



Enabling Grids for E-scienceE

Data Management System Practicals

Salvatore Scifo

INFN Catania

BioinfoGrid

Bari, March 8th - 10th 2006

www.eu-egee.org



Information Society





Practicals on Data Management

LFC and lcg-utils

LCG File Catalog Management

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

Replica Management

lcg-cp	Copies a grid file to a local destination
lcg-cr	Copies a file to a SE and registers the file in the catalog
lcg-del	Deletes one file
lcg-rep	Replication between SEs and registration of the replica to the catalog
lcg-gt	Gets the TURL for a given SURL and transfer protocol
lcg-sd	Sets file status to “Done” for a given SURL in a SRM request

File Catalog Interaction

lcg-aa	Adds an alias in LFC for a given GUID
lcg-ra	Removes an alias in LFC for a given GUID
lcg-rf	Registers in LFC a file placed in a SE
lcg-uf	Unregisters in LFC a file placed in a SE
lcg-la	Lists the alias for a given SURL, GUID or LFN
lcg-lg	Gets the GUID for a given LFN or SURL
lcg-lr	Lists the replicas for a given GUID, SURL or LFN

- Check the content of the environment variable **\$LFC_HOST**:

```
[localhost]/home/gildauser/globus > echo $LFC_HOST
lfc.gilda.ct.infn.it
```

- Ensure that you have your certificates in the **.globus** directory inside your home:

```
[localhost]/home/gildauser/.globus > ll
total 8
-rw-r--r--  1 gildauser gildauser  1644 Jan  4 13:15 usercert.pem
-r-----  1 gildauser gildauser  1925 Jan  4 15:22 userkey.pem
```

- If you don't have a proxy or it has expired, create it with the following command:

```
$ voms-proxy-init --voms gilda
```

- Listing the entries of a LFC directory

```
lfc-ls [-cdiLlRTu] [--comment] path
```

where *path* specifies the LFC pathname (mandatory)

- Remember that LFC has a directory tree structure
- */grid/<VO_name>/<My_DIR>*
- All members of a given VO have read-write permissions under their directory
- *-l* (it is a lowercase “L”) outputs long listing
- *-R* lists the contents of directories recursively (it has a non trivial cost for directory with a lot of files)
- You can set *LFC_HOME* to use relative paths:
 - setting *LFC_HOME=/grid/gilda/myDir*
 - */grid/gilda/myDir/myFile* becomes *myFile*

```
$ lfc-ls -l /grid/gilda
```

```
...
-rw-rw-r-- 1 4401 4400 0 Jun 21 2005 user.example
lrwxrwxrwx 1 4401 4400 0 Aug 26 03:32 user.example-link -> user.example
lrwxrwxrwx 1 4467 4400 0 Feb 10 03:25 user.example.link -> user.example
lrwxrwxrwx 1 4407 4400 0 Aug 26 03:31 user.example.tokyo33 /grid/gilda/user.example
-rw-rw-r-- 1 4401 4400 0 Jun 21 2005 user.example2
drwxr-xr-x 2 4466 4400 0 Feb 10 06:07 user04
drwxrwxrwx 7 4473 4400 0 Feb 10 05:59 user23
-rw-rw-r-- 1 4401 4400 115 Jun 21 2005 valencia15.ejemplo
...
```

```
$ export LFC_HOME=/grid/gilda/
```

```
$ lfc-ls -l user.example
```

```
-rw-rw-r-- 1 4401 4400 0 Jun 21 2005 /grid/gilda/user.example
```

- **Creating directories in the LFC**

```
lfc-mkdir [-m mode] [-p] path
```

where *path* specifies the LFC pathname (*lfn:/grid/gilda/bari/tutorial.txt*)

-m allow permissions setting (*not symbolic mode is allowed*)

- Remember that you must create the directory before registering any file in the catalog.

- Examples:

```
$ lfc-mkdir /grid/gilda/bari/<mydir>
```

- You can just check the directory with:

```
$ lfc-ls -l /grid/gilda/bari
```

- **Adding/deleting metadata information**
 - `lfc-setcomment <path> <comment>`
 - `lfc-delcomment <path>`
- `lfc-setcomment` adds a *comment* associated with a file/directory in the LFC Catalog.
 - **If the file/directory already has a comment associated it will be replaced by the new one.**
- `lfc-delcomment` deletes a comment previously added.
 - **If there is no comment associated with the file, this command returns the error “No such file or directory”**

- **Example:**

```
lfc-setcomment /grid/gilda/user.example "Hello World"
```

- **Check it with:**

```
lfc-ls --comment /grid/gilda/user.example
```

```
-rw-rw-r-- 1 4401 4400 0 Jun 21 2005 /grid/gilda/user.example Hello World
```

- **Example:**

```
lfc-delcomment /grid/gilda/user.example
```

- **Check your job with:**

```
lfc-ls -l --comment /grid/gilda/user.example
```

```
-rw-rw-r-- 1 4401 4400 0 Jun 21 2005 /grid/gilda/user.example
```

- **Creating a symbolic link**

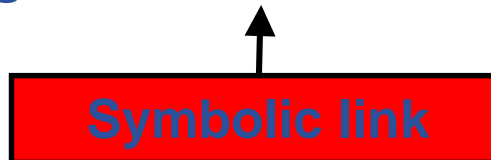
```
lfc-ln -s file linkname
```

```
lfc-ln -s directory linkname
```

Creates a link to the specified *file* or *directory* with *linkname*

```
$ lfc-ln -s /grid/gilda/user.example
  /grid/gilda/bari/linkToUser.example
```

Symbolic link



Original file



Let's check the link using `lfc-ls` with long listing (`-l`):

```
$ lfc-ls -l /grid/gilda/bari
```

```
lrwxrwxrwx  1 4432  4400          0 Feb 16 16:48 linkToUser.ex ->
/grid/gilda/user.example
```

Exercise No.1:

- Log onto an UI and initialize your proxy credentials if not already done
- Set up properly the environment variables to use `lfc-gilda.ct.infn.it` catalog
- Have a look inside the catalog
- Create a directory with your surname, under the directory `/grid/gilda/bari/<your_dir>`
- Set a symbolic link to the directory that you created
- Add a comment to that directory and verify it



- The LCG Data Management tools (usually called **lcg-utils**) allow users to:
 - copy files between **UI**, **CE**, **WN** and **SE**;
 - register entries in the File Catalog;
 - replicate files between **SEs**.
- Check if **LCG_GFAL_INFOSYS** environment variable is correctly set to the local GILDA Information Index (BDII):

```
export LCG_GFAL_INFOSYS=grid004.ct.infn.it:2170
```

- **Adding an alias for a given GUID**

```
lcg-aa --vo vo guid lfn
```

- **vo** specifies the Virtual Organization the user belongs to
- **guid** specifies the Grid Unique Identifier of the file you want to add the alias
- **lfn** specifies the new alias

- Example:

```
$ lcg-aa --vo gilda guid:bf95f82e-de21-4452-a4b5-b9d40a94ee2c lfn:/grid/gilda/bari/alias.txt
```

- **Checking if alias exists for a given file**

```
lcg-la --vo vo guid | lfn | surl
```

- Example:

```
$ lcg-la --vo gilda lfn:/grid/gilda/bari/alias.txt
lfn:/grid/gilda/bari/test.txt
lfn:/grid/gilda/bari/alias.txt
```

- Upload a file to a SE and register it into the catalog

```
lcg-cr -d dest_file | dest_host [-g guid] [-l lfn]
[-v | --verbose] --vo vo src_file
```

- **dest_host** is the fully qualified hostname of the destination SE
 - **dest_file** is a valid SURL (both *sfn://* or *srm://* format are valid)
 - **guid** specifies the Grid Unique Identifier. (*If this option is not present, a GUID is generated internally*)
 - **lfn** specifies the Logical File Name associated with the file
 - **vo** specifies the Virtual Organization the user belongs to
 - **src_file** specifies the source file name (both *file:///* or *gsiftp:///* are valid)
- Remember that if you register a new file, the corresponding destination directory must be already created in the catalog.

- **Example:**

```

$ lcg-cr -v -d grid009.ct.infn.it -l lfn:/grid/gilda/bari/test.txt
  --vo gilda file:/home/bari/prova.txt
Using grid catalog type: lfc
Source URL: file:/home/bari/prova.txt
File size: 353
Destination specified: grid009.ct.infn.it
Destination URL for copy:
  gsiftp://grid009.ct.infn.it/flatfiles/SE00/gilda/generated/2006-0...
# streams: 1
Alias registered in Catalog: lfn:/grid/gilda/bari/test.txt
Transfer took 1200 ms
Destination URL registered in Catalog:
  sfn://grid009.ct.infn.it/flatfiles/SE00/gilda/generated/2006-0...
guid:bf95f82e-de21-4452-a4b5-b9d40a94ee2c

```

To discover which SEs the user is allowed to use, remember you can use `lcg-infosites` command:

```
lcg-infosites --vo gilda se
```

The output is a list of SEs and related information on available/used space.

Exercise No.2:

- verify that your **LCG_GFAL_INFOSYS** environment variable is correctly set up
- create a dummy file in your home directory
- list the available storage elements
- copy and register the dummy file into the directory you created
- add an alias to the uploaded file
- check if the alias was created correctly



- **Replica a given file and register it in the Catalog**

```
lcg-rep -d dest_file | dest_host [-v | --verbose]
--vo vo src_file
```

- **dest_host** is the fully qualified hostname of the destination SE
- **dest_file** is a valid SURL (both *sfn://* or *srm://* are valid)
- **vo** specifies the Virtual Organization the user belongs to
- **src_file** it can be an LFN, GUID or SURL. (**If SURL, sfn:// or srm:// is allowed**).

```
$ lcg-rep -v -d grid005.iucc.ac.il --vo gilda \
lfn:/grid/gilda/bari/test.txt
Using grid catalog type: lfc
Source URL: lfn:/grid/gilda/bari/test.txt
File size: 186
Destination specified: grid005.iucc.ac.il
Source URL for copy: gsiftp://gilda-se-01.pd.infn.it/shared/...
Destination URL for copy: gsiftp://grid005.iucc.ac.il/storage/.../filebe...
# streams: 1
Transfer took 10980 ms
Destination URL registered in LRC:
sfn://grid005.iucc.ac.il/storage/gilda/generated/2006-02-20/filebe...
```

- Listing replicas for a given LFN, GUID or SURL

```
lcg-lr --vo vo file
```

where:

- **vo** specifies the Virtual Organization the user belongs to
- **file** it can be an LFN, GUID or SURL. (*If SURL, sfn:// or srm:// is allowed*).

- Example:

```
$ lcg-lr --vo gilda lfn:/grid/gilda/bari/note.txt
sfn://grid-se.bio.dist.unige.it/flatfiles/SE00/gilda/generated...
sfn://grid009.ct.infn.it/flatfiles/SE00/gilda/generated/2005-0...
```

we get the same output using the GUID of the file:

```
$ lcg-lr --vo gilda guid:4c10a8e3-2244-4c38-...
```

- **Deleting replicas**

```
lcg-del [ -a ] [ -s se ] [ -v | --verbose ] --vo vo file
```

- **a** is used to delete all replicas of the given file
- **se** specifies the SE from which you want to remove the replica
- **vo** specifies the Virtual Organization the user belongs to
- **file** it can be an LFN, GUID or SURL. (*If SURL, sfn:// or srm:// is allowed*)

- **Example:**

- delete one replica

```
$ lcg-del --vo gilda -s grid009.ct.infn.it
lfn:/grid/gilda/bari/alias.txt
```

- delete all the replicas

```
$ lcg-del -a --vo gilda lfn:/grid/gilda/bari/test.txt
```

- let's check if the previous command was successful

```
$ lcg-lr --vo gilda lfn:/grid/gilda/bari/test.txt
lcg_lr: No such file or directory
```

- or by `lfc-ls /grid/gilda/bari` (you will not see anymore test.txt and its alias)

- **Downloading a Grid file from a SE to a local destination**

```
lcg-cp [ -v | --verbose ] --vo vo src_file dest_file
```

- **vo** specifies the Virtual Organization the user belongs to
- **src_file** it can be an LFN, GUID or SURL. (*If SURL, sfn:// or srm:// is allowed*)
- **dest_file** specifies the destination. The protocol can be *file:///* or *gsiftp:///*

- **Example:**

```
$ lcg-cp -v --vo gilda lfn:/grid/gilda/bari/test.txt
file:/home/bari/test.txt
```

```
Source URL: lfn:/grid/gilda/bari/test.txt
```

```
File size: 353
```

```
Source URL for copy:
```

```
gsiftp://grid009.ct.infn.it/flatfiles/SE00/gilda/generated/2006...
```

```
Destination URL: file:/home/bari/test.txt
```

```
# streams: 1
```

```
Transfer took 1360 ms
```

Exercise No.3:

- Create two replicas of the file you previously uploaded (you could also use the alias to point it out)
- Check if the operation was successful
- Download the file back to your UI
- Delete only one replica and verify that
- Delete all the replicas and verify that
- Verify if the entry is still present in the catalog



- **GOAL:**

Submit a job that does data management: it will retrieve a file previously registered into the catalog.

- **Steps to follow up:**

- Create a new file in your UI and put some data into it
- Choose a SE to upload the file to (hint: use **lcg-infosites**) and use the appropriate command to accomplish at this operation

```
(lcg-cr -v --vo gilda -l
lfn:/grid/gilda/bari/<choose an lfn> -d <an SE
host> file:`pwd`/<your new file>)
```

- create a script.sh file with the following content:

```
#!/bin/sh
/bin/hostname
#Change the LFN_NAME to download from the Catalog.
echo "Start to download.."
lcg-cp --vo gilda lfn:/grid/gilda/bari/<lfn you choose> file:`pwd`/output.dat
echo "Done.."
```

- **Create the JobWithData.jdl:**

```
Type = "job";
JobType = "Normal";

Executable = "/bin/sh";
Arguments = "script.sh";

VirtualOrganisation = "gilda";

StdOutput = "std.out";
StdError = "std.err";

InputSandbox = {"script.sh"};
OutputSandbox = {"std.out", "std.err", "output.dat"};
```

- **Submit it to the grid**
- **Retrieve the output and verify the content of output.dat**



Practicals on Catalog gLite - FiReMan

Summary of the Fireman Catalog commands

glite-catalog-ls	Lists file/directory entries in a directory
glite-catalog-mkdir	Creates a directory
glite-catalog-mv	Renames a file/directory
glite-catalog-rm glite-catalog-rmdir	Removes a file/directory
glite-catalog-getreplica	Gets all replicas associated with a file/GUID
glite-catalog-touch glite-catalog-create	Creates a new entry in the catalog/update the modification time
glite-catalog-find	Finds entries based on their name pattern
glite-seindex-list	Lists all SEs having a replica of the given file

Summary of the Fireman Catalog commands

<code>glite-catalog-chmod</code>	Change access mode of the Fireman file/directory.
<code>glite-catalog-stat</code> <code>glite-catalog-getguid</code>	List the details of a file – all attributes, replicas. Return the associated GUID.
<code>glite-catalog-setattr</code> <code>glite-catalog-getattr</code> <code>glite-catalog-setschema</code>	Set/Get metadata attribute Set the metadata schema of a given directory
<code>glite-catalog-setacl</code> <code>glite-catalog-getacl</code> <code>glite-catalog-getdefacl</code> <code>glite-catalog-setdefacl</code> <code>glite-catalog-setdefperm</code>	Set/Get file/directory access control lists Set/Get the default ACL Set the default permission
<code>glite-catalog-symlink</code>	Makes a symbolic link to a file. Directory symlinks are not supported by design.

- **Browsing the contents of a directory**

```
glite-catalog-ls [-h][-q][-s SERVICE][-V][-v][-c] [-d][-g][-l][-R][-S][-t][-u][-x] LFN
```

where: **LFN** (Logical File Name) is the absolute path of the file/directory to list.

- **Main Options:**

- l request long output
- v increase the verbosity level
- R request recursive listing
- s sort by size
- t sort by modification time
- g print GUIDs in the long listing
- h print a short help
- s **SERVICE** specifies the service endpoint to use
- d list the directory node itself instead of its contents

- Example:

```
$glite-catalog-ls -l /satimages
```

```
-pdrwl-gspdrwl-gspdrwl-gs 9950k 2006-01-21 14:31:14 /satimages/20060719.jpg  
-pdrwl-gspdrwl-gspdrwl-gs 3462k 2006-01-23 11:37:48 /satimages/20060720.jpg  
-pdrwl-gspdrwl-gspdrwl-gs 5154k 2006-01-23 11:41:25 /satimages/20060721.jpg  
-pdrwl-gspdrwl-gspdrwl-gs 5692k 2006-01-23 11:45:01 /satimages/20060722.jpg  
-pdrwl-gspdrwl-gspdrwl-gs 5774k 2006-01-23 10:28:04 /satimages/20060723.jpg  
-pdrwl-gspdrwl-gspdrwl-gs 5676k 2006-01-23 11:48:14 /satimages/20060724.jpg  
-pdrwl-gspdrwl-gspdrwl-gs 6094k 2006-01-23 11:32:31 /satimages/20060725.jpg
```

```
-pdrwl-gspdrwl-gspdrwl-gs 9950k 2006-01-21 14:31:14 /satimages/20060719.jpg
```

- The first letter shows the type of the entry:
 - '-' for **regular files**
 - 'd' for **directories**
 - 'l' for **symbolic links**
 - 'v' for **virtual directories**.
- it follows **3 series of 8 flags**, respectively:
 - for the **owner**
 - the **group**
 - and **others**
 - **p** indicates the **permission to change attribute**
 - **d** gives **rights to delete the entry**.
 - It follows **permissions to read, write, list, execute**.
 - The last two flags are reserved for **metadata use**; They will give the rights to **get or set** the metadata but are currently unused.

- **Directory creation**

```
glite-catalog-mkdir [-h][-q][-s URL][-V][-v][-p]  
[-c] DIR
```

Where **DIR** is the name of the directory to create (in LFN format)

- p create missing parent directory

- c inherit the permission of the parent directory

Example:

```
glite-catalog-mkdir /bari
```

- **Deleting a directory**

```
glite-catalog-rmdir [-h][-q][-s URL][-V][-v] DIR
```

The specified directory **must be empty** or the command will fail

Example:

```
glite-catalog-rmdir /bari
```

- Getting file information

```
glite-catalog-stat [-h][-q][-s SERVICE][-V][-v][-r] LFN
```

Lists all information about a file or dir in the catalog. It includes LFN, GUID, owner/group with basic permission, list of ACLs (if any), SURL for every replica.

```
$ glite-catalog-stat /bari/new
```

```
LFN:                /bari/new
  Created:           2006-02-17 10:20:34.000
  Modified:          2006-02-17 10:20:34.000
  Size:              0
  Type:              File
  Expires:           Never
GUID:                3dd02562-5131-49ff-95ff-361e501f27bf
  Created:           2006-02-17 10:20:34.000
  Modified:          2006-02-17 10:20:34.000
  Size:              0
  Checksum:          <none>
  Status:            0
User:                /C=IT/O=GILDA/OU=Personal Certificate/L=INFN-Catania/CN=<User Name>...
Group:               egee-group
User rights:         pdrwlxgs
Group rights:        p---l-g-
Other rights:        p---l-g-
```

- **Locating files matching a specified pattern in the specified directory**

```
glite-catalog-find [-h] [-q] [-s SERVICE] [-V] [-v]
[-n LIMIT] PATTERN DIR
```

Where:

- **-n LIMIT** return no more than *LIMIT* results. The default is 100
- **PATTERN** is the file name pattern to look for. Recognized wildcards are **?** matching any single character, and ***** matching any string
- **DIR** Name of the directories where to look for files matching *PATTERN*.

- **Example:**

```
$ glite-catalog-find tes* /
/scifo/test
/pisa/grandi/test
/elmsheus/test
/karlsruhe46/test
```

Exercise No.4:

- Create a directory in the catalog named `/bari/<your surname>`
- Create a file in this directory using:
`glite-catalog-create <filename>`
- Recursively list all the files of the directory `/bari` using the `-R` flag
- Get all the information about the file you created:
`glite-catalog-stat -v /bari/<your dir>/<filename>`
- Find your file in the catalog using the `glite-catalog-find` command:
`glite-catalog-find <file> /bari`



- **Uploading a file to a SE**

- `glite-put <localfilename> <lfn>[-m <mode>][-c <config>]`

- `<localfilename>` is the name of the local file you want to upload
 - `<lfn>` is the logical file name you want to assign to that file
 - `-c <config>` use an alternative config file, overriding the glite-io-client configuration

Example: \$ glite-put JobWithData.jdl /bari/JobWithData.jdl

```
1140452058 INFO glite-io-client-put : Start File Transfer
[glite-put] Total 0.00 MB      |=====| 100.00 % [0.0 Mb/s]
1140452061 INFO glite-io-client-put : File Transfer Completed
```

Transfer Completed:

```
LFN                : /bari/JobWithData.jdl
GUID               : 00867928-ead2-13f9-9e7c-c1ced08dbeef
SURL               : srm://egee016.cnaf.infn.it:8443/srm/managerv1?SFN...
Data Written [bytes] : 337
Eff.Transfer Rate[Mb/s] : 0.000044
```

- **Downloading a file from the local SE**
 - `glite-get <lfn> <localfilename> [-c <config>]`
 - `<lfn>` is the logical file name of the file you want to download and `<localfilename>` is the name of the destination file
 - `-c <config>` use an alternative config file, overriding the glite-io-client configuration

Example: `$ glite-get /bari/JobWithData.jdl test`

```
1140452921 INFO glite_get : Start File Transfer
[glite-get] Total 0.00 MB      |=====| 100.00 % [0.0 Mb/s]
1140452922 INFO glite_get : File Transfer Completed
```

Transfer Completed:

```
LFN                : /bari/JobWithData.jdl
GUID               : 00867928-ead2-13f9-9e7c-c1ced08dbeef
SURL               : srm://egee016.cnaf.infn.it:8443/srm/managerv1?SFN...
Data Written [bytes] : 337
Eff.Transfer Rate[Mb/s] : 0.000051
```

- Removing the given file from the local Storage Element and delete the entry from the File Catalog

– `glite-rm <lfn> [-c <config>]`

Example: `$ glite-rm /bari/JobWithData.jdl`

Unlink Completed:

```
File           : /bari/JobWithData.jdl
Time [s]       : 4.162000
```

Exercise No.5:

- Create a dummy file in your home directory
- Upload the file you created into the directory `/bari/<your surname>` using `glite-put`
- List the files in this directory using `glite-catalog-ls`
- Retrieve the file you stored in the catalog using the `glite-get` command
- Remove all files in your directory from the catalog
- Delete your directory from the FiReMan catalog



- **LFC, gfal, lcg-utils**

- <http://indico.cern.ch/contributionDisplay.py?contribId=278&sessionId=7&confId=0>

- **gLite homepage**

- <http://www.glite.org>

- **FiReMan catalog user guide**

- <https://edms.cern.ch/file/570780/1/EGEE-TECH-570780-v1.0.pdf>

- **gLite-I/O user guide**

- <https://edms.cern.ch/file/570771/1.1/EGEE-TECH-570771-v1.1.pdf>

