

# Table of Contents

<b>Generic Installation and Configuration Guide for gLite 3.1.....</b>	<b>1</b>
<b>Support.....</b>	<b>2</b>
<b>Introduction to Manual Installation and Configuration.....</b>	<b>3</b>
<b>Installing the Operating System.....</b>	<b>5</b>
Scientific Linux 4 (CERN).....	5
Using SL4 compatible distributions other than CERN Scientific Linux.....	5
Node synchronization, NTP installation and configuration.....	6
Cron and logrotate.....	6
<b>Host Certificates.....</b>	<b>7</b>
<b>Oracle.....</b>	<b>8</b>
<b>Installing the Middleware.....</b>	<b>9</b>
Repositories.....	9
The middleware repositories.....	9
The Certification Authority repository.....	9
jpackage and the JAVA repository.....	10
The DAG repository.....	10
Installations.....	11
Note on the installation of the MON and VOMS/MySQL nodes.....	12
Updates.....	12
Normal updates.....	12
Important note on automatic updates.....	12
Upgrading from gLite 3.0.....	13
<b>Configuring the Middleware.....</b>	<b>14</b>
Using the YAIM configuration tool.....	14
Configuring multiple node types on the same physical host.....	14
<b>Installing and Configuring a batch system.....</b>	<b>15</b>
The Torque/PBS batch system.....	15
The WN for Torque/PBS.....	15
The lcg-CE for Torque/PBS.....	15
Standalone Torque server.....	15
Known issues.....	15
The SGE batch system.....	16
The cream CE for SGE.....	16
The lcg-CE for SGE.....	16
Configure lcg-CE and SGE Qmaster in the same physical machine.....	16
Configure lcg-CE and SGE Qmaster in different machines.....	16
Link the lcg-CE with a running SGE Qmaster server.....	18
The WN for SGE.....	19
The LSF batch system.....	19
The WN for LSF.....	20
The lcg-CE for LSF.....	20
Note on site-BDII for LSF.....	20
The Condor batch system.....	21

# Table of Contents

<b>Known issues.....</b>	<b>22</b>
<b>Note on hostname syntax.....</b>	<b>23</b>
<b>Firewalls.....</b>	<b>24</b>
<b>Documentation.....</b>	<b>25</b>

# Generic Installation and Configuration Guide for gLite 3.1

**Note that gLite 3.1 is being phased out. Check which services are still supported in 3.1 under the gLite web pages [↗](#).**

This document is addressed to Site Administrators responsible for middleware installation and configuration. It is a generic guide to manual installation and configuration for any supported node types. This guide is for gLite release 3.1, if you configure gLite 3.0 services please check the previous version of this guide.

# Support

Please open a GGUS ticket if you experience any Installation or Configuration problem.

Your contact point for technical support is your ROC <http://egee-sa1.web.cern.ch/egee-sa1/roc.html> but if you need to contact the release team, please send a mail to [gd-release-team@cern.ch](mailto:gd-release-team@cern.ch).

# Introduction to Manual Installation and Configuration

This document is addressed to Site Administrators responsible for middleware installation and configuration. It is a generic guide to manual installation and configuration for any supported node types. It provides a fast method to install and configure the gLite middleware version 3.1 for the following node types :

gLite 3.1 32bits:

- **glite-AMGA\_oracle**
- **glite-AMGA\_postgres**
- **glite-BDII**
- **glite-CREAM**
- **glite-FTM**
- **glite-GLEXEC\_wn**
- **glite-LB**
- **glite-LFC\_mysql**
- **glite-LFC\_oracle**
- **glite-LSF\_utils**
- **glite-MON**
- **glite-MPI\_utils**
- **glite-PX**
- **glite-SCAS**
- **glite-SE\_dcachel\_admin\_gdbm**
- **glite-SE\_dcachel\_admin\_postgres**
- **glite-SE\_dcachel\_info**
- **glite-SE\_dcachel\_pool**
- **glite-SE\_dpm\_disk**
- **glite-SE\_dpm\_mysql**
- **glite-SGE\_utils**
- **glite-TORQUE\_client**
- **glite-TORQUE\_server**
- **glite-TORQUE\_utils**
- **glite-UI**
- **glite-VOBOX**
- **glite-VOMS\_mysql**
- **glite-VOMS\_oracle**
- **glite-WMS**
- **glite-WN**
- **lcg-CE**

gLite 3.1 64bits:

- **glite-LFC\_mysql**
- **glite-LFC\_oracle**
- **glite-SE\_dcachel\_admin\_gdbm**
- **glite-SE\_dcachel\_admin\_postgres**
- **glite-SE\_dcachel\_info**
- **glite-SE\_dcachel\_pool**
- **glite-SE\_dpm\_disk**
- **glite-SE\_dpm\_mysql**
- **glite-TORQUE\_client**
- **glite-WN**

Note that glite-WN in 64bits is installed in compatibility mode and it also includes 32bit versions of the middleware clients.

The supported installation method for SL4 is the `yum` tool. Please note that **YAIM IS NOT SUPPORTING INSTALLATION** you have to configure yum repositories yourself and install the metapackages using your preferred way.

The configuration is performed by the YAIM tool. For a description of YAIM check YAIM guide

YAIM can be used by Site Administrators without any knowledge of specific middleware configuration details. They must define a set of variables in some configuration files, according to their site needs.

# Installing the Operating System

## Scientific Linux 4 (CERN)

The OS version of gLite Middleware version 3.1 is Scientific Linux 4 (SL4). For more information in SLC4, please check:

<http://www.scientificlinux.org>

The sources and the images (iso) to create CDs for SLC4 can be found in this site:

<ftp://ftp.scientificlinux.org/linux/scientific/4x>

Middleware testing has been mostly carried out on CERN Scientific Linux:

<http://linuxsoft.cern.ch/>

but it should be able to run on any binary compatible distribution.

## Using SL4 compatible distributions other than CERN Scientific Linux

The deployment team is based at CERN and it uses the local variant of SL which is called Scientific Linux CERN (SLC). gLite is supported on SL4 as well as SLC4. Other binary compatible distributions will be supported on a best effort basis. Most of the packages needed by gLite are provided either by the SL repository or by the gLite repository. However a few additional needed packages are provided by the SLC repository.

You can find in the gLite web page [↗](#) the list of needed RPMS for each metapackage and release.

If some dependencies are available in SLC but not in other distributions, you should add the CERN OS repository to your yum configuration, configuring yum so that **your local OS repository has priority**. In this way only the missing packages will be taken from CERN.

One way to achieve this is to use the `yum-protectbase` plugin and mark your OS repositories with `protect=1` (this can be sometimes the default). Then add the SLC repository in non-protected mode with `protect=0`. E.g. you can set up the yum repository via:

```
[sl-base]
baseurl=http://linuxsoft.cern.ch/scientific/4x/i386/SL/RPMS
enabled=1
protect=1
```

```
[slc-base]
baseurl=http://linuxsoft.cern.ch/cern/slc4X/i386/yum/os
enabled=1
protect=0
```

```
[slc-update]
baseurl=http://linuxsoft.cern.ch/cern/slc4X/i386/yum/updates
enabled=1
protect=0
```

## Node synchronization, NTP installation and configuration

A general requirement for the gLite nodes is that they are synchronized. This requirement may be fulfilled in several ways. If your nodes run under AFS they are most likely already synchronized. Otherwise, you can use the NTP protocol with a time server.

Instructions and examples for a NTP client configuration are provided in this section. If you are not planning to use a time server on your machine you can just skip this section.

Use the latest ntp version available for your system. If you are using APT, an apt-get install ntp will do the work.

- Configure the file `/etc/ntp.conf` by adding the lines dealing with your time server configuration such as, for instance:

```
restrict <time_server_IP_address> mask 255.255.255.255 nomodify notrap noquery
server <time_server_name>
```

Additional time servers can be added for better performance results. For each server, the hostname and IP address are required. Then, for each time-server you are using, add a couple of lines similar to the ones shown above into the file `/etc/ntp.conf`.

- Edit the file `/etc/ntp/step-tickers` adding a list of your time server(s) hostname(s), as in the following example:

```
137.138.16.69
137.138.17.69
```

- If you are running a kernel firewall, you will have to allow inbound communication on the NTP port. If you are using iptables, you can add the following to `/etc/sysconfig/iptables`

```
-A INPUT -s NTP-serverIP-1 -p udp --dport 123 -j ACCEPT
-A INPUT -s NTP-serverIP-2 -p udp --dport 123 -j ACCEPT
```

Remember that, in the provided examples, rules are parsed in order, so ensure that there are no matching REJECT lines preceding those that you add. You can then reload the firewall

```
# /etc/init.d/iptables restart
```

- Activate the ntpd service with the following commands:

```
# ntpdate <your ntp server name>
# service ntpd start
# chkconfig ntpd on
```

- You can check ntpd's status by running the following command

```
# ntpq -p
```

## Cron and logrotate

Many middleware components rely on the presence of `cron` (including support for `/etc/cron.*` directories) and `logrotate`. You should make sure these are available on your system.

# Host Certificates

**All nodes except UI, WN and BDII require the host certificate/key files to be installed.** Contact your national Certification Authority (CA) to understand how to obtain a host certificate if you do not have one already. Instructions to obtain a CA list can be found here:

- <http://grid-deployment.web.cern.ch/grid-deployment/lcg2CAlist.html>

Once you have obtained a valid certificate:

- `hostcert.pem` - containing the machine public key
- `hostkey.pem` - containing the machine private key

make sure to place the two files in the target node into the `/etc/grid-security` directory and check the access right for `hostkey.pem` is only readable by root and that the public key, `hostcert.pem`, is readable by everybody.

# Oracle

Some node types require the oracle instant clients in order to communicate with an Oracle DB. In that case you have to install them manually, please, visit the Oracle web page [to](#) download the necessary rpms. Normally you would require:

- oracle-instantclient-basic
- oracle-instantclient-sqlplus

This is needed if you are installing a VOMS oracle, LFC oracle, AMGA oracle and FTS oracle.

# Installing the Middleware

Please before you proceed further **make sure** that Java is installed in your system. As of SL4 the `yum` package manager is considered to be the default installation tool. The repository will continue to support APT but you should be aware of potential problems using this package manager in a multiarch (32 and 64 bit) environment.

The middleware packages will be released first for 32 bit and subsequently for 64 bit.

As the installation is not supported by YAIM, you have to install the metapackages on your own.

## Repositories

For a successful installation, you will need to configure your package manager to reference a number of repositories (in addition to your OS);

- the middleware repositories
- the CA repository
- the jpackage repository
- DAG
- SLC (if you are using SL, you will need to pick up a couple of extra packages provided only by the CERN version of SL - see above)

## The middleware repositories

gLite is distributed in multiple yum repositories. Each node type has its own independent repository. These repositories contain only the relevant rpms for each node type. To save space, all the rpms are stored in a directory called `generic` (with no `repodata`) and there are symbolic links to the packages in `generic` from the different repositories.

gLite 3.1 repository can be found under:

<http://linuxsoft.cern.ch/EGEE/gLite/R3.1/>

To use yum, `wget` the yum repository for the node type you want to install from the following web address and copy it in `/etc/yum.repos.d`:

- <http://grid-deployment.web.cern.ch/grid-deployment/glite/repos/3.1> 

**Note** that installation of several node types in the same physical host is not recommended. The repositories of each node type may not be synchronised for the same package and this can cause problems.

## The Certification Authority repository

The most up-to-date version of the list of trusted Certification Authorities (CA) is needed on your node. As the list and structure of the Certification Authorities (CA) accepted by the LCG project can change independently of the middleware releases, the rpm list related to the CAs certificates and URLs has been decoupled from the standard gLite/LCG release procedure.

**Please note that the lcg-CA metapackage and repository is no longer maintained.** The lcg-CA repository should be now replaced by the EGI trustanchors repository. All the details on how to install the CAs can be found in EGI IGTF  release pages.

## jpackage and the JAVA repository

**IMPORTANT NOTE:** jpackage repo files are broken for SL4. Please, either create your own local copy of jPackage or use the rpm lists from [www.glite.org](http://www.glite.org) to manually install the needed jPackage packages.

You should install Java JDK 1.5.0 on your system before installing the middleware. Download it from SUN java web site (1.5 or greater is required). [http://java.sun.com/javase/downloads/index\\_jdk5.jsp](http://java.sun.com/javase/downloads/index_jdk5.jsp), or from [here](#).

You can install the RPM supplied on SUNs web site but for several reasons it is recommended to use the jpackage build of JDK. This details are described [here](#). JAVA installation for gLite 3.1

The glite 3.1 distribution takes as many dependencies as possible from jpackage, so you should set up your package manager to reference jpackage 5, as described in the 'Configuration of Yum' section in JAVA installation for gLite 3.1

You can reference a **jpackage** repository with: (for example in `jpackage.repo`);

- <http://grid-deployment.web.cern.ch/grid-deployment/glite/repos/3.1/jpackage.repo>

```
[jpackage5-generic]
name=JPackage 5, generic
baseurl=http://mirrors.dotsrc.org/jpackage/5.0/generic/free/
enabled=1
protect=1
pgpkey=http://www.jpackage.org/jpackage.asc
pgpcheck=1

[jpackage5-generic-nonfree]
name=JPackage 5, generic non-free
baseurl=http://mirrors.dotsrc.org/jpackage/5.0/generic/non-free/
enabled=1
protect=1
pgpkey=http://www.jpackage.org/jpackage.asc
pgpcheck=1
```

## The DAG repository

DAG is a maintained repository which provides a number of packages not available through Scientific Linux. If you have installed the CERN version of Scientific Linux, you will find that the relevant file is already installed in `/etc/yum.repos.d`. Otherwise, please use the following

```
[main]
[dag]
name=DAG (http://dag.wieers.com) additional RPMS repository
baseurl=http://linuxsoft.cern.ch/dag/redhat/el4/en/$basearch/dag
pgpkey=http://linuxsoft.cern.ch/cern/slc4X/$basearch/docs/RPM-GPG-KEY-dag
pgpcheck=1
enabled=1
```

### Important Note

In a limited number of cases, DAG provides rpms already represent in the OS. In this case, DAG rpms are of a higher version. Normally the OS is protected from having its rpms upgraded (`protect=1`). Then there are two solutions:

- You install the relevant rpm by hand *before* installing the meta-package. For example

```
# wget 'http://linuxsoft.cern.ch/dag/redhat/el4/en/i386/RPMS.dag/perl-SOAP-Lite-0.69-1.el4.rf.noarch.rpm'
# yum localinstall perl-SOAP-Lite-0.69-1.el4.rf.noarch.rpm
```

- You can remove the `protect=1` (temporarily) from the yum configuration for your OS (not recommended). Currently this operation is required in the following cases:

meta-package	rpm
glite-FTM	perl-SOAP-Lite-0.69-1.el4.rf.noarch.rpm
glite-SE_dpm_mysql	perl-SOAP-Lite-0.69-1.el4.rf.noarch.rpm
glite-SE_dpm_disk	perl-SOAP-Lite-0.69-1.el4.rf.noarch.rpm
glite-AMGA_postgres	postgresql-odbc-08.02.0500-7.4.slc4.i386.rpm
glite-CREAM	perl-XML-SAX-0.96-1.el4.rf.noarch.rpm

## Installations

Here it is an example on how to install a service node (ex. a UI):

```
cd /etc/yum.repos.d/
wget http://grid-deployment.web.cern.ch/grid-deployment/glite/repos/3.1/glite-UI.repo
yum update
yum install glite-UI
```

The table below lists the available meta-packages and the associated repo file name in <http://grid-deployment.web.cern.ch/grid-deployment/glite/repos/3.1>.

Node Type	meta-package name	repo file
AMGA	glite_AMGA_oracle	glite_AMGA_oracle.repo
AMGA	glite_AMGA_postgres	glite_AMGA_postgres.repo
BDII	glite-BDII	glite-BDII.repo
cream CE	glite-CREAM	glite-CREAM.repo
dCache Storage Element	glite-SE_dcach_admin_gdbm	glite-SE_dcach_admin_gdbm.repo
dCache Storage Element	glite-SE_dcach_admin_postgres	glite-SE_dcach_admin_postgres.repo
dCache Storage Element	glite-SE_dcach_info	glite-SE_dcach_info.repo
dCache Storage Element	glite-SE_dcach_pool	glite-SE_dcach_pool.repo
DPM disk	glite-SE_dpm_disk	glite-SE_dpm_disk.repo
DPM Storage Element (mysql)	glite-SE_dpm_mysql	glite-SE_dpm_mysql.repo
FTA	glite-FTA_oracle	glite-FTA_oracle.repo
FTM	glite-FTM	glite-FTM.repo
FTS	glite-FTS_oracle	glite-FTS_oracle.repo
LB	glite-LB	glite-LB.repo
LCG CE	lcg-CE	lcg-CE.repo
LCG File Catalog server with mysql	glite-LFC_mysql	glite-LFC_mysql.repo
LCG File Catalog server with oracle	glite-LFC_oracle	glite-LFC_oracle.repo
LSF batch server utils	glite-LSF_utils	glite-LSF_utils.repo
MON-Box	glite-MON	glite-MON.repo
MPI utils	glite-MPI_utils	glite-MPI_utils.repo
MyProxy	glite-PX	glite-PX.repo
TORQUE client	glite-TORQUE_client	glite-TORQUE_client.repo
TORQUE server	glite-TORQUE_server	glite-TORQUE_server.repo
TORQUE batch server utils	glite-TORQUE_utils	glite-TORQUE_utils.repo

SGE batch server utils	glite-SGE_utils	glite-SGE_utils.repo
SLCS client	glite-SLCS_client	glite-SLCS_client.repo
User Interface	glite-UI	glite-UI.repo
VO agent box	glite-VOBOX	glite-VOBOX.repo
VOMS server with mysql	glite-VOMS_mysql	glite-VOMS_mysql.repo
VOMS server with oracle	glite-VOMS_oracle	glite-VOMS_oracle.repo
WMS	glite-WMS	glite-WMS.repo
Worker Node	glite-WN	glite-WN.repo

For the TAR UI and TAR WN, please check the following wiki pages:

- TAR UI Installation and Configuration
- TAR WN Installation and Configuration

## Note on the installation of the MON and VOMS/MySQL nodes

In order to install the MON box or the VOMS/MySQL node you need to install manually the MySQL server. Please, run the following command:

```
yum install mysql-server
```

## Updates

### Normal updates

Updates to gLite 3.1 will be released regularly.

If an update has been released, a `yum update` should be all that is required to update the rpms. If you want to update the 64bit WN, you need to run `yum groupupdate glite-WN` in order to properly get new dependencies as well.

**NOTE** that even if the recommendation is to use `yum update`, some sys admins are used to run `yum update metapackage-name`. This doesn't work in the last production releases due to a change in the way the dependencies are specified in the metapackage.

If reconfiguration of any kind is necessary, just run the following command (**don't forget to list all node types installed in your host**):

```
/opt/glite/yaim/bin/yaim -c -s site-info.def -n <node_type> [ -n <node_type> ... ]
```

### Important note on automatic updates

Several site use auto update mechanism. Sometimes middleware updates require non-trivial configuration changes or a reconfiguration of the service. This could involve database schema changes, service restarts, new configuration files, etc, which makes it difficult to ensure that automatic updates will not break a service. Thus

**WE STRONGLY RECOMMEND NOT TO USE AUTOMATIC UPDATE PROCEDURE OF ANY KIND**

on the gLite middleware repositories (you can keep it turned on for the OS). You should read the update docs and do the upgrade manually when an update has been released!

## Upgrading from gLite 3.0

As gLite 3.1 is the first release of the middleware for SL4, there is no supported upgrade path from gLite 3.0 on SL3.

# Configuring the Middleware

## Using the YAIM configuration tool

For a detailed description on how to configure the middleware with YAIM, please check the YAIM guide.

The necessary YAIM modules needed to configure a certain node type are automatically installed with the middleware. However, if you want to install YAIM rpms separately, you can install the repository of the node type you are interested in, as explained in the section [Middleware repositories](#) and then run `yum install glite-yaim-node-type`. This will automatically install the YAIM module you are interested in together with `yaim core`, which contains the core functions and utilities used by all the YAIM modules.

In order to know what's the latest version of YAIM running in production, you can check the [YAIM status page](#) where each yaim module is listed.

## Configuring multiple node types on the same physical host

**Note** that installation and configuration of several node types in the same physical host is not recommended. The repositories of each node type are now independent and may not be synchronised for the same package, which can cause problems.

# Installing and Configuring a batch system

## The Torque/PBS batch system

```
cd /etc/yum.repos.d/  
wget http://grid-deployment.web.cern.ch/grid-deployment/glite/repos/3.1/glite-TORQUE_server.repo  
wget http://grid-deployment.web.cern.ch/grid-deployment/glite/repos/3.1/glite-TORQUE_client.repo  
yum update  
yum install the_necessary_metapackage_
```

### The WN for Torque/PBS

After fetching the `glite-WN` repository (see above) use the following commands for the 64bit architecture:

```
yum groupinstall glite-WN  
yum install glite-TORQUE_client
```

and the following commands for the 32bit architecture:

```
yum install glite-WN  
yum install glite-TORQUE_client
```

In order to configure a Torque WN, you have to specify all the configuration target in one line:

```
yaim -c -s site-info.def -n glite-WN -n TORQUE_client
```

### The lcg-CE for Torque/PBS

- Install the `lcg-CE`, `glite-TORQUE_utils` and the `glite-TORQUE_server` metapackage. In order to configure this, you have to specify all the configuration target in one line:

```
yaim -c -s site-info.def -n lcg-CE -n TORQUE_server -n TORQUE_utils
```

- Without a Torque head node: Install the `lcg-CE` and the `glite-TORQUE_utils` metapackage, then run

```
yaim -c -s site-info.def -n lcg-CE -n TORQUE_utils
```

### Standalone Torque server

- Install the `glite-TORQUE_server` metapackage. Configure it by running:

```
yaim -c -s site-info.def -n TORQUE_server -n TORQUE_utils
```

### Known issues

1. The startup level for `globus-gatekeeper` and `globus-griftp` on `lcg-CE` is enabled, please run "

```
chkconfig globus-gatekeeper on  
chkconfig globus-gridftp on
```

In case that you rebooted your `lcg-CE`, please restart `glite-lb-locallogger` by

```
/opt/glite/etc/init.d/glite-lb-locallogger start
```

- If you are installing and configuring a standalone TORQUE\_server, please remove `config_gip_sched_plugin_pbs` from `/opt/glite/yaim/node-info.d/glite-torque_utils` before running YAIM to configure it.

## The SGE batch system

**DISCLAIMER:** The SGE/gLite integration software was the result of the collaboration between 3 institutions: LIP, CESGA and LeSC. You use this software at your own risk. It may be not fully optimized or correct and therefore, should be considered as experimental. There is no guarantee that it is compatible with the way in which your site is configured.

For questions related to SGE and LCG/gLite interaction, you can use the `project-eu-egge-batchsystem-sge@cernNOSPAMPLEASE.ch` mailing list.

### The cream CE for SGE

SGE support for the CREAM CE is only supported for the gLite 3.2 framework.

### The lcg-CE for SGE

#### Configure lcg-CE and SGE Qmaster in the same physical machine

- Install SGE rpms (require **openmotif**, **pdksh** and **xorg-x11-xauth** packages available in CERN SLC repositories). These rpms will install SGE files under `/usr/local/sge/pro`:

```
# yum localinstall sge-ckpt-V62u1-1.i386.rpm sge-parallel-V62u1-1.i386.rpm sge-utils-V6
```

- Install the **lcg-CE** and **glite-SGE\_utils** meta rpm packages. This last meta rpm package depends on **glite-info-dynamic-sge**, **glite-apel-sge**, **lcg-jobmanager-sge** and **glite-yaim-sge-utils**. `glite-info-dynamic-sge` depends on **perl-XML-Twig** `>= 3.0` available from CERN SLC or standard SL repositories. The `glite-yaim-sge-utils` will configure the SGE environment for a lcg-CE but it will only work properly if a **glite-yaim-lcg-ce** `>= 4.0.3-3` version is also used to configure the lcg-CE. If all the repositories are correctly set, the necessary software packages should be installed using:

```
# yum install lcg-CE glite-SGE_utils
```

- Download and install the **SGE server yaim interface** from ETICS repository:

```
# wget http://eticsoft.web.cern.ch/eticsoft/repository/org.glite/org.glite.yaim.sge-s
# yum localinstall glite-yaim-sge-server-4.1.1-0.noarch.rpm
```

- Configure the lcg-CE, SGE\_server and SGE\_utils services:

```
# /opt/glite/yaim/bin/yaim -c -s siteinfo/site-info.def -n lcg-CE -n SGE_server -n SGE_
```

#### Configure lcg-CE and SGE Qmaster in different machines

- Install the following SGE rpms in the CE (require **openmotif**, **pdksh** and **xorg-x11-xauth** packages available in CERN SLC repositories). These rpms will install SGE files under `/usr/local/sge/pro`:

```
# yum localinstall sge-utils-V62u1-1.i386.rpm sge-V62u1-1.i386.rpm
```

- Install the **lcg-CE** and **glite-SGE\_utils** meta rpm packages. This last meta rpm package depends on **glite-info-dynamic-sge**, **glite-apel-sge**, **lcg-jobmanager-sge** and **glite-yaim-sge-utils**. **glite-info-dynamic-sge** depends on **perl-XML-Twig >= 3.0** available from the DAG repository. The **glite-yaim-sge-utils** will configure the SGE environment for a lcg-CE but it will only work properly if a **glite-yaim-lcg-ce >= 4.0.3-3** version is also used to configure the lcg-CE. If all the repositories are correctly set, the necessary software packages should be installed using:

```
# yum install lcg-CE glite-SGE_utils
```

- Configure the lcg-CE service (in `siteinfo/site-info.def` the `BATCH_SERVER` variable should point to the machine where your SGE Qmaster will run)

```
# /opt/glite/yaim/bin/yaim -c -s siteinfo/site-info.def -n lcg-CE -n SGE_utils
```

- Install all SGE rpms in the machine where the SGE Qmaster is supposed to run (require **openmotif**, **pdksh** and **xorg-x11-xauth** packages available in CERN SLC repositories). These rpms will install SGE files under `/usr/local/sge/pro`:

```
# yum localinstall sge-ckpt-V62u1-1.i386.rpm sge-parallel-V62u1-1.i386.rpm sge-utils-V62u1-1.i386.rpm
```

- Download the **SGE server yaim interface** from ETICS repository and install it in the machine where the SGE Qmaster is supposed to run (NOTE: **glite-yaim-core** and **glite-version** rpms must be installed. If you do not want to set up the gLite repositories in your SGE Qmaster machine, you can download the last version of these rpms browsing through <http://linuxsoft.cern.ch/EGEE/gLite/R3.1/generic/sl4/i386/>)

```
# wget http://eticsoft.web.cern.ch/eticsoft/repository/org.glite/org.glite.yaim.sge-server-4.1.1-0.noarch.rpm
# yum localinstall glite-yaim-sge-server-4.1.1-0.noarch.rpm
```

- Configure the SGE Qmaster server service. The SGE Qmaster queues, user set lists and exec node list will be built according to the information declared in `site-info.def` for `QUEUES`, `VOs` and `WN_LIST`, respectively. In the case where SGE Qmaster is in a dedicated machine, the current version of **glite-yaim-core** prevents the detection of a `siteinfo/group.d` directory structure for VO groups. Therefore, a unique configuration file has to be defined (containing information for all VO groups) in `siteinfo/site-info.def` otherwise the configuration will hang while running yaim.

```
# /opt/glite/yaim/bin/yaim -c -s siteinfo/site-info.def -n SGE_server
```

- In the SGE Qmaster, declare the CE as an allowed submission machine:

```
# qconf -as <CE.MY.DOMAIN>
```

- If you have control of the SGE Qmaster, be sure that in the Qmaster configuration you have the following setting: **execd\_params INHERIT\_ENV=false**. This setting allows to propagate the environment of the submission machine (CE) into the execution machine (WN). It should be there by default if you use the `sge-server yaim` plugin. If not, you can add it using:

```
# qconf -mconf
```

- If you don't have control of the SGE Qmaster and if you need to load an environment in the WNs not present by default, you can do it defining the path to a script in the Job Manager configuration file. This script will be executed in the WN, setting the proper environment. As an example, if you want to load the gLite grid environment in the WN, which by default could not be there, define `$GRID_ENV = '/etc/profile.d/grid_env.sh'` in `/opt/globus/lib/perl/Globus/GRAM/JobManager/lcgsge.conf`. The same mechanism can be set to enable interoperability with other Grid projects.
- If you run the `SGE_server` configuration (to set SGE cluster queues, scheduler and global configurations) more than once, only the first action is valid. This is done to prevent overwriting the local site administrator local configuration tuning. In such cases, a warning is sent during the configuration procedure and a standard configuration template is stored in `/tmp` (which can be uploaded manually by the site administrator).

### Link the lcg-CE with a running SGE Qmaster server

- You should ensure that you are using the same SGE version for client and server tools, and that the SGE installation paths are the same in the CE and in the SGE Qmaster server.
- Install the SGE client tools in the CE. For the SGE version described in this manual the following rpms should be deployed (require **openmotif**, **pdksh** and **xorg-x11-xauth** packages available in CERN SLC repositories). These rpms will install SGE files under `/usr/local/sge/pro`:

```
# yum localinstall sge-utils-V62u1-1.i386.rpm sge-V62u1-1.i386.rpm
```

- Install the **lcg-CE** and **glite-SGE\_utils** meta rpm packages. This last meta rpm package depends on **glite-info-dynamic-sge**, **glite-apel-sge**, **lcg-jobmanager-sge** and **glite-yaim-sge-utils**. `glite-info-dynamic-sge` depends on **perl-XML-Twig >= 3.0** available from CERN SLC or SL repositories. The `glite-yaim-sge-utils` will configure the SGE environment for a lcg-CE but it will only work properly if a **glite-yaim-lcg-ce >= 4.0.3-3** version is also used to configure the lcg-CE. If all the repositories are correctly set, the necessary software packages should be installed using:

```
# yum install lcg-CE glite-SGE_utils
```

- Change the following variables in the **site-info.def** file:

```
BATCH_SERVER="SGE Qmaster FQN"
BATCH_BIN_DIR="Directory where the SGE binary client tools are installed in the CE" Ex
```

- Set the following variables in **siteinfo/services/glite-sge\_utils.pre**:

```
SGE_ROOT="The SGE instalation dir". Ex:/usr/local/sge/pro
SGE_CELL="SGE cell definition".Ex:default
SGE_QMASTER="SGE qmaster port". Ex: 536
SGE_EXECD="SGE execd port". Ex: 537
SGE_SPOOL_METH="SGE spooling method". Ex: classic
```

- Configure the lcg-CE service.

```
/opt/glite/yaim/bin/yaim -c -s siteinfo/site-info.def -n lcg-CE -n SGE_utils
```

- If your SGE QMASTER is listening in a different port from 536, than include the following line in the SGE JM configuration file (`/opt/globus/lib/perl/Globus/GRAM/JobManager/lcgsge.conf`) after the

\$SGE\_BIN\_PATH definition :

```
$SGE_QMASTER      = '<SGE qmaster por>';
```

- In the SGE Qmaster, declare the CE as an allowed submission machine:

```
qconf -as <CE.MY.DOMAIN>
```

- If you have control of the SGE Qmaster, be sure that in the Qmaster configuration you have the following setting: **execd\_params INHERIT\_ENV=false**. This setting allows to propagate the environment of the submission machine (CE) into the execution machine (WN). It should be there by default if you use the sge-server yaim plugin. If not, you can add it using:

```
qconf -mconf
```

- If you don't have control of the SGE Qmaster and if you need to load an environment in the WNs not present by default, you can do it defining the path to a script in the Job Manager configuration file. This script will be executed in the WN, setting the proper environment. As an example, if you want to load the gLite grid environment in the WN, which by default could not be there, define **\$GRID\_ENV = '/etc/profile.d/grid\_env.sh'** in /opt/globus/lib/perl/Globus/GRAM/JobManager/lcgsge.conf. The same mechanism can be set to enable interoperability with other Grid projects.

## The WN for SGE

- Install the following SGE rpms (require **openmotif**, **pdksh** and **xorg-x11-xauth** packages available in CERN repositories). These rpms will install SGE files under /usr/local/sge/pro:

```
# yum localinstall sge-parallel-V62u1-1.i386.rpm sge-V62u1-1.i386.rpm sge-utils-V62u1-1
```

- Install the \*glite-WN\*

```
# yum install glite-WN
```

- Download the **SGE client yaim interface** from ETICS repository and install it in the machine where the SGE client is supposed to run:

```
# yum http://eticssoft.web.cern.ch/eticssoft/repository/org.glite/org.glite.yaim.sge-cl
```

- Configure the glite-WN and SGE client services:

```
# /opt/glite/yaim/bin/yaim -c -s siteinfo/site-info.def -n WN -n SGE_client
```

## The LSF batch system

You have to make sure that the necessary packages for submitting jobs to your LSF batch system are installed on your CE. By default, the packages come as tar balls. At CERN they are converted into rpms so that they can be automatically rolled out and installed in a clean way (in this case using Quattor).

Since LSF is a commercial software it is not distributed together with the gLite middleware. Visit the Platform's LSF home page [for further information](#). You'll also need to buy an appropriate number of license keys before you can use the product.

The documentation for LSF is available on Platform Manuals [web page](#). You have to register in order to be able to access it.

For questions related to LSF and LCG/gLite interaction, you can use the [project-eu-egee-batchsystem-lsf@cernNOSPAMPLEASE.ch](mailto:project-eu-egee-batchsystem-lsf@cern.ch) mailing list.

## The WN for LSF

Apart from the LSF specific configurations settings there is nothing special to do on the worker nodes. Just use the plain WN configuration target.

```
./yaim -c -s site-info.def -n glite-WN
```

## The lcg-CE for LSF

There is some special configuration settings you need to apply when configuring your LSF batch system. The most important variables to set in YAIM's `site-info.def` file are:

```
JOB_MANAGER="lcglsf"
TORQUE_SERVER="machine where the gLite LSF log file parser runs"
BATCH_LOG_DIR="/path/to/where/the/lsf/accounting/and/event/files/are"
BATCH_BIN_DIR="/path/to/where/the/lsf/executables/are"
BATCH_VERSION="LSF_6.1"
CE_BATCH_SYS="lsf"
```

For gLite installations you may use the gLite LSF log file parser daemon to access LSF accounting data over the network. The daemon needs to access the LSF event log files which you can find on the master (or some common file system which you may use for fail over). By default, yaim assumes that the daemon runs on the CE in which case you have to make sure that the event log files are readable from the CE. But note that it is not a good idea to run the LSF master service on the CE.

Make sure that you are using `lcg-info-dynamic-lsf-2.0.36` or newer.

To configure your lcg-CE use:

```
./yaim -c -s site-info.def -n lcg-CE -n LSF_utils
```

## Note on site-BDII for LSF

When you configure your site-BDII you have to populate the `[vomap]` section of the `/opt/lcg/etc/lcg-info-dynamic-scheduler.conf` file yourself. This is because LSF's internal group mapping is hard to figure out from yaim, and to be on the safe side the site admin has to crosscheck. Yaim configures the `lcg-info-dynamic-scheduler` in order to use the LSF info provider plugin which comes with meaningful default values. If you would like to change it edit the `/opt/glite/etc/lcg-info-dynamic-lsf.conf` file. After YAIM configuration you have to list the LSF group - VOMS FQAN - mappings in the `[vomap]` section of the `/opt/lcg/etc/lcg-info-dynamic-scheduler.conf` file.

As an example you see here an extract from CERN's config file:

```
vomap :
  grid_ATLAS:atlas
```

```

grid_ATLASSGM:/atlas/Role=lcgadmin
grid_ATLASPRD:/atlas/Role=production
grid_ALICE:alice
grid_ALICESGM:/alice/Role=lcgadmin
grid_ALICEPRD:/alice/Role=production
grid_CMS:cms
grid_CMSSGM:/cms/Role=lcgadmin
grid_CMSPRD:/cms/Role=production
grid_LHCB:lhcb
grid_LHCBSGM:/lhcb/Role=lcgadmin
grid_LHCBPRD:/lhcb/Role=production
grid_GEAR:gear
grid_GEARSGM:/gear/Role=lcgadmin
grid_GEANT4:geant4
grid_GEANT4SGM:/geant4/Role=lcgadmin
grid_UNOSAT:unosat
grid_UNOSAT:/unosat/Role=lcgadmin
grid_SIXT:sixt
grid_SIXTSGM:/sixt/Role=lcgadmin
grid_EELA:eela
grid_EELASGM:/eela/Role=lcgadmin
grid_DTEAM:dteam
grid_DTEAMSGM:/dteam/Role=lcgadmin
grid_DTEAMPRD:/dteam/Role=production
grid_OPS:ops
grid_OPSSGM:/ops/Role=lcgadmin
module_search_path : ../lrms:../ett

```

For further details see the `/opt/glite/share/doc/lcg-info-dynamic-lsf` file.

## The Condor batch system

To get the condor middleware go to the Condor home page<sup>?</sup>. You have to make sure that the necessary condor packages are installed on the CEs and on the WNs. On the site-BDII YAIM configures the `lcg-info-dynamic-scheduler` to use the condor info provider plugin.

A guide on how to set up a Condor batch system with lcg-CE or creamCE as Condor Submitter is available at <https://twiki.cern.ch/twiki/bin/view/EGEE/InstallationInstructionsForCondor>.

You can use the `project-eu-egge-batchsystem-condor@cernNOSPAMPLEASE.ch` mailing list if you have problems concerning gLite and Condor interaction.

# Known issues

Known issues are published in the gLite web pages [↗](#) everytime there's a new 3.1 update. Please, read carefully the release notes for each update.

YAIM maintains a list of Known issues in the Known Issues section of the YAIM guide.

## Note on hostname syntax

The WLCG middleware assumes that hostnames are case sensitive. Site administrators **MUST** not choose mix case hostnames because of that. Actually all hostnames **MUST** be in lowercase since most of the WLCG middleware depends on Globus and in particular on the `globus_hostname` function that lowercases all hostnames. If hostnames are assigned using mix cases or uppercases, then any middleware that will compare hostnames as returned by the `globus_hostname` function and as provided by clients will fail.

# Firewalls

No automatic firewall configuration is provided by this version of the configuration scripts. If your nodes are behind a firewall, you will have to ask your network manager to open a few "holes" to allow external access to some service nodes. A complete map of which port has to be accessible for each service node is maintained in CVS;

[http://jra1mw.cvs.cern.ch:8180/cgi-bin/jra1mw.cgi/org.glite.site-info.ports/doc/?only\\_with\\_tag=HEAD](http://jra1mw.cvs.cern.ch:8180/cgi-bin/jra1mw.cgi/org.glite.site-info.ports/doc/?only_with_tag=HEAD) , or you can have a look to it's html version.

# Documentation

For further documentation you can visit:

- the LCG Directory [↗](#)
- the LCG Troubleshooting Guide
- Sysadmin wiki [↗](#)
- or just find The Answer [↗](#) :-)

---

This topic: LCG > GenericInstallGuide310

Topic revision: r111 - 2011-03-10 - MariaALANDESPRADILLO



Copyright &© 2008-2020 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.  
Ideas, requests, problems regarding TWiki? Send feedback