Introduction to HPC Using zcluster at GACRC

Georgia Advanced Computing Resource Center
University of Georgia
Zhuofei Hou, HPC Trainer
zhuofei@uga.edu
Outline

• What is GACRC?
• What is HPC Concept?
• What is zcluster?
• How does zcluster operate?
• How to work with zcluster?
What is GACRC?

Who Are We?
- Georgia Advanced Computing Resource Center
- Collaboration between the Office of Vice President for Research (OVPR) and the Office of the Vice President for Information Technology (OVPIT)
- Guided by a faculty advisory committee (GACRC-AC)

Why Are We Here?
- To provide computing hardware and network infrastructure in support of high-performance computing (HPC) at UGA

Where Are We?
- [http://gacrc.uga.edu](http://gacrc.uga.edu) (Web)  [http://wiki.gacrc.uga.edu](http://wiki.gacrc.uga.edu) (Wiki)
- [http://gacrc.uga.edu/help/](http://gacrc.uga.edu/help/) (Web Help)
- [https://wiki.gacrc.uga.edu/wiki/Getting_Help](https://wiki.gacrc.uga.edu/wiki/Getting_Help) (Wiki Help)
<table>
<thead>
<tr>
<th>Colleges &amp; Schools</th>
<th>Depts</th>
<th>PIs</th>
<th>Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>Franklin College of Arts and Sciences</td>
<td>14</td>
<td>117</td>
<td>661</td>
</tr>
<tr>
<td>College of Agricultural &amp; Environmental Sciences</td>
<td>9</td>
<td>29</td>
<td>128</td>
</tr>
<tr>
<td>College of Engineering</td>
<td>1</td>
<td>12</td>
<td>33</td>
</tr>
<tr>
<td>School of Forestry &amp; Natural Resources</td>
<td>1</td>
<td>12</td>
<td>31</td>
</tr>
<tr>
<td>College of Veterinary Medicine</td>
<td>4</td>
<td>12</td>
<td>29</td>
</tr>
<tr>
<td>College of Public Health</td>
<td>2</td>
<td>8</td>
<td>28</td>
</tr>
<tr>
<td>College of Education</td>
<td>2</td>
<td>5</td>
<td>20</td>
</tr>
<tr>
<td>Terry College of Business</td>
<td>3</td>
<td>8</td>
<td>22</td>
</tr>
<tr>
<td>School of Ecology</td>
<td>1</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>School of Public and International Affairs</td>
<td>1</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>College of Pharmacy</td>
<td>2</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td><strong>Centers &amp; Institutes</strong></td>
<td>9</td>
<td>19</td>
<td>59</td>
</tr>
<tr>
<td><strong>TOTALS:</strong></td>
<td><strong>49</strong></td>
<td><strong>233</strong></td>
<td><strong>1029</strong></td>
</tr>
<tr>
<td>Centers &amp; Institutes</td>
<td>PIs</td>
<td>Users</td>
<td></td>
</tr>
<tr>
<td>----------------------------------------------------------</td>
<td>-----</td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td>Center for Applied Isotope Study</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Center for Computational Quantum Chemistry</td>
<td>3</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Complex Carbohydrate Research Center</td>
<td>6</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>Georgia Genomics Facility</td>
<td>1</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Institute of Bioinformatics</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Savannah River Ecology Laboratory</td>
<td>3</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>Skidaway Institute of Oceanography</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Center for Family Research</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Carl Vinson Institute of Government</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>19</strong></td>
<td><strong>59</strong></td>
<td></td>
</tr>
</tbody>
</table>
Concept of High Performance Computing (HPC)

Serial:
- Serial problem cannot be broken
- Discrete instructions executed sequentially
- Only 1 instruction executed at any moment on a single processor

Parallel:
- Problem broken into parallel parts can be solved concurrently
- Instructions executed simultaneously on multiple processors
- Synchronization/communication employed
- Shared-memory multithreaded job or MPI job (Message Passing Interface)
What is zcluster?

- Cluster Structural Diagram
- Cluster Overview
- Computing Resources
- Storage Environment
GACRC Linux HPC Zcluster Structural Diagram

Zcluster Connection Speed:
- 10 Gbit Ethernet
- 1 Gbit Ethernet

Aggregator Switch

- Icebreaker Storage (sn3)
- Icebreaker Storage (sn4)
- Panasas Storage
  - /home
  - /usr/local
  - /db
- Transfer Nodes xfer.gacr.cuga.edu
- Compute Nodes on Rack 1-11
  - Node index: c1-1, c1-2, ...., c11-1, c11-2, ....
  - CPU: Intel Xeon/AMD Opteron multicore
  - GPU: NVIDIA Tesla (Fermi/Kepler)
  - /iscratch

Login Node zcluster.rcc.uga.edu

User

World
What is zcluster – Cluster Overview

GACRC zcluster is a Linux high-performance computing (HPC) cluster:

• OS: 64-bit Red Hat Enterprise Linux 5 (RHEL 5)

• Login Node: zcluster.rcc.uga.edu → Interactive Node: compute-14-7/9
  Transfer Node: xfer.gacrc.uga.edu

• Internodal Communication: 1Gbit network
  compute nodes ↔ compute nodes
  compute nodes ↔ storage systems
What is zcluster – 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 zcluster:
  ➢ Sun Grid Engine (SGE)
  ➢ Queueing commands: `qsub`, `qstat`, `qdel`, `qsj`, `qacct`
What is zcluster – Computing Resources

<table>
<thead>
<tr>
<th>Queue</th>
<th>Queue Name</th>
<th>Total Nodes</th>
<th>Cores/Node</th>
<th>Max Threads</th>
<th>RAM(GB)/Node</th>
<th>Processor</th>
<th>NVIDIA GPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regular</td>
<td>rcc-30d</td>
<td>45</td>
<td>12</td>
<td>6</td>
<td>48</td>
<td>Intel Xeon</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td></td>
<td>150</td>
<td>8</td>
<td></td>
<td>16</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>High Memory</td>
<td>rcc-m128-30d</td>
<td>1</td>
<td>8</td>
<td>5</td>
<td>128</td>
<td>Intel Xeon</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>8</td>
<td></td>
<td>192</td>
<td></td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10</td>
<td>12</td>
<td></td>
<td>256</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>rcc-m512-30d</td>
<td>2</td>
<td>32</td>
<td>8</td>
<td>512</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Multi Core</td>
<td>rcc-mc-30d</td>
<td>4</td>
<td>32</td>
<td>32</td>
<td>64</td>
<td>AMD Opteron</td>
<td></td>
</tr>
<tr>
<td>Interactive</td>
<td>interq</td>
<td>2</td>
<td>48</td>
<td></td>
<td>132</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>GPU</td>
<td>rcc-sgpu-30d</td>
<td>2</td>
<td>8</td>
<td>N/A</td>
<td>48</td>
<td>Intel Xeon</td>
<td>4 Tesla S1070 cards</td>
</tr>
<tr>
<td></td>
<td>rcc-mgpu-30d</td>
<td>2</td>
<td>12</td>
<td></td>
<td>48</td>
<td>Intel Xeon</td>
<td>9 Tesla (Fermi) M2070 cards</td>
</tr>
<tr>
<td></td>
<td>rcc-kgpu-30d</td>
<td>2</td>
<td>12</td>
<td></td>
<td>96</td>
<td></td>
<td>32 Tesla (Kepler) K20Xm cards</td>
</tr>
</tbody>
</table>

**Total peak performance: 23 Tflops**
Layer 1: Node
- A standalone “computer in a box”
- Multiple processors, e.g. 2, sharing memory
- Local disk storage, network interface, etc.
- Networked into a cluster

Layer 2: Processor
- A single computing component
- Multicore processor, e.g. 4 cores

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

rcc-30d 16GB Node

16 GB Shared RAM

8 cores and 16 GB RAM per node

Local Storage /Iscratch

I/O Bus
What is zcluster – Storage Environment

• Home directory ➔ /home/groupname/username/
  ➢ Mounted and visible on all nodes, with a quota of ~100GB
  ➢ Any directory on /home has snapshot backups
    ➢ /home/abclab/jsmith/.snapshot
    ➢ Completely invisible, however, user can “cd” into it and then “ls”:

```
zhuofei@zcluster:~$ ls -a
.
..  .bash_profile  .emacs.d  .fontconfig  .maple_history  MPLs  scripts  test.sh
  .bashrc  .ENV_file  .gnuplot_history  .Mathematica  openMPs  serializers  .viminfo
  .bash_history  downloads  exe  .history  .mc  .profile  sht  .Xauthority
  .bash_logout  .emacs  .flexlmrc  .lesshst  .mozilla  Pthreads  .ssh  ➔ .snapshot is NOT shown here!
zhuofei@zcluster:~$ cd .snapshot ➔ can “cd” into .snapshot
zhuofei@zcluster:~/.snapshot$ ls ➔ then “ls” to list its contents
2015.06.21.00.00.01.weekly  2015.06.27.01.00.01.daily  2015.06.28.01.00.01.daily  2015.06.30.01.00.01.daily
2015.06.26.01.00.01.daily  2015.06.28.00.00.01.weekly  2015.06.29.01.00.01.daily
```

What is zcluster – Storage Environment

• Local scratch ➔ /lscratch/username/
  ➢ On local disk of each compute node ➔ node-local storage
  ➢ rcc-30d 8-core nodes: ~18GB, rcc-30d 12-core nodes: ~370GB
  ➢ No snapshot backup
  ➢ Usage Suggestion: If your job writes results to /lscratch, job submission script should move the data to your home or escratch before exit

• Ephemeral Scratch ➔ /escratch4/zhuofei/zhuofei_Jul_01/
  ➢ Use make_escratch from Login to create working subdirectory .../username_mmm_dd/
  ➢ Accessible from Login, Transfer, Interactive, and Compute nodes
  ➢ Each user 4TB quota, No snapshot backup!
  ➢ To be deleted after 37 days
## What is zcluster – Storage Environment

<table>
<thead>
<tr>
<th>Filesystem</th>
<th>Role</th>
<th>Quota</th>
<th>Accessible from</th>
<th>Intended Use</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>/home/abclab/username/</td>
<td>Home</td>
<td>100GB</td>
<td>zcluster.rcc.uga.edu (Login)</td>
<td>Highly static data being used frequently</td>
<td>Snapshots</td>
</tr>
<tr>
<td>/escratch4/username/username_mmm_dd/</td>
<td>Scratch</td>
<td>4TB</td>
<td>xfer.gacrc.uga.edu (Transfer) Interactive nodes (Interactive) compute nodes (Compute)</td>
<td>Temporarily storing large data being used by jobs</td>
<td>make_escratch to create daily; Auto deleted in 37 days!</td>
</tr>
<tr>
<td>/lscratch/username/</td>
<td>Local Scratch</td>
<td>18 ~ 370GB</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>Variable</td>
<td>xfer.gacrc.uga.edu (Transfer)</td>
<td>Long-term data storage</td>
<td>Group sharing possible</td>
</tr>
</tbody>
</table>

Note:  
1. /usr/local : Software installation directory  
   /db : bioinformatics database installation directory  
2. use qlogin from Login node to log on Interactive node
## What is zcluster – Storage Environment

<table>
<thead>
<tr>
<th>6 Main Functions</th>
<th>On/From Node</th>
<th>Related Filesystem</th>
</tr>
</thead>
<tbody>
<tr>
<td>Login Landing</td>
<td>Login</td>
<td>/home/abclab/username/ (Home)</td>
</tr>
<tr>
<td></td>
<td>Transfer</td>
<td>/home/username/ (Transfer Home) *</td>
</tr>
<tr>
<td>Batch Job Submitting</td>
<td>Login or Interactive</td>
<td>/escratch4/username/username_mmm_dd/ (Scratch) (Suggested!)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/home/abclab/username/ (Home)</td>
</tr>
<tr>
<td>Interactive Job Running</td>
<td>Interactive</td>
<td>/escratch4/username/username_mmm_dd/ (Scratch)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/home/abclab/username/ (Home)</td>
</tr>
<tr>
<td>Data Transferring, Archiving, and</td>
<td>Transfer</td>
<td>/escratch4/username/username_mmm_dd/ (Scratch)</td>
</tr>
<tr>
<td>Compressing</td>
<td></td>
<td>/panfs/pstor.storage/home/abclab/username/ (Home) *</td>
</tr>
<tr>
<td>Job Data Temporarily Storing</td>
<td>Compute</td>
<td>/escratch4/username/username_mmm_dd/ (Scratch)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/lscratch/username/ (Local Scratch)</td>
</tr>
<tr>
<td>Long-term Active Data Storing</td>
<td>Login or Transfer</td>
<td>/project/abclab/</td>
</tr>
</tbody>
</table>
How does zcluster operate?
GACRC zcluster Operational Diagram

1 Gbit Ethernet

**Login Node**

ssh username@zcluster.rcc.uga.edu with zcluster password
(for script editing, batch job submission)

**Transfer Node**

ssh username@xfer.gacr.c.uga.edu with UGA MyID password
(for data transferring, compression, and packaging)
Workflow 1 – Home as Job Working Space

1. Linux/Mac user:
   `ssh userID@zcluster.rcc.uga.edu`

   Windows user:

2. `mkdir ./workDir`

3. `cd ./workDir`

4. Linux/Mac user:
   `scp input userID@copy.rcc.uga.edu:~/workDir`

   Windows user:

5. `nano ./sub.sh`

   ```bash
   #!/bin/bash
   cd `pwd`
   /usr/local/samtools/latest/samtools...
   ```

   Login

6. `$ qsub -q rcc-30d sub.sh`

Other qsub options:
- `-l mem_total=20`: use 48GB rcc-30d nodes
- `-pe thread 4`: request 4 cores for 4 threads

Note: `inputDate` could be files or data folder (`scp -r`)
Workflow 1 – Home as Job Working Space

1. Log on to zcluster Login node: ssh userID@zcluster.rcc.uga.edu
2. Create a working subdirectory in home: mkdir ./workDir
3. Change directory to workDir: cd ./workDir
4. Transfer data to workDir using scp or SSH Client File Transfer (with tar or gzip)
5. Make a zcluster job submission script: nano ./sub.sh
6. Submit job: qsub -q rcc-30d ./sub.sh

Useful qsub options: -l mem_total=20g : use 48GB high-RAM rcc-30d nodes
- pe thread 4 : request 4 cores for 4 threads, max 6 is suggested!
Workflow 2 – Global Scratch as Job Working Space

1. Linux/Mac user:
   `ssh userID@zcluster.rcc.uga.edu`

   Windows user:

2. `make_escratch`

3. `cd /escratch4/userID/userID_Jul_01`

4. Linux/Mac user:
   `scp input(userID@copy.rcc.uga.edu:/escratch4/userID/userID_Jul_01`

   Windows user:

5. `nano ./sub.sh`

   `#!/bin/bash`
   `cd `pwd``
   `/usr/local/samtools/latest/samtools...`

6. `$ qsub -q rcc-30d sub.sh`

Other qsub options:
- `-l mem_total=20g`: use 48GB rcc-30d nodes
- `-pe thread 4`: request 4 cores for 4 threads

Note: inputDate could be files or data folder (`scp -r`)
Workflow 2 – Global Scratch as Job Working Space

1. Log on to zcluster Login node: `ssh userID@zcluster.rcc.uga.edu`
2. Create a working subdirectory on global scratch: `make_escratch`
3. Change directory to `userID_Jul_01`: `cd /escratch4/userID/userID_Jul_01`
4. Transfer data to `userID_Jul_01` using `scp` or `SSH Client File Transfer` (with `tar` or `gzip`)
5. Make a zcluster job submission script: `nano ./sub.sh`
6. Submit job: `qsub -q rcc-30d ./sub.sh`

Useful qsub options: `-l mem_total=20g` : use 48GB high-RAM rcc-30d nodes

   `-pe thread 4` : request 4 cores for 4 threads, max 6 is suggested!
How to work with zcluster?

Before we start:

• To get zcluster to be your best HPC buddy

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

GACRC Support: https://wiki.gacrc.uga.edu/wiki/Getting_Help
How to work with zcluster?

To submit a ticket to us?

- **Job Troubleshooting:**
  - Please tell us details of your question or problem, including but not limited to:
    - Your user name
    - Your job ID
    - Your working directory
    - The queue name and command you used to submit the job

- **Software Installation:**
  - Specific name and version of the software
  - Download website
  - Supporting package information if have

Note:
It’s USER’s responsibility to make sure the correctness of datasets being used by jobs!
How to work with zcluster?

• You are not alone on cluster... Each user is sharing finite computing resources, e.g., CPU cycles, RAM, disk storage, network bandwidth, with other researchers:

    \textit{What you do may affect others on the cluster}

- Do NOT run jobs on login node \implies use the queues or the interactive nodes
- Do NOT use login node to move data into/out of cluster \implies use Transfer xfer.gacrc.uga.edu
- NO multi-threaded job running with only 1 core requested \implies threads \# = cores \# requested
- NO large memory job running on regular nodes \implies HIGHMEM queue
- NO long job running on interactive node \implies 12 hours
- NO small memory job running on large memory nodes \implies Saving memory for others
How to work with zcluser?

- Start with zcluser
- Connect and Login
- Transfer Files Using Transfer Node
- Software Installed
- Run Interactive Jobs
- Submit Batch Jobs
  - How to submit *serial*, *threaded*, and *MPI* batch jobs; useful qsub options
  - How to check job status, cancel a job
  - How to check memory usage of a job
Start with zcluster

- You need a User Account: username@zcluster.rcc.uga.edu
- Procedure: https://wiki.gacrc.uga.edu/wiki/User_Accounts
  - A UGA faculty member (PI) may register a computing lab: http://help.gacrc.uga.edu/labAcct.php
  - The PI of a computing lab may request user accounts for group members: http://help.gacrc.uga.edu/userAcct.php
- User receives a welcome email once the account is ready
- User uses passwd to change initial temporary password to a permanent one upon the first time of login
Connect and Login

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

```
ssh zhuofei@zcluster.rcc.uga.edu
```

or

```
ssh -X zhuofei@zcluster.rcc.uga.edu
```

(\(^1\) - `X` is for *X windows application* running on the cluster with its UGI to be forwarded to local

(\(^2\) On Windows, use a *SSH client* to open the connection (next page))

• Logging in: You will be prompted for your `zcluster password`:

```
zhuofei@zcluster.rcc.uga.edu’s password: □□
```

(\(^3\) On Linux/Mac, when you type in the password, the prompt blinks and does not move)

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

```
zhuofei@zcluster:~$ 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 Files Using Transfer Node xfer.gacrc.uga.edu

✓ `ssh username@xfer.gacrc.uga.edu` with your **UGA MyID password**

✓ Landing directory: `/home/username`

✓ Move data into/out of zcluster (`scp`, `sftp`, `rsync`, **SSH Secure Shell File Transfer, FileZilla**)

✓ Compress or package data on zcluster (`tar`, `gzip`)

✓ Transfer data between zcluster and Sapelo (`cp`, `mv`)

✓ Filesystems you can access:
  - `/home/username/` : Transfer home (landing spot)
  - `/panfs/pstor.storage/home/abclab/username/` : zcluster home
  - `/escratch4/username/` : zcluster scratch
  - `/project/abclab/` : long-term active data storage

✓ 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 \[\rightarrow\] scp/sftp/rsync \[\rightarrow\] Transfer (xfer.gacrc.uga.edu)

• On Linux, Mac or cygwin on Windows : \texttt{scp (-r) [Source] [Target]}

\textit{E.g. 1}: working on local machine, from Local \(\rightarrow\) zcluster global scratch

\begin{verbatim}
scp ./file zhuofei@xfer.gacrc.uga.edu:/escratch4/zhuofei/zhuofei_Jul_1/
scp -r ./folder/ zhuofei@xfer.gacrc.uga.edu:/escratch4/zhuofei/zhuofei_Jul_1/
\end{verbatim}

\textit{E.g. 2}: working on local machine, from zcluster global scratch \(\rightarrow\) Local

\begin{verbatim}
scp zhuofei@xfer.gacrc.uga.edu:/escratch4/zhuofei/zhuofei_Jul_1/file ./
scp -r zhuofei@xfer.gacrc.uga.edu:/escratch4/zhuofei/zhuofei_Jul_1/folder/ ./
\end{verbatim}

• On Window: \texttt{SSH Secure Shell File Transfer, FileZilla, WinSCP} (next page)
Transfer Files Using Transfer Node xfer.gacrc.uga.edu

1. Quick Connect
2. Host Name: xfer.gacrc.uga.edu
3. User Name: chuser
4. Port Number: 22
5. Transfer node!
6. Transfer (xfer.gacrc.uga.edu)
7. Connected to xfer.gacrc.uga.edu - /resc/hres/zhuser/zhuser_jul_06

SSH client
Software Installed

- Perl, Python, Java, awk, sed, C/C++ and Fortran compilers
- Matlab, Maple, R
- Many Bioinformatics applications: NCBI Blast+, Velvet, Trinity, TopHat, MrBayes, SoapDeNovo, Samtools, RaxML, etc.
- RCCBatchBlast (RCCBatchBlastPlus) to distribute NCBI Blast (NCBI Blast+) searches to multiple nodes.
- Many Bioinformatics Databases: NCBI Blast, Pfam, uniprot, etc.
- For a complete list of applications installed: https://wiki.gacrc.uga.edu/wiki/Software
Run Interactive Jobs

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

```
zhuofei@zcluster:~$ qlogin
Your job 1391816 ("QLOGIN") has been submitted
waiting for interactive job to be scheduled ...
Your interactive job 1391816 has been successfully scheduled.
...
compute-14-7.local$
```

    ➡️ Now I am on compute-14-7, which is an interactive node

• Current maximum runtime is **12** hours

• When you are done, remember to **exit** the session!

• Detailed information, like interactive parallel job? Go to:

  https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_zcluster
Submit Batch Jobs

• Components you need to submit a batch job:
  ➢ **Software** already installed on zcluster
  ➢ **Job submission script** to run the software, and
    ✓ Specify working directory
    ✓ Export environment variables, e.g.,
      PATH (searching path for executables)
      LD_LIBRARY_PATH (searching paths for shared libraries)

• Common commands you need:
  ➢ `qsub` with specifying queue name, cores to be requested
  ➢ `qstat`, `qdel`
  ➢ `qsj`, `qacct`
Submit Batch *Serial* Job

- **Step 1**: Create a job submission script *st.sh* running Samtools:
  
  ```bash
  #!/bin/bash
  cd /escratch4/zhuofei/zhuofei_Feb_1
  time /usr/local/samtools/latest/samtools <command> [options]
  ```

  - Linux default shell (*bash*)
  - Specify and enter (*cd*) working directory (*/escratch4/zhuofei/zhuofei_Feb_1*)
  - Run samtools with *time* command to measure amount of time it takes to run the application

- **Step 2**: Submit *st.sh* to the queue:

  ```bash
  $ qsub -q rcc-30d st.sh
  ```

  OR

  ```bash
  $ qsub -q rcc-30d -l mem_total=20g st.sh
  ```

  - Submit a job to the queue *rcc-30d* with 16GB RAM/Node
  - OR
  - Submit a job to the queue *rcc-30d* with 48GB RAM/Node
Submit Batch **Threaded** Job

- **Step 1:** Create a job submission script `blastn.sh` running NCBI Blast +:

  ```bash
  #!/bin/bash
  cd /escratch4/zhuofei/zhuofei_Feb_1
  time /usr/local/ncbiblast+/latest/bin/blastn -num_threads 4 [options]  ➔ Run blastn with 4 threads
  ```

- **Step 2:** Submit `blastn.sh` to the queue:

  ```bash
  $ qsub -q rcc-30d -l mem_total=20g -pe thread 4 blastn.sh
  ```

  - to the queue rcc-30d with **48GB RAM/Node**
  - **4 cores requested**

  **Number of Threads = Number of Cores Requested**

  **Note:**
  Please use the `rcc-mc-30d` queue, if using threads **more than 8**!
Submit Batch **MPI** Job

**Step 1:** Create a job submission script *raxml.sh* running RAxML:

```bash
#!/bin/bash

cd /escratch4/zhuofei/zhuofei_Dec_25

export MPIRUN=/usr/local/mpich2/1.4.1p1/gcc 4.5.3/bin/mpirun ➔ Define and export environment variable *(MPIRUN)*

$MPIRUN -np $NSLOTS /usr/local/raxml/latest/raxmlHPC-MPI-SSE3 [options] ➔ Run RAxML with 20 MPI processes (*-np $NSLOTS*)

**Step 2:** Submit *raxml.sh* to the queue:

```
$ qsub -q rcc-30d -pe mpi 20 raxml.sh
```

20 cores requested, $NSLOTS will be assigned to 20 automatically, before the job submission script is interpreted.
Useful `qsub` Command Options

<table>
<thead>
<tr>
<th><code>qsub</code> options</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>-q queue_name</td>
<td>Defines the queue to run your job, e.g. -q rcc-30d</td>
</tr>
<tr>
<td>-l mem_total=20g</td>
<td>Request a compute node with <strong>at least</strong> 20GB total physical RAM installed</td>
</tr>
<tr>
<td>-pe thread 4</td>
<td>Request 4 cores for a threaded job with 4 threads; maximum of 6 on rcc-30d</td>
</tr>
<tr>
<td>-pe mpi 20</td>
<td>Request 20 cores for a MPI job with 20 MPI processes, maximum of <strong>75</strong> on rcc-30d</td>
</tr>
<tr>
<td>-cwd</td>
<td>Run in current working directory</td>
</tr>
<tr>
<td>-M <a href="mailto:MyID@uga.edu">MyID@uga.edu</a></td>
<td>Defines the email address to send an email notification</td>
</tr>
<tr>
<td>-m ea</td>
<td>Send an email notification when job ends or aborts</td>
</tr>
<tr>
<td>-N name</td>
<td>Defines the name of a job</td>
</tr>
</tbody>
</table>
Check and Cancel Jobs

- To check the status of your jobs: **qstat**

<table>
<thead>
<tr>
<th>Job-ID</th>
<th>Prior</th>
<th>Name</th>
<th>User</th>
<th>State</th>
<th>Submit/Start at</th>
<th>Queue</th>
<th>Slots</th>
<th>Ja-task-ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>9707321</td>
<td>0.50766</td>
<td>sub1.sh</td>
<td>jsmith</td>
<td>r</td>
<td>01/28/2016 13:39:23</td>
<td><a href="mailto:rcc-30d@compute-7-12.local">rcc-30d@compute-7-12.local</a></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>9707322</td>
<td>0.50383</td>
<td>sub2.sh</td>
<td>jsmith</td>
<td>Eqw</td>
<td>01/28/2016 13:39:23</td>
<td><a href="mailto:rcc-30d@compute-7-12.local">rcc-30d@compute-7-12.local</a></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>9707323</td>
<td>0.00000</td>
<td>sub3.sh</td>
<td>jsmith</td>
<td>qw</td>
<td>01/28/2016 13:39:28</td>
<td></td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

- To cancel your job with a JobID: **qdel**

<table>
<thead>
<tr>
<th>Job-ID</th>
<th>Prior</th>
<th>Name</th>
<th>User</th>
<th>State</th>
<th>Submit/Start at</th>
<th>Queue</th>
<th>Slots</th>
<th>Ja-task-ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>9707321</td>
<td>0.50766</td>
<td>sub1.sh</td>
<td>jsmith</td>
<td>r</td>
<td>01/28/2016 13:39:23</td>
<td><a href="mailto:rcc-30d@compute-7-12.local">rcc-30d@compute-7-12.local</a></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>9707323</td>
<td>0.00000</td>
<td>sub3.sh</td>
<td>jsmith</td>
<td>qw</td>
<td>01/28/2016 13:39:28</td>
<td></td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
Check Memory Usage

- For a running job: `qsj`

```bash
$ qsj 9707368
=============================================
job_number: 9707368
owner: s_110
cwd: /escratch4/s_110/s_110_Jan_28
hard_queue_list: rcc-30d
script_file: sub.sh
......
usage 1: cpu=00:01:27, mem=0.96498 GBs, io=0.00014, vmem=73.734M, maxvmem=75.734M
```

- For a finished jobs: `qacct`

```bash
$ qacct -j 970732
===============================
qname rcc-30d
hostname compute-7-12.local
jobname sub.sh
jobnumber 9707323
......
cpu 183.320
mem 2.021
io 0.000
maxvmem 6.530G
```

Total Memory
Thank You!