Resources

The scheduler's job is to make sure the cluster runs as many jobs as possible within the shortest possible time. It does this by planning when to run jobs and on what nodes. It's a bit like the computer game Tetris - the scheduler is trying to pack all of the bricks so that no gaps exist. An important part of this process is telling the scheduler what resources a job requires (or the shape of the Tetris brick) so it can plan its placement.

When you submit a job with qsub, the job is given some default requirements. For example, it assumes that you only need one node and they you'll only be using one core on that node. This has a direct impact on your job, because if you need 2 nodes and you only ask for 1, then you won't get the second node. Alternatively, if you ask for 1 core on a 16-core machine, but then use all of them, the scheduler will think that there are free cores and run more jobs on the node - leading all of the jobs to run more slowly.

There are many resources that you can request of the scheduler. The full list is available in the pbs_resources man page (type man pbs_resources on the submit host). Here we will cover the two most requested resources; the number of nodes, and the number of processes per node. Thankfully, these are also the two resources you can most easily predict.

Requesting resources is not the same as using them. You can request 10 nodes and 160 cores, but if your code isn't programmed to use them (via technologies such as MPI and OpenMPI) then you won't be able to use them.

Requesting Resources

You request the type of resources with the -l option to qsub. The argument takes a list of resources and their values. For example, you can request that you want to run on two nodes as follows:

```
$ qsub -l nodes=2 example1.sh
```

If you want to say that your job requires 8 cores on each node, you specify the ppn resource (processes-per-node) as well as the number of nodes:

```
$ qsub -l nodes=2:ppn=8 example1.sh
```

If you want to run your job on a specific host (this is useful with Interactive jobs) then you can specify the node name:

```
$ qsub -l nodes=avafat01.grid.fe.up.pt -I
```

If you know how long your job will take, you can specify the wall clock time (or simply wallclock) in hours:minutes:seconds, for example this job will run for a maximum of 2 hours:

```
$ qsub -l walltime=02:00:00 example1.sh
```

× If your job doesn't finish before the wallclock limit, the scheduler will kill it. All jobs are given a default wallclock limit.

You can combine requests for multiple resources by either specifying them separately, or separating them with commas. So both of these commands do for the same thing:

```
$ qsub -l walltime=02:00:00 -l nodes=1 example1.sh
$ qsub -l walltime=02:00:00,nodes=1 example1.sh
```

Viewing Resources

You can see the resources requested and allocated to a job using the qstat command with the -f argument:

```
$ qsub -l walltime=02:00:00,nodes=1:ppn=1 example1.sh
63198.maui.grid.fe.up.pt
$ qstat -f 63198.maui.grid.fe.up.pt
Job Id: 63198.maui.grid.fe.up.pt
    Job_Name = example1.sh
    Job_Owner = jbarber@submit.grid.fe.up.pt
    job_state = R
    queue = batch
    server = maui.grid.fe.up.pt
    Checkpoint = u
    ctime = Thu May 30 11:09:28 2013
    Error_Path = submit.grid.fe.up.pt:/homes/jbarber/example1.sh.e63198
    exec_host = ava13.grid.fe.up.pt/5
    Hold_Types = n
    Join_Path = n
    Keep_Files = n
    Mail_Points = a
    mtime = Thu May 30 11:09:28 2013
    Output_Path = submit.grid.fe.up.pt:/homes/jbarber/example1.sh.o63198
    Priority = 0
    qtime = Thu May 30 11:09:28 2013
    Rerunable = True
    Resource_List.nodect = 1
    Resource_List.nodes = 1:ppn=1
    Resource_List.walltime = 02:00:00
    session_id = 19278
    Variable_List = PBS_O_QUEUE=batch,PBS_O_HOME=/homes/jbarber,
    PBS_O_LANG=en_US.UTF-8,PBS_O_LOGNAME=jbarber,
    PBS_O_PATH=/usr/lib64/qt-3.3/bin:/usr/local/bin:/bin:/usr/bin:/usr/lo
    cal/sbin:/usr/sbin:/sbin:/homes/jbarber/bin,
    PBS_O_MAIL=/var/spool/mail/jbarber,PBS_O_SHELL=/bin/bash,
    PBS_O_HOST=submit.grid.fe.up.pt,PBS_SERVER=maui.grid.fe.up.pt,
    PBS_O_WORKDIR=/homes/jbarber
    etime = Thu May 30 11:09:28 2013
    submit_args = -l walltime=02:00:00,nodes=1:ppn=1 example1.sh
    start_time = Thu May 30 11:09:28 2013
    Walltime.Remaining = 7196
    start_count = 1
    fault_tolerant = False
    submit_host = submit.grid.fe.up.pt
    init_work_dir = /homes/jbarber
```

From:
https://grid.fe.up.pt/dokuwiki/ - **GRID FEUP**

Permanent link:
https://grid.fe.up.pt/dokuwiki/doku.php?id=documentation:resources

Last update: **2016/05/25 12:51**