administer user accounts on Windows hosts.
Using Grid Engine in a Microsoft Windows Environment	Learn how to use Sun Grid Engine in a Microsoft Windows environment.
How to Add Windows Hosts Later	Learn how to add Microsoft Windows hosts at a later point in time.

Managing Users on Windows Hosts

It is possible to administer user accounts on all Windows hosts individually. Each Windows Host has an authentication center which validates user names and corresponding user rights. User accounts which are defined on a Windows workstation are referred to here as local user accounts or local users.

Each Windows Host has its own local domain, and each Windows Server has the ability to make that domain available to other hosts. Account names within a local domain and account names within a server domain can collide. To avoid such collisions, you must specify the correct user account by providing the domain name as a prefix to the user account name followed by a + (plus sign) character.

Windows User Example

The following is an example that illustrates the potential complexity of Windows host accounts interacting with Windows Domain accounts. Suppose Windows Workstation host named CRUNCH has a local user account named Peter. This Windows Workstation is part of the domain named ENGINEERING. This domain is provided by a Windows Server which also has a user account named Peter. In this example, the ENGINEERING domain is the default domain of the host named CRUNCH. The following table shows the possible results of what would happen if a person tried to log in to CRUNCH.

Table – Using Domain Accounts

Login Name	Result
CRUNCH+Peter	Peter is logged in with his account as a local user of the machine CRUNCH.
ENGINEERING+Peter	Peter is logged in with the account provided by the Windows Server hosting the ENGINEERING domain.
Peter	This approach is equivalent to using ENGINEERING+Peter because CRUNCH has ENGINEERING as its default domain. Otherwise, the local account would be used.

Each domain has a special user account that provides superuser access. The default name for that account is `Administrator`. For native Windows, the members of the Administrators group and of the Domain Admins group in the server domain also have superuser access. However, for Interix, only the user Administrator of the local domain is the superuser of the local host.

The local Administrator can start applications in an account without knowing the password of the user for that account. However, the application would not be able to access network resources because even the local Administrator is not fully trusted by the network, unlike the Unix super user `root`. Therefore, the Sun Grid Engine administrator uses the `sgepasswd` tool to register the users' passwords, as explained in Using Grid Engine in a Microsoft Windows Environment.

UNIX User Management

UNIX has no equivalent to the Windows domain concept. With UNIX, each user has a local account and is authenticated as a local account even if the underlying account information lies on an LDAP or NIS server. The UNIX super user `root` is similar to the local Windows super user `Administrator`. The UNIX super user can start applications and processes on behalf of UNIX accounts without knowing each corresponding password.

Using Sun Grid Engine in a Microsoft Windows Environment

The Grid Engine execution environment starts jobs on behalf of the submitting user. The execution daemon (sge_execd) on UNIX hosts runs as root so that it can start jobs on behalf of all users.

On Windows hosts, the execution daemon runs as the local Administrator user so that it can start jobs on behalf of users without knowing their password, but these jobs would not have the permissions to access network resources. Only fully authenticated users can access network resources. For a full authentication, the user's password is needed. Therefore, all users who want to submit jobs to a Windows execution host have to register their passwords with Grid Engine. The execution daemon still needs to run as the local Administrator to have the permissions to do several administrative tasks.

Registering Windows User Passwords

Users who want to start Grid Engine jobs on Windows execution hosts use the sgepasswd client application to register their Windows passwords. The following example shows Peter who has a user account in the domain ENGINEERING. Because ENGINEERING is the principal domain of the Windows execution host CRUNCH, Peter does not need to register his password for a specific domain. This should be the default in any properly set up single domain environment. In multiple domain environments, it might be necessary to register the password explicitly for a specific domain.

> Note
> You must run the sgepasswd command on a non-Windows host.

```
> sgepasswd
   Changing password for Peter
   New password:
   Re-enter new password:
   Password changed
```

Using the sgepasswd Command

The sgepasswd command changes the Grid Engine password file sgepasswd(5). This file contains a list of user names and their Windows passwords in encrypted form.

You can use sgepasswd to perform the following tasks:

- To add a new entry for your user account.
- To change your existing password, if you know your stored password.

> Caution
> If Grid Engine tries to run several of your jobs at once on a Windows execution host and is unable to access a correct password for your account, the Windows intrusion detection system could disable your account. To keep your account from being disabled, you must prevent your pending jobs from being run before you attempt to change your Windows user password. Once you have changed your password using sgepasswd on a non-Windows host and then on your Windows domain, you can allow your jobs to be run again.

Additionally, the root user can change or delete the password entries for other user accounts. sgepasswd is only available on non-Windows hosts.

The sgepasswd uses one of the following syntaxes:

```
sgepasswd [[ -D <domain> ] -d <user> ]

sgepasswd [ -D <domain> ] [ <user> ]
```

This command supports the following options:

-D domain	By default, sgepasswd adds or modifies the current UNIX user name without a domain specification. You can use this switch to add a domain specification in front of the current user name. Consult your Microsoft Windows documentation for more information about domain users.
-d user	Only root can use this parameter to delete entries from the sgepasswd(5) file.
-help	Prints a listing of all options.

Additionally, the following environment variables affect the operation of this command.

SGE_CERTFILE	Specifies the location of public key file. By default, sgepasswd uses the file $SGE_ROOT/$SGE_CELL/common/sgeCA/certs/cert.pem.
SGE_KEYFILE	If set, this specifies the location of the private key file. The default file is /var/sgeCA/port$SGE_QMASTER_PORT/$SGE_CELL/private/key.pem.
SGE_RANDFILE	If set, this specifies the location of the rand.seed file. The default file is /var/sgeCA/port$SGE_QMASTER_PORT/$SGE_CELL/private/rand.seed.

Adding Windows Hosts to Existing Grid Engine Systems

If you have a running Grid Engine system on which Windows support is not enabled, you can enable the support manually. The following steps provide a Windows-enabled Grid Engine system that allows additional Windows execution hosts.

How to Add Windows Hosts Later

1. Copy Windows binaries to the $SGE_ROOT directory.

2. Type the following command:

   ```
   qconf -mconf
   ```

   Set the execd_params to enable_windomacc=true.

3. Type the following command:

   ```
   qconf -am <win_admin_name>
   ```

4. Run the following command:

   ```
   $SGE_ROOT/util/sgeCA/sge_ca -init -days 365
   ```

5. For a CSP installation, run the following command:

```
$SGE_ROOT/util/sgeCA/sge_ca -user <win_admin_name>
```

6. Type the following command:

   ```
   qconf -ah <new_win_hosts>
   ```

7. Copy certificates to each Windows host.
8. Set the owner of the certificates to **ADMINUSER**.
   Use a command similar to the following example:

   ```
   chown -R foo:bar /var/sgeCA/port <SGE_QMASTER_PORT>
   ```

9. Run normal exec daemon installation on each execution host.

Other Installation Issues

Additional considerations for installing Sun Grid Engine software are identified in this section. These include the following topics:

Topic	Description
How to Verify and Install Linux Motif Libraries	Learn how to verify and install Linux Motif libraries.
How to Install the Software on a System With IPMP	Describes how to install the Sun Grid Engine software on hosts with the Solaris Operating Environment IP Multipathing (IPMP) technology.

How to Verify and Install Linux Motif Libraries

On newer Linux systems, the `libXm.so.2` Motif libraries are not always installed, which results in the inability to run the precompiled Linux qmon binary.

To correct this problem, follow these steps:

1. Check if the libraries are already present.

   ```
   % ls -l /usr/X11R6/lib/libXm*
   ```

   If the `/usr/X11R6/lib/libXm.so.2` points to a `libXm.so.2.x` version, you are done. Note that a symbolic link to

/usr/X11R6/lib/libXm.so.3 does not work.
If the libraries are not present, then continue following these steps.

2. Download the corresponding openmotif libraries from http://www.ist.co.uk/DOWNLOADS/motif_download.html or from the SUSE 9.1 distribution (an additional **rpm** file called **openmotif21-*** is available).

3. Install the missing libraries as root.
   For SUSE 9.1, you install the openmotif21-* package like any other package. For packages downloaded from http://www.ist.co.uk, install the libraries as shown in the following example.

```
# rpm -i --prefix /tmp/test --force \
      openmotif-2.1.31-2_IST-JDS2003.i386.rpm
# cd /tmp/test/OpenMotif-2.1.31/lib
# cp libXm.so.2.1 /usr/X11R6/lib
# cd /usr/X11R6/lib
# ln -s libXm.so.2.1 libXm.so.2
```

4. Test **qmon**.

```
% ldd `which qmon`
```

How to Install the Software on a System with IPMP

This section describes how to install the Grid Engine software on hosts with the Solaris Operating Environment IP Multipathing (IPMP) technology.

What Is IP Multipathing?

IP Multipathing is a technology that allows TCP/IP interfaces to be grouped for failover and load balancing purposes. If an interface within an IP Multipathing group fails, the interface is disabled and its IP address is relocated to another interface in the group. Outbound IP traffic is distributed across the interfaces of a group. For further details on IP Multipathing, refer to the Solaris Operating Environment documentation at http://docs.sun.com/app/docs/doc/816-4554/ipmptm-1.

Issues Between IPMP and Grid Engine

When starting the Grid Engine daemons on a machine where the main interface is part of an IPMP group, error messages appear. When the IPMP load balancing distributes the connections across the interfaces in the group, the IP packets show up at the receiving end as coming from a different host from the one associated with the main interface. For example, on a machine with three interfaces named qfe0, qfe1, and qfe3, where the IP addresses for these interfaces are 10.1.1.1, 10.1.1.2, and 10.1.1.3 respectively, IPMP would need an extra address for each interface for testing. However, that requirement is ignored in this example. Each of these addresses has a host name associated with it. The hosts table looks like the following example:

```
10.1.1.1 sge
10.1.1.2 sge-qfe1
10.1.1.3 sge-qfe2
```

The machine's host name is sge. When a connection is established from sge to another machine, it might go through sge, sge-qfe1, or sge-qfe2. Upon installation, Grid Engine will only recognize sge. When Grid Engine receives a connection request from sge-qfe2, it closes the connection because the request is not from one of the authorized (or known) nodes.

To solve this problem, use the `host_aliases` files to "tell" Grid Engine that `sge`, `sge1`, and `sge-qfe2` are all from the same machine. See the `sge_h_aliases` man page for details. The `host_aliases` file in this case would look like this:

```
sge sge-qfe1 sge-qfe2
```

> Note
> If you make any changes to the `$SGE_ROOT/$SGE_CELL/common/host_aliases` file, you must stop and restart all running Grid Engine daemons (`sge_qmaster` and `sge_execd`). To do this, log in as root to all your Grid Engine hosts and enter these commands:
>
> ```
> /etc/init.d/sgemaster stop
> /etc/init.d/sgeexecd stop
> /etc/init.d/sgemaster start
> /etc/init.d/sgeexecd start
> ```

Installing the Grid Engine Master Node With IPMP

There are two ways that you can fix this problem:

- Ignore the error messages during installation. This method is operating system independent (except for MS Windows).
- Temporarily disable IPMP on the interface associated with the machine's host name. This method only works on systems running at least Version 8 of the Solaris OS.

Ignoring the Error Messages

To ignore the error messages, follow these steps:

1. Run the **`inst_sge -m`** command while ignoring the error messages during the start up of the daemons.

2. Shut down the daemons with the **`/etc/init.d/sgemaster stop`** and **`/etc/init.d/sgemaster stop`** commands.
   Due to the networking errors, some daemons fail to shutdown and must be killed with the `kill -9` command. To see which daemons failed to shutdown use this command: `ps -e | grep sge_`.

3. Install the **`host_aliases`** file in the **`$SGE_ROOT/$SGE_CELL/common`** directory.

4. Restart the daemons with the **`/etc/init.d/sgemaster start`** and **`/etc/init.d/sgeexecd start`** commands.

Temporarily Disabling IPMP

To temporarily disable IPMP, follow these steps:

1. Identify the interface associated with the machine's host name.

2. Verify that the interface has IPMP enabled by using the **`ifconfig`** interface **`| grep groupname`** command.

3. Take note of the group name.

4. Disable IPMP with this command: **`ifconfig`** interface **`group ""`** .

5. Install the Grid Engine master node.

6. Install the **`host_aliases`** file in the **`$SGE_ROOT/$SGE_CELL/common`** directory.

7. Restart the daemons with the with the **`/etc/init.d/sgemaster`** and **`/etc/init.d/sgeexecd`** commands.

8. Re-enable IPMP using the following command: **`ifconfig`** interface ***`group`*** _IPMP group.

Installing a Grid Engine on an Execution Host With IPMP

Once the `host_aliases` file is installed and the Grid Engine daemons are restarted, you can simply start the execution host installation without further problems.

Enabling Administrative and Submit Hosts With IPMP

You have two choices when enabling these hosts with IPMP:

- Follow the same procedure used for the execution host (updating the `host_aliases` file before installation).
- Add all the host names associated with the administrative or submit host with one of the following commands:
    - For the administrative host:

      ```
      qconf -ah <hostname> <alias 1> <alias 2> ...
      ```

    - For the submit host:

      ```
      qconf -as <hostname> <alias 1> <alias 2> ...
      ```