HPC 101

---

01 Jun 2016

1. Components of the cluster management system
    - Login nodes
    - Data transfer nodes
    - Compute nodes
    - Home directory space
    - Scratch space
    - 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
    - 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 things

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

6. 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.

7. Copying data to and from Flux
   - Command line (Mac and Linux)

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

   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

8. Software modules
   - Software modules enable you to control which software packages are available to use to prevent conflicts and control versions.
   - To find modules that are available

```
module available
module av
module av anaconda
```

- `module` commands need to be run once after logging in and before submitting a job to the queue. That insures that the settings are communicated to all the nodes (using the `-V` PBS flag).
- If you find you need the same module(s) all the time, you can create a file that will load them automatically at login.

```
nano ${HOME}/privatemodules/default
module rm intel-comp openmpi
module load anaconda2
module list
```

- Modules can have dependencies

```
module list
Currently Loaded Modulefiles:
  1) moab      2) torque    3) modules   4) use.own
module load gromacs/4.6.5
```

- Modules can have conflicts

```
module rm gcc gsl mkl gromacs
module load intel-comp
module load openmpi
```

- Switching versions can be easier

```
module swap openmpi/1.8.2/gcc/4.8.0 \
    openmpi/1.6.5/intel/14.0.2
module swap openmpi
```

- flux-utils

```
cancel-my-jobs
freealloc
```

9. Now we have the pieces to run software that is already runnable, e.g., things that are installed on Flux and don't require anything additional to run, like R, Stata, Matlab.
    - Example of developing a working script. Supose we are embarking on a study of plant phylogeny. We have some data already stored in an R library. As the study progresses, the same statistics will need to be run as each round of data is added. So, first we determine, using a small sample, on the login node which commands are needed and to check that we have syntax correct.

```
module load R/3.1.1
R

library(datasets)
data(iris)
summary(iris)
```

- Now that we have that working, we need to put that into a script and check that the batch version works. Enter those commands into `iris.R`

```
mkdir R-examples
cd R-examples
R CMD BATCH --no-restore --no-save iris.R
```

- Next, we need to make a PBS script to run the batch job. Copy your `template.pbs` to `iris.pbs`, fill in the necessary options to run the R script and submit it.

10. Compiling software
    - Serial or shared memory software – one physical machine
    - Use the compiler directly

```
cp -r /scratch/data/workshops/hpc101/code-examples .
icc -o hello hello.c
./hello
```

    - To change compiler

```
module load gcc
gcc -o hello hello.c
./hello
```

    - For this simple example, `gcc` and `intel-comp` can coexist, however it is good practice to unload the unused compiler if you can.
    - Software you compile yourself is run the same way as commercial software
    - Parallel or MPI-based software – more than one physical machine
    - Uses compiler scripts instead of using the compiler directly
    - Compiler and MPI version must match

```
mpicc -o hello mpi-hello.c
mpirun -np 4 hello
```

    - You must load the correct (matching) openmpi to run a program compiled with it
    - When run from inside a batch job, you do not need `-np N` unless you use fewer processors than requested by the PBS script.
    - To change compilers, you have to change both the compiler and the openmpi

```
module list
module rm intel-comp openmpi
module load gcc/4.7.0 openmpi/1.6.5/gcc/4.7.0
mpicc -o hello mpi-hello.c
mpirun -np 4 hello
```

11. Use `make`
    - Many packages that you have to compile will come with a `Makefile`, or you will run `./configure` to create a `Makefile`.
    - `make` is a program that enables you to create reliable steps for compiling, testing, and installing softare. It can also help with portability – moving software from one type of computer to another.
    - Why does this matter? Repeatability. Reliability. Convenience. To compensate for memory lapses.
    - `make` will automatically look for `Makefile`, then `makefile`, or you can tell it explicitly to use a file with `-f <filename>`