Integrating the IEKP Linux cluster as a Tier-2/3 prototype centre into the LHC Computing Grid

Volker Büge[1,2], Ulrich Felzmann[1], Christopher Jung[1,2],
Hartmut Stadie[3], Anja Vest[1]
[1] Institut für Experimentelle Kernphysik, Universität Karlsruhe
[2] Institut für Wissenschaftliches Rechnen, Forschungszentrum Karlsruhe
[3] DESY, Hamburg

January 2006

Abstract

The local Linux cluster at the "Institut für Experimentelle Kernphysik" (IEKP), University of Karlsruhe, is shared between working groups of the high energy physics experiments AMS, CDF, and CMS and has successfully been integrated into the SAM grid of CDF and the LHC computing grid, LCG, for CMS while it still supports local users. This shared usage of the cluster causes heterogeneous software environments, grid middleware and access policies. Within the LCG, the IEKP site realises the concept of a Tier-2/3 prototype centre. The installation procedure and setup of the LCG middleware has been modified according to the local conditions and is described in detail in this note. With its dedicated configuration, the IEKP site offers the full grid functionality such as data transfers, CMS software installation, and grid based physics analyses. The need for prioritisation of certain user groups has been satisfied by supporting different Virtual Organisations.

1 Introduction

High energy physics experiments at the proton-proton collider LHC need to deal with huge data production rates and event sizes. Therefore, the demand for computing power and mass storage has significantly increased and this development will continue in future. For example, the CMS detector is designed to be read out 40 million times each second. Assuming an event size of about 1.5 MB leads to 60 TB of data per second. This enormous flow is reduced by the CMS trigger system to a final recording rate of 150 Hz or 225 MB per second [1] which has to be stored online for later processing and data reconstruction. Corresponding Monte Carlo simulations have to be generated and stored as well. The access to the data needs to be ensured for the analysis groups all over the world. Because these data samples are very large it is not possible to transfer dedicated data sets to each user performing an analysis. Rather the analysis jobs are not executed at the user's workstation but at a site where the desired data is available. The LHC experiments cope with these challenges via grid technologies. Since the analysis groups are distributed all over the world, mass storage and computing power are organised as well in a decentralised structure.

In the last years, several grid concepts have been developed for these purposes, e. g. the Scandinavian initiative NorduGrid [2], the American Grid3 [3], or the LHC specific grid, the LHC Computing Grid (LCG) [4]. The key of the LHC offline computing model [1] is a multi-tier hierarchical structure. One central Tier-0 centre located at CERN accepts raw data from the online data acquisition system of each experiment. It archives these data and performs a first pass reconstruction. The Tier-0 centre also distributes raw and processed data to the Tier-1 centres which are responsible for data archiving, reconstruction, reprocessing and other data-intensive analysis tasks. The Tier-1 centres provide regional tasks for users in their local community. Furthermore, they offer user support and the coordination of the associated Tier-2 and Tier-3 centres. The next layer consists of a numerous set of Tier-2 centres which are smaller but have substantial computing resources. These centres have to offer capacity for analyses, calibration activities, and Monte Carlo simulation. Access to larger data sets and secure storage of new data that they produce is provided by the Tier-1 centres. Finally, the Tier-3 centres offer resources that can be used interactively by local groups. In addition, they provide small computing and storage capacities for the collaborations. Within this architecture, the user's workstation is called Tier-4.

This note describes the installation and the setup of a Tier-2/3 prototype centre within the LHC Computing Grid at the "Institut für Experimentelle Kernphysik" (IEKP) at the University of Karlsruhe. Since the beginning of 2004, it is one of several sites within the LCG.

2 The LCG Components

The software framework of the LHC computing grid is still under development. In one of the next releases, the LCG software will be merged with gLite [5], the next generation middleware project of the organisation "Enabling Grids for E-sciencE" (EGEE) [6]. Each grid user is a member of at least one Virtual Organisations (VO). This abstract entity groups users and resources in the same administrative domain. One has to distinguish between grid-wide and site-wide services. The latter are services each site has to provide in order to offer computing power and disk space to a Virtual Organisation. In contrast, the grid-wide services are the infrastructure of a VO and therefore provided centrally by the VO itself. An overview of these different services and their interactions is given in this section.

2.1 Grid-wide Services

- **VO server**
  The VO server is the registry office of a Virtual Organisation. On this LDAP[1] server a list of all users belonging to a VO and their role and permissions are stored. At present, the VO server distinguishes only two kinds of members: normal grid users and the software managers. This structure of user permissions will be changed in the near future. The successor of the VO server will be the Virtual Organisation Membership Service (VOMS) server that is able to handle a more precise management of user permissions and allows users to be members in different VOs. With this server the user will be able to select its role and VO when initialising the grid-proxy, a temporary electronic passport for the user.

- **Replica Location Service:**
  When a file is copied to the grid, the Replica Location Service (RLS) gives it a global unique identifier, the GUID. All replicas of that file have this same identifier. Since the GUID is a long and incomprehensible identification string, there is also the possibility to set Logical File Names (LFN) as synonyms for the GUID. The RLS exists only once per VO and maps the logical file names to the physical file names via the GUID. Therefore it is one of the most crucial services for data management within a Virtual Organisation.

- **Information Service:**
  To manage the information on available resources on one hand and users on the grid on the other hand, a multi-layer information service is established:

  - Grid Resource Information Service (GRIS):
    This service runs locally on the grid portal machines of each site. It publishes site specific information via the GLUE[2] schema, e. g. the installed software or the number of free CPUs.
  - Grid Information Index Server (GIIS):
    This server collects the information of the local GRIS via LDAP.
  - Berkeley Database Information Index (BDII):
    The BDII is the central collection point for all information. It acquires the information from all site GIIS servers via LDAP.

- **Resource Broker**
  The Resource Broker (RB) uses information from the Information System to distribute incoming job requests to the accessible resources. It scans the user's job request for its requirements and searches through the Information System for sites that offer the resources the user desires. When the RB has found a matching site, its Job Submission Service (JSS) submits the job and all files the job needs to the corresponding site. The Logging and Bookkeeping Service (LB) controls the status of the job and can be accessed by the user. Finally, the job output is transferred to the Resource Broker and stored there until it is retrieved by the user. A Resource Broker can accommodate multiple Virtual Organisations and in turn, a single VO can have access to multiple Resource Brokers.

[1] LDAP is an acronym for Lightweight Directory Access Protocol. It is a lightweight client-server protocol for accessing directory services over the TCP/IP protocol or other connection oriented transfer services.

[2] The Grid Laboratory for a Uniform Environment (GLUE) is a schema for information description

2.2 Site-wide services

- **User Interface**
  The User Interface (UI) is the access point to the grid for the user. Several programs are installed on the UI, e. g. for submitting a job to the grid, getting information on the job status or retrieving the output of the job. The User Interface is also the portal for file access. The user can copy and register files as well as replicate existing files to other destinations. The authentication of the user is realised by the grid-proxy which has to be initialised before working on the grid.

- **Computing Element**
  The Computing Element (CE) is the portal to the local batch system of a site. It is not – as the name suggests – the place where jobs are computed; these are the Worker Nodes (WN) of the local batch system. The main reason to put this component in between the user and the Worker Nodes is simple: the CE offers a layer of abstraction, i. e. the user does not have to deal with different batch systems on different sites.

  When a job is submitted to the CE, the middleware looks up in the grid-mapfile[3], whether the user, authenticated by the grid-proxy, is allowed to run his job on this site or not. Furthermore, the grid-mapfile contains the information to which local account the user is mapped. At present there are three possibilities for this mapping:

  - The user has no local account (standard):
    The job is mapped to a generic mapping account. These are usually numbered accounts of the type VO-name plus a three digits number, e. g. cms001.
  - The user has a local account:
    The local site manager decides whether the job is mapped to the user's local account or a generic mapping account.
  - The user is the software manager of the VO:
    On each site, a dedicated software directory is accessible via an environment variable[4]. The VO software manager, who is mapped to an account with special privileges, is authorised to install software in this directory. Here, all other members of the VO only have read permissions.

  The mapping is described in detail in chapter 4.6.1. In a future release of LCG, an improved management of permissions in the form of VOMS will be available.

- **Worker Node**
  The CE submits a grid job via the local batch system to the Worker Node (WN) and copies the input data for the job to a certain subdirectory of the user's login directory on the WN. The grid job has access to the files stored in this directory as well as to files stored on the grid and the web via the gridftp protocol and the wget tool, respectively. After the job is finished, the CE sends the job output back to the Resource Broker.

- **Storage Element**
  The Storage Element (SE) is the grid node used for data storage. Like the Computing Element, it can be a gateway to the local storage system, but it also can offer disk space itself. Since the SE is a Globus Gridftp service, it supports by default the gsiftp protocol. Other protocols like SRM [7] can be added by the local site manager. Since LCG version 2.4.0, also dCache [8] is

[3] /etc/grid-security/grid-mapfile
[4] $VO_CMS_SW_DIR for users of the VO CMS

provided. At smaller sites like the IEKP, only disk space is available whereas larger sites like GridKa [9] offer disk and tape space as well as tape archival.

- **Monitoring Box**
  The Monitoring Box (MON) publishes information about the functionality of all affiliated grid site nodes. The monitoring tool used is the Relational Grid Monitoring Architecture (R-GMA).

3 The IEKP Linux cluster

The Linux cluster located at the IEKP consists of two parts, the inner network and the outer network. It comprises portal machines, file servers, computing nodes and a controlling machine. The cluster is internally called "EKPplus" and its setup has already been described in [10]. Operating experience has lead to further studies which are depicted in detail in [11]. Since May 2005, the LCG components CE, SE, UI and MON have been successfully integrated into the EKPplus cluster. The network architecture and the cluster component specifications are described in this chapter.

3.1 Network architecture

The inner network of the EKPplus cluster hosts the computing nodes "ekpplusXXX", several file servers "ekpfsX" and a dedicated cluster control machine named "ekpplusctl". This control machine takes care about the local users, manages the job queues for the batch system and provides the root file system for the computing nodes.

The outer network consists of publicly accessible portals which serve as testbeds for the development of analysis software. To accomplish this task for local users of the CDF, CMS and AMS working groups, experiment specific software is installed and kept up-to-date on the five different portals assigned to the respective experiments. Via multiple Ethernet cards, the portals are also connected to the inner private network. Thus, they offer access to the file servers and the usage of the Worker Nodes via the local batch system.

Apart from the monitoring host "ekp-lcg-mon" which is only connected to the outer network, all other LCG components – Computing Element "ekp-lcg-ce", Storage Element "ekp-lcg-se" and User Interface "ekp-lcg-ui" – are fully integrated into the EKPplus cluster. Therefore, they are connected to the outer and the inner network, like all portal machines. The User Interface provides a grid access point for local users and allows them to submit jobs to the grid. The Storage Element stores the VO specific software and re-exports some file-space of the file servers reserved for LCG purposes. The Computing Element forwards the received LCG jobs to the cluster control machine "ekpplusctl" which enqueues the job in the local batch system. In this manner, the internal computing nodes "ekpplusXXX" need to be and are considered as LCG Worker Nodes since the Computing Element acts only as a gateway between the the LHC Computing Grid and the existing local Linux cluster.

The whole EKPplus cluster is connected to the IEKP desktop network by a 1 GBit connection. It is currently protected against attacks from the outside by a so called "bridged firewall" named "ekpplusnat". The former firewall which was described in [12] was replaced in April 2004. The necessary modifications in the firewall configuration required by the LCG middleware is described in chapter 4.3. Figure 1 depicts the overall architecture of the EKPplus cluster and the integration of the LCG components.

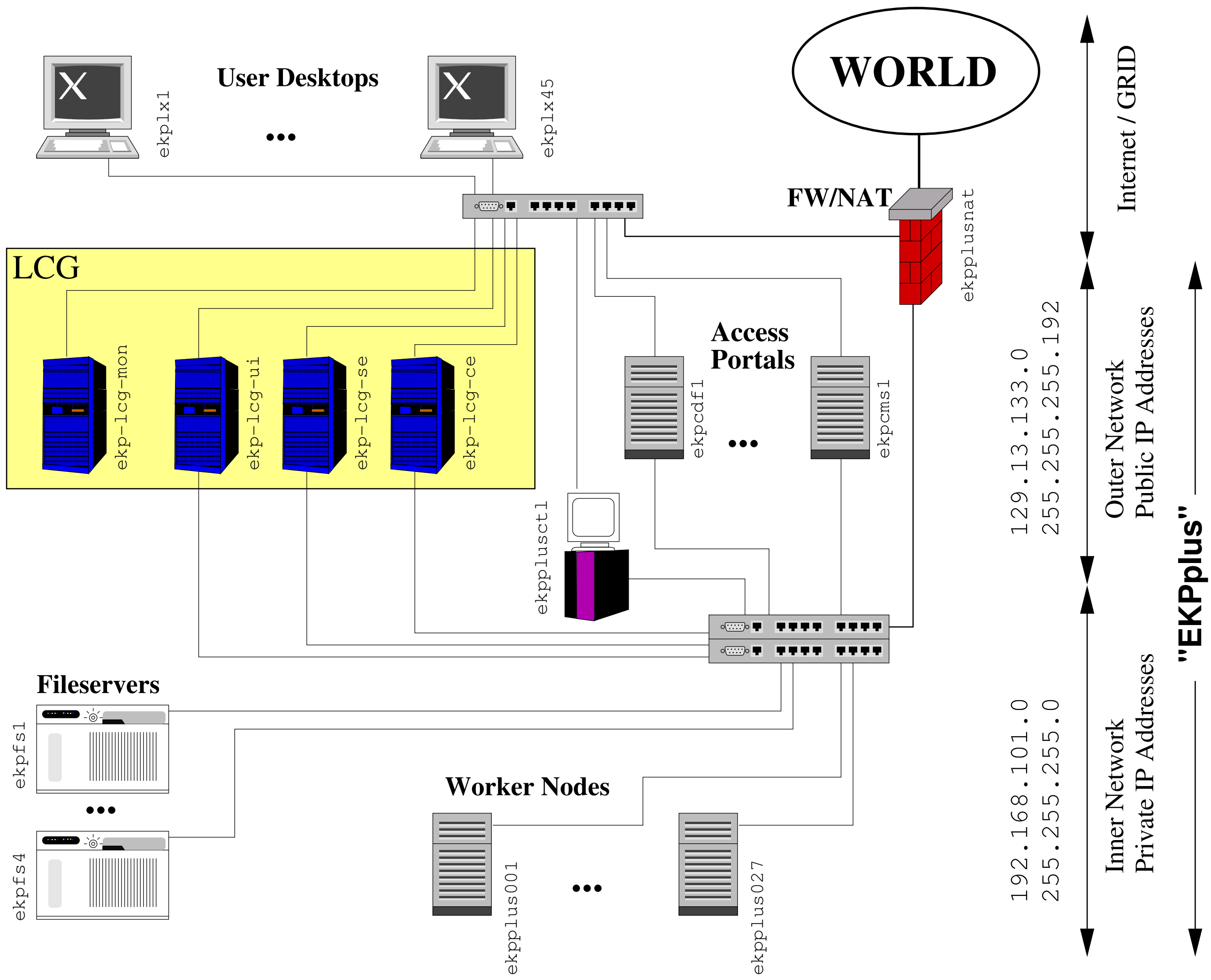

Figure 1: A schematic overview of the architecture of the EKPplus cluster and the integration of the LCG components CE, SE, UI and MON.

3.2 Cluster component specifications

The specification of the local cluster components and the LCG site services hosted at the IEKP are described in detail in the following.

- Five file servers provide a disk space of about 20 TB in total. At the moment, about 1 TB can be used for the storage of official datasets of a VO accessible for grid users.
- The Storage Element comprises an AMD Athlon™ XP 1600+ processor with 512 MB SDRAM. Its disk space offered to grid users for writing and reading 'private' datasets is 240 GB.
- The local batch system consists of 27 computing nodes. The first 18 nodes ekpplus001 – ekpplus018 mainly provide AMD Athlon 2600+ processors, the nine nodes ekpplus019 – ekpplus027 are AMD Dual Opteron machines. All nodes have 1 GB memory per CPU and a local disk space of 40 GB per CPU. Since the computing nodes support the LCG software, they can be used as grid Worker Nodes (see chapter 2.2) and are shared between local and grid users. The Opteron machines run a 32-bit operating system; an upgrade to a 64-bit operating system is foreseen in the near future.

- Both, the Computing Element and the User Interface comprise an AMD Athlon™ processor with 1,4 GHz and 512 MB SDRAM.

- The Monitoring Box is an AMD Athlon™ XP 2100+ processor with 1.7 GHz and 1 GB SDRAM.

Since the load of the LCG components is minor, a virtualisation on a single capable server will be implemented in the near future. This virtualisation can improve the utilisation of resources as well as security aspects.

The local monitoring system used at the IEKP is Ganglia [13] which is based on a hierarchical design.

4 Installation and setup of LCG at the IEKP

The installation procedure and setup of the LCG middleware has been modified according to the local conditions of the IEKP Linux cluster and is described in detail in this chapter. It is specified in the site configuration file

```
ekp-lcg-ui:/opt/lcg/yaim/examples/site-info_260.def
```

provided by the configuration and installation tool *yaim* (see chapter 4.2.2), where the number 260 refers to the used version of the middleware, LCG 2.6.0.

The need for a prioritisation of certain user groups has been satisfied by supporting different Virtual Organisations. Since January 2005, an infrastructure for the VO 'dcms' is setup at DESY (see chapter 5). This VO comprises and supports all CMS members registered at the German grid sites of Aachen, Hamburg and Karlsruhe. Thus, all members of the VO dcms are also members of the VO cms. The IEKP currently supports the following LCG VOs:

- cms: VO for the members of the CMS Collaboration

- dcms: VO for the German members of the CMS Collaboration

- dteam: VO for the LCG deployment team

- dech: VO for the EGEE Federation DE/CH

The VO dech is supported mainly for the setup of a testbed used during the participation in the GridKa school at the Forschungszentrum Karlsruhe in September 2005 [14].

4.1 History

Up to LCG version 2.2.0, the installation of the LCG software had been performed using LCFGng [15]. This system can automatically install and configure a large number of nodes. The configuration of the grid node types to be installed is specified in macro files on an LCFGng server, a dedicated installation node. Nevertheless, LCFGng could not cope with the heterogeneous hardware and software environment at the IEKP, since at each installation the operating system RedHat 7.3 was newly installed. Moreover, the Worker Node software installation had to be done with the "manual installation", which was offered by the LCG software additionally to the LCFGng method.

4.2 Generic installation

For the installation of the LCG software, the generic installation method has been chosen. It is provided for Scientific Linux (SL) since LCG version 2.3.0. This successor of the older RedHat 7.3 operating system is based on the RedHat Enterprise Server. It is recompiled and maintained by CERN and Fermilab. The release of the operating system used at the IEKP is Scientific Linux CERN Edition (SLC) 3.0.6. On the Worker Nodes a slightly modified release is used. The current software release of the grid middleware is LCG 2.6.0. In this section, the procedure of the generic installation is described. A more detailed documentation can be found on the web:

```
http://grid-deployment.web.cern.ch/grid-deployment/documentation/\
       LCG2-Manual-Install/
```

At the IEKP, user accounts and home directories are exported by the ekpplusctl via the NIS[5] and NFS[6] services. Using the generic installation, the mapping accounts for the grid user are generated automatically on the CE and the SE. Since the NIS protocol is used to export all users from one central machine, no local accounts should be generated. Thus, some extra configuration steps described in chapter 4.2.3 have to be executed.

The Worker Nodes are controlled by a PBS/Torque [16] batch system on the cluster control machine ekpplusctl. Since the WNs are already configured and used by other users, the needed LCG software components have to be installed manually as described in chapter 4.4.3.

4.2.1 Java

Before installing the LCG middleware one has to make sure that Java is installed on the system. The recommended J2SDK 1.4.2 rpm package can be downloaded from the SUN java web site

```
http://java.sun.com/j2se/1.4.2/download.html
```

4.2.2 Yaim, the installation and configuration tool

The current installation of the LCG software of the grid nodes SE, CE, UI and MON at the IEKP has been performed with the configuration and installation tool yaim. It is a configurable tool and offers a flexible method for the installation process. Moreover, an installation server is no longer needed. The latest version of yaim can be found at

```
http://www.cern.ch/grid-deployment/gis/yaim/
```

The yaim version number corresponds to the version of LCG it intends to configure. In order to work with yaim, it first has to be downloaded using wget, then it has to be installed on each target node:

[5]The Network Information Service (NIS) provides information like login names, passwords, host names and IP numbers to all machines on the network. It has been developed by Sun and was formerly known as “Sun Yellow Pages”.

[6]The Network File System (NFS) allows machines to mount disk partitions on a remote machine like a local hard drive.

```
wget http://www.cern.ch/grid-deployment/gis/yaim/\
                         lcg-yaim-2.6.0-10.noarch.rpm
```

```
rpm -ivh lcg-yaim-2.6.0-10.noarch.rpm
```

Executing these commands, the yaim software is installed in the newly created directory

```
ekp-lcg-<node>:/opt/lcg/yaim/
```

where `<node>` means 'se', 'ce', 'ui' or 'mon'. This directory contains the needed scripts as well as example configuration files for the installation. Since both, the configuration files and the scripts are relevant for all different grid node types, the directory `/opt/lcg/yaim/` can be exported from one machine to all other grid nodes. At the IEKP, this directory is exported from the UI to the SE, CE and MON.

4.2.3 The site configuration files

The file

```
ekp-lcg-ui:/opt/lcg/yaim/examples/site-info_260.def
```

contains the site specific configuration. It is based on the example file `site-info.def` placed in the same directory. The configuration is performed by key-value pairs. The full version of the `site-info_260.def` file adapted for the grid type nodes at the IEKP can be found in appendix A. This site configuration file is used by the configuration scripts located in the directory

```
ekp-lcg-ui:/opt/lcg/yaim/functions/
```

In the CE configuration context, a file which contains a plain list of the managed batch computing nodes needs to be adapted. This configuration file is named

```
ekp-lcg-ui:/opt/lcg/yaim/examples/wn-list_260.def
```

The path and the name of this file are pointed by the variable `WN_LIST` in the site configuration file. For test purposes a dedicated grid Worker Node named

```
ekplcgwn0.physik.uni-karlsruhe.de
```

has been configured. It is the only content of the `wn-list_260.def` file. Since the batch nodes of the local cluster are managed by the controlling machine ekpplusctl, the CE cannot be an additional PBS server for the Worker Nodes. Therefore, the PBS/Torque server has to be specified by the variable `TORQUE_SERVER` in the site configuration file and is set to

```
TORQUE_SERVER=ekpplusctl.ekpplus.cluster
```

A plain list of the grid users' generic mapping accounts belonging to a VO supported at the IEKP, their IDs, user groups and VOs are specified in a configuration file called

```
ekp-lcg-ui:/opt/lcg/yaim/examples/users_260_ekp_cmsgrid.conf
```

The format of this file is

```
UID:LOGIN:GID:GROUP:VO:SGM_FLAG
```

It is read out by the variable `USERS_CONF` in the site configuration file. At the IEKP, the local user accounts and home directories are exported by the ekpplusctl via NIS and NFS to all other machines. In order to prevent yaim from also creating these mapping accounts, an empty configuration file

```
ekp-lcg-ui:/opt/lcg/yaim/examples/users_260_ekp_empty.conf
```

is used which is read out by the introduced variable `USERS_CONF_EMPTY`. In the two configuration scripts

```
/opt/lcg/yaim/functions/config_edgusers
/opt/lcg/yaim/functions/config_users
```

the former variable `USERS_CONF` is replaced by the new variable `USERS_CONF_EMPTY`. With these changes, yaim does not create local user accounts anymore.

4.2.4 The RPM installation tool APT

The proposed installation and configuration method is based on the Debian advanced packaging tool (APT) [17] and on a set of shell scripts built within the yaim framework. If APT is not installed it can be downloaded from

```
ftp://ftp.scientificlinux.org/linux/scientific/305/i386/SL/\
       RPMS/apt-0.5.15cnc6-4.SL.i386.rpm
```

In order to perform the middleware and Certification Authorities (CA) installation (see chapters 4.2.5 and 4.2.6), the variables `LCG_REPOSITORY` and `CA_REPOSITORY` have to be configured in the site configuration file `site-info_260.def` as follows:

```
LCG_REPOSITORY=''rpm http://grid-deployment.web.cern.ch/\
                 grid-deployment/gis apt/LCG-2_6_0/sl3/en/\
                 i386 lcg_sl3 lcg_sl3.updates''

CA_REPOSITORY=''rpm http://grid-deployment.web.cern.ch/\
                grid-deployment/gis apt/LCG_CA/en/i386 lcg''
```

For the dependencies of the middleware to be installed, it has to be ensured that APT can find and download the specific operating system rpms. For this purpose the site administrators have to install an rpm called 'apt-sourceslist', or else create an appropriate file in the `/etc/apt/sources.list.d` directory. In order to cover all known dependencies solved by the apt-get command, at least the lists `lcg.list` and `lcg-ca.list` should be present in `/etc/apt/sources.list.d/`.

4.2.5 Middleware installation

After having adapted the configuration file `site-info_260.def`, the installation of the different nodes (CE, WN, UI, MON and SE) with the desired middleware packages is done by executing the command

```
/opt/lcg/yaim/scripts/install_node \
    /opt/lcg/yaim/examples/site-info_260.def <meta-package>
```

Here, the WN installed is the test Worker Node `ekplcgwn0` (see chapter 4.2.3) and `<meta-package>` represents the different node types

```
lcg-CE_torque
lcg-WN_torque
lcg-UI
lcg-MON
lcg-SE_classic
```

For example, the installation of a Computing Element including the Torque scheduler is performed by the command:

```
/opt/lcg/yaim/scripts/install_node \
    /opt/lcg/yaim/examples/site-info_260.def lcg-CE-torque
```

There is a known installation conflict between the 'torque-clients' rpm and the 'postfix' mail client [18]. In order to workaround the problem it is proposed to either uninstall postfix or to remove the file `/usr/share/man/man8/qmgr.8.gz` from the target node. The latter method has been used during the installation procedure at the IEKP.

4.2.6 Certification Authorities installation

In order to authenticate with Grid resources, a user needs to have a digital X.509 certificate issued by a Certification Authority (CA) trusted by LCG. The installation of the up-to-date version of the Certification Authorities is automatically done by the middleware installation described in chapter 4.2.5. Anyway, as the list and structure of the CAs 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 LCG release procedure. In order to ascertain the version number of the latest set of CA rpms, the web page

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

should be consulted. An upgrade of the CA list of the nodes to the latest version can be achieved with the command:

```
apt-get update && apt-get -y install lcg-CA
```

To keep the CA configuration up-to-date, a periodic upgrade procedure of the CA on the installed nodes is applied by running the above command via a cron job every eight hours.

4.2.7 Host certificates

Among other nodes which are not supported at the IEKP, the CE, SE and MON require the host certificate and host key files before the installation is started. Instructions to obtain a CA rpm list can be found in

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

Once a valid certificate is obtained, i. e. a file `hostcert.pem` containing the machine public key and a file `hostkey.pem` containing the machine private key, these two files have to be placed on the target node into the directory `/etc/grid-security`. In order to obtain a new host certificate, the following steps have to be executed:

1. If a host certificate already exists, move the existing files `hostcert.pem` to `hostcert.pem-old` and `hostkey.pem` to `hostkey.pem-old`.

2. Run the command

   ```
   grid-cert-request -host <fullyQualifiedDomainName_of_your_host> \
                        -dir <cert_request_path> -int
   ```

   Fill in the fields:

   ```
   Level 0 Organization [GermanGrid]: GermanGrid
   Level 0 Organizational Unit [EKP]: <your_organisation>
   Name []: <fullyQualifiedDomainName_of_your_host>
   ```

3. Sign the new `hostcert_request` with your personal user certificate and user key

   ```
   openssl smime -sign -in <cert_request_path>/hostcert_request.pem \
   -out signed-hostcert_request.pem \
   -signer ~/.globus/usercert.pem \
   -inkey ~/.globus/userkey.pem -text
   ```

   The usercert and userkey can usually be found in `~/.globus` in the user's login directory.

4. Send the signed request to the CA with:

   ```
   mail -s "hostreq" gridka-ca@iwr.fzk.de \
                       < signed-hostcert_request.pem
   ```

   or as attachment with any email program.

5. In order to have a valid host certificate until the expiration date of the old one, the final step is to copy the new files `hostkey.pem` and `hostcert_request.pem` to `hostkey.pem-new` and `hostcert_request.pem-new`, respectively. Then, the old certificates have to be renamed again as mentioned under 1.

4.2.8 Middleware configuration

The general procedure to configure the middleware packages which have been installed on the considered nodes via the procedure described in chapter 4.2.5, is to run the command:

```
/opt/lcg/yaim/scripts/configure_node \
     /opt/lcg/yaim/examples/site-info_260.def <node_type>
```

where `<node_type>` describes the different nodes types:

```
CE_torque
WN_torque
UI
MON
SE_classic
```

4.3 Firewall Configuration

Since the LCG components are fully integrated into the EKPplus net (129.13.133.0 / 255.255.255.192), they are placed behind the firewall "ekpplusnat" of the institute. To allow external access to the services run by LCG, some ports of the firewall have to be opened to these dedicated hosts. Actually the whole internal campus net is maintained and therefore protected by the Universitäts-Rechenzentrum Karlsruhe (RZ). The higher-ranking firewall has been switched off for the EKPplus net completely to ease debugging and reduce administration.

Every node behind the firewall is allowed to connect to the outside world with almost no restrictions. In order to conserve the number of IP addresses used, the Worker Nodes are connected to the private network segment exclusively. To provide the required outbound connectivity from the Worker Nodes, the institute's firewall translates the private addresses of the respective Worker Node to the IP address of its own public interface.

A complete list of the accessible services required by the LHC experiment software can be found at the following URL:

```
http://lcgdeploy.cvs.cern.ch/cgi-bin/lcgdeploy.cgi/lcg2/docs/\
       lcg-port-table.pdf
```

In table 1, the necessary inbound ports needed by the LCG middleware are summarised.

4.4 Node-specific installation and configuration steps

In this section, the configuration steps that are needed to complete the configuration in addition to the automatic configuration scripts provided by yaim are listed. Some of these configuration aspects are common for several or all node types and should be executed by yaim.

[9] locally configurable using the GLOBUS_TCP_PORT_RANGE environment variable

[10] not mandatory, but useful

Node	Port	Service
SE	80	SRM - http (apache)
SE	2135, 2136	GRIS, Extended GRIS
SE	2811	GridFTP Control
SE	3147	RFIO
SE	6375	SE / apache-ssl
SE	8080,8088	SE (tomcat)
SE	8443	gLite File Transfer Service
SE	20000 - 25000	GridFTP data [9]
CE	2119	Globus & EDG Gatekeepers (GRAM)
CE	2135	GRIS
CE	2170, 2171	BDII (ldap)
CE	2811	GridFTP Control
CE	9002	locallogger (logd)
CE	20000 - 25000	GridFTP data [9]
MON	80	http (apache)
MON	2169	GOUT (ldap)
MON	8080, 8088, 8443	R-GMA (tomcat)
UI	22	SSH [10]

Table 1: This table describes the inbound network connectivity from the nodes required by the LHC experiment software.

4.4.1 All grid node types aspects

- In general, on the various node types the VOs are reflected as Unix groups. In order to map certificate subjects of the jobs users have submitted uniquely, generic mapping accounts are used. Those accounts are named as the groups with a number appended. A total of 50 accounts per pool have been created on the ekpplusctl and are exported to all nodes. For each mapping account, in `ekp-lcg-<node>:/etc/grid-security/gridmapdir/` a file is created by the configuration script

  ```
  ekp-lcg-<node>:/opt/lcg/yaim/functions/config_mkgridmap
  ```

- To ensure that all LCG machines have always the same time, a NTP[7] server is used. This service synchronises the machine time with a given timer server. At the IEKP, the time servers of the Universitäts-Rechenzentrum Karlsruhe[8] can be used as well as the cluster control machine ekpplusctl which acts also as a time server since it is time synchronised with the RZ servers. The NTP configuration file `/etc/ntp.conf` has to be modified and should contain the lines

  ```
  # --- OUR TIMESERVERS -----
  ekpplusctl.ekpplus.cluster
  ```

  Before the NTP service is started, it is recommended to synchronise the machine time manually. This can be done using the program ntpdate with the ekpplusctl as time server:

  ```
  ntpdate ekpplusctl
  ```

[7]Network Time Protocol

[8]ntp0.rz.uni-karlsruhe.de, ntp1.rz.uni-karlsruhe.de and ntp2.rz.uni-karlsruhe.de

This synchronisation will only work when the NTPD service is disabled. After having successfully adjusted the time of the node, the NTPD service can be started. To enable that this service will be started within the boot procedure of the node, the service is activated with the chkconfig command:

```
service ntpd start
chkconfig ntpd on
```

The status of the NTP can be checked by running the command

```
ntpq -p
```

- In the directory `ekp-lcg-<node>:/opt/edg/externals/lib/`, there should exist the three symbolic links

  ```
  libswigtcl8.so.0 -> libswigtcl8.so.0.0.0
  libswigpy.so.0 -> libswigpy.so.0.0.0
  libswigpl.so.0 -> libswigpl.so.0.0.0
  ```

  which are needed during the middleware configuration. If these links do not exist, they have to be created manually.

4.4.2 CE_torque

The Computing Element is installed with the procedure described in section 4.2.5 after having finished the adaption of the configuration file `site-info_260.def`. Some important information about the CE are summarised in this section

- The globus-mds GRIS has to report the presents of the VOs for each batch queue. Accordingly, a line like

  ```
  GlueCEAccessControlBaseRule: VO:dcms
  ```

  is added to the file `ekp-lcg-ce:/opt/edg/var/etc/ce-static.ldif` for each VO.

- The number of CPUs published by the LCG site is defined in the file

  ```
  ekp-lcg-ce:/opt/lcg/libexec/lcg-info-dynamic-pbs
  ```

  by the line

  ```
  $TotalCPU += $num_pro;
  ```

  where `$num_pro` is the number of processes which can be run on each CPU. In order to publish less Worker Nodes than the local batch system contains, one might modify the variable `$num_pro`. At the IEKP, currently 20 Worker Nodes are published to the grid.

- On the CE, as well as on the SE, the host keys of the WNs have to be added to the file

  ```
  ekp-lcg-ce:/etc/ssh/ssh_known_hosts
  ```

- All information about incoming jobs and grid users are stored in the log-files

  ```
  ekp-lcg-ce:/var/log/messages
  ekp-lcg-ce:/var/log/globus-gatekeeper.log
  ekp-lcg-ce:/var/log/globus-gridftp.log
  ```

4.4.3 WN_torque

To ease maintenance and possible upgrades, all Worker Nodes have an identical root file-system which is hosted on the central cluster control machine and exported to each Worker Node read-only. In addition, a separate home directory is exported write-enabled for each node to store host specific configuration like ssh host keys or log files for the local batch system. The root file-system which contains Scientific Linux CERN 3 was created with some modified scripts supported by the CERN nbtools project [19].

Apart from the `/tmp/` directory and the swap space, the Worker Nodes do not own a local file system. Therefore, the LCG software had to be installed into a special directory at the cluster control machine. This was done by executing the necessary steps in a chroot environment. First, the meta rpm package

```
lcg-WN-torque
```

which only contains dependencies, was installed. Then, the yaim site configuration script was run to perform the basic installation settings needed by LCG. Because the Torque package for the Worker Nodes delivered with LCG did not worked properly, it was recompiled from sources and installed manually. Furthermore, the following settings have to be considered:

- Since the Worker Nodes and the Storage Element communicate using the GridFTP protocol, the transaction cannot take place connecting through the internal private net. Hence, the public IP address for the Storage Element must be made available to the Worker Nodes. Otherwise the GridFTP session fails during the authentication.
- SSH has to be allowed to use host based authentication between the Worker Nodes and the Computing Element. This can be achieved by adding the following lines to the system-wide configuration file `ssh_config`:

  ```
  RhostsAuthentication yes
  PasswordAuthentication yes
  EnableSSHKeysign yes
  HostbasedAuthentication yes
  ```

  In addition, the host-names of the Computing Element and the Storage Element have to be mentioned in the system-wide configuration file `shosts.equiv`.
- The latest Certificate Revocation Lists (CRL) has to be known by the Worker Nodes. A cron job which runs several times a day and copies the relevant certificates using afs from CERN to the corresponding globus directory takes care about it.

4.4.4 SE_classic

The installation of the LCG middleware packages on the Storage Element is performed similarly to the installation on the CE. In addition, appropriate directories for the supported VOs have to be created and are defined in the site configuration file:

```
ekp-lcg-ui:/opt/lcg/yaim/examples/site-info_260.def
```

- The official datasets of a VO can be stored in a subdirectory located in

  ```
  ekp-lcg-se:/grid/data/
  ```

  The subdirectories are named `cms/`, `cmsgrid/`, `dcms/`, `dech/`, and `dteam/`, according to the supported VOs and can be read by each grid user belonging to the VO.

- When a grid user replicates a "private" file, it is stored in the directory

  ```
  ekp-lcg-se:/grid/users/
  ```

  with the corresponding subdirectories. Here, the subdirectories are created by the yaim configuration script

  ```
  ekp-lcg-se:/opt/lcg/yaim/functions/config_seclassic
  ```

- The VO related software is installed by the software managers in the directory

  ```
  ekp-lcg-se:/grid/sw/
  ```

  and the according subdirectories which are created by the yaim configuration script

  ```
  ekp-lcg-se:/opt/lcg/yaim/functions/config_sw_dir
  ```

4.4.5 UI

On the User Interface the users obtain their grid proxies. Moreover, it provides all programs for the job submission and data management on the grid. The installation and configuration of the UI is again done with yaim using the commands introduced in section 4.2.5. In the directory

```
ekp-lcg-ui:/etc/grid-security/certificates/
```

all allowed Certification Authorities have to be present. The UI does not provide any information to the GIIS. Nevertheless, the coordinates of the Resource Broker, the Logging Host and the Proxy Server have to be defined on the UI. These machines are hosted at DESY and specified in the site configuration file by the entries:

```
RB_HOST=grid-rb0.desy.de
PX_HOST=grid-pxy.desy.de
BDII_HOST=grid-bdii.desy.de
```

4.4.6 MON

The Monitoring Box is not connected to the private net of the EKPplus cluster. Since the ekp-plusctl can only act as NTP-server for internal IP addresses, the time server of the Universitäts-Rechenzentrum is used on the used Monitoring Box, i. e. the `ekp-lcg-mon:/etc/ntp.conf` file contains:

```
# --- OUR TIMESERVERS -----
restrict 129.13.96.2 mask 255.255.255.255 nomodify notrap noquery
server ntp1.rz.uni-karlsruhe.de
```

4.5 Cluster Control Host

The cluster control machine – named "ekpplusctl" – is not part of the LCG components itself, because no pure LCG services are running on it. Nevertheless, it hosts the root file-systems mounted by the Worker Nodes and runs the resource manager and the scheduler for the local batch system. Moreover, it takes care about user accounts and groups including the generic mapping accounts and exports the login directories.

The underlying operating system for the ekpplusctl is Scientific Linux CERN 3, as well as for the LCG hosts. A detailed description of the batch system and the user management is given in the following chapter, a short summary of the configuration steps to carry out is given here:

- Generic user accounts and generic group accounts used by LCG users are created on the ekpplusctl and are announced to the Worker Nodes using NIS. To prevent batch jobs from creating ssh back doors, the file attributes for the respective `.ssh` subdirectories are set to "immutable". This blocks certain user accounts from installing "authorized_keys" files. Furthermore, it is important that pool account users can not set up "at jobs" or "cron jobs". Adding the pool accounts to the files `/etc/at.deny` and `/etc/cron.deny`, respectively, refuses the creation of such jobs.

- The host-names of the Computing Element and the Storage Element are added to the system-wide configuration file `shosts.equiv` to enable the user to copy files without password authentication.

- A cron job mentioned above updates the certificates for the Worker Nodes every eight hours.

- In addition to the existing local queues, a corresponding queue for every supported VO is created.

4.6 Accounting and Billing

The advantage of a batch system is the possibility to submit many jobs at the same time to the system. At the IEKP, the scheduler used to send a job to the next free Worker Node is the flexible system MAUI [20]. In order to take the user's history of CPU time usage into account, MAUI supports the fair share principle: every user, including grid users, has an account with a certain amount of priority credits. By submitting a job to the batch system, the user loses credit points corresponding to the CPU time of the job. These points are restored after a defined time period. The advantage of this system is that a user with more priority credits left is prioritised compared to a user having spent all these points. Nevertheless, a user with no points left or even a negative number of points has access to the batch system when a Worker Node is not used by any other, more prioritised users. The scheduler MAUI is able to manage both, user and group fair share. Since the grid mapping accounts can not be identified with certain users, only group fair share is applied on these accounts.

The treatment of grid users depending on their affiliation is organised by mapping them to different accounts and user groups and by the configuration of the queues they can use. Both, the mapping and the queues are described in the next sections.

4.6.1 Mapping

Each grid user presents a Distinguished Name (DN) contained in a digital certificate on job submission, e. g.

```
/O=GermanGrid/OU=EKP/CN=Anja Vest
```

A user being a member of the CMS collaboration and coming to the IEKP LCG site, using either the VO 'cms' or 'dcms', is mapped to a dedicated account depending on its affiliation. This mapping is steered by the file

```
ekp-lcg-ce:/etc/grid-security/grid-mapfile
```

If the grid user is not a member of a German institute, he gets mapped to a generic account cms001 – cms050 which are members of the user group cmsgrid. A grid user from Aachen or Hamburg using the VO dcms is mapped to a generic account dcms001 – dcms050 which are members of both, the user groups dcms and cmsgrid. A grid user from Karlsruhe is mapped to its local account which is specified on the CE and the SE each within the file

```
ekp-lcg-ce:/opt/edg/etc/grid-mapfile-local
```

The local user is a member of the local user group cms and the user groups dcms and cmsgrid.

The mapping is controlled by the cron job

```
ekp-lcg-<node>:/etc/cron.d/lcg-expiregridmapdir
```

running on all grid node types. On the Compute Element, the hard links between the certificate DN and the generic local account which can be found in

```
ekp-lcg-ce:/etc/grid-security/gridmapdir/
```

are deleted after 48 hours if no jobs of the user are in the batch queue.

All mapping accounts have corresponding storage directories on the Storage Element. According to their affiliation, the grid users can use the IEKP capacities with different priorities which are managed by fair share targets. In order to priotise the local IEKP users, the fair share credits of the local user groups are set to 30. The grid mapping account groups dcms and dech have a fair share target of 10, whereas the grid user group cmsgrid has fair share credits of 5. The fair share target for the user group dteam is 70 to ensure that the LCG deployment team is always able to run its site functional test jobs, even if the site is loaded with jobs. The handling of the grid users at the IEKP depending on their affiliation is summarised in table 2. All fair share targets are listed in table 3 ordered by their value.

	CMS members	DCMS members	IEKP-CMS members
Mapping accounts	cms001 – cms050	dcms001 – dcms050	local account
Group membership	cmsgrid	dcms + cmsgrid	cms + dcms + cmsgrid
Storage directories	/grid/users/cmsgrid/	/grid/users/dcms/	/grid/users/cms/
Priority	low priority	medium priority	high priority

Table 2: Mapping of the different grid users depending on their affiliation.

User group	Fair share target
dteam	70
cms	30
cdf	30
ams	10
dcms	10
dech	10
cmsgrid	5

Table 3: Fair share targets of the user groups at the IEKP.

4.6.2 Queues

In addition to the local batch queues, one grid queue according to each VO supported by the IEKP LCG site is configured. These queues are dedicated to jobs submitted via the grid and have the same names as the respective VOs:

```
ekp-lcg-ce.physik.uni-karlsruhe.de:2119/jobmanager-lcgpbs-cms
ekp-lcg-ce.physik.uni-karlsruhe.de:2119/jobmanager-lcgpbs-dcms
ekp-lcg-ce.physik.uni-karlsruhe.de:2119/jobmanager-lcgpbs-dech
ekp-lcg-ce.physik.uni-karlsruhe.de:2119/jobmanager-lcgpbs-dteam
```

The dteam queue is mainly used for site functional tests and has therefore a lower CPU and wall time than the cms, dcms or dech queue. The grid queues have the following properties:

- The cms, dcms and dech queues are limited to 24 hours of CPU time and 36 hours of wall time and have a medium priority (nice 10).
- The dteam queue is limited to 2 hours of CPU time and 4 hours wall time and also has a medium priority (nice 10).
- The maximum number of running jobs for all queues is 15.
- The maximum number of jobs per user of each VO is 9.
- The queue lengths are normalised to the CPU capacities (1kSPEC CINT2000).

The grid queues have been set up with the queue manager daemon ’qmgr’ (example: queue cms):

```
create queue cms
set queue cms queue_type = Execution
set queue cms Priority = 20
set queue cms max_queuable = 500
set queue cms max_running = 15
set queue cms resources_max.cput = 23:59:00
set queue cms resources_max.nice = 10
set queue cms resources_max.walltime = 36:05:00
set queue cms resources_min.nice = 10
set queue cms resources_default.neednodes = grid
set queue cms resources_default.nice = 10
set queue cms resources_default.nodect = 0
set queue cms acl_group_enable = True
set queue cms acl_groups = cms
set queue cms acl_groups += dcms
set queue cms max_user_run = 9
set queue cms enabled = True
set queue cms started = True
```

The line `’set queue cms acl_groups += dcms’` ensures that members of the VO dcms which are mapped to generic dcms accounts also can run on the cms grid queue. Thus, dcms grid users can use the two queues

```
ekp-lcg-ce.physik.uni-karlsruhe.de:2119/jobmanager-lcgpbs-cms
ekp-lcg-ce.physik.uni-karlsruhe.de:2119/jobmanager-lcgpbs-dcms
```

4.7 Test of the installation

A detailed description of how to test a LCG site can be found at the URL

```
http://grid-deployment.web.cern.ch/grid-deployment/documentation/\
     LCG2-Site-Testing/
```

5 New DESY-hosted VO dcms

In this chapter, the configuration to enable the DESY-hosted VO dcms at the IEKP LCG site is summarised. The installation of a new VO can be done via ’yaim’. The site configuration file has been edited for the creation of the new VO dcms and the relevant entries are:

```
VO_DCMS_SW_DIR=$VO_SW_DIR/dcms
VO_DCMS_DEFAULT_SE=$SE_HOST
VO_DCMS_STORAGE_DIR=$CE_CLOSE_SE1_ACCESS_POINT/dcms
VO_DCMS_QUEUES="dcms"
VO_DCMS_USERS=ldap://grid-vo.desy.de/ou=dcms,ou=vo,o=desy,c=de
```

The dcms software directory

Node	Name	Needed on node type
VO	ldap://grid-vo.desy.de/ou=dcms,ou=vo,o=desy,c=de	CE,SE
RB	grid-rb0.desy.de	UI
BDII	grid-bdii.desy.de	UI,WN
PXY	grid-pxy.desy.de	UI,WN
GIIS	grid-giis.desy.de	BDII

Table 4: The relevant coordinates for the DESY-hosted VOs

```
ekp-lcg-se:/grid/sw/dcms/
```

is currently empty since the relevant software for all cms users is already stored in the cms software directory

```
ekp-lcg-se:/grid/sw/cms/
```

The relevant coordinates for the DESY-hosted VOs are summarised in table 4. For the users of the new VO local user groups as well as local generic accounts have been created. In the configuration file

```
ekp-lcg-ce:/opt/edg/etc/edg-mkgridmap.conf
```

the entries for the VO dcms need to be placed *before* the CMS server:

```
# Map VO members  dcmssgm
group ldap://grid-vo.desy.de/ou=dcmssgm,ou=vo,o=desy,c=de dcmssgm

# Map VO members  dcms
group ldap://grid-vo.desy.de/ou=dcms,ou=vo,o=desy,c=de .dcms
...
# A list of authorised users
auth ldap://grid-vo.desy.de/ou=people,o=desy,c=de
...
```

The software manager dcmssgm is defined analogously to cmssgm, but currently not in use.

On the UI, the DESY Resource Broker is specified in the configuration file

```
ekp-lcg-ui:/opt/edg/etc/dcms/edg_wl_ui.conf:

[
  VirtualOrganisation = "dcms";
  NSAddresses = "grid-rb0.desy.de:7772";
  LBAddresses = "grid-rb0.desy.de:9000";
  ## HLR location is optional.
  # HLRLocation = "fake HLR Location"
  ## MyProxyServer is optional.
  MyProxyServer = "grid-pxy.desy.de"
]
```

The listing of the VO dcms looks as follows:

```
dn: ou=dcms,ou=vo,o=desy,c=de
ou: dcms
cn: VO dcms
description: VO dcms
owner: cn=Andreas Gellrich,ou=admins,o=desy,c=de
member: cn=Anja Vest,ou=people,o=desy,c=de
member: cn=Christopher Jung,ou=people,o=desy,c=de
member: cn=Volker Buege,ou=people,o=desy,c=de
...
```

whereas the listing of the VO dcmssgm looks as:

```
dn: ou=dcmssgm,ou=vo,o=desy,c=de
objectClass: groupOfNames
ou: dcmssgm
cn: VO dcmssgm
description: VO dcmssgm
owner: cn=Andreas Gellrich,ou=admins,o=desy,c=de
member: cn=Klaus Rabbertz,ou=people,o=desy,c=de
```

Acknowledgements

We are grateful to Prof. Günter Quast for his support and to the IEKP system administrators for their work in constructing and maintaining the IEKP Linux cluster, namely Yves Kemp, Michal Kreps, Christian Sander and Patrick Schemitz. Furthermore, we thank the dcms LCG administrators Andreas Gellrich, Mathias de Riese, Matthias Kirsch and Andreas Nowack for their support during the LCG middleware installation procedure. We wish to thank the German Bundesministerium für Bildung und Forschung BMBF for financial support, and we are grateful for the technical support by the Universitäts-Rechenzentrum Karlsruhe.

A The yaim site configuration file

`ekp-lcg-ui:/opt/lcg/yaim/examples/site-info_260.def:`

```
MY_DOMAIN=physik.uni-karlsruhe.de

CE_HOST=ekp-lcg-ce.$MY_DOMAIN
SE_HOST=ekp-lcg-se.$MY_DOMAIN
RB_HOST=grid-rb0.desy.de
PX_HOST=grid-pxy.desy.de
BDII_HOST=grid-bdii.desy.de
MON_HOST=ekp-lcg-mon.$MY_DOMAIN
REG_HOST=lcgic01.gridpp.rl.ac.uk
```

```
## Only for sites with Experiment Software Area under AFS
GSSKLOG=no
GSSKLOG_SERVER=my-gssklog.$MY_DOMAIN

## Change this if your torque server is not on the CE
TORQUE_SERVER=ekpplusctl.ekpplus.cluster

WN_LIST=/opt/lcg/yaim/examples/wn-list_260.conf
USERS_CONF=/opt/lcg/yaim/examples/users_260_ekp_cmsgrid.conf
USERS_CONF_EMPTY=/opt/lcg/yaim/examples/users_260_empty.conf
FUNCTIONS_DIR=/opt/lcg/yaim/functions

## Pick the apt-get sources appropriate to your OS
LCG_REPOSITORY="'rpm http://linuxsoft.cern.ch LCG/apt/LCG-2_6_0/\
                sl3/en/i386 lcg_sl3 lcg_sl3.updates' \
                'rpm http://grid-deployment.web.cern.ch/\
                grid-deployment/gis \
                apt/LCG-2_6_0/sl3/en/i386 lcg_sl3 lcg_sl3.updates'"

CA_REPOSITORY="rpm http://grid-deployment.web.cern.ch/\
               grid-deployment/gis apt/LCG_CA/en/i386 lcg"

## For the relocatable distribution,
## ensure that INSTALL_ROOT is set correctly
INSTALL_ROOT=/opt

## You will probably want to change these too for the relocatable dist
OUTPUT_STORAGE=/tmp/jobOutput
JAVA_LOCATION="/usr/java/j2sdk1.4.2_08"

## Set this to '/dev/null'
## if you want to turn off yaim installation of cron jobs
CRON_DIR=/etc/cron.d

GLOBUS_TCP_PORT_RANGE="20000 25000"

MYSQL_PASSWORD=secret

APEL_DB_PASSWORD="APELDB_PWD"

GRID_TRUSTED_BROKERS=" "

GRIDMAP_AUTH="ldap://lcg-registrar.cern.ch/ou=users,o=registrar,\
              dc=lcg,dc=org \
              ldap://grid-vo.desy.de/ou=people,o=desy,c=de \
              ldap://rb.scai.fraunhofer.de/ou=people,o=dech,dc=lcg,dc=org"

GRIDICE_SERVER_HOST=$SE_HOST

SITE_EMAIL=lcgadmin@ekp.uni-karlsruhe.de
SITE_NAME=ekplcg2
```

```
SITE_LOC="Karlsruhe, Germany"
SITE_LAT=49.01055556
SITE_LONG=8.41138889
SITE_WEB="http://www-ekp.physik.uni-karlsruhe.de/~lcgadmin/"
SITE_TIER="TIER 3"
SITE_SUPPORT_SITE="http://www.uni-karslruhe.de"

## SE_classic should be 'disk', SE_dpm or SE_dcache should be 'srm_v1'
SE_TYPE=disk

JOB_MANAGER=lcgpbs

CE_BATCH_SYS=torque
CE_CPU_MODEL=PIII
CE_CPU_VENDOR=athlon
CE_CPU_SPEED=1400
CE_OS=SLC
CE_OS_RELEASE=3.0.6
CE_MINPHYSMEM=513
CE_MINVIRTMEM=1025
CE_SMPSIZE=1
CE_SI00=381
CE_SF00=0
CE_OUTBOUNDIP=TRUE
CE_INBOUNDIP=FALSE
CE_RUNTIMEENV="LCG-2 LCG-2_1_0 LCG-2_1_1 LCG-2_2_0 LCG-2_3_0 LCG-2_3_1 \
               LCG-2_4_0 LCG-2_6_0 R-GMA"
CE_CLOSE_SE="SE1"
CE_CLOSE_SE1_HOST=$SE_HOST
CE_CLOSE_SE1_ACCESS_POINT=/grid/users
# CE_CLOSE_SE2_HOST=another-se.$MY_DOMAIN
# CE_CLOSE_SE2_ACCESS_POINT=/somewhere

## dCache-specific settings, ignore if you are not running d-cache
DCACHE_ADMIN="my-admin-node"
DCACHE_POOLS="my-pool-node1:/pool-path1 my-pool-node2:/pool-path2"
## Optional
# DCACHE_PORT_RANGE="20000,25000"

## SE_dpm-specific settings. Ignore if you are not running a DPM
DPMDATA=$CE_CLOSE_SE1_ACCESS_POINT
## The database user
DPMMGR=the-dpm-db-user
## The database user password
DPMUSER_PWD=the-dpm-db-pwd
DPMFSIZE=200M
## Set this if you are building a DPM yourself
## and/or if you need a default DPM for the lcg-stdout-mon
DPM_HOST=$SE_HOST
DPMPOOL=the_dpm_pool_name
## Optional
# DPM_PORT_RANGE="20000,25000"
```

```
FTS_SERVER_URL="https://fts.${MY_DOMAIN}:8443/path/glite-data-transfer-fts"

BDII_HTTP_URL="http://grid-deployment.web.cern.ch/grid-deployment/gis/\
               lcg2-bdii/dteam/lcg2-all-sites.conf"
## list of the services provided by the site
BDII_REGIONS="CE SE"
BDII_CE_URL="ldap://$CE_HOST:2135/mds-vo-name=local,o=grid"
BDII_SE_URL="ldap://$SE_HOST:2135/mds-vo-name=local,o=grid"
BDII_RB_URL="ldap://$RB_HOST:2135/mds-vo-name=local,o=grid"
BDII_PX_URL="ldap://$PX_HOST:2135/mds-vo-name=local,o=grid"
BDII_VOBOX_URL="ldap://$VOBOX_HOST:2135/mds-vo-name=local,o=grid"

VOS="dcms cms dteam dech"
QUEUES="dcms cms dteam dech"

VO_SW_DIR=/grid/sw

VO_DCMS_SW_DIR=$VO_SW_DIR/dcms
VO_DCMS_DEFAULT_SE=$SE_HOST
VO_DCMS_STORAGE_DIR=$CE_CLOSE_SE1_ACCESS_POINT/dcms
VO_DCMS_QUEUES="dcms"
VO_DCMS_USERS=ldap://grid-vo.desy.de/ou=dcms,ou=vo,o=desy,c=de

VO_CMS_SW_DIR=$VO_SW_DIR/cms
VO_CMS_DEFAULT_SE=$SE_HOST
VO_CMS_STORAGE_DIR=$CE_CLOSE_SE1_ACCESS_POINT/cmsgrid
VO_CMS_SGM=ldap://grid-vo.nikhef.nl/ou=lcgadmin,o=cms,dc=eu-datagrid,dc=org
VO_CMS_USERS=ldap://grid-vo.nikhef.nl/ou=lcg1,o=cms,dc=eu-datagrid,dc=org
VO_CMS_VOMS_SERVERS="vomss://lcg-voms.cern.ch:8443/voms/cms?/cms/"

VO_DECH_SW_DIR=$VO_SW_DIR/dech
VO_DECH_DEFAULT_SE=$SE_HOST
VO_DECH_STORAGE_DIR=$CE_CLOSE_SE1_ACCESS_POINT/dech
VO_DECH_QUEUES="dech"
VO_DECH_SGM=ldap://rb.scai.fraunhofer.de/ou=lcgadmin,o=dech,dc=lcg,dc=org
VO_DECH_USERS=ldap://rb.scai.fraunhofer.de/ou=lcg1,o=dech,dc=lcg,dc=org

VO_DTEAM_SW_DIR=$VO_SW_DIR/dteam
VO_DTEAM_DEFAULT_SE=$SE_HOST
VO_DTEAM_STORAGE_DIR=$CE_CLOSE_SE1_ACCESS_POINT/dteam
VO_DTEAM_QUEUES="dteam"
VO_DTEAM_SGM=ldap://lcg-vo.cern.ch/ou=lcgadmin,o=dteam,dc=lcg,dc=org
VO_DTEAM_USERS=ldap://lcg-vo.cern.ch/ou=lcg1,o=dteam,dc=lcg,dc=org
VO_DTEAM_VOMS_SERVERS="vomss://lcg-voms.cern.ch:8443/voms/dteam?/dteam/"
```

B Important files or directories on the grid nodes

```
ekp-lcg-ce:/etc/grid-security/gridmapdir/
ekp-lcg-ce:/opt/edg/etc/edg-mkgridmap.conf
```

```
 ekp-lcg-ce:/opt/lcg/libexec/lcg-info-dynamic-pbs

ekp-lcg-se:/etc/grid-security/gridmapdir/
ekp-lcg-se:/opt/edg/etc/edg-mkgridmap.conf

ekp-lcg-ui:/opt/edg/etc/edg_wl_ui_cmd_var.conf
ekp-lcg-ui:/opt/edg/etc/myvo/edg_wl_ui.conf
ekp-lcg-ui:/etc/profile.d/lcgenv.sh
ekp-lcg-ui:/opt/edg/var/etc/edg-replica-manager/edg-replica-manager.conf

ekp-lcg-mon:/etc/ntp.conf
```

where ’myvo’ can be one of the VOs supported at the IEKP.

C Working on the Grid

In this section, a brief introduction of how to submit a job and manage data storage on the grid is given. A more detailed overview on these topics with some examples can be found in [21].

C.1 Executing a Job on the Grid

The Job Description Language (JDL) [22] has been developed in order to provide the Resource Broker with all information it needs to decide to which site the job can be sent to for execution as well as information for the Worker Node. All these information are specified in one file. The minimum content of such a jdl-file is the following:

```
Executable     ="ShellScript.sh";
InputSandbox   ={"ShellScript.sh"};
StdOutput      ="std.out";
StdError       ="std.err";
OutputSandbox  ={"std.out","std.err"};
```

This example file is called from here on ”anyname.jdl”. It informs the Resource Broker, that the file ShellScript.sh has to be transferred to and executed on the Worker Node. The standard output and error are written into the files std.out and std.err, respectively.

In order to authenticate to the Resource Broker and to be authorised to use the site the job will be sent to for execution, the user has to initialise his grid-proxy:

```
grid-proxy-init
```

The command

```
edg-job-list-match --vo cms anyname.jdl
```

provides the user with a list of all CEs that are matching the requirements of the job. The submission of the job to one of these CEs is done by the command

```
edg-job-submit --vo cms -r SITE:QUEUE -o job.txt anyname.jdl
```

The optional flag `-r` can be used to name a special site and queue for the job to be executed on. This option as well as the determination of the VO (`--vo cms`) can also be specified in the `anyname.jdl` file. The job identifier can be stored in a file specified by the flag `-o`. Information on the status of the job are available via

```
edg-job-status -i job.txt
```

and the output of the job, when having finished successfully, can be retrieved using

```
edg-job-get-output -i job.txt --dir ./
```

This command stores the files of the Output Sandbox in the directory specified by the flag `--dir`.

C.2 Data Storage on the Grid

Information like the name and free space of a Storage Element can be obtained using

```
lcg-infosites --vo cms --is se
```

Having found the name of a Storage Element, a file can be copied and registered with the command

```
lcg-cr --vo cms -l lfn:<mylogicalfilename> -d <SE>\
        file:<path and filename>
```

If the file is often needed, a replica of it can be transferred to other Storage Elements. This is done by

```
lcg-rep --vo cms -d <SE 2> lfn:<mylogicalfilename>
```

A copy of a file stored on the grid can be retrieved with the command

```
lcg-cp --vo cms lfn:<mylogicalfilename> file:‘pwd‘/<localcopyname>
```

Finally, all replicas of a file can be deleted using

```
lcg-del --vo cms -a GUID
```

C.3 CRAB

For an easier handling of the job submission, the CRAB (CMS Remote Analysis Builder) [23] tool can be used. It interacts directly with the grid middleware so that the user does not have to deal with the LCG commands described above.

References

[1] I. Bird *et al.*, “LHC computing Grid. Technical design report”, CERN-LHCC-2005-024 2

[2] P. Eerola *et al.*, “The NorduGrid architecture and tools”, eConf **C0303241** (2003) MOAT003, http://www.nordugrid.org/ 2

[3] Grid3 Homepage: http://www.ivdgl.org/grid2003/ 2

[4] LHC Computing Grid Homepage: http://lcg.web.cern.ch/LCG/ 2

[5] gLite Homepage: http://glite.web.cern.ch/glite/ 2

[6] Enabling Grids for E-sciencE Homepage: http://public.eu-egee.org/ 2

[7] The Storage Resource Manager Collaboration Homepage: http://www-isd.fnal.gov/srm/ 4

[8] dCache Homepage: http://dcache.desy.de/ 4

[9] J. van Wezel and H. Marten, “GridKa: The Grid Computing Centre Karlsruhe for applications from high-energy physics”, Nachr. Forsch. zentr. Karlsruhe **36** (2004) 138. http://www.gridka.de/ 5

[10] P. Schemitz and M. Stanitzki, “Aufbau eines Datenanalyse-Clusters”, IEKP-KA/2003-12, http://www-ekp.physik.uni-karlsruhe.de/ clusmon/EKPplus/ekpplus-iekp-ka2003-12.pdf 5

[11] Y. Kemp, PhD thesis in preparation, will be available as IEKP-KA/2006-1 5

[12] P. Schemitz, “Eine Hard- und Software-Umgebung für Physik-Analysen bei CDF II”, PhD thesis, IEKP-KA/03-01, http://www-ekp.physik.uni-karlsruhe.de/pub/web/thesis/schemitz_phd.ps.gz 5

[13] Ganglia Monitoring System Homepage: http://ganglia.sourceforge.net/ 7

[14] GridKa school 2005 Homepage: http://gks05.fzk.de/ 7

[15] LCFGng Homepage: http://www.lcfg.org/ 7

[16] Tera-scale Open-source Resource and QUEue manager TORQUE Homepage: http://www.clusterresources.com/products/torque/ 8

[17] T. L. Williams, “A Postprocessor For And Automatic Programmed Tool (Apt) Processor”, Y-2013, http://www.debian.org/doc/manuals/apt-howto/index.en.html 10

[18] LCG Operations – Bugs: bug #5509 http://savannah.cern.ch/bugs/?func=detailitem&item_id=5509 11

[19] CERN nbtools project Homepage: http://linux.web.cern.ch/linux/scientific3/docs/netboot/ 16

[20] Maui Cluster Scheduler Homepage: http://www.clusterresources.com/products/maui/ 18

[21] Volker Büge, “Aufbau eines Grid-Standorts zum Einsatz der CMS-spezifischen Software und Messung von Parametern des W- und Z-Bosons am LHC”, Diploma thesis, IEKP-KA/2005-21, http://www-ekp.physik.uni-karlsruhe.de/pub/web/thesis/iekp-ka2005-21.pdf 27

[22] LCG-2 user scenario, CERN-LCG-GDEIS-498081,
http://public.eu-egee.org/files/UserScenario2.pdf 27

[23] Crab Homepage: http://cmsdoc.cern.ch/cms/ccs/wm/www/Crab/ 28