Introduction to batch computing on the Flux cluster: Torque PBS

---

07 Oct 2016

1. Components of the cluster management system
    - Login nodes
    - Compute nodes
    - Home directory space
    - Scratch space
    - Data transfer nodes
    - Batch job management software – Torque
    - Scheduling software – Moab
    - Billing system and account management
2. What you need to use the system
    - Login account
    - Allocation: Could be provided by college (COE and LSA) or Dept or PI or Rackham
3. Creating a first batch job and running it
    - create a batch file with a working script

```
$ mkdir hpc101
$ cd hpc101
$ nano hello.sh
----
echo "Hello, world"
----
```

    - batch file structure: preamble and job portion
    - The absolute, bare minimum that must be in a PBS script

```
#PBS -A hpc101_flux
#PBS -q flux
./hello.sh
```

    That will almost never be enough, but that's the minimum. Where's the money and which line do I stand in to pay. + What are the available options? + batch options that should be specified and in some preferred order

```
cp /scratch/data/workshops/hpc101/preamble.txt .
####  PBS preamble

#PBS -N job_name_no_spaces

#PBS -M uniqname@umich.edu
#PBS -m abe
#PBS -j oe

#PBS -l nodes=1:ppn=1,mem=1gb,walltime=00:15:00
#PBS -V

#PBS -A hpc101_flux
#PBS -l qos=flux
#PBS -q flux

####  End PBS preamble
```

- To create a PBS script, add the preamble to any runnable script

```
$ cat preamble.txt hello.sh > script.pbs
```

- Some other things are useful to have in the script, so

```
$ cp /scratch/data/workshops/hpc101/template.pbs .
$ nano template.pbs
```

- Note the if statements and the note about which modules are needed (we'll come back to those in a bit). Complete the information for `template.pbs`
- For now, let's run what we have

```
$ qsub template.pbs
```

- batch job manager commands

```
$ qsub <PBSscript.pbs>
$ qstat -u $USER
$ qstat <JobID>
$ qdel <JobID>
```

4. Scheduler and how it is different from the batch manager
    - scheduler determines the order in which things run and instructs the batch manager to start jobs.
    - scheduler commands typically begin with an 'M', but not always

```
$ mdiag -u $USER
```

shows which allocations can be used

```
$ mdiag -a hpc101_flux
```

shows procs and memory for an allocation

5. How to check on jobs and accounts

```
$ checkjob -v <JobID>
$ showq -w acct=<AccountName>
```

6. How to check on allocations

```
$ freealloc account_name
```

Maybe you are like me, submit a bunch of jobs, then realize that you didn't load the modules first. Aargh! You can use

```
$ cancel_my_jobs
```

to delete *all* your currently running or queued jobs. This wraps the `qdel` command with some options and error checking so you don't generate a ton of e-mail to us that you don't have permission to delete everyone else's jobs, too.

1. Running an interactive job
   - What an interative job is
   - Run it when 1) you need more time or memory or threads than would be polite or allowed on a login node and/or 2) you need to run interactively with processors on more than one physical machine.
   - To run an interactive job, you can put all the PBS options on the qsub command

     ```
     $ qsub -I -V -l nodes=2:ppn=12,pmem=2gb,walltime=1:00:
     00 \
          -A hpc101_flux -l qos=flux -q flux -j oe <pbs_scr
     ipt>
     ```

   or you can add the `-I` option to `qsub` with a file

   ```
   $ qsub -I <pbs_script>
   ```

   Note: Resize your terminal window to the size you want *before* your submit an interactive job.

2. Copying data to and from Flux using the command line (Mac and Linux)

   ```
   $ sftp flux-xfer.arc-ts.umich.edu
   ```

   which will give you an interactive prompt, or

```
$ scp my_file flux-xfer.arc-ts.umich.edu:
$ scp my_data_file flux-xfer.arc-ts.umich.edu:data/
$ scp -r my_data_dir flux-xfer.arc-ts.umich.edu:my_study
```

- GUI tools, e.g., WinSCP, CyberDuck FileZilla