High Performance Computing (HPC) Club Training Session

Xinsheng (Shawn) Qin
24 Oct 2016
Outline

• The Hyak Supercomputer
• Log in to Hyak
• Submitting your first job
• The backfill queue
• Transfer files between your PC and Hyak
• Bash scripting
  • ~/.bashrc
The Hyak Supercomputer

- A typical **node** has 16 processor **cores** and 128GB of memory
- All the nodes run CentOS 6 Linux
- ~10000 cores in total
- Club members have access to 1028 cores
- Node types
  - Computing nodes
  - Login nodes
    - Submitting jobs
    - Transferring files
Log in to Hyak

Remotely Accessing a Linux System via SSH protocol

- Use a SSH client program
  - Linux & Mac: just open a terminal
    - Mac: application->utilities->terminal
    - Ubuntu: Search for “terminal” in your applications
  
  **usage:** `ssh [options] user@IP-address`

  **e.g**

  `ssh xsqin@hyak.washington.edu`
  `ssh xsqin@americano.amath.washington.edu`
  `ssh shawn@128.95.220.141`

  Then enter your password and PRN from your token

- Windows: putty
Log in to Hyak

- putty
- More info here: http://wiki.cac.washington.edu/display/hyakusers/Logging+In
Linux command basis

• Show current directory: `pwd`
• `$HOME` directory
• Show contents in current directory: `ls [options]`
• change current directory: `cd`
  • `cd absolute_path`
  • `cd relative_path`
• Create a new folder: `mkdir folder_name`
• Remove a file/folder: `rm [options]`
• Rename a file/folder: `mv old_name new_name`
Submitting your first job

- Node types:
  - logging node: used to submit jobs. Don’t execute heavy tasks on it
  - computing node: where your jobs are running
- Using qsub
  - qsub [options] command-for-running-your-job
  - e.g. qsub -l walltime=2:00:00 -I -V matlab
- PBS (Portable Batch System) Jobscripts
  - instructions for the scheduler
  - setting up the work environment
  - executing your production program
- http://wiki.cac.washington.edu/display/hyakusers/Hyak+Job+Scheduler
  #HyakJobScheduler-ImportantInformation
Submitting your first job

qsub run.sh
Submitting your first job

Practice:
In your terminal, clone a small code for practicing by typing:
git clone https://github.com/xinshengqin/HPCC_training_session.git
Then do:
• Compile the c code: mpicc test.c -o test
• Change working directory, output directory and group in run.sh
• Submit the job: qsub run.sh
Managing Jobs in Your Queues

• Check the status of your job
  checkjob <job_id>
  qstat -f <job_id>

• Queue Status
  • showq -w qos=<your_group>
  • showq -w user=<your_netid>
  • e.g. showq -w qos=stf

• To cancel a job
  mjobctl -c <job_id>
  qdel <job_id>
Backfill queue & interactive jobs

• Backfill queue
  • make use of idle CPUs throughout the cluster
  • subject to immediate preemption by node owner’s jobs
  • no interactive jobs
  • preempted (canceled and requested) at least every 4 hours

• Interactive jobs
  • qsub -l -V -W group_list=hyak-stf -l walltime=2:00:00
Managing your files

• Transfer files from your local desktop to the server
  • Linux & Mac: SCP
    • scp [options] <source directory> <target directory>
  • Windows: WinSCP
    • https://winscp.net/eng/download.php

• Where should I put my job outputs?
  • Gscratch space: fast, temporary
    • /gscratch/<your group>/<your netid>
    • e.g. /gscratch/stf/xsqin
  • Lolo: slow, long-term storage
    • /lolo/archive/hyak/<your group>/<netid>
    • /lolo/archive/hyak/stf/xsqin
Bash scripting

- Some useful setup that I put in my $HOME/.bashrc
  git clone https://github.com/xinshengqin/HPCC_training_session.git