GACRC Sapelo2 Cluster New User Training

Georgia Advanced Computing Resource Center (GACRC)
Enterprise Information Technology Services (EITS)
The University of Georgia
Outline

• GACRC

• Sapelo2 Cluster
  ➢ Cluster Diagram
  ➢ Overview
  ➢ Five Directories
  ➢ Four Computational Queues
  ➢ Software Modules

• Batch Job Submission Workflow

• qlogin Commands for interactive sessions

• GACRC Wiki and User Support
GACRC

- A high-performance-computing (HPC) center at the UGA
- 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

Wiki: [http://wiki.gacrc.uga.edu](http://wiki.gacrc.uga.edu)
Web Site: [http://gacrc.uga.edu](http://gacrc.uga.edu)
Please Note: You need to connect to the UGA network using VPN when accessing from outside of the UGA main campus.

UGA VPN: https://eits.uga.edu/access_and_security/infosec/tools/vpn/
Two Nodes:
1. Login node for batch job workflow: MyID@sapelo2.gacrc.uga.edu
2. Transfer node for data transferring: MyID@xfer.gacrc.uga.edu

Five Directories:
1. **Home**: Login landing spot; 200GB quota; Backed-up
2. **Scratch**: High-speed storage for temp files needed for current jobs; NO quota; NOT backed-up
3. **Work**: High-speed storage for input files needed for repeated jobs; per group quota of 500GB and max 100,000 single files; NOT backed-up
4. **Project**: Temporary data parking; per group quota of 1TB; Backed-up (ONLY accessible from Transfer node!)
5. **Local Scratch**: Local storage on each individual compute node; 200GB quota; NOT backed-up

Four Computational Queues: batch, highmem_q, gpu_q, groupBuyin_q
## Five Directories

[https://wiki.gacrc.uga.edu/wiki/Disk_Storage](https://wiki.gacrc.uga.edu/wiki/Disk_Storage)

<table>
<thead>
<tr>
<th>Directory</th>
<th>Name</th>
<th>Quota</th>
<th>Accessible from</th>
<th>Intended Use</th>
<th>Backed-up</th>
<th>Important Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>/home/MyID</td>
<td>Home</td>
<td>200GB</td>
<td>Login, Transfer</td>
<td>Static data, e.g. 1. Scripts, source codes 2. Local software</td>
<td>Yes</td>
<td>Not for storing data of your jobs!</td>
</tr>
<tr>
<td>/scratch/MyID</td>
<td>Scratch</td>
<td>No Limit</td>
<td>Compute</td>
<td>Temporary files needed for current running jobs</td>
<td>No</td>
<td>Clean up when your job finished! Subject to “30-day purge” policy</td>
</tr>
<tr>
<td>/work/abclab</td>
<td>Work</td>
<td>500GB</td>
<td>Transfer</td>
<td>Input files needed for repeated jobs</td>
<td>No</td>
<td>Clean up any old data! Group sharing is possible</td>
</tr>
<tr>
<td>/project/abclab</td>
<td>Project</td>
<td>1TB</td>
<td>Transfer</td>
<td>Temporary data parking</td>
<td>Yes</td>
<td>Group sharing is possible</td>
</tr>
<tr>
<td>/lscratch</td>
<td>Local</td>
<td>200GB</td>
<td>Compute</td>
<td>Jobs with heavy disk I/O operations</td>
<td>No</td>
<td>Clean up when job exits from node! Data are persistent</td>
</tr>
</tbody>
</table>
Accessing Directories from Nodes

- /home/MyID
- /scratch/MyID
- /work/abclab

Login

Interactive

Transfer

Duo Authentication

User

Duo Authentication

Please Note:
Local scratch is accessible on each individual compute node; it is not shown here.
More about scratch file system "30-day purge" policy

https://wiki.gacrc.uga.edu/wiki/Disk_Storage#Scratch_file_system

Any file that is not accessed or modified by a compute job in a time period no longer than 30 days will be automatically deleted off the /scratch file system.

Measures circumventing this policy will be monitored and actively discouraged.

- You have a list of those purgeable files located at /usr/local/var/lustre_stats/$USER.over30d.files.lst
- You are suggested to copy files from /scratch to /project or outside of GACRC
- You should first move all unnecessary files and folders to /scratch/trash/$USER
- The fastest way to save your old files is to copy them to /project area, using the fpsync utility on xfer.gacrc.uga.edu
- When use tar to archive data on /scratch, please do not use its z option (compression option). Once you achieved data by tar, you can compress it using gzip later, for example, tar cf archiveName.tar ./folder; gzip archiveName.tar
<table>
<thead>
<tr>
<th>Queue</th>
<th>Total Nodes</th>
<th>RAM(GB)/Node</th>
<th>Max Mem(GB)/Single-node job</th>
<th>Cores/Node</th>
<th>Processor Type</th>
<th>GPU Cards/Node</th>
<th>InfiniBand</th>
</tr>
</thead>
<tbody>
<tr>
<td>batch</td>
<td>35</td>
<td>192</td>
<td>184</td>
<td>32</td>
<td>Intel Xeon Skylake</td>
<td></td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>31</td>
<td>64</td>
<td>58</td>
<td>28</td>
<td>Intel Xeon Broadwell</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>42</td>
<td>128</td>
<td>120</td>
<td>48</td>
<td>AMD Opteron</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>63</td>
<td>128</td>
<td>120</td>
<td></td>
<td>AMD EPYC (41</td>
<td>22)</td>
<td></td>
</tr>
<tr>
<td>highmem_q</td>
<td>7</td>
<td>1024</td>
<td>990</td>
<td>28</td>
<td>64</td>
<td>Intel Xeon Broadwell (3) AMD EPYC (4)</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>17</td>
<td>512</td>
<td>502</td>
<td>32</td>
<td>48</td>
<td>AMD EPYC (15) AMD Opteron (2)</td>
<td></td>
</tr>
<tr>
<td>gpu_q</td>
<td>4</td>
<td>192</td>
<td>184</td>
<td>32</td>
<td>Intel Xeon Skylake (EDR)</td>
<td>1 NVDIA P100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>128</td>
<td>120</td>
<td>16</td>
<td>Intel Xeon (QDR)</td>
<td>8 NVDIA K40m</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>96</td>
<td>90</td>
<td>12</td>
<td></td>
<td>7 NVDIA K20Xm</td>
<td></td>
</tr>
<tr>
<td>groupBuyin_q</td>
<td>variable</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Software Environment

1. Software names are long and have a EasyBuild toolchain name associated to it

2. Complete module name: Name/Version-toolchain, e.g., Python/3.6.4-foss-2018a

3. Software names are case-sensitive!
   - `module avail`: List all available software modules installed on cluster
   - `module load moduleName`: Load a module into your working environment
   - `module list`: List modules currently loaded
   - `module unload moduleName`: Remove a module from working environment
   - `module spider pattern`: Search module names matching a pattern (case-insensitive)
1. Log on to Login node using MyID and password, and two-factor authentication with Archpass Duo:
   ```
   ssh MyID@sapelo2.gacrc.uga.edu
   ```
2. On Login node, change directory to your scratch space: `cd /scratch/MyID`
3. Create a working subdirectory for a job: `mkdir ./workDir`
4. Change directory to `workDir`: `cd ./workDir`
5. Transfer data from local computer to `workDir`: use `scp` or `SSH File Transfer` to connect Transfer node
   Transfer data on cluster to `workDir`: log on to Transfer node and then use `cp` or `mv`
6. Make a job submission script in `workDir`: `nano ./sub.sh`
7. Submit a job from `workDir`: `qsub ./sub.sh`
8. Check job status: `qstat_me` or Cancel a job: `qdel JobID`
Step1: Log on to Login node - Mac/Linux using ssh
https://wiki.gacrc.uga.edu/wiki/Connecting#Connecting_to_Sapelo2

1. Open Terminal utility

2. Type command line: `ssh MyID@sapelo2.gacrc.uga.edu`

3. You will be prompted for your MyID password

4. Sapelo2 access requires ID verification using two-factor authentication with Archpass Duo. If you are not enrolled in Archpass Duo, please refer to

   https://eits.uga.edu/access_and_security/infosec/tools/archpass_duo/ on how to enroll
Step1 (Cont.) - Mac/Linux using ssh

Use Terminal utility on Mac or Linux!

Log on
Input MyID password!

Select Duo login option!

I am on sapelo2-sub2!
Step1 (Cont.) - Windows using SSH Secure Utilities

1. Download and install SSH Secure Utilities: [http://eits.uga.edu/hardware_and_software/software/](http://eits.uga.edu/hardware_and_software/software/)

2. You can use PuTTY as an alternative: [https://www.putty.org/](https://www.putty.org/)
Please Note:
Authentication Method needs to be set as Keyboard Interactive in default <profile Setting>
Step 1 (Cont.) - Windows using SSH Secure Utilities

1. Log in to your Windows computer.

2. Click on the Start button.

3. Click on "Computer".

4. Click on "SSH Secure Shell".

5. Enter the Host Name: `sapelo2.gacrc.uga.edu`.

6. Enter the User Name: `MyID`.

7. Enter the Port Number: `22`.

8. Enter your UGA MyID password and click OK.

Host Name: `sapelo2.gacrc.uga.edu`
User Name: `MyID`
Port Number: `22`
Step 1 (Cont.) - Windows using SSH Secure Utilities

9. Enter “push” and click OK

10. Verify login using Duo, then you will log on!
Step2: On Login node change directory to global scratch

- Once you logged on, your current directory will be your **home directory**
  
  ```
  zhuofei@sapelo2-sub2 ~$ pwd
  /home/zhuofei
  ← this is my home directory!
  ```

- Use `cd` command to change your current directory to `/scratch/MyID`
  
  ```
  zhuofei@sapelo2-sub2 /$ cd /scratch/zhuofei
  zhuofei@sapelo2-sub2 zhuofei$ pwd
  /scratch/zhuofei
  ← this is my scratch space!
  ```

- Use `ls` command to take a look in `/scratch/MyID`
  
  ```
  zhuofei@sapelo2-sub2 zhuofei$ ls
  user_test  workDir_Alex  workDir_bk
  ```
Step 3 - 4: Create and cd to a working subdirectory

- Use `mkdir` command to make a subdirectory in `/scratch/MyID`

  ```
  zhuofei@sapelo2-sub2 zhuofei$ mkdir workDir
  zhuofei@sapelo2-sub2 zhuofei$ ls
  user_test   workDir    workDir_Alex  workDir_bk
  ```

- Use `cd` command to change your current directory to `/scratch/MyID/workDir`

  ```
  zhuofei@sapelo2-sub2 zhuofei$ cd workDir
  zhuofei@sapelo2-sub2 workDir$ ls
  zhuofei@sapelo2-sub2 workDir$ `it is empty!`
Step5: Transfer data from local computer to workDir - Mac/Linux
https://wiki.gacrc.uga.edu/wiki/Transferring_Files

1. You need to connect to cluster’s Transfer node (xfer.gacrc.uga.edu)
2. Open Terminal utility on local computer to use scp (-r) [Source] [Target]

E.g. 1: working on local computer, from Local ➔ workDir on cluster

```
scp ./file zhuofei@xfer.gacrc.uga.edu:/scratch/zhuofei/workDir/
scp -r ./folder/ zhuofei@xfer.gacrc.uga.edu:/scratch/zhuofei/workDir/
```

E.g. 2: working on local computer, from workDir on cluster ➔ Local

```
scp zhuofei@xfer.gacrc.uga.edu:/scratch/zhuofei/workDir/file .
scp -r zhuofei@xfer.gacrc.uga.edu:/scratch/zhuofei/workDir/folder/ .
```
Step 5 (Cont.) - Windows using SSH Secure Utilities

1. You need to connect to cluster’s **Transfer node** (xfer.gacrc.uga.edu)

2. Use **SSH File Transfer** on **local computer** (alternative FileZilla or WinSCP)

3. Steps 1-10 are the same as steps on page 14-15, except for **Host Name** in step

   Host Name: xfer.gacrc.uga.edu

4. Once you log on, use File Transfer of SSH Secure Utilities, as shown on next page
Step 5 (Cont.) - Windows using SSH Secure Utilities

1. Click to open File Transfer

2. Change local and remote paths

3. Drag data between local computer and remote Transfer node
Step5 (Cont.): Transfer data on cluster to workDir

• Log on to Transfer node (xfer.gacrc.uga.edu)
  ✓ Mac/Linux: ssh MyID@xfer.gacrc.uga.edu (page 12)
  ✓ Windows: use SSH Secure Utilities (page 19)

• Landing folder: /home/MyID (Home)

• You can transfer data between following directories on cluster using cp or mv:
  1. /home/MyID (Home)
  2. /scratch/MyID (Scratch)
  3. /work/abclab (Work)
  4. /project/abclab (Project)

• Most file systems on Transfer are auto-mounted upon the first time full-path access, e.g., cd /project/abclab/
Step6: Make a job submission script in workDir
https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sapelo2#Job_submission_Scripts

$ nano sub.sh

nano is a small and friendly text editor on Linux.

Ctrl-x to save file and quit from nano
Step 6 (Cont.)

1. Sample script on GACRC Wiki Software page: [https://wiki.gacrc.uga.edu/wiki/Bowtie2-Sapelo2](https://wiki.gacrc.uga.edu/wiki/Bowtie2-Sapelo2)

2. Modify it as needed for your computing

To run this example, you need to copy 3 files into your current working dir:

- `cp /usr/local/training/sub.sh`
- `cp /usr/local/training/myreads.fq`
- `cp -r /usr/local/training/index`

```bash
#PBS -S /bin/bash
#PBS -q batch
#PBS -N bowtie2_test
#PBS -l nodes=1:ppn=1
#PBS -l mem=2gb
#PBS -l walltime=1:00:00

#PBS -M zhuofei@uga.edu
#PBS -m ae

cd $PBS_O_WORKDIR

module load Bowtie2/2.3.3-foss-2016b

time bowtie2 -p 1 -x ./index/lambda_virus -U ./myreads.fq -S output.sam
```
Step 7: Submit a job from workDir using `qsub`

https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sapelo2#How_to_submit_a_job_to_the_batch_queue

```
zhuofei@sapelo2-sub2 workDir$ pwd
/scratch/zhuofei/workDir
zhuofei@sapelo2-sub2 workDir$ ls
index  myreads.fq  sub.sh
zhuofei@sapelo2-sub2 workDir$ qsub sub.sh
11943.sapelo2
```

**sub.sh** is 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
Step 8: Check job status using `qstat_me`

https://wiki.gacrc.uga.edu/wiki/Monitoring_Jobs_on_Sapelo2

```
zhuofei@sapelo2-sub2 workDir$ 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>11943.sapelo2</td>
<td>bowtie2_test</td>
<td>zhuofei</td>
<td>00:06:40</td>
<td>C</td>
<td>batch</td>
</tr>
<tr>
<td>11944.sapelo2</td>
<td>bowtie2_test</td>
<td>zhuofei</td>
<td>00:05:17</td>
<td>R</td>
<td>batch</td>
</tr>
<tr>
<td>11946.sapelo2</td>
<td>bowtie2_test</td>
<td>zhuofei</td>
<td>00:12:51</td>
<td>R</td>
<td>batch</td>
</tr>
<tr>
<td>11947.sapelo2</td>
<td>bowtie2_test</td>
<td>zhuofei</td>
<td>0</td>
<td>R</td>
<td>batch</td>
</tr>
<tr>
<td>11948.sapelo2</td>
<td>bowtie2_test</td>
<td>zhuofei</td>
<td>0</td>
<td>Q</td>
<td>batch</td>
</tr>
</tbody>
</table>
```

**R**: job is running

**C**: job completed (or canceled or crashed) and is not longer running. Jobs stay in this state for 24 hours.

**Q**: job is pending, waiting for resources to become available

Note: “Time Use” is the CPU time, instead of the wall-clock time of your job staying on cluster!
Step 8 (Cont.): Cancel job using `qdel`  
https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sapelo2#How_to_delete_a_running_or_pending_job

```
zhuofei@sapelo2-sub2 workDir$ qdel 11947
zhuofei@sapelo2-sub2 workDir$ qstat_me
```

<table>
<thead>
<tr>
<th>Job ID</th>
<th>Name</th>
<th>User</th>
<th>Time Use S Queue</th>
</tr>
</thead>
<tbody>
<tr>
<td>11943.sapelo2</td>
<td>bowtie2_test</td>
<td>zhuofei</td>
<td>00:06:40 C batch</td>
</tr>
<tr>
<td>11944.sapelo2</td>
<td>bowtie2_test</td>
<td>zhuofei</td>
<td>00:05:17 R batch</td>
</tr>
<tr>
<td>11946.sapelo2</td>
<td>bowtie2_test</td>
<td>zhuofei</td>
<td>00:12:51 R batch</td>
</tr>
<tr>
<td>11947.sapelo2</td>
<td>bowtie2_test</td>
<td>zhuofei</td>
<td>00:00:09 C batch</td>
</tr>
<tr>
<td>11948.sapelo2</td>
<td>bowtie2_test</td>
<td>zhuofei</td>
<td>0 Q batch</td>
</tr>
</tbody>
</table>

Job 11947 status is changed from R to C  
C status will stay in list for ~24 hour
Step8 (Cont.): Check Job using `qstat -n -u MyID`
https://wiki.gacrc.uga.edu/wiki/Monitoring_Jobs_on_Sapelo2

```
zhuofei@sapelo2-sub2 workDir$ qstat -n -u zhuofei

dispatch.ecompute:

<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>Elap Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>12175.sapelo2</td>
<td>zhuofei</td>
<td>batch</td>
<td>bowtie2_test</td>
<td>132442</td>
<td>1</td>
<td>1</td>
<td>2gb</td>
<td>01:00:00</td>
<td>R 00:23:44</td>
</tr>
<tr>
<td>n238/0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12176.sapelo2</td>
<td>zhuofei</td>
<td>batch</td>
<td>bowtie2_test</td>
<td>67226</td>
<td>1</td>
<td>1</td>
<td>2gb</td>
<td>01:00:00</td>
<td>R 00:20:44</td>
</tr>
<tr>
<td>n237/0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12177.sapelo2</td>
<td>zhuofei</td>
<td>batch</td>
<td>bowtie2_test</td>
<td>119643</td>
<td>1</td>
<td>1</td>
<td>2gb</td>
<td>01:00:00</td>
<td>R 00:05:44</td>
</tr>
</tbody>
</table>

Note: “Elap Time” is the wall-clock time, instead of the CPU time, which `qstat_me` can give you!

---

29
GACRC SAPELO2 CLUSTER NEW USER TRAINING WORKSHOP

6/22/2020
Step 8 (Cont.): Check Job using `qstat -f JobID`  

https://wiki.gacrc.uga.edu/wiki/Monitoring_Jobs_on_Sapelo2

```bash
zhuofei@sapelo2-sub2 workDir$ qstat -f 12175
Job Id: 12175.sapelo2
  Job_Name = bowtie2_test
  Job_Owner = zhuofei@sapelo2-sub1.ecompute
  resources_used.cput = 00:20:08
  resources_used.mem = 157064kb
  resources_used.vmem = 287900kb
  resources_used.walltime = 00:20:10
  job_state = R
  queue = batch
  ......  
  Error_Path = sapelo2-sub1.ecompute:/scratch/zhuofei/workDir/bowtie2_test.e12175
  Output_Path = sapelo2-sub1.ecompute:/scratch/zhuofei/workDir/bowtie2_test.o12175
  ......  
  Resource_List.nodes = 1:ppn=1
  Resource_List.walltime = 1:00:00
  Resource_List.mem = 2gb
  ......  
  submit_args = sub.sh
  ......  
  init_work_dir = /scratch/zhuofei/workDir
```

6/22/2020
Summary: Obtain Job Details

https://wiki.gacrc.uga.edu/wiki/Monitoring_Jobs_on_Sapelo2

Option 1:

```
qstat -n -u MyID  for running/pending/completed jobs on cluster
qstat -nr -u MyID for running jobs on cluster
qstat -nrt -u MyID for running jobs on cluster, showing each array job element
```

Option 2:

```
qstat -f JobID   for detail info of a running jobs or finished job within 24 hours
```

Option 3:

Email notification from finished jobs (completed, canceled, or crashed), if using:

```
#PBS -M MyID@uga.edu
#PBS -m ae
```
1. Linux/Mac user: 
   `ssh MyID@sapelo2.gacrc.uga.edu`

   Windows user: 
   ![Diagram Image]

5. Linux/Mac user: 
   `scp file MyID@xfer.gacrc.uga.edu:/scratch/MyID/workDir`

   Windows user: 
   ![Diagram Image]

2. `cd /scratch/MyID`

3. `mkdir ./workDir`

4. `cd ./workDir`

6. `nano ./sub.sh`

   #PBS -/bin/bash
   #PBS -q batch
   #PBS -N bowtie2_test
   #PBS -l nodes=1:ppn=1
   #PBS -l mem=2gb
   #PBS -l walltime=1:00:00
   #PBS -M yourMyID@uga.edu
   #PBS -m ae

   cd $PBS_O_WORKDIR
   module load Bowtie2/2.3.3-foss-2016b
   ...

7. `$ qsub sub.sh`

8. `$ qstat me or qdel JobID`

6/22/2020

GACRC SAPEL02 CLUSTER NEW USER TRAINING WORKSHOP
qlogin Commands

https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sapelo2 - How to open an interactive session

1. Type qlogin commands from Login node to open Interactive node:
   - `qlogin_intel`: Start an interactive session on an Intel node
   - `qlogin_amd`: Start an interactive session on an AMD node
   - `qlogin`: start an interactive job on either type of nodes

2. Type `exit` command to quit and back to Login node
GACRC Wiki [http://wiki.gacrc.uga.edu](http://wiki.gacrc.uga.edu)


Running Jobs: [https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sapelo2](https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sapelo2)

Monitoring Jobs: [https://wiki.gacrc.uga.edu/wiki/Monitoring_Jobs_on_Sapelo2](https://wiki.gacrc.uga.edu/wiki/Monitoring_Jobs_on_Sapelo2)

Job Submission Queue: [https://wiki.gacrc.uga.edu/wiki/Job_Submission_Queue](https://wiki.gacrc.uga.edu/wiki/Job_Submission_Queue)

Software: [https://wiki.gacrc.uga.edu/wiki/Software](https://wiki.gacrc.uga.edu/wiki/Software)

Transfer File: [https://wiki.gacrc.uga.edu/wiki/Transferring_Files](https://wiki.gacrc.uga.edu/wiki/Transferring_Files)

Linux Command: [https://wiki.gacrc.uga.edu/wiki/Command_List](https://wiki.gacrc.uga.edu/wiki/Command_List)

Training: [https://wiki.gacrc.uga.edu/wiki/Training](https://wiki.gacrc.uga.edu/wiki/Training)

User Account Request: [https://wiki.gacrc.uga.edu/wiki/User_Accounts](https://wiki.gacrc.uga.edu/wiki/User_Accounts)
GACRC Support
https://uga.teamdynamix.com/TDClient/Requests/ServiceCatalog?CategoryID=11593

➢ **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

Please note to make sure the correctness of datasets being used by your jobs!
GACRC Service Catalog

Services (11)

Account Creation
For a research group’s PI to request user accounts for group members on the GACRC computing systems.

Class Account Creation
For an instructor to request user accounts for students attending a course that will need to use GACRC computing systems.

Class Account Modification
For instructors to request changes to be made in previously requested class account.

Computing Lab Modification/Deletion

General Internal

General Support
Report issues and request help with GACRC systems, except for software installation requests and account/lab creation requests.

Lab Creation
For a research group’s PI to register a computing lab on the GACRC computing systems.

Modify/Delete Account
For PIs to request changes in or deletion of user accounts on GACRC computing systems.

Software Installation/Update
Request software and common application database (e.g. NCBI blast databases) installation and upgrade.

My Recent Requests

- home directory is not fully provisioned: ss57215
- GACRC Sapeko2 New Lab/Use Account Request 2018-11-14_preTraining
- GACRC Sapeko2 Cluster New Lab/Use Account Request 2018-11-05_preTraining
- provision 5 user accounts for ugahebo desk group
- GACRC Sapeko2 New Lab/Use Account Request 2018-10-22_preTraining

View All Recent Requests

Popular Services

- EITS Help Desk Support Request
- MyID Account Request
  - Change Request
- 62 Restricted VPN Access
- Terry Classroom & Meeting Room Support

View All Popular Services

My Recently Visited Services

- Modify/Delete Account
- Class Account Creation
Click to request
Appendix: Examples of Batch Serial/Threaded/MPI Job Scripts

https://wiki.gacrc.uga.edu/wiki/Sample_Scripts

- 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
       - maximum wallclock time
    2. load software using `ml load` (for cluster software)
    3. run any Linux commands you want to run, e.g., `pwd`, `mkdir`, `cd`, `echo`, etc.
    4. run the software
       - **Input data** for analysis, if have

- Common queueing commands you need:
  - `qsub`, `qstat_me`, `qstat`, `qdel`
  - `qstat -f`, `showq`
Example 1: **Serial job script running NCBI Blast+ using 1 CPU**

```
#PBS -S /bin/bash  # Linux default shell (bash)
#PBS -q batch     # Queue name (batch)
#PBS -N testBlast # Job name (testBlast)
#PBS -l nodes=1:ppn=1 # Number of nodes (1), number of cores (1), node feature is NOT needed!
#PBS -l mem=20gb # Maximum amount of RAM memory (20 GB) is enforced by the cluster!
#PBS -l walltime=48:00:00 # Maximum wall-clock time (48 hours) for the job, default 6 minutes

cd $PBS_O_WORKDIR # Compute node will use the directory from which the job is submitted as the working directory, i.e., /lustre1/MyID/workDir

ml load BLAST+/2.6.0-foss-2016b-Python-2.7.14 # Load the module of ncbiblast+, version 2.6.0

time blastn [options] ... # Run blastn with ‘time’ command to measure the amount of time it takes to run the application
```

https://wiki.gacrc.uga.edu/wiki/BLAST%2B-Sapelo2
Example 2: Threaded job script running NCBI Blast+ using 4 CPUS

```bash
#PBS -S /bin/bash
#PBS -q batch
#PBS -N testBlast
#PBS -l nodes=1:ppn=4
#PBS -l mem=20gb
#PBS -l walltime=480:00:00
#PBS -M jsmith@uga.edu
#PBS -m ae
#PBS -j oe

cd $PBS_O_WORKDIR

ml load BLAST+/2.6.0-foss-2016b-Python-2.7.14

time blastn -num_threads 4 [options] ...  ➔ Run blastn with 4 threads (-num_threads 4)
```

- Number of nodes (1), number of cores (4)
  
  Number of cores requested (4) = Number of threads (4)

- Email address to receive a notification for computing resources
- Send email notification when job aborts (a) or terminates (e)
- Standard error file (testBlast.e12345) will be merged into standard out file (testBlast.o12345)
*Example 3: MPI job script running RAxML using 2 full nodes*

```bash
#PBS -S /bin/bash
#PBS -q batch
#PBS -N testRAxML
#PBS -l nodes=2:ppn=28
#PBS -l walltime=120:00:00
#PBS -l mem=100gb

Number of nodes (2), number of cores (28)
Total cores requested = 2 \times 28 = 56

We suggest, Number of MPI Processes (50) ≤ Number of cores requested (56)

cd $PBS_O_WORKDIR

ml load RAxML/8.2.11-foss-2016b-mpi-avx

To run raxmlHPC-MPI-AVX, MPI version using OpenMPI

mpirun --np 50 raxmlHPC-MPI-AVX [options]

Run raxmlHPC-MPI-AVX with 50 MPI processes (--np 50), default 56
```
Guideline Tips

• Do NOT use Login node to run CPU/memory intensive tasks directly ➔ submit job to queue!

• Do NOT use Login Node to transfer data between your local computer and cluster ➔ use Transfer node!

• Do NOT use Home for storing job data ➔ use /scratch/MyID

• Do NOT park data in Scratch or Local Scratch ➔ clean up when job finished or exits from node

• Do NOT park data permanently in Project ➔ download data to your local drive

• NO large memory job running on batch queue ➔ use highmem_q

• NO small memory job running on highmem_q queue ➔ use batch

• In general, number of threads you want to run with a parallel job = number of cores (ppn) 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.
Thank You!

**Telephone Support**

EITS Help Desk: 706-542-3106

Monday – Thursday: 7:30 a.m. – 7:30 p.m.

Friday: 7:30 a.m. – 6 p.m.

Saturday – Sunday: 1 p.m. – 7 p.m.

**Georgia Advanced Computing Resource Center**

101-108 Computing Services building

*University of Georgia*

*Athens, GA 30602*

[https://gacrc.uga.edu/](https://gacrc.uga.edu/)