



# LHC COMPUTING GRID

## HOW TO TEST A GRID SITE

---

<i>Document identifier:</i>	<b>LCG-GIS-TST-SBS</b>
<i>EDMS id:</i>	<b>n/a</b>
<i>Version:</i>	<b>v3.0.0-1</b>
<i>Date:</i>	<b>May 15, 2006</b>
<i>Section:</i>	<b>GIS</b>
<i>Document status:</i>	<b>Final</b>
<i>Author(s):</i>	Piotr Nyczyk, Patricia Mendez Lorenzo, Antonio Retico, Min Tsai, Alessandro Usai ( <a href="mailto:support-lcg-deployment@cern.ch">&lt;support-lcg-deployment@cern.ch&gt;</a> )
<i>File:</i>	<b>LCG2-Site-Testing</b>

---

*Abstract: These notes will assist you in testing a freshly installed site.*

---



---

## CONTENTS

<b>1. NOTICE</b> .....	<b>3</b>
<b>2. INTRODUCTION</b> .....	<b>4</b>
<b>3. TESTING THE UI (AND INDIVIDUAL WNS)</b> .....	<b>5</b>
3.1. CREATE A PROXY .....	5
3.2. LOCAL TESTING USING SITE FUNCTIONAL TESTS (SFT).....	5
3.3. RUN SIMPLE JOBS .....	6
3.4. DATA MANAGEMENT TOOLS .....	11
<b>4. TESTING THE CE AND WNS</b> .....	<b>14</b>
<b>5. TESTING THE SE</b> .....	<b>22</b>
<b>6. TESTING THE R-GMA</b> .....	<b>23</b>



---

## 1. NOTICE

*Please be aware that this document is currently being updated and the text below refers to earlier versions of the middleware. That said, quite a few of the tests are still useful so the present version of the doc is being retained until a glite 3.0 version is available.*



---

## 2. INTRODUCTION

This is a collection of basic commands that can be run to test the correct setup of a site. These tests are not meant to be a replacement of the test tools provided by LCG certification team. They are instead a collection of quick and non-invasive functional tests suitable to be run in order to be sure that the site configuration has been correctly performed.

The tests in this chapter should enable the site administrator to verify the basic functionality of the site.

There are currently available tools for:

- Testing the UI
- Testing the CE and WNs
- Testing the SE
- Testing R-GMA

Not included in this release:

- Testing the RB
- Testing the BDII
- Testing the Proxy



### 3. TESTING THE UI (AND INDIVIDUAL WNS)

The main tools used on a UI are:

1. Tools to manage certificates and create proxies
2. Local testing using Site Functional Tests (SFT)
3. Tools to deal with the submission and status retrieval of jobs (UI only)
4. Client tools of the data management. These include tools to transport data and to query the replica location service (deprecated by SFT)

Log to a UI and run the following tests (all the commands used in the examples should be in your path).

In some cases individual Worker Nodes (WN) can be tested in this way as well. This is quite important as it allows to detect potential misconfiguration on one of many WNs. But in order to use this procedure on WNs several additional steps must be taken before proceeding:

1. Use regular user account instead of root (eg. “su - dteam001”).
2. Put your personal certificate in the home directory and create a proxy.
3. Don't forget to destroy the proxy and remove your personal certificate after testing.

#### 3.1. CREATE A PROXY

```
> grid-proxy-init
Your identity: /C=CH/O=CERN/OU=GRID/CN=Markus Schulz 1319
Enter GRID pass phrase for this identity:
Creating proxy ..... Done
Your proxy is valid until: Mon Apr 5 20:53:38 2004
```

#### 3.2. LOCAL TESTING USING SITE FUNCTIONAL TESTS (SFT)

Starting from version 2.6.0 LCG release contains “client” part of Site Functional Tests (SFT). The new version of SFT allows to perform local testing of UI and also WN. To test your UI make sure that the valid grid proxy is created and then use the following command:

```
> /opt/lcg/sft/sftests local-test
Looking for good central SE:
Found good SE: lxn1183.cern.ch

Starting tests:
Running test: sft-softver
Running test: sft-caver
Publishing results of test: sft-softver to local directory
End of test: sft-softver, publishing finished: OK
Running test: sft-rgma
```



```
Running test: sft-csh
Running test: sft-lcg-rm
Publishing results of test: sft-csh to local directory
End of test: sft-csh, publishing finished: OK
Publishing results of test: sft-caver to local directory
End of test: sft-caver, publishing finished: OK
Publishing results of test: sft-lcg-rm to local directory
End of test: sft-lcg-rm, publishing finished: OK
```

After more less 30 seconds if your machine has *links* browser installed you should see the report, which should look like this:

(1/23)

Site Functional Tests - local report for node lxb1921.cern.ch

```
lqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqk
x  Test      x Result  x Summary  x
tqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqu
x sft-softver x OK      x LCG-2_6_0 x
tqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqu
x sft-caver  x WARNING x          x
tqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqu
x sft-rgma   x OK      x          x
tqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqu
x sft-csh    x ERROR   x          x
tqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqu
x sft-lcg-rm x OK      x OK       x
mqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqj
```

Overall summary: ERROR

Otherwise you will see the message where to find the report:

Report available in `/tmp/sft-local-results_test.html`

Use any web browser to see the contents of this file.

The report is an HTML document which you can easily navigate to find the details of tests and potential failures.

### 3.3. RUN SIMPLE JOBS

Check that `globus-job-run` works.

Choose a CE that is known to work.

At this purpose you can use the CE at CERN. Its name can be found in the GOC DB.

(`<support-lcg-deployment@cern.ch>`) in our example we use `lxn1181.cern.ch`

```
> globus-job-run lxn1181.cern.ch /bin/pwd
/home/dteam002
```

What can go wrong with this most basic test? If your VO membership is not correct you might be not in the grid-mapfile. In this case you will see some errors that refer to grid security.



Next is to see if the UI is correctly configured to access a RB. Create the following files for these tests:

testJob.jdl this contains a very basic job description.

```
Executable = "testJob.sh";
StdOutput = "testJob.out";
StdError = "testJob.err";
InputSandbox = {"/testJob.sh"};
OutputSandbox = {"testJob.out", "testJob.err"};
#Requirements = other.GlueCEUniqueID == "lxnl181.cern.ch:2119/jobmanager-lcgpbs-short";
```

The "Requirements" tag in the jdl, commented out in the example, means that you want to run the job on a specific CE In order to get a list of computational resources available to your VO you can also query the information system:

```
> lcg-infosites --vo dteam ce

*****
These are the related data for dteam: (in terms of CPUs)
*****

#CPU   Free   Total Jobs   Running Waiting ComputingElement
-----
20     20     1             0         1     ce01.pic.es:2119/jobmanager-torque-dteam
40     40     0             0         0     ceitep.itep.ru:2119/jobmanager-torque-dteam
52     52     0             0         0     ce.prd.hp.com:2119/jobmanager-pbs-dteam
8      8      2             0         2     ce01.lip.pt:2119/jobmanager-torque-dteam
24     22     0             0         0     lcgce.psn.ru:2119/jobmanager-torque-dteam
7      6      2             1         1     ce00.inta.es:2119/jobmanager-torque-dteam
3      3      0             0         0     ce001.imbm.bas.bg:2119/jobmanager-pbs-long
24     24     0             0         0     ingvar.nsc.liu.se:2119/jobmanager-torque-dteam
2      1      1             1         0     lcg03.gsi.de:2119/jobmanager-torque-dteam
332    33     3             3         0     lcg06.gsi.de:2119/jobmanager-lcglsf-dteam

[...]

88     2      74            60        14     bohr0001.tier2.hep.man.ac.uk:2119/jobmanager-lcgpbs-infinite
3      0      0             0         0     eksp-lcg-ce.physik.uni-karlsruhe.de:2119/jobmanager-torque-dtea
4      0      0             0         0     grid-ce.physik.uni-wuppertal.de:2119/jobmanager-pbs-short
10     0      9             9         0     virgo-ce.roma1.infn.it:2119/jobmanager-lcgpbs-infinite
4      0      0             0         0     grid-ce.physik.uni-wuppertal.de:2119/jobmanager-pbs-medium
26     22     1             1         0     testbed001.phys.sinica.edu.tw:2119/jobmanager-torque-dteam
2      2      0             0         0     accip43.physik.rwth-aachen.de:2119/jobmanager-torque-dteam
0      0      0             0         0     bigmac-lcg-ce.physics.utoronto.ca:2119/jobmanager-lcgcondor-dt
```

If not specified the BDII defined in default (through LCG\_GFAL\_INFOSYS) will be queried. Otherwise you can specify the BDII you want to query including the argument "-is" followed by the name of the BDII.

```
> lcg-infosites --vo dteam ce --is <any BDII>
```

In the GOC DB you can identify the BDII for the production and the test zone <sup>1</sup>.

If you specify a verbose level 1, only the names of the queues will be printed:

```
> lcg-infosites --vo dteam -v 1 ce
```

And if you specify a verbose level 2, information dealing with the operational system Ram Memory and the processor of each CE will be printed as follows:

```
*****
These are the related data for dteam: (in terms of CEs)
*****

RAMMemory   Operating System   System Version   Processor   CE Name
-----
524288      Redhat             3                PIV         CE.pakgrid.org.pk
768         SL                 3                PIII        accip43.physik.rwth-aachen.de
2016        Redhat             1SMPFriFeb2010   Intel(R) Xeon(TM) CPU2.80GHz   atlasce.lnf.infn.it
2015        Redhat             1SMPFriFeb2010   Intel(R) Xeon(TM) CPU2.80GHz   atlasce01.na.infn.it
512         Redhat             3                PIII        bfa.tier2.hep.man.ac.uk

[.....]
```

<sup>1</sup>Convince yourself that this is the address of a working BDII that you can reach. For instance, if you decided to use the CERN BDII lcg-bdii.cern.ch you can run

```
ldapsearch -LLL -x -H ldap://lcg-bdii.cern.ch:2170 -b "mds-vo-name=local,o=grid"
```



As long as the `leg-infosites` command is not working it makes no sense to conduct further tests.

### **testJob.sh contains a very basic test script**

```
#!/bin/bash
date
hostname
echo "*****"
echo "env | sort"
echo "*****"
env | sort
echo "*****"
echo "mount"
echo "*****"
mount
echo "*****"
echo "rpm -q -a | sort"
echo "*****"
/bin/rpm -q -a | sort

sleep 20
date
```

run the following command to see which sites can run your job. (If you are not member of the *dteam* vo you can use your one)

```
> edg-job-list-match --vo dteam testJob.jdl
```

the output should look like:

```
Selected Virtual Organisation name (from --vo option): dteam
Connecting to host lxnl177.cern.ch, port 7772

*****
                COMPUTING ELEMENT IDs LIST
The following CE(s) matching your job requirements have been found:

                *CEId*
CE.pakgrid.org.pk:2119/jobmanager-torque-dteam
accip43.physik.rwth-aachen.de:2119/jobmanager-torque-dteam
alexander.it.uom.gr:2119/jobmanager-torque-dteam
bfa.tier2.hep.man.ac.uk:2119/jobmanager-lcgpbs-infinite
bfa.tier2.hep.man.ac.uk:2119/jobmanager-lcgpbs-long
bfa.tier2.hep.man.ac.uk:2119/jobmanager-lcgpbs-short
bigmac-lcg-ce.physics.utoronto.ca:2119/jobmanager-lcgcondor-dteam
boalice5.bo.infn.it:2119/jobmanager-lcgpbs-cert
boalice5.bo.infn.it:2119/jobmanager-lcgpbs-infinite

[...]

t2-ce-01.mi.infn.it:2119/jobmanager-lcgpbs-infinite
t2-ce-01.mi.infn.it:2119/jobmanager-lcgpbs-long
grid012.ct.infn.it:2119/jobmanager-lcgpbs-infinite
goliath25.farm.particle.cz:2119/jobmanager-lcgpbs-long
ce001.m45.ihep.su:2119/jobmanager-pbs-infinite
grid002.ca.infn.it:2119/jobmanager-lcgpbs-short
```





```
grid002.ca.infn.it:2119/jobmanager-lcgpbs-cert
grid002.ca.infn.it:2119/jobmanager-lcgpbs-infinite
grid002.ca.infn.it:2119/jobmanager-lcgpbs-long
cmsboce1.bo.infn.it:2119/jobmanager-lcglsf-cert
cmsboce1.bo.infn.it:2119/jobmanager-lcglsf-short
```

\*\*\*\*\*

If an error is reported rerun the command using the `-debug` option. Common problems are related to the RB that has been configured to be used as the default RB for the node. To test if the UI works with a different RB you can run the command using configuration files that overwrite the default settings. Configure the two files to use for the test a known working RB. The RB at CERN that can be used is: `lxn1177.cern.ch` The file that contains the VO dependent configuration has to contain the following:

```
lxn1177.vo.conf
```

```
[
VirtualOrganisation = "dteam";
NSAddresses = "lxn1177.cern.ch:7772";
LBAddresses = "lxn1177.cern.ch:9000";
## HLR location is optional. Uncomment and fill correctly for
## enabling accounting
#HLRLocation = "fake HLR Location"
## MyProxyServer is optional. Uncomment and fill correctly for
## enabling proxy renewal. This field should be set equal to
## MYPROXY_SERVER environment variable
MyProxyServer = "myproxy.cern.ch"
]
```

and the common one:

```
lxn1177.conf
```

```
[
rank = - other.GlueCEStateEstimatedResponseTime;
requirements = other.GlueCEStateStatus == "Production";
RetryCount = 3;
ErrorStorage = "/tmp";
OutputStorage = "/tmp/jobOutput";
ListenerPort = 44000;
ListenerStorage = "/tmp";
LoggingTimeout = 30;
LoggingSyncTimeout = 30;
LoggingDestination = "lxn1177.cern.ch:9002";
# Default NS logger level is set to 0 (null)
# max value is 6 (very ugly)
NSLoggerLevel = 0;
DefaultLogInfoLevel = 0;
DefaultStatusLevel = 0;
DefaultVo = "dteam";
]
```

Then run the list match with the following options:

```
> edg-job-list-match -c 'pwd'/lxn1177.conf --config-vo 'pwd'/lxn1177.vo.conf testJob.jdl
```



If this works you should have investigate the configuration of the RB that is selected by default from your UI or the associated configuration files.

If the job-list-match is working you can submit the test job using:

```
> edg-job-submit --vo dteam testJob.jdl
```

The command returns some output like:

```
Selected Virtual Organisation name (from --vo option): dteam
Connecting to host lxn1177.cern.ch, port 7772
Logging to host lxn1177.cern.ch, port 9002
```

```
*****
                                JOB SUBMIT OUTCOME
The job has been successfully submitted to the Network Server.
Use edg-job-status command to check job current status. Your job identifier (edg_jobId) is:

- https://lxn1177.cern.ch:9000/0b6EdeF6dJlnHkKByTkc_g
```

In case the output of the command has a significant different structure you should rerun it and add the `-debug` option. Save the output for further analysis.

Now wait some minutes and try to verify the status of the job using the command:

```
edg-job-status https://lxn1177.cern.ch:9000/0b6EdeF6dJlnHkKByTkc_g
```

repeat this until the job is in the status: Done (Success)

If the job doesn't reach this state, or gets stuck for longer periods in the same state you should run a command to access the logging information. Please save the output.

```
edg-job-get-logging-info -v 1 https://lxn1177.cern.ch:9000/0b6EdeF6dJlnHkKByTkc_g
```

Assuming that the job has reached the desired status please try to retrieve the output:

```
edg-job-get-output https://lxn1177.cern.ch:9000/0b6EdeF6dJlnHkKByTkc_g
```

```
Retrieving files from host: lxn1177.cern.ch ( for https://lxn1177.cern.ch:9000/
0b6EdeF6dJlnHkKByTkc_g )
```

```
*****
                                JOB GET OUTPUT OUTCOME

Output sandbox files for the job:
- https://lxn1177.cern.ch:9000/0b6EdeF6dJlnHkKByTkc_g
have been successfully retrieved and stored in the directory:
/tmp/jobOutput/markusw_0b6EdeF6dJlnHkKByTkc_g
```

Check that the given directory contains the output and error files.

One common reason for this command to fail is that the access privileges for the jobOutput directory are not correct, or the directory has not been created.

If you encounter a problem rerun the command using the `-debug` option.



### 3.4. DATA MANAGEMENT TOOLS

Test that you can reach an external SE. Run the following simple command to list a directory at one of the CERN SEs.

```
edg-gridftp-ls gsiftp://castorgrid.cern.ch/castor/cern.ch/grid/dteam
```

You should get a long list of files.

If this command fails it is very likely that your firewall setting is wrong.

In order to see which resources you can see via the information system you should run:  
%endverbatim

```
> lcg-infosites --vo dteam se
```

```
*****
```

```
These are the related data for dteam: (in terms of SE)
```

```
*****
```

Avail Space(Kb)	Used Space(Kb)	Type	SEs
823769996,	1760568	disk	seitep.itep.ru
176870676,	3776680	disk	lcgse.psn.ru
68185000,	4830436	disk	se01.lip.pt
221473672,	4232672	disk	castorgrid.pic.es
69374252,	1636296	disk	lcg04.gsi.de
27929516,	33084	disk	se00.inta.es

```
[...]
```

1000000000000,	500000000000	mss	castorsrm.ifaes.es
471844384,	1173380848	disk	grid-se.physik.uni-wuppertal.de
26375676,	1016344	disk	accip41.physik.rwth-aachen.de
289937964,	513772988	n.a	bigmac-lcg-se.physics.utoronto.ca
1000000000000,	500000000000	mss	castorsrm.cern.ch
721160000,	61710000	disk	se002.m45.ihep.su
1000000000000,	500000000000	mss	castorsrm.ific.uv.es
17658520089,	10128743911	disk	dcache.gridpp.rl.ac.uk
52428800,	0	disk	zam420.zam.kfa-juelich.de

You can use a particular BDII, if needed, using the `-is` option as described above. As a crosscheck you can try to repeat the test with one of the BDIIs at CERN. In the GOC DB you can identify the BDII for the production and the test zone.

This option admits a verbose level 1 just to see the names of the SEs.

`lcg-infosites` informs as well about the close SEs for each CE using the option `closeSE`:

```
> lcg-infosites --vo dteam closeSE
```

```
Name of the CE: ceitep.itep.ru:2119/jobmanager-torque-dteam
```

```
Name of the close SE: seitep.itep.ru
```

```
Name of the CE: ce.prd.hp.com:2119/jobmanager-pbs-dteam
```



Name of the close SE: se.prd.hp.com

Name of the CE: ce01.lip.pt:2119/jobmanager-torque-dteam

Name of the close SE: se01.lip.pt

[...]

Options to see the endpoints of the LRC and RMC are included and in the last release an option LFC allows to retrieve the name of the machine hosting the new LCG catalog.

Finally a option "tag" allows you to get the software tags published in each CE:

Name of the TAG: VO-dteam-pm2

Name of the CE:cel.egee.fr.cgg.com

Name of the TAG: VO-dteam-dteam1

Name of the CE:grid-ce.physik.uni-wuppertal.de

[...]

The script includes a help option (lcg-infosites -help).

As long as the lcg-infosites and the edg-gridftp-ls commands are not working it makes no sense to conduct further tests.

Assuming that this functionality is well established the next test is to use the lcg-utils in order to copy a local file from the UI to a SE and register the file with the replica location service.

Create a file in your home directory. To make tracing this file easy the file should be named according to the scheme:

```
testFile.<SITE-NAME>.txt
```

the file should be generated using the following script:

```
#!/bin/bash
echo "*****"
echo "hostname: " `hostname` " date: " `date`
echo "*****"
```

In the following examples we will use the following values:

SE: castorgrid.cern.ch

VO: dteam

file: testFile.mysite.txt

The destination storage element (option -d) is not needed if the environment variable VO\_<VO-NAME>\_DEFAULT\_SE is set up.

the command to cp the file to the SE is:

```
> lcg-cr -v --vo dteam -l testFile.mysite.txt.`date +%m.%d.%y:%H:%M:%S` -d castorgrid.cern.ch file://`pwd`/testFile.mysite.txt
Using grid catalog type: edg
Source URL: file:///afs/cern.ch/user/a/aretico/testFile.mysite.txt
File size: 158
Destination specified: castorgrid.cern.ch
Destination URL for copy: gsiftp://castorgrid.cern.ch/castor/cern.ch/grid/dteam/generated/2005-03-31/filebb981eb0-abac-4ce2-976c-83e82043e038
# streams: 1
Alias registered in Catalog: lfn:testFile.mysite.txt.03.31.05:18:19:52
Transfer took 770 ms
Destination URL registered in Catalog: sfn://castorgrid.cern.ch/castor/cern.ch/grid/dteam/generated/2005-03-31/filebb981eb0-abac-4ce2-976c-83e82043e038
guid:c6d5348b-73d5-487e-bf80-4ba07400a5da
```



The command, if everything is setup correctly, returns a line with:

```
guid:c6d5348b-73d5-487e-bf80-4ba07400a5da
```

Save the guid and the expanded lfn for further reference . We will refer to these as YourGUID and YourLFN.

In case this command failed you should keep the output and analyze it with your support contact. There could be various reasons for this command to fail.

Now we check that the RLS knows about your file. This is done by using the `lcg-lr` command from the `lcg-utils`.

The syntax is:

```
lcg-lr -v --vo YourVO lfn:YourLFN
```

**Example:**

```
> lcg-lr -v --vo dteam lfn:testFile.mysite.txt.03.31.05:18:19:52  
sfn://castorgrid.cern.ch/castor/cern.ch/grid/dteam/generated/2005-03-31/filebb981eb0-abac-4ce2-976c-83e82043e038
```

as before, report problems to your primary site.

If the RLS knows about the file the next test is to transport the file back to your UI. For this we use the `lcg-cp` command.

The syntax is:

```
lcg-cp -v --vo YourVO lfn:YourLFN file:DestFile
```

**Example:**

```
> lcg-cp -v --vo dteam lfn:testFile.mysite.txt.03.31.05:18:19:52 file://`pwd`/testBack.txt  
Source URL: lfn:testFile.mysite.txt.03.31.05:18:19:52  
File size: 158  
Source URL for copy: gsiftp://castorgrid.cern.ch/castor/cern.ch/grid/dteam/generated/2005-03-31/filebb981eb0-abac-4ce2-976c-83e82043e038  
Destination URL: file:///afs/cern.ch/user/a/aretico/testBack.txt  
# streams: 1  
Transfer took 650 ms
```

this should create in the current working directory a file named `testBack.txt`. List this file.

With this you tested most of the core functions of your UI. Many of these functions will be used to verify the other components of your site.



## 4. TESTING THE CE AND WNS

We assume that you have setup a local CE running a batch system. On most sites the CE provides two major services. For the information system the CE runs the site GIIS. The site GIIS is the top node in the hierarchy of the site and via this service the other resources of the site are published to the grid.

To test the working of the site GIIS you can run an ldap query of the following form. Inspect the output with some care. Are the computing resources (queues, etc. ) correctly reported? Can you find the local SE?. Do these numbers make sense?

```
ldapsearch -LLL -x -H ldap://lxn1181.cern.ch:2135 -b "mds-vo-name=cernlcg2,o=grid"
```

replace lxn1181.cern.ch with your site's GIIS hostname and cernlcg2 with the name that you have assigned to your site GIIS.

If nothing is reported try to restart the MDS service on the CE.

Now verify that the GRIS on the CE is operating correctly: Here again the command for the CE at CERN.

```
ldapsearch -LLL -x -H ldap://lxn1181.cern.ch:2135 -b "mds-vo-name=local,o=grid"
```

One common reason for this to fail is that the information provider on the CE has a problem. Convince yourself that MDS on the CE is up and running. Run on the CE the qstat command. If this command doesn't return there might be a problem with one of the worker nodes WNs, or the batch system. Have a look at the following link that covers some aspects on trouble shooting PBS and Torque on the GRID. <http://goc.grid.sinica.edu.tw/gocwiki/TroubleShootingHistory>

The next step is to verify that you can run jobs on the CE. For the most basic test no registration with the information system is needed. However tests can be run much easier if the resource is registered in the information system. For these tests the testZone BDII and RB have been setup at CERN. Forward your site GIIS name and host name to the deployment team for registration.

Initial tests that work without registration.

First tests from a UI of your choice:

As described in the subsection covering the UI tests the first test is a test of the fork jobmanger.

```
> globus-job-run <YourCE> /bin/pwd
```

Frequent problems that have been observed are related to the authentication. Check that the CE has a valid host certificate and that your DN can be found in the grid-mapfile.

Next logon to your CE and run a local PBS job to verify that PBS is working. Change your id to a user like dteam001. In the home directory create the following file:

```
test.sh
-----
#!/bin/bash

echo "Hello Grid"
```

run: qsub test.sh this will return a job ID of the form: 16478.lxn1181.cern.ch you can use qstat to monitor the job. However it is very likely that the job has finished before your have queried the status. PBS will place two files in your directory:

```
test.sh.o16478 and test.sh.e16478 These contain the stdout and stderr
```



Now try to submit to one of your PBS queues that are available on the CE. The following command is an example for a site that runs a PBS without shared home directories. The short queue is used. It can take some minutes until the command returns.

```
globus-job-run <YourCE>/jobmanager-lcgpbs -queue short /bin/hostname  
lxshare0372.cern.ch
```

The next test submits a job to your CE by forcing the broker to select the queue that you have chosen. You can use the testJob JDL and script that has been used before for the UI tests.

```
edg-job-submit --debug --vo dteam -r <YourCE>:2119/jobmanager-lcgpbs-short \  
testJob.jdl
```

The `--debug` option should only be used if you have been confronted with problems.

Follow the status of the job and as before try to retrieve the output. A quite common problem is that the output can't be retrieved. This problem is related to some inconsistency of ssh keys between the CE and the WN. See <http://goc.grid.sinica.edu.tw/gocwiki/TroubleShootingHistory> and the CE/WN configuration.

If your UI is not configured to use a working RB you can, as described in the UI testing subsection use configuration files to use the testZone RB.

For further tests get registered with the testZone BDII. As described in the subsection on joining LCG2 you should send your CE's hostname and the site GIIS name to the deployment team.

The next step is to take the testJob.jdl that you have created for the verification of your UI. Remove the comment from the last line of the file and modify it to reflect your CE.

```
Requirements = other.GlueCEUniqueID == "<YourCE>:2119/jobmanager-lcgpbs-short";
```

Now repeat the `edg-job-list-match --vo dteam testJob.jdl` command known from the UI tests. This output should just show one resource.

The remaining tests verify that core of the data management is working from the WN and that the support for the experiment software installation as described in <https://edms.cern.ch/file/412781/SoftwareInstallation.pdf> is working correctly. The tests you can do to verify the later are limited if you are not mapped to software manager for your VO. To test the data management functions your local default SE has to be setup and tested. Of course you can assume the SE working and run the tests before testing the SE.

Add an argument to the JDL that allows to identify the site. The jdl file should look like:

```
testJob_SW.jdl  
  
Executable = "testJob.sh";  
StdOutput = "testJob.out";  
StdError = "testJob.err";  
InputSandbox = {"/testJob.sh"};  
OutputSandbox = {"testJob.out", "testJob.err"};  
Requirements = other.GlueCEUniqueID == "lxn1181.cern.ch:2119/jobmanager-lcgpbs-short";  
Arguments = "CERNPBS" ;
```

replace the name of the site and the CE and queue names to reflect your settings.

The first script to run collects some configuration information from the WN and test the user software installation area.



testJob.sh

```
#!/bin/bash
echo "+++++"
echo "      $1      " `hostname` " " `date`
echo "+++++"
echo "the environment on the node"
echo " "
env | sort
echo "+++++"
echo "+++++"
echo "+++++"
echo "software path for the experiments"
env | sort | grep _SW_DIR
echo "+++++"
echo "+++++"
echo "+++++"
echo "mount"
mount
echo "+++++"
echo "===== "
echo "verify that the software managers of the supported VOs can \
write and the users read"
echo "DTEAM ls -l " $VO_DTEAM_SW_DIR
ls -dl $VO_DTEAM_SW_DIR
echo "ALICE ls -l " $VO_ALICE_SW_DIR
ls -dl $VO_ALICE_SW_DIR
echo "CMS ls -l " $VO_CMS_SW_DIR
ls -dl $VO_CMS_SW_DIR
echo "ATLAS ls -l " $VO_ATLAS_SW_DIR
ls -dl $VO_ATLAS_SW_DIR
echo "LHCB ls -l " $VO_LHCB_SW_DIR
ls -dl $VO_LHCB_SW_DIR
echo "===== "
echo "===== "
echo "view the default SE for the mail VOs"
echo "DTEAM default SE = " $VO_DTEAM_DEFAULT_SE
echo "ALICE default SE = " $VO_ALICE_DEFAULT_SE
echo "CMS default SE = " $VO_CMS_DEFAULT_SE
echo "ATLAS default SE = " $VO_ATLAS_DEFAULT_SE
echo "LHCB default SE = " $VO_LHCB_DEFAULT_SE
echo "===== "
echo "===== "
echo "    echo "view the default BDII "
echo "LCG_GFAL_INFOSYS =" $LCG_GFAL_INFOSYS
echo "===== "
echo "===== "
echo "===== "
echo "===== "
echo "===== "
echo "rpm -q -a | sort "
rpm -q -a | sort
echo "===== "
date
```





Run this job as described in the subsection on testing UIs. Retrieve the output and verify that the environment variables for the experiment software installation is correctly set and that the directories for the VOs that you support are mounted and accessible.

Please keep the output of this job as a reference. It can be helpful if problems have to be located.

Next we test the data management. For this the default SE should be working. The following script will do some operations similar to those used on the UI.

We first test that we can access a remote SE via simple gridftp commands. Then we test that the lcg utils have access to the information system. This is followed by exercising the data moving capabilities between the WN, the local SE and between a remote SE and the local SE. Between the commands we run small commands to verify that the RLS service knows about the location of the files.

Submit the job via `edg-job-submit` and retrieve the output. Read the file containing `stdout` and `stderr`. Keep the files for reference.

Here now a listing of `testJob.sh`:

```
#!/bin/bash

TEST_ID='hostname -f`-`date +%y%m%d%H%M`
REPORT_FILE=report
rm -f $REPORT_FILE
FAIL=0
user=`id -un`
echo "Test Id: $TEST_ID"
echo "Running as user: $user"

if [ "$1" == "x" ]; then
    echo "Usage: $0 <VO>"
    exit 1
else
    VO=$1
fi

echo "default BDII= $LCG_GFAL_INFOSYS"
#grep mds.url= /opt/edg/var/etc/edg-replica-manager/edg-replica-manager.conf
echo
echo "Can we see the SE at CERN?"
set -x
edg-gridftp-ls --verbose gsiftp://castorgrid.cern.ch/castor/cern.ch/grid/$VO > /dev/null
result=$?
set +x

if [ $result == 0 ]; then
    echo "We can see the SE at CERN."
    echo "ls CERN SE: PASS" >> $REPORT_FILE
else
    echo "Error: Can not see the SE at CERN."
    echo "ls CERN SE: FAIL" >> $REPORT_FILE
    FAIL=1
fi

echo
echo "Can we see the information system?"
set -x
```



```
lcg-infosites --vo $VO all
result=$?
set +x

if [ $result == 0 ]; then
    echo "We can see the Information System."
    echo "lcg-infosites: PASS" >> $REPORT_FILE
else
    echo "Error: Can not see the Information System."
    echo "lcg-infosites: FAIL" >> $REPORT_FILE
    FAIL=1
fi
```

The following test is using the lcg data management tools.

```
#!/bin/bash
echo "*****"
echo "    Test of the LCG Data Management Tools    "
echo "*****"
TEST_ID=`hostname -f`-`date +%y%m%d%H%M`
REPORT_FILE=report
FAIL=0
user=`id -un`
echo "Test Id: $TEST_ID"
echo "Running as user: $user"

if [ "x$1" == "x" ]; then
    echo "Usage: $0 <VO>"
    exit 1
else
    VO=$1
fi

echo "LCG_GFAL_INFOSYS      " $LCG_GFAL_INFOSYS
echo "VO_DTEAM_DEFAULT_SE   " $VO_DTEAM_DEFAULT_SE

echo
echo "Can we see the information system?"
set -x
ldapsearch -H ldap://$LCG_GFAL_INFOSYS -x -b o=grid | grep numE
result=$?
set +x

if [ $result == 0 ]; then
    echo "We can see the Information System."
    echo "LDAP query: PASS" >> $REPORT_FILE
else
    echo "Error: Can not see the Information System."
    echo "LDAP query: FAIL" >> $REPORT_FILE
    FAIL=1
fi

lfname=testFile.$TEST_ID.txt
rm -rf $lfname
```



```
cat <<EOF > $lfname
*****
Test Id: $TEST_ID

File used for the lcg-utils test
*****

EOF
myLFN="lcg-utils-test-$TEST_ID.`date +%m.%d.%y:%H:%M:%S`"

echo
echo "Copy a local file to the default SE and register it with an lfn."
set -x
lcg-cr -v --vo $VO -d $VO_DTEAM_DEFAULT_SE -l lfn:$myLFN file://`pwd`/$lfname
result=$?
set +x

if [ $result == 0 ]; then
    echo "Local file copied the the default SE."
    echo "Copy file to default SE: PASS" >> $REPORT_FILE
else
    echo "Error: Could not copy the local file to the default SE."
    echo "Copy file to default SE: FAIL" >> $REPORT_FILE
    FAIL=1
fi

echo
echo "List the replicas."
set -x
lcg-lr --vo $VO lfn:$myLFN
result=$?
set +x

if [ $result == 0 ]; then
    echo "Replica listed."
    echo "LCG List Replica: PASS" >> $REPORT_FILE
else
    echo "Error: Can not list replicas."
    echo "LCG List Replica: FAIL" >> $REPORT_FILE
    FAIL=1
fi

lf2=$lfname.2
rm -rf $lf2

echo
echo "Get the file back and store it with a different name."
set -x
lcg-cp -v --vo $VO lfn:$myLFN file://`pwd`/$lf2
result=$?
diff $lfname $lf2
set +x
```



```
if [ $result == 0 ]; then
    echo "Got get file."
    echo "LCG copy: PASS" >> $REPORT_FILE
else
    echo "Error: Could not get the file."
    echo "LCG copy: FAIL" >> $REPORT_FILE
    FAIL=1
fi

if [ "x`diff $lfname $lf2`" == "x" ]; then
    echo "Files are the same."
else
    echo "Error: Files are different."
    FAIL=1
fi

echo
echo "Replicate the file from the default SE to the CASTOR service at CERN."
set -x
lcg-rep -v --vo $VO -d castorgrid.cern.ch lfn:$myLFN
result=$?
lcg-lr --vo $VO lfn:$myLFN
set +x

if [ $result == 0 ]; then
    echo "File replicated to Castor."
    echo "LCG Replicate: PASS" >> $REPORT_FILE

else
    echo "Error: Could not replicate file to Castor."
    echo "LCG Replicate: FAIL" >> $REPORT_FILE
    FAIL=1
fi

echo
echo "3rd party replicate from castorgrid.cern.ch to the default SE."
set -x
ufilesfn=`lcg-lr --vo $VO lfn:TheUniversalFile.txt | grep lxn1183`
lcg-rep -v --vo $VO -d $VO_DTEAM_DEFAULT_SE $ufilesfn
result=$?
lcg-lr --vo $VO lfn:TheUniversalFile.txt
set +x

if [ $result == 0 ]; then
    echo "3rd party replicate succeeded."
    echo "LCG 3rd party replicate: PASS" >> $REPORT_FILE
else
    echo "Error: Could not do 3rd party replicate."
    echo "LCG 3rd party replicate: FAIL" >> $REPORT_FILE
    FAIL=1
fi

rm -rf TheUniversalFile.txt
```



```
echo
echo "Get this file on the WN."
set -x
lcg-cp -v --vo $VO lfn:TheUniversalFile.txt file://`pwd`/TheUniversalFile.txt
result=$?
set +x

if [ $result == 0 ]; then
    echo "Copy file succeeded."
    echo "LCG copy: PASS" >> $REPORT_FILE
else
    echo "Error: Could not copy file."
    echo "LCG copy: FAIL" >> $REPORT_FILE
    FAIL=1
fi

defaultSE=VO_DTEAM_DEFAULT_SE

# Here we have to use a small hack. In case that we are at CERN we will never remove the
if [ $defaultSE = lxn1183.cern.ch ]
then
    echo "I will NOT remove the master copy from: " $defaultSE
else
    echo
    echo "Remove the replica from the default SE."
    set -x
    lcg-del -v --vo $VO -s $defaultSE lfn:TheUniversalFile.txt
    result=$?
    lcg-lr --vo $VO lfn:TheUniversalFile.txt
    set +x

    if [ $result == 0 ]; then
        echo "Deleted file."
        echo "LCG delete: PASS" >> $REPORT_FILE
    else
        echo "Error: Could not do Delete."
        echo "LCG delete: FAIL" >> $REPORT_FILE
        FAIL=1
    fi
fi

echo "Cleaning Up"
rm -f $lfname $lf2 TheUniversalFile.txt

if [ $FAIL = 1 ]; then
    echo "LCG Data Manager Test Failed."
    exit 1
else
    echo "LCG Data Test Passed."
    exit 0
fi
```



## 5. TESTING THE SE

If the tests described to test the UI and the CE on a site have run successful then there is no additional test for the SE needed. We describe here some of the common problems that have been observed related to SEs.

In case the SE can't be found by the edg-replica-manager tools the SE GRIS might be not working, or not registered with the site GIIS.

To verify that the SE GRIS is working you should run the following ldapsearch. Note that the hostname that you use should be the one of the node where the GRIS is located. For mass storage SEs it is quite common that this is not the SE itself.

```
ldapsearch -LLL -x -H ldap://lxn1183.cern.ch:2135 -b "mds-vo-name=local,o=grid"
```

If this returns nothing or very little the MDS service on the SE should be restarted. If the SE returns some information you should carefully check that the VOs that require access to the resource are listed in the GlueSAAccessControlBaseRule field. Does the information published in the GlueSEAccessProtocolType fields reflect your intention? Is the GlueSEName: carrying the extra "type" information?

The next major problem that has been observed with SEs is due to a mismatch with what is published in the information system and what has been implemented on the SE.

Check that the gridmap-file on the SE is configured to support the VOs that are published the GlueSAAccessControlBaseRule fields.

Run a ldapsearch on your site GIIS and compare the information published by the local CE with what you can find on the SE. Interesting fields are: GlueSEName, GlueCESEBindSEUniqueID, GlueCESEBindCEAccesspoint

Are the access-points for all the supported VOs created and is the access control correctly configured?

The current version of lcg-infosites does not provide information on access points on the SE. A possible alternative is to manually go through ldapsearch results.

In order to test the gsiftp protocol in a convenient way you can use the edg-gridftp-ls and edg-gridftp-mkdir commands. You can use the globus-url-copy command instead. The -help option describes the syntax to be used.

Run on your UI and replace the host and accesspoint according to the report for your SE:

```
edg-gridftp-ls --verbose gsiftp://lxn1183.cern.ch/storage
drwxrwxr-x   3 root    dteam      4096 Feb 26 14:22 dteam
```

and:

```
edg-gridftp-ls --verbose gsiftp://lxn1183.cern.ch/storage/dteam
drwxrwxr-x  17 dteam003 dteam      4096 Apr  6 00:07 generated
```

if the globus-gridftp service is not running on the SE you get the following message back: error a system call failed (Connection refused)

If this happens restart the globus-gridftp service on your SE.  
Now create a directory on your SE.

```
edg-gridftp-mkdir gsiftp://lxn1183.cern.ch/storage/dteam/t1
```

Verify that the command ran successful with:

```
edg-gridftp-mkdir gsiftp://lxn1183.cern.ch/storage/dteam/t1
edg-gridftp-ls --verbose gsiftp://lxn1183.cern.ch/storage/dteam/
```

Verify that the access permissions for all the supported VOs are correctly set.



## 6. TESTING THE R-GMA

R-GMA comes with two testing scripts. The first script can be used on any node that has the R-GMA client installed (CE, SE, RB, UI, MON). To start the test, use the following command:

```
> $RGMA_HOME/bin/rgma-client-check
```

This script will check that R-GMA has been configured correctly. This is accomplished by publishing data using various APIs and verifying that the data is available via R-GMA. Successful test output should look like this:

```
*** Running R-GMA client tests on rgmaclient.server.org ***

Checking C API: Done.
Success
Checking C++ API: Success
Checking Python API: Success
Checking Java API: Success

Checking for safe arrival of tuples, please wait... Success

*** R-GMA client test successful ***
```

The second script is intended to run only on an R-GMA MON box. It checks if the R-GMA server is configured correctly and tries to connect to the servlets. To run the script, use:

```
> $RGMA_HOME/bin/rgma-server-check
```

Successful output should look like:

```
*** Running R-GMA server tests on lxn1193.cern.ch ***

Checking servlets...
Connecting to http://lxn1193.cern.ch:8080/R-GMA/ConsumerServlet:OK
Connecting to streaming port 8088 on lxn1193.cern.ch:OK
Connecting to http://lxn1193.cern.ch:8080/R-GMA/StreamProducerServlet:OK
Connecting to http://lxn1193.cern.ch:8080/R-GMA/LatestProducerServlet:OK
Connecting to http://lxn1193.cern.ch:8080/R-GMA/DBProducerServlet:OK
Connecting to http://lxn1193.cern.ch:8080/R-GMA/CanonicalProducerServlet:OK
Connecting to http://lxn1193.cern.ch:8080/R-GMA/ArchiverServlet:OK

*** R-GMA server test successful ***
```

If either test fails. The common reasons are that the servlets are down or that there is a firewall problem (make sure ports 8080 and 8088 are open). Check that the values for the Servlets and Registry are correct in the file `/opt/edg/var/edg-rgma/rgma.conf`. Take the URL from this file, append `/getStatus` to the end and use a browser to connect to the servlet. If you can not connect to the servlet, check the URL again. Try to restart the tomcat server on the MON node

```
/opt/edg/etc/init.d/edg-tomcat4 restart
```

If there are problems starting the servlets some information can be found in the tomcat logs.

```
/var/tomcat4/logs/catalina.out
```