



Enabling Grids for E-scienceE

Job Submission Tutorial

(material from INFN Catania)

www.eu-egee.org





Enabling Grids for E-science

Find out what are the best fitting resources for your computational request.

Suppose to have a "job" that can be expressed by the following JDL

```
(vi $HOME/Catania/JobSubmission/hostname.jdl)
[ Type="Job";
  JobType="Normal";
  Executable = "/bin/sh";
  Arguments = "start_hostname.sh";
  StdError = "stderr.log";
  StdOutput = "stdout.log";
  InputSandbox = "start_hostname.sh";
  OutputSandbox = {"stderr.log", "stdout.log"};
  RetryCount = 7; ]
```

www.eu-egee.org





The script `start_hostname.sh` is sent to the CE and it is passed as an argument to the executable `"/bin/sh"`. The script has two instructions:

```
#!/bin/sh  
sleep 5  
hostname -f
```

Run the `glite-job-list-match` command to find out the appropriate resources list.

```
$ glite-job-list-match hostname.jdl
```

COMPUTING ELEMENT IDs LIST

The following CE(s) matching your job requirements have been found:

CEId

- cn01.be.itu.edu.tr:2119/jobmanager-lcglsf-infinite
- cn01.be.itu.edu.tr:2119/jobmanager-lcglsf-long
- cn01.be.itu.edu.tr:2119/jobmanager-lcglsf-short
- dgt01.ui.savba.sk:2119/jobmanager-lcgpbs-short
- gilda-ce-01.pd.infn.it:2119/jobmanager-lcgpbs-long
- gilda-ce-01.pd.infn.it:2119/jobmanager-lcgpbs-short
- glite-ce.ct.infn.it:2119/blah-pbs-long
- grid-ce.bio.dist.unige.it:2119/jobmanager-lcgpbs-long
- grid-ce.bio.dist.unige.it:2119/jobmanager-lcgpbs-short
- grid002.mporzio.astro.it:2119/jobmanager-lcgpbs-infinite
- grid002.mporzio.astro.it:2119/jobmanager-lcgpbs-long
- grid002.mporzio.astro.it:2119/jobmanager-lcgpbs-short
- grid006.cecalc.ula.ve:2119/jobmanager-lcgpbs-infinite
- grid010.ct.infn.it:2119/jobmanager-lcgpbs-infinite
- grid010.ct.infn.it:2119/jobmanager-lcgpbs-long
- grid010.ct.infn.it:2119/jobmanager-lcgpbs-short
- grid011f.cnaf.infn.it:2119/jobmanager-lcgpbs-long
- grid011f.cnaf.infn.it:2119/jobmanager-lcgpbs-short
- glite-ce.ct.infn.it:2119/blah-pbs-short
- gilda-ce-01.pd.infn.it:2119/jobmanager-lcgpbs-infinite
- grid011f.cnaf.infn.it:2119/jobmanager-lcgpbs-infinite
- grid-ce.bio.dist.unige.it:2119/jobmanager-lcgpbs-infinite
- gildace.ct.astro.it:2119/jobmanager-lcgpbs-infinite
- gildace.ct.astro.it:2119/jobmanager-lcgpbs-long
- gildace.ct.astro.it:2119/jobmanager-lcgpbs-short
- glite-ce.ct.infn.it:2119/blah-pbs-infinite
- gildace.ct.astro.it:2119/jobmanager-lcgpbs-short

www.eu-egee.org



**Repeat this exercise on LCG UI available using the
command**

- edg-job-list-match

finding the appropriate command line to use.

Exercise 2

=====

Let's now run our "hostname" job on a CE (letting the WM choose the best suitable CE for execution). Consider again the `hostname.jdl` and `start_hostname.sh`.

```
$ glite-job-submit -o myjobid hostname.jdl
```



Enabling Grids for E-scienceE

Where myjobid contains:

\$ cat myjobid

###Submitted Job Ids###

**[https://glite-
rb.ct.infn.it:9000/zGFA9ntyOY8gIIISLPG94cg](https://glite-rb.ct.infn.it:9000/zGFA9ntyOY8gIIISLPG94cg)**

www.eu-egee.org



**Follow now the job life through job-status command
in order to discover which the selected
CE is.**

\$ glite-job-status -i myjobid

or

\$ glite-job-status -i myjobid -v 3

Or

\$ glite-job-logging-info -v 2 -i myjobid



Enabling Grids for E-scienceE

Repeat all using LCG commands as listed below:

edg-job-submit -o myjobid hostname.jdl

edg-job-status -i myjobid -v 3

edg-job-get-logging-info -v 2 -i myjobid

www.eu-egee.org



Retrieve job output

(Check that directory JobOutput exists before running command)

```
$ glite-job-output -i myjobid --dir ./JobOutput
```

Exercise 4

=====

Let's now force the WM to choose a given CE as destination for the job. It can be done

in two different ways. First of all, try the glite-job-submit option

`--resource ce_id`

`-r ce_id`

If the command is launched with this option, the job-ad sent to the NS contains a line of the type

`SubmitTo = ce_id` and the job is submitted by the WMS to the resource identified by `ce_id` without going

through the match-making process. The standard format for a CEId is:

`<full hostname>:<port number>/jobmanager-<service>-<queue name>`

So:

```
$ glite-job-submit -r grid010.ct.infn.it:2119/jobmanager-lcgpbs-long -o myjobid  
hostname.jdl
```

As done before, let's inspect the content of myjobid file:

```
$ cat myjobid
```

```
###Submitted Job Ids###
```

```
https://glite-rb.ct.infn.it:9000/zGFA9ntyOY8gIISLPG94cg
```

```
https://glite-rb.ct.infn.it:9000/HRtARhjGQArbw9EeW9hBtQ
```

```
[pappalar@glite-tutor:~/Catania/JobSubmission]$
```

It now contains two files! Did you expect that? :) After a short while, in order to investigate the status use `glite-job-status` and choose 2 when asked.

```
$ glite-job-status -i myjobid -v 3
```

```
1 : https://glite-rb.ct.infn.it:9000/zGFA9ntyOY8gIISLPG94cg
```

```
2 : https://glite-rb.ct.infn.it:9000/HRtARhjGQArbw9EeW9hBtQ
```

```
a : all
```

```
q : quit
```

www.eu-egee.org



Enabling Grids for E-science

Try now to get to the same result through setting "requirements" attribute inside the JDL file. Change our hostname.jdl according to what is shown below and save it as hostnamereq.jdl.

```
(vi $HOME/Catania/JobSubmission/hostnamereq.jdl)
```

```
[ Type="Job";
```

```
JobType="Normal";
```

```
Executable = "/bin/sh";
```

```
Arguments = "start_hostname.sh";
```

```
StdError = "stderr.log";
```

```
StdOutput = "stdout.log";
```

```
InputSandbox = "start_hostname.sh";
```

```
OutputSandbox = {"stderr.log", "stdout.log"};
```

```
Requirements = other.GlueCEUniqueld == "grid010.ct.infn.it:2119/jobmanager-  
lcpbs-long";
```

```
RetryCount = 7; ]
```

www.eu-egee.org



- Try out with the `glite-job-list-match`. That will help us to understand whether we correctly set our criteria.

`$ glite-job-list-match hostnamereq.jdl`



Enabling Grids for E-scienceE

Let's immediately go to the glite-job-submit!!!

\$ glite-job-submit -o myjobid hostnamereq.jdl

www.eu-egee.org





Enabling Grids for E-scienceE

```
$ cat myjobid
```

```
###Submitted Job Ids###
```

```
https://glite-rb.ct.infn.it:9000/zGFA9ntyOY8gIISLPG94cg
```

```
https://glite-rb.ct.infn.it:9000/HRtARhjGQArbw9EeW9hBtQ
```

```
https://glite-rb.ct.infn.it:9000/CWCwBjRvNvg8EYLCvYpHPg
```

Go ahead as usual until job status is done and then retrieve the Output.

www.eu-egee.org



Exercise 6

=====

If you're interested into selecting the best CE in a set whose items respect a given criteria, instead of into a specific CE, you need to set up a filter through Requirements attribute and also a Rank criteria to get the entries ordered.

In the following example, the JDL provides a filter based on LRMS type, explicitly asking for "PBS".

In the selected set, the best Ranked CE will be chosen, that is the one having the highest number of Free CPUs.

Save this JDL as hostnamerank.jdl.

```
(vi $HOME/Catania/JobSubmission/hostnamerank.jdl)
[ Type="Job";
  JobType="Normal";
  Executable = "/bin/sh";
  Arguments = "start_hostname.sh";
  StdError = "stderr.log";
  StdOutput = "stdout.log";
  InputSandbox = "start_hostname.sh";
  OutputSandbox = {"stderr.log", "stdout.log"};
  Requirements = other.GlueCEInfoLRMSType == "PBS";
  Rank = other.GlueCEStateFreeCPUs;
  RetryCount = 7; ]
```

\$ glite-job-list-match hostnamerank.jdl

Exercise 8

=====

Run an ls command on the selected resource.

```
(vi $HOME/Catania/JobSubmission/scriptls.jdl)
```

```
[ Executable = "ls.sh";
```

```
Arguments = "-alt";
```

```
StdError = "stderr.log";
```

```
StdOutput = "stdout.log";
```

```
InputSandbox = "ls.sh";
```

```
OutputSandbox = {"stderr.log", "stdout.log"}; ]
```

The executable is ls.sh that produces the execution of /bin/ls -alt on the resource, as shown:

```
#!/bin/sh
```

```
/bin/ls
```

Exercise 9

=====

Suppose you have your own c program source code and that you'll be willing to execute it on the selected resource. How to do it?

Very simple: just compile and transfer it as executable properly setting the JDL file content.



source.c is a C program that prints a Welcome message on the resource. See the listing:

```
(vi $HOME/Catania/JobSubmission/source.c)
```

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

```
int main(int argc, char **argv)
```

```
{
```

```
    char *name = argv[1];
```

```
    /*system("clear");*/
```

```
    printf("\n\n\n");
```

```
    printf("Hello %s!\n",name);
```

```
    printf("Welcome to Biomed, Clermont-Ferrand \n\n\n");
```

```
    exit(0);
```

```
}
```



Enabling Grids for E-science

Let's compile it, typing simply 'make'. Through makefile you can see in your directory, you'll obtain c_exe executable file.

Just have a look at the JDL indicating that the executable must be transferred and executed.

```
(vi $HOME/Catania/JobSubmission/c_exe.jdl)
```

```
Type = "Job";
```

```
JobType = "Normal";
```

```
Executable = "/bin/sh";
```

```
StdOutput = "std.out";
```

```
StdError = "std.err";
```

```
OutputSandbox = {"std.err","std.out"};
```

```
InputSandbox = {"startExe.sh","c_exe"};
```

```
Arguments = "startExe.sh";
```

```
RetryCount = 7;
```

www.eu-egee.org



Information Society





Enabling Grids for E-scienceE

So:

```
$ glite-job-submit -o myjobid c_exe.jdl
```

www.eu-egee.org

