Access Policy.
Batch system of the calculus servers

For jobs that require more resources (CPU time, memory or space on disk) than those limited by the interactive shell, the batch system must be used. This system is Sun Grid Engine. The way to send jobs implies knowing beforehand the maximum values of CPU time, number of processors, architecture, memory and space on disk that the job is going to require. This also allows a better use of the system resources by all the users.
The command to send jobs to the batch system in the servers is qsub, followed by a list of the resources the job needs. For instance, let us suppose that we want to send a Gaussian job that does not require more than 2 Gigabytes of memory and 10 Gigabytes of scratch, and we estimate an approximated execution time of no more than 24 hours, using one processor. If the entry file for Gaussian is called script.com and it can be found in the “pruebas” directory, the way to send this job to queue would be:
qsub -l num_proc=1,s_rt=24:00:00,s_vmem=2G,h_fsize=10G
cd $HOME/pruebas
g98 < script.com
control+D
If there is no mistake, we will get this kind of message:
Your job 492 ("STDIN") has been submitted
The number 492 is what is called job identifier or JOBID and it allows us to identify our job within the batch system (for instance, in order to use the command qstat and to know whether it is being executed, in-queue, etc. or not). It is also important to indicate this number in consults by phone or e-mail to the Systems Department staff.
The way to specify resources is with the option -l, followed by the resources that are being required separated with a “,”. It is essential to leave a blank space between the option “-l” and the resource list. The resources must be:

Resource	Meaning	Units	Minimum value	Maximum value
num_proc	Number of processors required for the job			
SVG				
FT				

1

16
(only with MPI)

256

s_rt

Maximum real time a job can last for

Time

-

200:00:00

200:00:00

s_vmem

Total amount of RAM memory required by a job

Size

112M-SVGD
112M-HPC
550M-SD

4G

2TB

h_fsize

Maximum space required by one single file created by the job

Size

-

120G

400GB

We must bear in mind that:
It is very important to leave a blank space between the option “-l” and the following resource, while there

should not be any other blank space separating the resources after that.
1. These values are maximum, so they cannot be broken. This means that if we believe that our job will last around 23 hours, we should put s_rt=24:00:00 in order to be sure of leaving a safety margin. After 24 hours the system will finish the job automatically, even though it would not have been finished.
2. The more these values are in keeping with the resources the job actually consumes, the more priority the job will have. If these resources are not enough for the job, it will be aborted due to lack of resources and it will be necessary to indicate the proper values. In general, we recommend applying for the closest resources, above the resources estimated. The reason is that the less resources are applied for, the more priority the job will have and the sooner it will run.
The format of the units for the resources is the following:
TIME: it specifies the maximum time period during which a resource can be used. The format is the following: [[hours:]minutes:]seconds, for example:
00:30:00 are 30 minutes
100:00:00 are 10 hours
1200 are 1200 seconds (20 minutes)
SIZE: it specifies the maximum size in Megabytes. It is expressed in the form whole[suffix]. The suffix acts as a multiplier defined in the following table:
K Kilo (1024) bytes
M Mega (1,048,576) bytes
G Giga (1,073,741,824) bytes
To sum up, let us show some examples for different jobs:
1. For a job that requires few memory consumption and execution time:
qsub -l num_proc=1,s_rt=00:10:00,s_vmem=100M,h_fsize=100M job.sh
2. A job that requires much execution time (80 hours) and few memory (256Mb is enough)
qsub -l num_proc=1,s_rt=80:00:00,s_vmem=256M,h_fsize=10M job.sh
3. A job with great memory requirements (4GByte) but few execution time:
qsub -l num_proc=1,s_rt=00:30:00,s_vmem=4G,h_fsize=10M job.sh
4. A job that generates a big result file (up to 20Gigabytes):
qsub -l num_proc=1,s_rt=30:00:00,s_vmem=500M,h_fsize=20G job.sh
5. A job that consumes 100 CPU hours, 1Gigabyte of memory and generates a 5-Gigabyte file:
qsub -l num_proc=1,s_rt=100:00:00,s_vmem=2G,h_fsize=5G job.sh
6. A parallel job with 8 processors and 10 hours of total execution time, 8Gigabytes of total memory and which generates a 10-Gigabyte file:
qsub -l num_proc=8,s_rt=10:00:00,s_vmem=8G,h_fsize=10G job.sh
If you need to use values superior to the limits of these resources, you must apply for the special queue, sending an e-mail to the address sistemas@cesga.es
Once we execute the command qsub and we obtain the identifier for the job, it passes to an appropriate queue for its execution. The job will wait in turn for the moment when the required resources are available, to go to execution, and finally the job will finish and it will disappear from the queue.
Checking the state of the jobs:
In order to check the state in which the jobs are, the command qstat can be used.

qstat
We will obtain an exit as the following one:

Job id

prior

name

user

state

at

Queue

master

489
0
carlosf
r
12/29/2003
19:49:05
Cola1
MASTER

The meanings of the fields are the following:
Job-ID: 489 is the value of the JOB-ID that was assigned to the PBS queue system. The JobID is a unique identifier valid for every job and it allows the monitoring of it.
Prior: indicates the priority with which the job is being executed.
Name: STDIN is the name of the job that was sent to queue. If a job was sent from the standard input (that is, writing the commands intensively when the job was sent), STDIN will appear. In the event of being a script, the name of the script will appear.
User: carlosf is the login of the user who sent the job to queue
State: “r” is the state in which the job is at the moment, and it indicates that it is running. The other possible states of a job are:
t: transferring the job in order to start running
s: temporally suspended in order to execute other foreground jobs
w: the job is in-queue, waiting for the necessary resources for it to run to be available, or because the user exceeded the limits.
Submit/start at: date and hour when the job was sent to queue or when it started running
Queue: cola 1 is the name of the queue to which the job was sent. The destination queue will depend on the resources requested.
Master: indicates the host from which the job was sent
For further information about how to use the batch system in an specific system check the corresponding user's guide:
- SVG user's guide

- Finisterrae user's guide