Introduction to HPC Using Sapelo at GACRC

Georgia Advanced Computing Resource Center
University of Georgia
Suchitra Pakala
pakala@uga.edu
Slides courtesy: Zhuofei Hou
Outline

• GACRC

• Sapelo Cluster at GACRC

• Job Submission Workflow

• Work with Sapelo Cluster

• Guideline and Practical Tips
We are a high-performance-computing (HPC) center at UGA.

We provide to the UGA research and education community an advanced computing environment:

- HPC computing and networking infrastructure located at the Boyd Data Center
- Comprehensive collection of scientific, engineering and business applications
- Consulting and training services

- [http://wiki.gacrc.uga.edu](http://wiki.gacrc.uga.edu) (GACRC Wiki)
- [https://wiki.gacrc.uga.edu/wiki/Getting_Help](https://wiki.gacrc.uga.edu/wiki/Getting_Help) (GACRC Support)
- [http://gacrc.uga.edu](http://gacrc.uga.edu) (GACRC Web)
Sapelo cluster at GACRC

- Cluster Diagram
- Cluster Overview
- Computing Resources
- Storage Environment
What is a Cluster?

When you work on cluster, 3 roles are working:

➢ User: to submit a job

➢ Queueing system: to dispatch a job to run on cluster

➢ Cluster: to run a job
Sapelo Cluster Diagram

- **Node**: Computer for a specific function on cluster, e.g., login node, work node
- **Queue**: Organization of **compute nodes** for a specific computing need
- **Cluster**: **Nodes + Drives**, all connected by network

**Diagram Details**:

- **Login Node (sapelo1)**
  - qlogin

- **Work Node (Interactive Node)**
  - Work Drive
    - /lustre1
    - No quota limit

- **Home Drive**
  - /home
  - 100 GB

- **Transfer Node (xfer)**
  - Project Drive
    - /project
    - 1 TB

- **Compute Nodes**
  - AMD Compute Nodes
  - HIGHMEM Compute Nodes
  - GPU Compute Nodes

**Batch Queue**

---

**UGA MyID**
The New GACRC Linux HPC Cluster Structural Diagram

Network Connection Speed:
- **InfiniBand**
- 10 Gbit Ethernet
- 1 Gbit Ethernet

---

**Icebreaker Storage**
- Node index: sn3
- `/project`
- `/home`
- `/usr/local`
- `/db`

**Seagate Storage**
- Lustre Parallel File System
- `/lustre1`

**Transfer Node**
- `xfer.gacrcc.uga.edu`
- Build Node `build1.gacrcc.uga.edu`
- **AMD Opteron 16-core processor, 64 GB RAM**

**Compute Nodes**
- Node index: n0, n1, n2, ..., n165, n166, n167
- **CPU:** AMD Opteron / Intel Xeon multicore processors
- **GPU:** NVIDIA Tesla (Kepler) K40m GPU cards

**Login Node (VMs)**
- `sapelo1.gacrcc.uga.edu`
- **Intel Xeon processor**

---

User

Aggregator Switch

Storage Switch

Top of Rack Switch

Login Switch
Cluster Overview

Sapelo is a Linux high performance computing (HPC) cluster:

- OS: 64-bit CentOS Linux 6.5
- You can log on to:
  - Login (username@sapelo1.gacrc.uga.edu): edit script, submit batch job (not suggested)
  - Transfer (username@xfer.gacrc.uga.edu): transfer, compress, package data
- Login `qlogin` Work Node: edit script, submit batch job, run interactive job
- Internodal Communication: *InfiniBand network*

compute nodes ⇔ compute nodes ⇔ storage systems
Sapelo Cluster Overview

• Batch-queueing System:
  ➢ Jobs can be started (submitted), monitored, and controlled
  ➢ Determine which compute node is the best place to run a job
  ➢ Determine appropriate execution priority for a job to run

• On Sapelo:
  ➢ Torque Resource Manager + Moab Workload Manager
  ➢ Queueing commands:
    * `qsub`, `qstat_me`, `qstat`, `qdel`  
    * `showjobs`, `checkjob`, `showq`
## Computing Resources

<table>
<thead>
<tr>
<th>Queue</th>
<th>Node Feature</th>
<th>Total Nodes</th>
<th>RAM(GB) /Node</th>
<th>Max RAM(GB) /Single-node Job</th>
<th>Cores /Node</th>
<th>Processor</th>
<th>GPU</th>
<th>GPU Cards /Node</th>
<th>InfiniBand</th>
</tr>
</thead>
<tbody>
<tr>
<td>batch</td>
<td>AMD</td>
<td>112</td>
<td>128</td>
<td>126</td>
<td></td>
<td>AMD Opteron</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
<td>256</td>
<td>252</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HIGHMEM</td>
<td></td>
<td>6</td>
<td>512</td>
<td>504</td>
<td>48</td>
<td>AMD Opteron</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
<td>1024 (1)</td>
<td>997</td>
<td></td>
<td>Intel Xeon</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>N/A</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>GPU</td>
<td></td>
<td>2</td>
<td>128</td>
<td>126</td>
<td>16</td>
<td>Intel Xeon</td>
<td></td>
<td>NVIDIA K40m</td>
<td></td>
</tr>
<tr>
<td>abcnode (buy-in)</td>
<td></td>
<td>85</td>
<td>variable</td>
<td>variable</td>
<td>variable</td>
<td>variable</td>
<td></td>
<td>variable</td>
<td></td>
</tr>
</tbody>
</table>
Layer 1: Node
- A standalone “computer in a box”
- Multiple processors, e.g. 4, sharing memory
- Local disk storage, network interface, etc.
- Networked into a cluster

Layer 2: Processor
- A single computing component
- Multicore processor, e.g. 12 cores

Layer 3: Core
- Central processing unit (CPU) reading and executing instructions independently
- Each core is assigned to a software thread

Node 167

128 GB Shared RAM

I/O bus

Local Storage /tmp

48 cores and 128 GB RAM per node
Storage Environment

Mainly, there are 3 different storage locations

- **Home directory** ➔ /home/username
  - With a quota of ~100GB
  - Any directory on /home has snapshot backups

- **Global Scratch directory** ➔ /lustre1/username
  - “No quota limit”
  - No snapshot backup
  - User’s responsibility to clean up data

- **Group project Storage directory** ➔ /project/abclab/
  - Created for a lab, e.g., abclab
  - Temporary data parking for non-current active data
## Storage Environment

<table>
<thead>
<tr>
<th>4 Filesystems</th>
<th>Role</th>
<th>Quota</th>
<th>Accessible from</th>
<th>Intended Use</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>/home/username/</td>
<td>Home</td>
<td>100GB</td>
<td>sapelo1.gacrc.uga.edu (Login)</td>
<td>Static data being used frequently, e.g., scripts, local software</td>
<td>Snapshots</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Interactive nodes (Interactive)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>xfer.gacrc.uga.edu (Transfer)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>build1.gacrc.uga.edu (Build)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>compute nodes (Compute)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>/lustre1/username/</td>
<td>Global Scratch</td>
<td>No Limit</td>
<td>Interactive nodes (Interactive)</td>
<td>Current data being read from/written into by jobs running on cluster</td>
<td>User to clean up! Subject to deletion in 30 days*</td>
</tr>
<tr>
<td>tmp/lscratch/</td>
<td>Local Scratch</td>
<td>N/A</td>
<td>Individual compute node</td>
<td>Jobs with heavy disk I/O</td>
<td>User to clean up!</td>
</tr>
<tr>
<td>/project/abclab/</td>
<td>Storage</td>
<td>1TB</td>
<td>xfer.gacrc.uga.edu (Transfer)</td>
<td>Temporary data parking for non-current active data</td>
<td>Group sharing possible</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Note: /usr/local/apps : Software installation directory ; /db : Bioinformatics database installation directory</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>* denotes component or policy to be implemented in the future</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Storage Environment – Data Storing Rule

I/O Performance

Fast

/lustre1/username/ ➜ Current Data being used by jobs running on cluster, i.e., job data

/home/username/ ➜ Static Data being used frequently, e.g., scripts and software

Slow

/project/abclab/ ➜ Non-current Active Data to be analyzed in the future, e.g., 1 month

User Local Computer ➜ Final Data, e.g., final outputs and results
Storage Environment - Accessing Rule of 123

1. `/home/username`
   - `ssh`
   - `qlogin`

2. `/lustre1/username`
   - `cd`
   - `interactive`
   - `cd`
   - `Landing`

3. `/project/abclab`
   - `cd`
   - `Transfer`
   - `ssh`
# Storage Environment

<table>
<thead>
<tr>
<th>7 Main Functions</th>
<th>Related Filesystem</th>
<th>Related Node</th>
</tr>
</thead>
<tbody>
<tr>
<td>Login Landing</td>
<td>/home/username (Home) <em>(Always!)</em></td>
<td>Login or Transfer</td>
</tr>
<tr>
<td>Batch Job Submitting</td>
<td>/home/username (Home)</td>
<td>Login</td>
</tr>
<tr>
<td></td>
<td>/lustre1/username (Global Scratch) <em>(Suggested!)</em></td>
<td>Interactive</td>
</tr>
<tr>
<td>Interactive Job Running</td>
<td>/lustre1/username (Global Scratch) /home/username (Home)</td>
<td>Transfer</td>
</tr>
<tr>
<td>Data Transferring, Archiving, Compressing</td>
<td>/lustre1/username (Global Scratch) /home/username (Home)</td>
<td></td>
</tr>
<tr>
<td>Long-term Active Data Storing</td>
<td>/project/abclab (Project Storage)</td>
<td></td>
</tr>
<tr>
<td>Job Data Temporarily Storing</td>
<td>/lustre1/username (Global Scratch) /tmp/lscratch (Local Scratch)</td>
<td>Compute</td>
</tr>
</tbody>
</table>
Job Submission Workflow

Job Working Space is Global Scratch: /lustre1/username/

- Why?
  No quota limit + The I/O fastest filesystem

- How to access?
  From Interactive Node, cd /lustre1/username/

- What need to pay attention?
  Clean up! Non-current Active Data ➔ /project/abclab/  
  Non-active Data ➔ local storage  
  using Transfer Node
Job Submission Workflow

1. Linux/Mac user:
   ssh username@sapelo1.gacrc.uga.edu

   Windows user:

2. qlogin

3. cd /lustre1/username

4. mkdir ./workDir

5. cd ./workDir

6. Linux/Mac user:
   scp file username@xfer.gacrc.uga.edu:/lustre1/username/workDir

   Windows user:

7. nano ./sub.sh

   #PBS -S /bin/bash
   #PBS -q batch
   #PBS -N testBlast
   #PBS -l nodes=1:ppn=1:AMD
   #PBS -l mem=20gb
   #PBS -l walltime=48:00:00
   cd $PBS_O_WORKDIR
   module load ncbiblast+/2.2.29
   blastn [options] > outputfile

8. $ qsub sub.sh

Note: -r option of scp command will recursively copy a directory
Job Submission Workflow

1. Log on to Sapelo Login node: `ssh username@sapelol.gacrc.uga.edu`
2. From Sapelo Login node, transfer to Interactive node by issuing the command: `qlogin`
3. From Interactive node: Change directory to global scratch: `cd /lustre1/username`
4. Create a working subdirectory on global scratch: `mkdir ./workDir`
5. Change directory to `workDir`: `cd ./workDir`
6. Transfer data to `workDir` using `scp` or SSH File Transfer (with `tar` or `gzip`)
7. Make a Sapelo job submission script: `nano ./sub.sh`
8. Submit job: `qsub ./sub.sh`
Work on Sapelo

Before we start:

GACRC Wiki: http://wiki.gacrc.uga.edu

GACRC Wiki Running Jobs: https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sapelo

GACRC Wiki Software: https://wiki.gacrc.uga.edu/wiki/Software

GACRC Wiki Linux Command: https://wiki.gacrc.uga.edu/wiki/Command_List

GACRC Support: https://wiki.gacrc.uga.edu/wiki/Getting_Help
Working With Sapelo

➢ Start with the Cluster
➢ Connect and Login
➢ Transfer Files Using Transfer Node
➢ Software Packages
➢ Run Interactive Jobs
➢ Run Jobs
  ✓ How to submit a job
  ✓ Job submission scripts for *serial*, *threaded*, and *MPI* batch jobs
  ✓ How to check job status, and cancel a job
  ✓ How to check memory usage of a job
User Account

• User Account: UGAMyID@sapelo1.gacrc.uga.edu

A valid official UGA MyID is a MUST to create a user account!

• To get a user account:

4. Welcome letter with whole package of information about your Sapelo user account
Connect and Login

• On Linux/Mac: use Terminal utility and `ssh` to your account:

  ```
  ssh pakala@sapelol.gacrc.uga.edu
  or
  ssh -x pakala@sapelol.gacrc.uga.edu
  (\textsuperscript{1} \textit{X} is for \textit{X windows application} running on the cluster with its GUI to be forwarded to local
  \textsuperscript{2} On Windows, use a \textit{SSH client} to open the connection (next page))
  ```

• Logging in: You will be prompted for your UGA MyID password:

  ```
  pakala@sapelol.gacrc.uga.edu’s password:
  (\textsuperscript{3} On Linux/Mac, when you type in the password, the prompt blinks and does not move)
  ```

• Logging out: `exit` to leave the system:

  ```
  [pakala@75-104 ~]$ exit
  ```
Connect and Login

1. To download:
   http://eits.uga.edu/hardware_and_software/software/
   with your UGA MyID and password
2. After connection is built, working environment is
   Linux, same as Linux/Mac users’
Transfer Node xfer.gacrc.uga.edu

✓ ssh username@xfer.gacrc.uga.edu with your UGA MyID password
✓ Landing directory: /home/username (Sapelo home)
✓ Move data into/out of Sapelo (scp, sftp, rsync, SSH File Transfer, FileZilla, WinSCP)
✓ Compress or package data on Sapelo (tar, gzip)
✓ Transfer data between Sapelo and zcluster (cp, mv)
✓ Filesystems you can access using full path:
  ➢ /home/username/: Sapelo home directory (landing directory)
  ➢ /lustre1/username/: Sapelo global scratch directory
  ➢ /project/abclab/: long-term active data storage for group
  ➢ /panfs/pstor.storage/home/abclab/username/: zcluster home directory
  ➢ /escratch4/username/: zcluster global scratch directory
✓ Most file systems on Transfer are auto-mounted upon the first time full-path access, e.g.,
  cd /project/abclab/
Transfer Files Using Transfer Node xfer.gacrc.uga.edu

User’s local scp/sftp/rsync Transfer (xfer.gacrc.uga.edu)

• On Linux, Mac or cygwin on Windows: scp [Source] [Target]

E.g. 1: working on local machine, from Local ➔ Sapelo global scratch

```
scp ./file pakala@xfer.gacrc.uga.edu:/lustrel/pakala/suchi_Sept_14/
scp -r ./folder/ pakala@xfer.gacrc.uga.edu:/lustrel/pakala/suchi_Sept_14/
```

E.g. 2: working on local machine, from Sapelo global scratch ➔ Local

```
scp pakala@xfer.gacrc.uga.edu:/lustrel/pakala/suchi_Sept_14/file ./
scp -r pakala@xfer.gacrc.uga.edu:/lustrel/pakala/suchi_Sept_14/folder/ ./
```

• On Window: SSH Secure Shell File Transfer, FileZilla, WinSCP (next page)
SSH Secure Client: Connecting to Transfer

1. Quick Connect
2. Connect to Remote Host
   - Host Name: xlarge usage
   - User Name: nphala
   - Port Number: 22
   - Authentication Method: Profile Settings
3. Click Connect
4. Connect to Remote Host
SSH Secure Client: Connecting to Transfer Node
SSH Secure: Connected to Home Directory
SSH Secure : Navigated to /lustre1/pakala
Software Packages

• The cluster uses **environment modules** to define the various paths for software packages

• Current number of modules installed is ~**600** and expanding daily!

• `module avail` to list all available modules (centralized installation)
**Software Packages**

```bash
[pakala@75-104 ~]$ module avail
```

<table>
<thead>
<tr>
<th>package</th>
<th>version</th>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>fftw/3.3.4/pgi149-omp183</td>
<td>(D) mscanc0.8</td>
<td></td>
</tr>
<tr>
<td>find_cns/07212016</td>
<td></td>
<td></td>
</tr>
<tr>
<td>flash/1.2.11</td>
<td>(D) mscancx-transposed/0.8</td>
<td></td>
</tr>
<tr>
<td>flex/2.6.0/gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>flex/2.6.0/pgi/14.10</td>
<td>(D) megahit/latest</td>
<td></td>
</tr>
<tr>
<td>freebayes/1.0.1</td>
<td>(D) megahit/1.0.0</td>
<td></td>
</tr>
<tr>
<td>gam-nos/1.1b</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gamsess/5Dec2014-1node</td>
<td>(D) mega/6.4.3</td>
<td></td>
</tr>
<tr>
<td>gatk/latest</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gatk/3.3.0</td>
<td>(D) meme/4.11.0</td>
<td></td>
</tr>
<tr>
<td>gatk/3.4.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gatk/3.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gatk/3.6</td>
<td>(D) meraculous/2.0.5</td>
<td></td>
</tr>
<tr>
<td>gblocks/0.91b</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gblocks/0.91b</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.7.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc/4.8.0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**StdEnv**

<table>
<thead>
<tr>
<th>package</th>
<th>version</th>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>lmod/5.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>setarg/5.8</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Where:

(D): Default Module

Use "module spider" to find all possible modules.

Use "module keyword key1 key2 ..." to search for all possible modules matching any of the "keys".
Software Packages

- **module list** to list which modules currently loaded:

```
[pakala@75-104 ~]$ module list
Currently Loaded Modules:
  1) moab/8.1.1   2) StdEnv
```

- **module load** to load the needed modules:

```
[pakala@75-104 ~]$ module load ncbiblast/2.2.26
[pakala@75-104 ~]$ module load bamtools/2.4.0
[pakala@75-104 ~]$ module list
Currently Loaded Modules:
  1) moab/8.1.1   3) ncbiblast/2.2.26
  2) StdEnv       4) bamtools/2.4.0
```

- **module unload** to remove the specific module:

```
[pakala@75-104 ~]$ module unload bamtools/2.4.0
[pakala@75-104 ~]$ module list
Currently Loaded Modules:
  1) moab/8.1.1   2) StdEnv   3) ncbiblast/2.2.26
```
Run Interactive Jobs

• To run an interactive job, using `qlogin` command from Login node:

```
[pakala@75-104 ~]$ qlogin
qsub: waiting for job 1329698.pbs.scm to start
qsub: job 1329698.pbs.scm ready
[pakala@n15 ~]$ qsub
[pakala@n15 ~]$ pwd
/home/pakala
[pakala@n15 ~]$ cd /lustre1/pakala
[pakala@n15 pakala]$ module load ncbiblast+/2.2.29
[pakala@n15 pakala]$ module load bowtie/1.1.1
[pakala@n15 pakala]$ module list

Currently Loaded Modules:
1) moab/8.1.1  3) ncbiblast/2.2.26  5) ncbiblast+/2.2.29
2) StdEnv       4) bamtools/2.4.0   6) bowtie/1.1.1
```

• When you are done, remember to `exit` the session
Run Batch Jobs

• Components you need to run a job:
  ➢ Software already installed (cluster software or the one installed by yourself)
  ➢ Job submission script to
    1. specify computing resources:
       ✓ number of nodes and cores
       ✓ amount of memory
       ✓ node’s feature
       ✓ maximum wallclock time
    2. load software using `module load` (for cluster software)
    3. run any Linux commands you want to run, e.g., `pwd`, `mkdir`, `cd`, `echo`, etc.
    4. run the software

• Common queueing commands you need:
  ➢ `qsub`, `qstat`, `qstat_me`, `qdel`
  ➢ `showjobs`, `checkjob`, `showq`
Submit Batch Jobs

```
[pakala@n15 AF293]$ pwd
/lustre1/pakala/AF293
[pakala@n15 AF293]$
[pakala@n15 AF293]$
[pakala@n15 AF293]$

1412941.pbs.scm
```

-qsub is to submit a job

-sub.sh is the job submission script to

1. specify computing resources:
2. load software using module load
3. run any Linux commands you want to run
4. run the software
Job Submission Script

- **Example 1:** *Serial job script* `sub.sh` running NCBI Blast +

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>#PBS -S /bin/bash</code></td>
<td>Linux shell (bash)</td>
</tr>
<tr>
<td><code>#PBS -q batch</code></td>
<td>Queue name (batch)</td>
</tr>
<tr>
<td><code>#PBS -N testBlast</code></td>
<td>Name of the job (testBlast)</td>
</tr>
<tr>
<td><code>#PBS -l nodes=1:ppn=1:AMD</code></td>
<td>Number of nodes (1), number of cores/node (1), node type (AMD)</td>
</tr>
<tr>
<td><code>#PBS -l mem=20gb</code></td>
<td>Maximum amount of physical memory (20 GB) used by the job</td>
</tr>
<tr>
<td><code>#PBS -l walltime=48:00:00</code></td>
<td>Maximum wall clock time (48 hours) for the job, default 6 minutes</td>
</tr>
<tr>
<td><code>cd $PBS_O_WORKDIR</code></td>
<td>Use the directory from which the job is submitted as the working directory</td>
</tr>
<tr>
<td><code>module load ncbiblast+/2.2.29</code></td>
<td>Load the module of ncbiblast+, version 2.2.29</td>
</tr>
<tr>
<td><code>time blastn [options] &gt; outputfile 2&gt;&amp;1</code></td>
<td>Run blastn with ‘time’ command to measure the amount of time it takes to run the application</td>
</tr>
</tbody>
</table>
Job Submission Script

- Example 2: Threaded job script `sub.sh` running NCBI Blast+ with 24 threads

```bash
#PBS -S /bin/bash
#PBS -q batch
#PBS -N testBlast
#PBS -l nodes=1:ppn=24:AMD
#PBS -l mem=50gb
#PBS -l walltime=480:00:00
cd $PBS_O_WORKDIR

module load ncbiblast+/2.2.29

time blastn -num_threads 24 [options] > outputfile 2>&1
```

- Number of nodes (1), number of cores/node (24), node type (AMD)

  \[ \text{Number of threads (24)} = \text{Number of cores requested (24)} \]

- Email address to receive a notification for computing resources

- Send email notification when job aborts (a) or terminates (e)

- Standard error file (testBlast.e1234) will be merged into standard out file (testBlast.o1234)

- Run blastn with 24 threads (-num_threads 24)
Job Submission Script

• Example 3: **MPI job script** `sub.sh` running RAxML with **94** MPI processes

```bash
#PBS -S /bin/bash
#PBS -q batch
#PBS -N testRAxML
#PBS -l nodes=2:ppn=48:AMD  ➔ Number of nodes (2), number of cores/node (48), node type (AMD)
#PBS -l walltime=480:00:00  ➔ Total cores requested = 2 × 48 = 96
#PBS -l mem=20gb
#PBS -j oe

cd $PBS_O_WORKDIR

module load raxml/8.1.20
module load intel/15.0.2
module load openmpi/1.8.3/intel/15.0.2

mpirun –np 94 raxmlHPC-MPI-AVX [options] > outputfile ➔ Run raxmlHPC-MPI-AVX with 94 MPI processes (–np 94)
```

*We suggest, Number of MPI Processes (94) ≤ Number of cores requested (96)*
#PBS -S /bin/bash
#PBS -q batch
#PBS -N testRAxML
#PBS -l nodes=2:ppn=27:AMD
#PBS -l walltime=480:00:00
#PBS -l mem=20gb
#PBS -j oe

```bash
cd $PBS_O_WORKDIR

# Context Sharing
CONTEXTS=$(/usr/local/bin/set_contexts.sh $PBS_NUM_PPN)
if [[ "$?" -eq "0" ]]; then
    export PSM_SHAREDCONTEXTS_MAX=$CONTEXTS
fi

module load raxml/8.1.20
module load intel/15.0.2
module load openmpi/1.8.3/intel/15.0.2

mpirun -np 50 raxmlHPC-MPI-AVX [options] > outputfile
```

⇒ ppn number (27) fewer than 48 MUST be a multiplier of 3!

New lines copied from GACRC Wiki

⇒ Run raxmlHPC-MPI-AVX with 50 MPI processes (–np 50)
Check Job Status (**qstat_me**) and Cancel a Job (**qdel**)

**Job status:**
- **R**: job is running
- **C**: job completed (or canceled or crashed) and is not longer running. Jobs stay in this state for 1h
- **Q**: job is pending, waiting for resources to become available

```
$ qstat_me
<table>
<thead>
<tr>
<th>Job ID</th>
<th>Name</th>
<th>User</th>
<th>Time Use</th>
<th>S</th>
<th>Queue</th>
</tr>
</thead>
<tbody>
<tr>
<td>481929.pbs</td>
<td>testJob1</td>
<td>jSmith</td>
<td>900:58:0</td>
<td>C</td>
<td>batch</td>
</tr>
<tr>
<td>481931.pbs</td>
<td>testJob2</td>
<td>jSmith</td>
<td>04:00:03</td>
<td>R</td>
<td>batch</td>
</tr>
<tr>
<td>481934.pbs</td>
<td>testJob3</td>
<td>jSmith</td>
<td></td>
<td></td>
<td>Q</td>
</tr>
</tbody>
</table>
```

```
$ qdel 481934
$ qstat
<table>
<thead>
<tr>
<th>Job ID</th>
<th>Name</th>
<th>User</th>
<th>Time Use</th>
<th>S</th>
<th>Queue</th>
</tr>
</thead>
<tbody>
<tr>
<td>481929.pbs</td>
<td>testJob1</td>
<td>jSmith</td>
<td>900:58:0</td>
<td>C</td>
<td>batch</td>
</tr>
<tr>
<td>481931.pbs</td>
<td>testJob2</td>
<td>jSmith</td>
<td>04:00:03</td>
<td>R</td>
<td>batch</td>
</tr>
</tbody>
</table>
| 481934.pbs| testJob3| jSmith |          |   | C     | ← Stay on list 1 hr
```

Stay on list 1 hr
Check All Jobs on Cluster (\texttt{qstat})

\begin{verbatim}
$ qstat
Job ID    Name                User               Time Use S Queue
-------------------------------------------------------------------------------
2392787.pbs           G_s-Butyl_0_opt     gtp72866           101:34:5 R batch
2392792.pbs           G_s-Butyl_20_opt    gtp72866           56:14:32 R batch
2393599.pbs           ...80-oz0.6-K220     rbchan             89:99:59 R batch
2395891.pbs           NoRiver_jul_2008    hluo76                    0 Q batch
2395893.pbs           NoRiver_jul_2012    hluo76                    0 Q batch
2396261.pbs           ...opyl_freq_005     gtp72866           31:55:56 R batch
2396297.pbs           ...opyl_freq_041     gtp72866           202:53:1 R batch
2400998.pbs           ...rrect_lab4.sh     xwwang              844:06:3 R batch
2406268.pbs           abundTrinity3       mpodio            0 Q batch
2423833.pbs           L80-307              jx57780          393:37:1 C batch
2423838.pbs           L80-312              jx57780          373:57:4 C batch
2513741.pbs           ...3012_ah100_00     byang              00:00:00 R batch
2513743.pbs           ...4012_ah070_01     byang              138:22:3 R batch
.
.
.
\end{verbatim}

Note: qstat command will give you a long list of all jobs from all users on cluster!
Check Jobs on Which Nodes (*qstat* -n -u MyID)

```
$ qstat -n -u mheiges
pbs.scm:

<table>
<thead>
<tr>
<th>Job ID</th>
<th>Username</th>
<th>Queue</th>
<th>Jobname</th>
<th>SessID</th>
<th>NDS</th>
<th>TSK</th>
<th>Req'd Memory</th>
<th>Req'd Time</th>
<th>S</th>
<th>Elap Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>2554422.pbs.scm</td>
<td>mheiges</td>
<td>batch</td>
<td>STDIN</td>
<td>395315</td>
<td>1</td>
<td>1</td>
<td></td>
<td>480:00:00</td>
<td>R</td>
<td>00:01:16</td>
</tr>
<tr>
<td>2554425.pbs.scm</td>
<td>mheiges</td>
<td>batch</td>
<td>STDIN</td>
<td>395393</td>
<td>1</td>
<td>1</td>
<td></td>
<td>480:00:00</td>
<td>R</td>
<td>00:01:16</td>
</tr>
<tr>
<td>2554432.pbs.scm</td>
<td>mheiges</td>
<td>batch</td>
<td>STDIN</td>
<td>395497</td>
<td>1</td>
<td>1</td>
<td>8gb</td>
<td>240:00:00</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>2554436.pbs.scm</td>
<td>mheiges</td>
<td>batch</td>
<td>STDIN</td>
<td>395690</td>
<td>1</td>
<td>1</td>
<td></td>
<td>480:00:00</td>
<td>R</td>
<td>00:01:15</td>
</tr>
<tr>
<td>2554467.pbs.scm</td>
<td>mheiges</td>
<td>batch</td>
<td>DistribJob</td>
<td>404662</td>
<td>1</td>
<td>1</td>
<td>8gb</td>
<td>240:00:00</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>2554468.pbs.scm</td>
<td>mheiges</td>
<td>batch</td>
<td>DistribJob</td>
<td>404695</td>
<td>1</td>
<td>1</td>
<td>8gb</td>
<td>240:00:00</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>2554469.pbs.scm</td>
<td>mheiges</td>
<td>batch</td>
<td>DistribJob</td>
<td>405046</td>
<td>1</td>
<td>1</td>
<td>8gb</td>
<td>240:00:00</td>
<td>C</td>
<td></td>
</tr>
</tbody>
</table>

Note: “Elap Time” is the wall-clock time, instead of the CPU time, which *qstat_me* can give you!
```
Check Computing Resources Used by a Job

Option 1: `qstat -f JobID` for running jobs or finished jobs in 1 hour

Option 2: `showjobs JobID` for finished jobs over 1 hour, but ≤ 7 days

➢ “showjobs” command should be run in a “qlogin” session, not on the login node.

Option 3: Email notification from finished jobs (completed, canceled, or crashed),

if using: #PBS -M pakala@uga.edu

#PBS -m ae
qstat -f JobID for running jobs or finished jobs in 1 hour

```
qstat -f 2199353
Job Id: 2199353.pbs.scm
    Job_Name = Blastplus_2
    Job_Owner = pakala@n14
resources_used.cput = 04:51:53
resources_used.energy_used = 0
resources_used.mem = 13408044kb
resources_used.vmem = 13933700kb
resources_used.walltime = 02:59:32
    job_state = C
    queue = batch
...  
  ctime = Mon Aug 28 10:38:45 2017
  Error_Path = n14:/lustre1/pakala/AF293/Blastplus_2.e2199353
  Output_Path = n14:/lustre1/pakala/AF293/Blastplus_2.o2199353
  Priority = 0
  Resource_List.mem = 45gb
  Resource_List.nodect = 1
  Resource_List.nodes = 1:ppn=2:AMD
  Resource_List.walltime = 24:00:00
  session_id = 105520
  Shell_Path_List = /bin/bash
```
showjobs  JobID  for finished jobs over 1 hour, but ≤ 7 days

$ showjobs 2189688
Job Id      : 2189688.pbs.scm
Job Name    : Blastplus_8
Output File : n14:/lustre1/pakala/AF293/Blastplus_8.o2189688
Error File  : n14:/lustre1/pakala/AF293/Blastplus_8.e2189688
Working Directory : /lustre1/pakala/AF293
Home Directory : /home/pakala
Submit Arguments : Blastplus_Trail_3.sh
User Name    : pakala
Group Name   : gacrc-instruction
Queue Name   : batch
Wallclock Limit : 1:00:00:00
Wallclock Duration : 00:44:02
CPUPTime     : 03:19:17
Memory Used  : 12.7Gb
Memory Limit : 45gb
vmem Used    : 14.6Gb
Submit Time  : Fri Aug 25 15:02:53 2017
End Time     : Fri Aug 25 16:58:35 2017
Exit Code    : 0
Master Host  : n98
Email notification from finished jobs

PBS Job Id: 700009.pbs.scm
Job Name: testJob
Exec host: n1/4-27
Execution terminated
Exit_status=0
resources_used.cput=00:05:12
resources_used.energy_used=0
resources_used.mem=410984kb
resources_used.vmem=6548516kb
resources_used.walltime=04:00:59
Error_Path: uga-2f0f976.scm:/home/zhuofei/MPIs/testJob.o700009
Output_Path: uga-2f0f976.scm:/home/zhuofei/MPIs/testJob.o700009

PBS Job Id: 700097.pbs.scm
Job Name: testJob
Exec host: n5/4-27
Execution terminated
Exit_status=271
resources_used.cput=00:11:22
resources_used.energy_used=0
resources_used.mem=412304kb
resources_used.vmem=6548524kb
resources_used.walltime=05:00:41
Error_Path: uga-2f0f976.scm:/home/zhuofei/MPIs/testJob.o700097
Output_Path: uga-2f0f976.scm:/home/zhuofei/MPIs/testJob.o700097
Check Queue Status

[pakala@n15 for_suman]$ showq

<table>
<thead>
<tr>
<th>JOBID</th>
<th>USERNAME</th>
<th>STATE</th>
<th>PROCs</th>
<th>REMAINING</th>
<th>STARTTIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>1412927</td>
<td>gent</td>
<td>Running</td>
<td>4</td>
<td>19:22:58:29</td>
<td>Tue Mar 7 13:35:02</td>
</tr>
<tr>
<td>1412944</td>
<td>pakala</td>
<td>Running</td>
<td>1</td>
<td>19:23:37:33</td>
<td>Tue Mar 7 14:14:06</td>
</tr>
<tr>
<td>1412945</td>
<td>j103308</td>
<td>Running</td>
<td>4</td>
<td>19:23:38:19</td>
<td>Tue Mar 7 14:14:52</td>
</tr>
</tbody>
</table>

370 active jobs
3406 of 7472 processors in use by local jobs (57.63%)
114 of 170 nodes active (67.06%)

<table>
<thead>
<tr>
<th>JOBID</th>
<th>USERNAME</th>
<th>STATE</th>
<th>PROCs</th>
<th>WCLIMIT</th>
<th>QUEUETIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>1410619</td>
<td>rzzmh</td>
<td>Idle</td>
<td>12</td>
<td>30:00:00:00</td>
<td>Thu Mar 2 17:02:57</td>
</tr>
<tr>
<td>1411044</td>
<td>apjaeger</td>
<td>Idle</td>
<td>42</td>
<td>28:00:00:00</td>
<td>Fri Mar 3 13:30:32</td>
</tr>
<tr>
<td>1412702</td>
<td>mesbahi</td>
<td>Idle</td>
<td>24</td>
<td>41:04:00:00</td>
<td>Tue Mar 7 10:08:01</td>
</tr>
</tbody>
</table>

81 eligible jobs

<table>
<thead>
<tr>
<th>JOBID</th>
<th>USERNAME</th>
<th>STATE</th>
<th>PROCs</th>
<th>WCLIMIT</th>
<th>QUEUETIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>1409386</td>
<td>lferreri</td>
<td>BatchHold</td>
<td>1</td>
<td>20:00:00:00</td>
<td>Tue Feb 28 13:16:59</td>
</tr>
</tbody>
</table>

1 blocked job

Total jobs: 452
Guideline Tips

• Do NOT use Login Node to run jobs ➔ Interactive Node or the queue
• Do NOT use Login Node upload or download data to/from cluster
• Do NOT use Login Node to transfer data to the home dir

• NO large memory job running on AMD nodes ➔ HIGHMEM nodes
• NO small memory job running on HIGHMEM nodes ➔ AMD nodes
• As a general rule, threads # = cores # requested
Practical Tips

- Each directory should not have too many files inside! A rule of thumb would be to try to keep no more than a few tens of thousands of files (<10000 would be even better) in any single directory which is accessed frequently.

All files are in ONE single dir! ❌

Files are organized in subdirs! ✓
Practical Tips

- Job name should have a specific computational meaning

  Good Examples:  
  
  ```bash
  #PBS -N blastn_dataSet1_trail2 ; #PBS -N M-10-1121
  ```

  Bad Examples:  
  
  ```bash
  #PBS -N job1 ; #PBS -N bowtie ; #PBS -N 20160930
  ```

- The stdout .o file and stderr .e file are to be written into files at the finishing time of a job.

  Redirect standard output and error of the application to a file, instead of letting it be written in the stdout .o file and stderr .e file of the job, e.g.:

  ```bash
  time application >file 2>&1
  ```

- Monitor job progress from time to time, to catch if a job gets stuck
How to Submit Tickets to GACRC

• For Installation/Downloading Software:
  ➢ User needs to provide the name, version (or latest), and website
  ➢ Applications need to be compatible with Linux
  ➢ Note – only FREE software will be installed

• For Troubleshooting:
  ➢ List the path of the working directory, path of the script that is producing errors, Job ID, and the command sent to the queue or interactive node
  ➢ No need to attach the script or huge error messages

• For Testing:
  ➢ Please have a sample dataset at your working directory, so that it can be used for debugging

• These steps will help us in responding quickly and efficiently
Useful Links

- GACRC Wiki: https://wiki.gacrc.uga.edu/wiki/Main_Page
- GACRC Software: https://wiki.gacrc.uga.edu/wiki/Software
- GACRC Support: http://gacrc.uga.edu/help/
- GACRC Training: https://wiki.gacrc.uga.edu/wiki/Training
THANK YOU for your patience 😊