



Enabling Grids for E-science

# *Laboratory: Hands-on using EGEE Grid and gLite middleware*

***Athanasia Asiki***

***[aassiki@cslab.ece.ntua.gr](mailto:aassiki@cslab.ece.ntua.gr)***

***Computing Systems Laboratory,  
National Technical University of Athens***

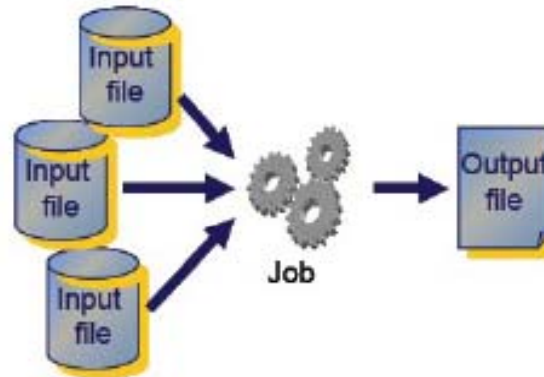
[www.eu-egee.org](http://www.eu-egee.org)



Information Society  
and Media







- **The execution of a typical Grid application follows this scenario:**
  - The user submits its application's job to the "Grid"
  - The job is being executed
  - The job's execution may include the processing of one or more **Input Files** stored in a Storage node
  - The job may produce one or more **Output Files**
  - The **Output Files** can be stored somewhere in the Grid system (perhaps in the Storage Element or in the User Interface)
  - The User can access the **Output Files** using the corresponding Grid mechanisms



## WORLDWIDE LHC COMPUTING GRID

### GLITE 3.1 USER GUIDE MANUALS SERIES

---

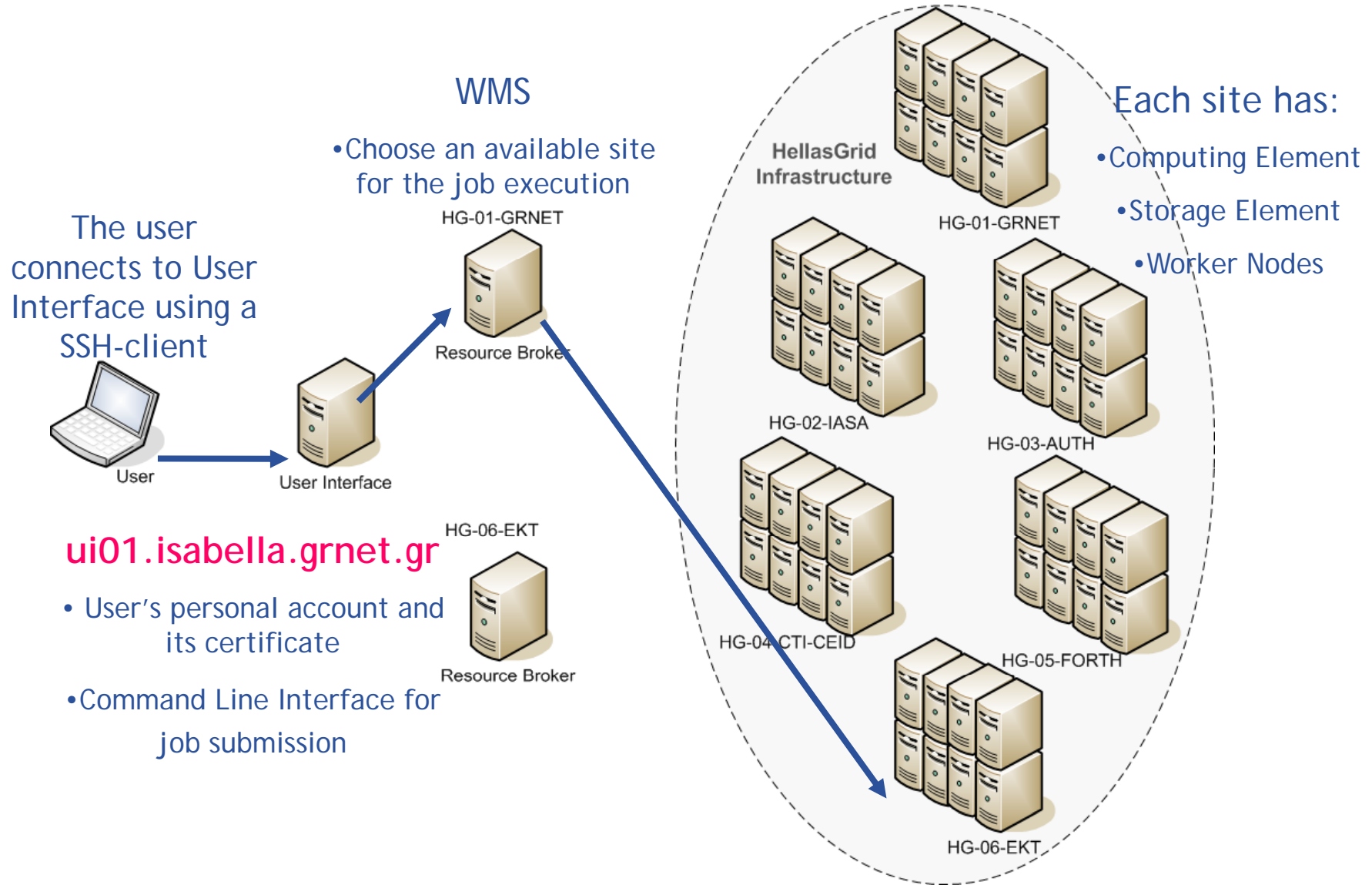
**Document Identifier:** CERN-LCG-GDEI8-722388  
**EDMS Id:** 722388  
**Version:** 1.2  
**Date:** March 7, 2008  
**Section:** Experiment Integration and Distributed Analysis  
**Document status:** DRAFT  
**Author(s):** Stephen Burke, Simone Campana, Patricia Méndez Lorenzo, Christopher Nater, Roberto Santinelli, Andrea Sciabà  
**File:** gLite-3-UserGuide

---

***Abstract:** This guide is an introduction to the WLCG/eGEE Grid and to the gLite 3.1 middleware from a user's point of view.*

---

<http://glite.web.cern.ch/glite/documentation/>



- **Creating a proxy certificate**
  - **voms-proxy-init --voms=hgdemo**
- **Listing Computing Elements that match a job description**
  - **glite-wms-job-list-match -a testJob1.jdl**
- **Submitting a job**
  - **glite-wms-job-submit -o jobld -a testJob1.jdl**
- **Retrieving the status of a job**
  - **glite-job-status -i jobld**
- **Retrieving the output of a job**
  - **glite-wms-job-output -i jobld**

- **Retrieving the status of a job**
  - `glite-wms-job-status --all`
  - `glite-wms-job-status --all -s CLEARED`
- **Retrieving job output to a specific directory**
  - `glite-wms-job-output -i jobId --dir ~/job1`
- **Retrieving logging information about submitted jobs**
  - `glite-wms-job-logging-info -i jobId`

- Create an appropriate .jdl file in order to run the executable file of myhostname.c in the Grid

- myhostname.c

```
#include <stdio.h>
```

```
#include <malloc.h>
```

```
#define BUF_SIZE 1000
```

```
int main( int argc, char *argv[]) {
```

```
    char *hostname;
```

```
    hostname = (char *) malloc(BUF_SIZE);
```

```
    gethostname(hostname,255);
```

```
    printf("host is %s\n",hostname);
```

```
    free(hostname);
```

```
    return 0;
```

```
}
```



- **Modify a copy of the testJob1.jdl**
  - `cp testJob1.jdl myhostname.jdl`
- **Compile:**
  - `gcc -o myhostname myhostname.c`
- **Identify the correct executable for the new job**

- **BuildandRun.sh**

```
#!/bin/sh  
gcc $1.c -o $1  
./$1
```

- ⇒ **Compile myhostname.c on the CE itself.**
- ⇒ **Use a bash script to run the compiler and then run the executable (BuildandRun.sh)**
- ⇒ **Modify testJob1.jdl file so that the script "buildandrun.sh" is run with the parameter "myhostname"**

- **Add the attributes to the JDL file**
  - `PerusalFileEnable = true;`
  - `PerusalTimeInterval = 30;`

- **Select files to be inspected**

- `glite-wms-job-perusal --set -f std.out -f std.err <JobID>`

- *Result:*

- Connecting to the service [https://wms03.egee-see.org:7443/glite\\_wms\\_wmproxy\\_server](https://wms03.egee-see.org:7443/glite_wms_wmproxy_server)*

- ===== glite-wms-job-perusal Success =====*

- Files perusal has been successfully enabled for the job:*

- <https://lb01.egee-see.org:9000/uZV2wC49XZF57jgQVsCBXw>*

- **Retrieve status**

- `glite-wms-job-perusal --get -f std.out <JobId>`

- **Job Collection: Submit a set of independent jobs**

## Preparation

- **Create a jdl directory**
  - `mkdir jdl/`
- **Move jdl files to the above directory**
  - `cp myhostname.jdl jdl/`
  - `cd jdl`
  - `cp myhostname.jdl pi.jdl`
  - `vi pi.jdl ...`

- **Submit job collection**

- `glite-wms-job-submit -o collec -a --collection jdl`

Connecting to the service [https://wms02.egee-see.org:7443/glite\\_wms\\_wmproxy\\_server](https://wms02.egee-see.org:7443/glite_wms_wmproxy_server)

```
===== glite-wms-job-submit Success =====
```

The job has been successfully submitted to the WMPProxy

Your job identifier is:

<https://wms02.egee-see.org:9000/59yfALHeA23KSTECrIHKyw>

The job identifier has been saved in the following file:

`/home/training/egee01/collec`

```
=====
```

Retrieve status

`glite-wms-job-status -i collec`

- **lcg-infosites** ⇒ obtain VO-specific information on existing Grid resources

**lcg-infosites --vo <vo> <option> -v <verbosity> -f <site> --is <bdii>**

where:

**--vo <vo>**: the name of the VO to which the information to print is related (mandatory)

**<option>**: specifies what information has to be printed. It can take the following values:

**ce**: the number of CPUs, running jobs, waiting jobs and CE names (global, no VO-specific information)

**se**: the names of the SEs supporting the VO, the type of storage system and the used and available space;

**-v 1: only the CE / SE names**

**-v 2: the cluster names, the amount of RAM, the operating system name and version and the processor model**

**all**: the information given by ce and se

**closeSE**: the names of the CEs supporting the VO and their close SEs

**tag**: the software tags published by each CE supporting the VO

**lfc**: the hostname of the LFC catalogues available to the VO

**lfcLocal**: the hostname of the local LFC catalogues available to the VO

**rb**: the hostname and port of the RBs available to the VO

**dli**: the Data Location Index servers available to the VO

**dliLocal**: the local Data Location Index servers available to the VO

**sitenames**: the names of all WLCG/EGEE sites;

- Obtaining information about computing resources

✓ [egee01@ui01 egee01]\$ **lcg-infosites --vo hgdemo ce**

valor del bdii: bdii.isabella.grnet.gr:2170

#CPU	Free	Total Jobs	Running	Waiting	ComputingElement
11	7	0	0	0	node001.grid.auth.gr:2119/jobmanager-pbs-hgdemo
20	19	0	0	0	ce02.marie.hellasgrid.gr:2119/jobmanager-pbs-hgdemo
64	12	0	0	0	ce01.isabella.grnet.gr:2119/jobmanager-pbs-hgdemo
118	53	0	0	0	ce01.marie.hellasgrid.gr:2119/jobmanager-pbs-hgdemo
118	53	0	0	0	glite-ce01.marie.hellasgrid.gr:2119/blah-pbs-hgdemo
122	110	0	0	0	ce01.afroditi.hellasgrid.gr:2119/jobmanager-pbs-hgdemo

[...]

- Obtaining information about storage resources

✓ [egee01@ui01 egee01]\$ **lcg-infosites --vo hgdemo se**

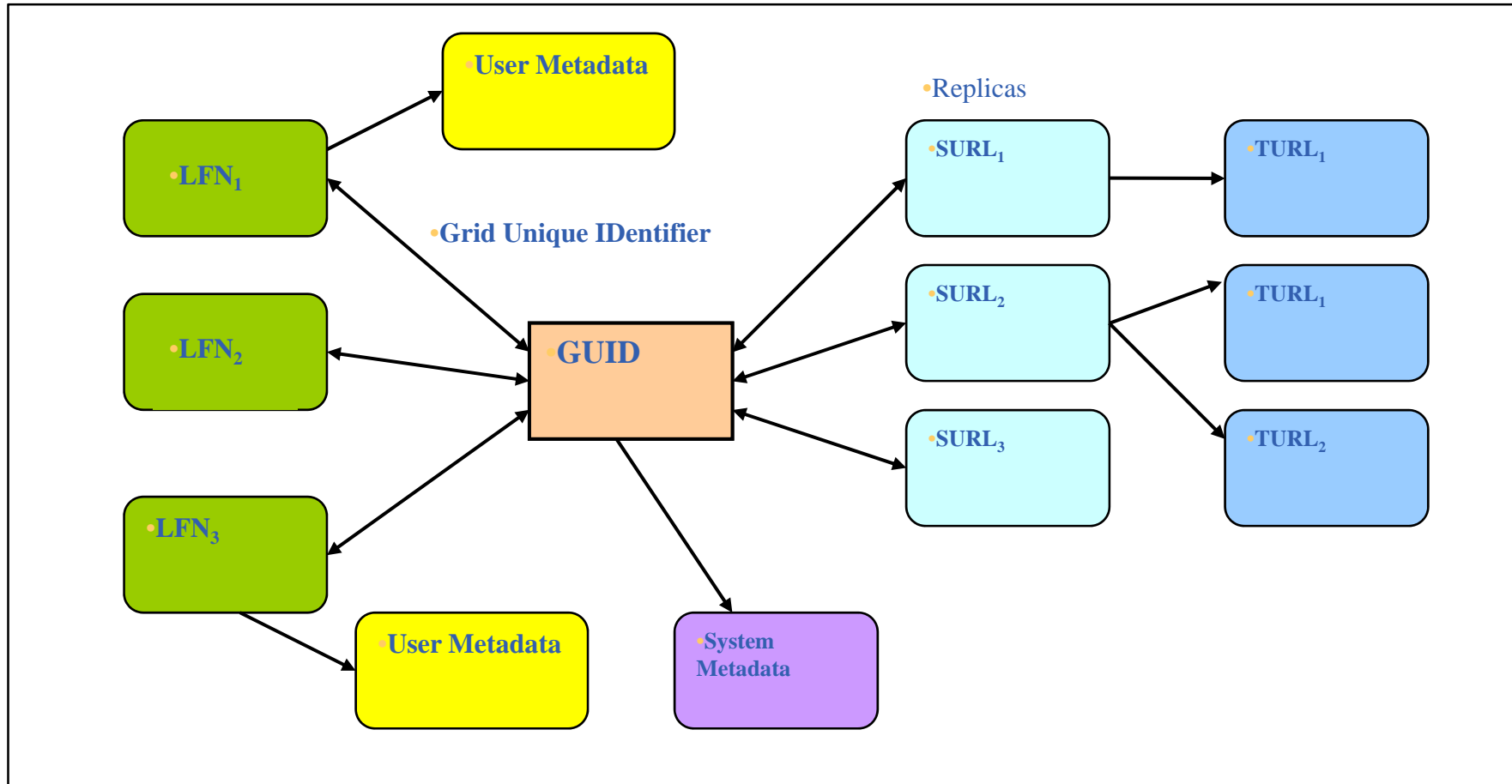
Avail Space(Kb)	Used Space(Kb)	Type	SEs
45766072	100254364	n.a	se.phy.bg.ac.yu
28651116	81503928	n.a	se001.grid.bas.bg
170660000	16280000	n.a	se02.marie.hellasgrid.gr
1780407808	3362175488	n.a	se01.isabella.grnet.gr
2730000000	200000000	n.a	se01.marie.hellasgrid.gr
2360000000	550000000	n.a	se01.afroditi.hellasgrid.gr
2500000000	190000000	n.a	se01.kallisto.hellasgrid.gr
2640000000	350000000	n.a	se01.ariagni.hellasgrid.gr

[...]



- **Listing the hostname of the LFC catalogues**
  - ✓ [egee01@ui01 egee01]\$ **lcg-infosites --vo hgdemo lfc**  
lfc.isabella.grnet.gr
- **Listing the software tags published by each CE supporting the VO**
  - ✓ [egee01@ui01 egee01]\$ **lcg-infosites --vo see tag**  
valor del bdii: bdii.isabella.grnet.gr:2170  
Name of the CE: g02.phy.bg.ac.yu  
VO-seegrid-vive-0.4.2  
VO-seegrid-vive-0.4.3  
[...]
- **Listing all WLCG/EGEE sitenames**
  - ✓ [egee01@ui01 egee01]\$ **lcg-infosites --vo hgdemo sitenames**  
[...]  
HG-01-GRNET  
HG-02-IASA  
HG-03-AUTH  
HG-04-CTI-CEID  
HG-05-FORTH  
HG-06-EKT  
[...]

- **Grid Unique Identifier (GUID)**
  - Identifies a file uniquely
  - *Example:* guid:ab993b98-8bc9-4984-901e-91290276090c
  
- **Logical File Name (LFN) (User Alias)**
  - Refers to a file instead of a GUID
  - lfn:<any\_string>
  - LFC catalogue: lfn:/grid/<MyVO>/<MyDirs>/<MyFile>
  - *Example:* lfn:/grid/hgdemo/test\_egee01/test\_file
  
- **Storage URL (SURL) (Physical File Name-PFN)**
  - Identifies a replica in a SE
  - <sfn|srm>://<SE\_hostname>/<some\_string>
  - *Example:* sfn://se01.isabella.grnet.gr/storage/hgdemo/generated/2007-04-20/filec4087974-dbaa-4890-91e2-3c105fa0a3df
  
- **Transport URL (TURL)**
  - A valid URI with the necessary information to access a file in a SE
  - <protocol>://<some\_string>
  - *Example:* gsiftp://se01.isabella.grnet.gr/storage/hgdemo/generated/2007-04-20/file1a08d327-d7dc-4d89-bb01-2c86f59eae37



- **File Catalogue (gLite) or LCG File Catalogue**
  - Maintains mappings between LFNs, GUID, SURLS
  - Local File Catalogue, holding only replicas stored at a given site
  - Global File Catalogue, containing information about all files in the Grid
  - Consists of a unique catalogue, where the LFN is the main key
  - System metadata are supported
- **Grid file**
  - Physically present in a SE
  - Registered in the file catalogue
- **High Level Tools (lcg\_util) ⇔ Consistency between files in the SEs and entries in the file catalogue**
- **Low level Data Management tools ⇔ Inconsistency between SEs physical files and catalogue entries**

- **lfc-ls**: List file / directory entries in a directory
- **lfc-mkdir**: Create directory
- **lfc-chmod**: Change access mode of a LFC file / directory
- **lfc-chown**: Change owner and group of a LFC file / directory
- **lfc-ln**: Make a symbolic link to a file / directory
- **lfc-rename**: Rename a file / directory
- **lfc-rm**: Remove a file / directory
- **lfc-setcomment**: Add / replace a comment
- **lfc-delcomment**: Delete the comment associated with a file / directory

# More advanced LFC commands

- **lfc-getacl**: Get file / directory access control lists
- **lfc-setacl**: Set file / directory access control lists
- **lfc-entergrpmap**: Defines a new group entry in the Virtual ID table
- **lfc-enterusrmap**: Defines a new user entry in Virtual ID table
- **lfc-modifygrpmap**: Modifies a group entry corresponding to a given virtual gid
- **lfc-modifyusrmap**: Modifies a user entry corresponding to a given virtual uid
- **lfc-rmgrpmap**: Suppresses group entry corresponding to a given virtual gid or group name
- **lfc-rmusrmap**: Suppresses user entry corresponding to a given virtual uid or user name.

- ***lcg\_util* tools**
  - Allow users to copy files between UI, CE, WN and a SE
  - Allow users to register entries in the file catalogue and replicate files between SEs
- **Replica Management**
- **lcg-cp**: Copies a Grid file to a local destination (*download*)
- **lcg-cr**: Copies a file to a SE and registers the file in the catalogue (*upload*)
- **lcg-del**: Deletes one file (either one replica or all replicas)
- **lcg-rep**: Copies a file from one SE to another SE and registers it in the catalogue (replicate)
- **lcg-gt**: Gets the TURL for a given SURL and transfer protocol

- **File Catalogue Interaction**

**lcg-aa:** Adds an alias in the catalogue for a given GUID

**lcg-ra:** Removes an alias in the catalogue for a given GUID

**lcg-rf:** Registers in the catalogue a file residing on an SE

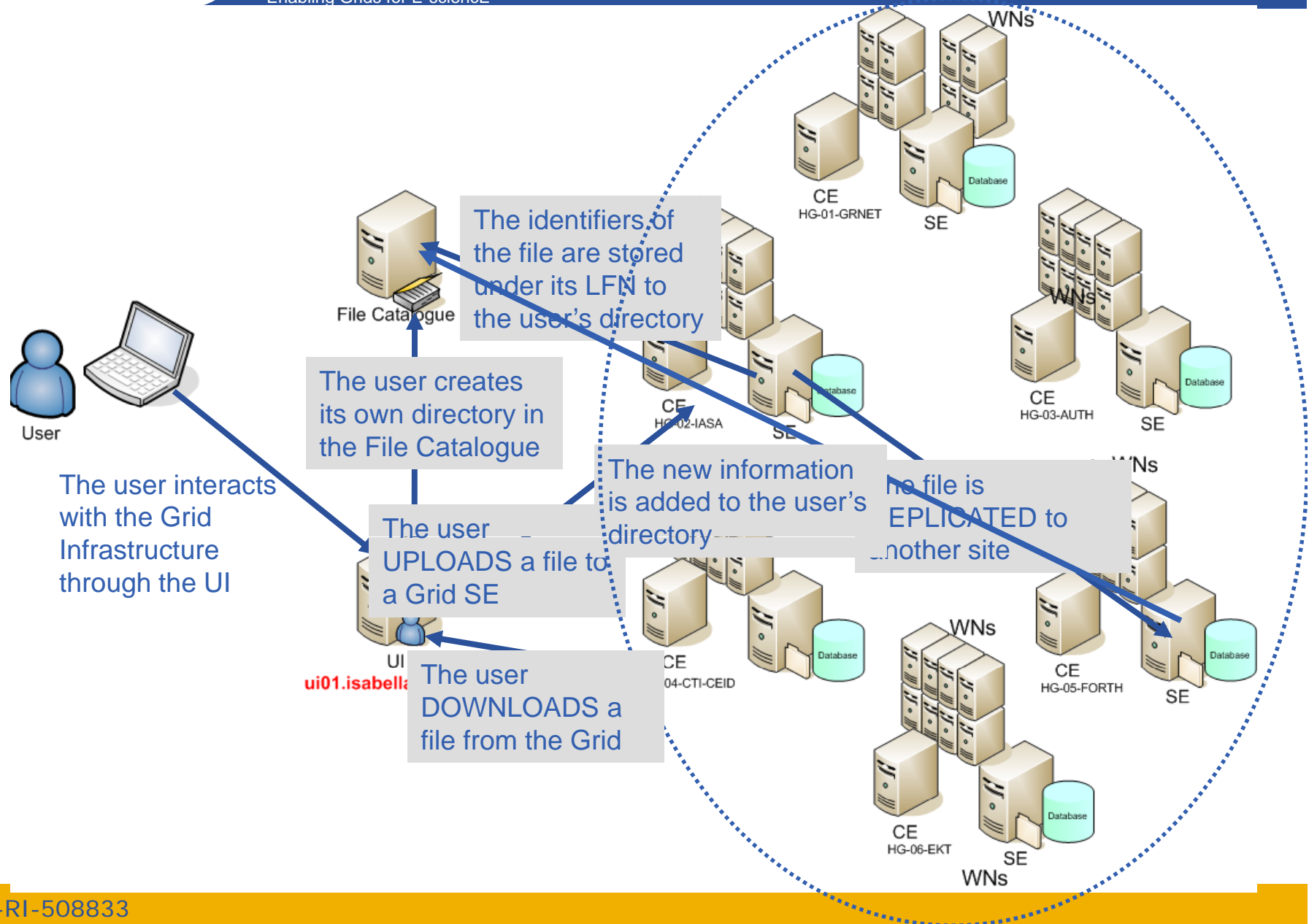
**lcg-uf:** Unregisters in the the catalogue a file residing on an SE

**lcg-la:** Lists the aliases for a given LFN, GUID or SURL

**lcg-lg:** Gets the GUID for a given LFN or SURL

**lcg-lr:** Lists the replicas for a given LFN, GUID or SURL





- The **LFC\_HOST** variable must contain the hostname of the machine providing the LFC service
  - ✓ [egee01@ui01 egee01]\$ **export LFC\_HOST=`lcg-infosites --vo hgdemo lfc`**
  - ✓ [egee01@ui01 egee01]\$ **export LCG\_CATALOG\_TYPE=lfc**
- The **LCG\_GFAL\_VO** variable must contain the name of the user's VO
  - ✓ [egee01@ui01 egee01]\$ **export LCG\_GFAL\_VO=hgdemo**

- **Creating a directory in the LFN namespace**

✓ [egee01@ui01 egee01]\$ **lfc-mkdir /grid/hgdemo/test\_(\$USER)**

where \$USER -> the username of each participant

- **Listing the entries of a LFC directory**

✓ [egee01@ui01 egee01]\$ **lfc-ls -l /grid/hgdemo/**  
 drwxrwxr-x 0 26158 32000 0 Apr 20 12:45 test\_egee01

- **Defining LFC\_HOME variable to point to the created directory**

 ✓ [egee01@ui01 egee01]\$ **export LFC\_HOME=/grid/hgdemo/test\_(\$USER)**

- **Removing LFNs from the LFC**

✓ [egee01@ui01 egee01]\$ **lfc-rm -r lfc:/grid/hgdemo/test\_(\$USER)**

- **Uploading a file to the grid with specific LFN and in a specific storage element**

✓ [egee01@ui01 egee01]\$ **lcg-cr file:\$PWD/sating.ppm -l lfn:sating guid:e20d1fa9-e35f-426d-aa72-cb99b18c2791**

✓ [egee01@ui01 egee01]\$ **lfc-ls /grid/hgdemo/test\_\${USER}/ image1**

- **Replicating a file**

✓ [egee01@ui01 egee01]\$ **lcg-rep -v lfn:sating -d se01.afroditi.hellasgrid.gr**

Using grid catalog type: lfc

Using grid catalog : lfc.isabella.grnet.gr

Source URL: lfn:/grid/hgdemo/egee01\_test/image1

File size: 33985

VO name: hgdemo

Destination specified: se01.kallisto.hellasgrid.gr

Source URL for copy: gsiftp://se01.isabella.grnet.gr/storage/hgdemo/generated/2007-09-18/filee8e4acdf-dd25-49e4-8ced-3d46aba13000

Destination URL for copy:

gsiftp://se01.kallisto.hellasgrid.gr/se01.kallisto.hellasgrid.gr:/data02/hgdemo/2007-09-18/file5625a538-ecf7-4e87-b22b-8ef861e4b30d.172458.0

- **Copying a file out of the Grid**

✓ [egee01@ui01 egee01]\$ **lcg-cp -t 100 lfn:sating file:\$PWD/copy\_image1**

- **Listing replicas**

- ✓ [egee01@ui01 egee01]\$ **lcg-lr --vo hgdemo lfn:satimg**  
 sfn://se01.isabella.grnet.gr/storage/hgdemo/generated/2007-09-18/filee8e4acdf-dd25-49e4-8ced-3d46aba13000  
 srm://se01.kallisto.hellasgrid.gr/dpm/kallisto.hellasgrid.gr/home/hgdemo/generated/2007-09-18/file5625a538-ecf7-4e87-b22b-8ef861e4b30d

- **Listing guides given the lfn**

- ✓ [egee01@ui01 egee01]\$ **lcg-lg lfn:satimg**  
 guid:e20d1fa9-e35f-426d-aa72-cb99b18c2791

- **Adding metadata information to LFC entries**

- ✓ [egee01@ui01 egee01]\$ **lfc-setcomment /grid/hgdemo/test\_(\$USER) /satimg "Created for the training"**

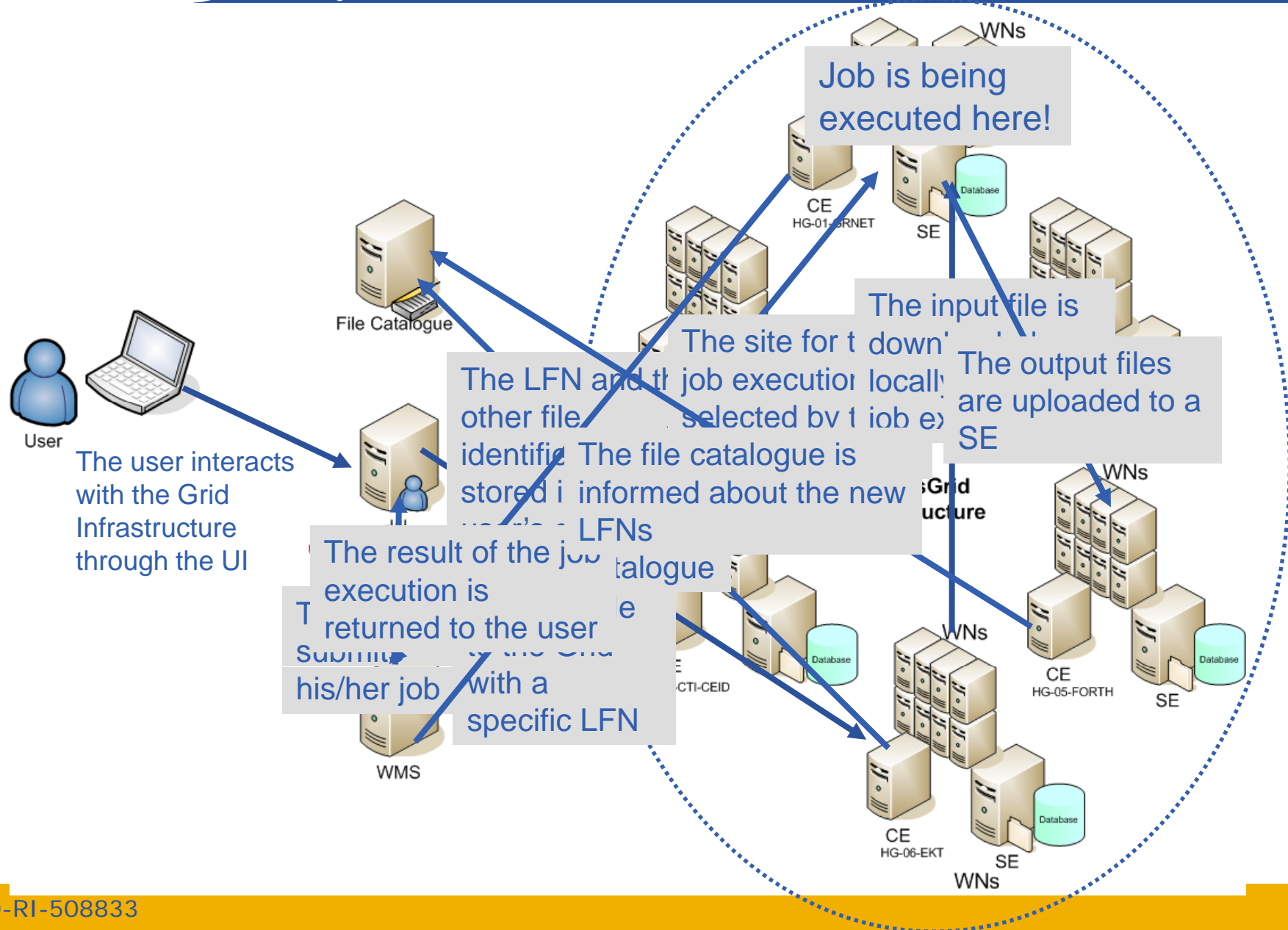
- **View metadata of a specific file**

- ✓ [egee01@ui01 egee01]\$ **lfc-ls --comment /grid/hgdemo/test\_(\$USER)/satimg**  
 image1 Created for the training

- **Removing metadata information to LFC entries**

- ✓ [egee01@ui01 egee01]\$ **lfc-delcomment /grid/hgdemo/test\_ \$USER/satimg**

# Job interacting with files



✓ [egee01@ui01 egee01]\$ **less testJob2.sh**

- **#!/bin/sh**

**echo Information about the Worker Node**

**echo Running at `hostname`**

**echo Date `date`**

**echo Enabling the executable flag for the application**

**chmod 755 \$PWD/compressjpeg**

**echo Fetching data from a Storage Element**

**lcg-cp --vo hgdemo \$1 file:\$PWD/local\_copy**

**echo Running executable of the user taking as input the transferred file**

**\$PWD/compressjpeg local\_copy > local\_compressed\_copy**

**echo Convert to jpeg \_ Unix command**

**pnmt/jpeg local\_copy > local\_copy.jpeg**

**ls -al**

**echo Registering output to the Storage Elements**

**lcg-cr --vo hgdemo -l \$2 file:\$PWD/local\_compressed\_copy**

**lcg-cr --vo hgdemo -l \$3 file:\$PWD/local\_copy.jpeg**

**echo Job is done!**

```

✓ [egee01@ui01 egee01]$ less testJob2.jdl
[
  Type = "job";
  JobType = "normal";
  RetryCount = 0;
  ShallowRetryCount = 3;
  Executable = "testJob2.sh";
  Arguments = "lfn:/grid/hgdemo/test_(username)/sating lfn:/grid/hgdemo/te
st_(username)/satingjpg lfn:/grid/hgdemo/test_(username)/greysating";
  InputSandbox = {"file:///home/training/(username)/testJob2.sh", "file:/
/home/training/(username)/compressjpeg"};
  StdOutput = "std.out";
  StdError = "std.err";
  OutputSandbox = {"std.out", "std.err", "local_copy_jpeg"};
  DataRequirements = {
    [InputData = {"lfn:/grid/hgdemo/test_(username)/sating"}];

    DataCatalogType = "DLI"];
  };
  DataAccessProtocol = {"gsiftp", "https"};
]
    
```



- **Uploading a file to the grid with specific LFN and in a specific storage element**
- ✓ [egee01@ui01 egee01]\$ **lcg-cr file:\$PWD/satimg.ppm -l lfn:satimg guid:594cf6b5-e3b7-49b5-a916-0d5e3054af17**
- ✓ [egee01@ui01 egee01]\$ **lfc-ls /grid/hgdemo/test\_(\$USER)/**  
image1  
satimg

- **Submit job**

```
glite-wms-job-submit -o jobld2 -a testJob2.jdl
```

- **Watch the job status**

```
watch "glite-wms-job-status -i jobld2"
```

- **Retrieve the job output**

```
glite-wms-job-output -i jobld2
```

- **Listing the entries of a LFC directory**

```
[egee01@ui01 egee01]$ lfc-ls -l /grid/hgdemo/test_($USER)
```

```
-rw-rw-r-- 1 26259 32000      1438745 Nov 04 22:36 greysatimg  
-rw-rw-r-- 1 26259 32000      33985 Nov 04 21:08 image1  
-rw-rw-r-- 1 26259 32000     14110553 Nov 04 22:32 satimg  
-rw-rw-r-- 1 26259 32000     1521142 Nov 04 22:36 satimgjpg
```

- Listing the entries of a LFC directory

```
[egee01@ui01 egee01]$ lcg-lr --vo hgdemo lfn:satimgjpg
```

```
[egee01@ui01 egee01]$ lcg-lr --vo hgdemo lfn:greysatimg
```

- Download output files

```
[egee01@ui01 egee01]$ lcg-cp -t 100 lfn:satimgjpg file:$PWD/local_satimgjpg
```

```
[egee01@ui01 egee01]$ lcg-cp -t 100 lfn:greysatimg file:$PWD/local_greysatimg
```

```
[
  Type = "job";
  JobType = "Parametric";
  Parameters = N;
  ParameterStart = 1;
  ParameterStep = 1;
  RetryCount = 0;
  ShallowRetryCount = 3;
  Executable = "testJob2.sh";
  Arguments = "Ifn:/grid/hgdemo/test_(username)/sating_PARAM_ Ifn:/grid/hg
demo/test_(username)/satingjpg_PARAM_ Ifn:/grid/hgdemo/test_(username)/greysatim
g_PARAM_";
  InputSandbox = {"file:///home/training/(username)/testJob2.sh", "file:/
/home/training/(username)/compressjpeg"};
  StdOutput = "std.out";
  StdError = "std.err";
  OutputSandbox = {"std.out", "std.err", "local_copy_jpeg"};
  DataRequirements = {
    [InputData = {"Ifn:/grid/hgdemo/test_(username)/sating_PARAM_",
    "Ifn:/grid/hgdemo/test_(username)/compressjpeg"};
    DataCatalogType = "DLI";}
  };
  DataAccessProtocol = {"gsiftp", "https"};
]
```

- **Uploading file to the grid with specific LFN and in a specific storage element**
- ✓ `[egee01@ui01 egee01]$ lcg-cr file:$PWD/satimg1.ppm -l lfn:satimg1 -d se01.isabella.grnet.gr`  
`guid:594cf6b5-e3b7-49b5-a916-0d5e3054af17`  
 ...  
 ➔ **All three subimages must be uploaded**
- ✓ `[egee01@ui01 egee01]$ lfc-ls /grid/hgdemo/test_ ($USER) /`  
`image1`  
`satimg`  
`satimg1`  
`satimg2`  
`satimg3`

- **Submit job**

```
glite-wms-job-submit -a -o parametric-ids.txt parametrictestJob.jdl
```

- **Watch the job status**

```
watch "glite-wms-job-status -i parametric-ids.txt"
```

- **Retrieve the job output**

```
glite-wms-job-output -i parametric-ids.txt
```

- **Deleting all replicas with the specific lfn**
  - ✓ [egee01@ui01 egee01]\$ **lcg-del -a lfn:image1**
  - ✓ [egee01@ui01 egee01]\$ **lcg-lr --vo hgdemo lfn:image1**  
lcg\_lr: No such file or directory
  
- **Delete LFC directory**
  - ✓ [egee01@ui01 egee01]\$ **lfc-ls /grid/hgdemo/test\_(\$USER)**
  - ✓ [egee01@ui01 egee01]\$ **lfc-rm -r /grid/hgdemo/test\_(\$USER)**
  - ✓ [egee01@ui01 egee01]\$ **lfc-ls /grid/hgdemo/**



**Thank you !**



- **Commandline Tutorial**

- [http://wiki.egee-see.org/index.php/Programming\\_from\\_the\\_Command\\_Line](http://wiki.egee-see.org/index.php/Programming_from_the_Command_Line)