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 and Overview
  - Five Directories
  - Five Computational Partitions
  - Software Environment
- Batch Job Submission Workflow
- Useful Commands: `squeue --me`, `sacct`, `sacct-gacrc`, `qlogin`
- GACRC Wiki and User Support
- Appendices
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
Help and Support: https://wiki.gacrc.uga.edu/wiki/Getting_Help
Web Site: http://gacrc.uga.edu
Kaltura Channel: https://kaltura.uga.edu/channel/GACRC/176125031
Sapelo2 Cluster

1. **Login/Submit Nodes**
sapelo2.gacrc.uga.edu

   - (ss-sub1, ss-sub2, ss-sub3)

2. **Transfer Nodes**
xfer.gacrc.uga.edu

   - (xfer4, xfer5, xfer6)

**Home Storage** /home/MyID
- 200 GB quota

**Scratch Storage** /scratch/MyID
- No quota

**Work Storage** /work/abclab
- 500 GB or 10^5 files quota

**Project Storage** /project/abclab
- 1 TB quota

**Compute Nodes**

- **Cluster**: Nodes + Storage, all connected by network
- **Node**: Computer with a particular purpose within the cluster

---

**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/](https://eits.uga.edu/access_and_security/infosec/tools/vpn/)
### 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>Login, Transfer</td>
<td>Temporary files needed for currently running jobs</td>
<td>No</td>
<td>Clean up when your job finishes! Subject to “30-day purge” policy</td>
</tr>
<tr>
<td>/work/abclab</td>
<td>Work</td>
<td>500GB $10^5$ files</td>
<td>Login, Transfer</td>
<td>Input files needed for repeated jobs</td>
<td>No</td>
<td>Clean up when your job finishes! Group sharing is possible</td>
</tr>
<tr>
<td>/project/abclab</td>
<td>Project</td>
<td>1TB (initial)</td>
<td>Transfer</td>
<td>Temporary data parking</td>
<td>Yes</td>
<td>Group sharing is possible</td>
</tr>
<tr>
<td>/scratch</td>
<td>Local Scratch</td>
<td>200GB - 800GB</td>
<td>Compute</td>
<td>Jobs with heavy disk I/O operations</td>
<td>No</td>
<td>Clean up when job exits from node!</td>
</tr>
</tbody>
</table>
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 old files is to copy them to /project area, using the fpsync utility on xfer.gacrc.uga.edu
➢ When you archive data using tar on /scratch, please do not use z option (compression option). After you achieved data by tar, you can use gzip to compress it.
Compute nodes are divided into groups called **partitions**. A **partition** is a collection of compute nodes for a particular computing need.

- **batch/batch\_30d**
  - For regular jobs

- **highmem\_p/highmem\_30d\_p**
  - For high memory jobs

- **gpu\_p/gpu\_30d\_p**
  - For GPU-enabled jobs

- **inter\_p**
  - For interactive jobs

\[ p = \text{“partition”} \]
\[ _{30d} = 30 \text{ day (time limit)} \]
Computational Partitions  
https://wiki.gacrc.uga.edu/wiki/Job_Submission_partitions_on_Sapelo2  
https://wiki.gacrc.uga.edu/wiki/Systems#Sapelo2

<table>
<thead>
<tr>
<th>Type</th>
<th>Partition</th>
<th>Time limit</th>
<th>Max jobs Running</th>
<th>Max jobs Submit</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regular</td>
<td>batch</td>
<td>7 days</td>
<td>250</td>
<td>10,000</td>
<td>Regular nodes</td>
</tr>
<tr>
<td></td>
<td>highmem_p</td>
<td></td>
<td>15</td>
<td>100</td>
<td>For running high memory jobs</td>
</tr>
<tr>
<td></td>
<td>gpu_p</td>
<td></td>
<td>18</td>
<td>20</td>
<td>For running GPU-enabled jobs</td>
</tr>
<tr>
<td>Long-term</td>
<td>batch_30d</td>
<td>30 days</td>
<td>1</td>
<td>2</td>
<td>30-day partition counterparts</td>
</tr>
<tr>
<td></td>
<td>highmem_30d_p</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>gpu_30d_p</td>
<td></td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Interactive</td>
<td>inter_p</td>
<td>2 days</td>
<td>3</td>
<td>20</td>
<td>Regular nodes, for interactive jobs.</td>
</tr>
<tr>
<td>Buy-in</td>
<td>name_p</td>
<td>variable</td>
<td></td>
<td></td>
<td>Partitions that target different groups' buy-in nodes. The name string is specific to each group.</td>
</tr>
<tr>
<td>Partition</td>
<td>Total Nodes</td>
<td>Max Mem(GB) /Single-node job</td>
<td>Cores /Node</td>
<td>Processor Type</td>
<td>GPU Cards /Node</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------</td>
<td>-----------------------------</td>
<td>-------------</td>
<td>------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>batch</td>
<td>~275</td>
<td>120</td>
<td>64</td>
<td>AMD EPYC</td>
<td>N/A</td>
</tr>
<tr>
<td>batch_30d</td>
<td></td>
<td>32</td>
<td></td>
<td>AMD Opteron</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>48</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>highmem_p</td>
<td>~30</td>
<td>500</td>
<td>32</td>
<td>AMD EPYC</td>
<td>N/A</td>
</tr>
<tr>
<td>highmem_30d_p</td>
<td></td>
<td>48</td>
<td></td>
<td>AMD Opteron</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>64</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>990</td>
<td>28</td>
<td>Intel Xeon Broadwell</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>48</td>
<td></td>
<td>AMD Opteron</td>
<td></td>
</tr>
<tr>
<td>gpu_p</td>
<td>~10</td>
<td>180</td>
<td>32</td>
<td>Intel Xeon Skylake</td>
<td>1 NVIDIA P100</td>
</tr>
<tr>
<td>gpu_30d_p</td>
<td></td>
<td>120</td>
<td>16</td>
<td>Intel Xeon</td>
<td>8 NVIDIA K40m</td>
</tr>
<tr>
<td></td>
<td></td>
<td>90</td>
<td>12</td>
<td></td>
<td>7 NVIDIA K20Xm</td>
</tr>
<tr>
<td>name_p</td>
<td></td>
<td>variable</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Software Environment
https://wiki.gacrc.uga.edu/wiki/Software

• Approximately 900 software modules are installed (as of March 2021)
• Most modules are compiled EasyBuild (EB) toolchains GCC-8.3.0 or foss-2019b.
• Name format: Name/Version-Toolchain, e.g., Python/3.8.2-GCCcore-8.3.0 (case-sensitive)
• Module commands:
  ➢ ml spider pattern: Search module names matching a pattern
  ➢ ml moduleName: Load a module into your working environment
  ➢ DO NOT LOAD/USE MODULES ON THE LOGIN/SUBMIT NODES! (ss-sub1, ss-sub2, ss-sub3, etc...)
  ➢ ml av: List all available software modules installed on cluster
  ➢ ml: List modules currently loaded
  ➢ ml -moduleName: Remove a module from working environment
  ➢ ml purge: Remove all modules from working environment
Important Tip using Software
https://wiki.gacrc.uga.edu/wiki/Available_Toolchains_and_Toolchain_Compatibility

• When you load more than one software modules, **toolchain compatibility** is the most important thing you need to pay attention to

• If you load more than one module and some toolchains are incompatible, your job will end up with failing dependencies or Lmod errors, such as:

  Lmod has detected the following error:
  These module(s) exist but cannot be loaded as requested

<table>
<thead>
<tr>
<th>Failed Modules</th>
<th>Working Modules</th>
</tr>
</thead>
<tbody>
<tr>
<td>ml Python/3.7.4-GCCcore-8.3.0 ml Perl/5.28.0-GCCcore-7.3.0</td>
<td>ml Python/3.7.4-GCCcore-8.3.0 ml Perl/5.30.0-GCCcore-8.3.0</td>
</tr>
<tr>
<td>ml Beast/2.6.3-foss-2019b ml Perl/5.28.0-GCCcore-7.3.0</td>
<td>ml Beast/2.6.3-foss-2019b ml Perl/5.30.0-GCCcore-8.3.0</td>
</tr>
</tbody>
</table>
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 WinSCP 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:
   
   ```
   sbatch ./sub.sh
   ```

8. Check job status: squeue --me or Cancel a job: scancel jobID
Step 1: Log on to Login node - Mac/Linux using ssh

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

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
Step 1 (Cont.) - Mac/Linux using ssh

Use Terminal utility on Mac or Linux!

```
[zhuofei@localhost ~]$ ssh zhuofei@sapelo2.gacrc.uga.edu
Password: 

Enter a passcode or select one of the following options:
1. Duo Push to XXX-XXX-5758
2. Phone call to XXX-XXX-5758
3. Phone call to XXX-XXX-1925
4. SMS passcodes to XXX-XXX-5758

Passcode or option (1-5): 1

Success. Logging you in...
Last login: Tue Sep 15 11:22:42 2020 from 128.192.75.65
zhuofei@ss-sub1 ~$
```
Step 1 (Cont.) - Windows using PuTTY

1. Download and install PuTTY: https://www.putty.org/

2. Detailed downloading and installation instructions:
   https://wiki.gacrc.uga.edu/wiki/How_to_Install_and_Configure_PuTTY

3. Detailed configuring and usage instructions:
   https://wiki.gacrc.uga.edu/wiki/How_to_Install_and_Configure_PuTTY#Configuring_PuTTY
Step1 (Cont.) - Windows using PuTTY

The first time you connect to login node, PuTTY will give you this security alert window. Please click "Yes"
Next you will enter your UGA MyID password and initiate DUO authentication procedure:

- **UGA MyID password**
- **DUO option**
Step2: On Login node change directory to global scratch

- Once you logged on, your current directory will be your **home directory**

  ```
  zhuofei@ss-sub1 ~$ pwd
  /home/zhuofei
  ← this is my home directory!
  ```

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

  ```
  zhuofei@ss-sub1 ~$ cd /scratch/zhuofei/
  zhuofei@ss-sub1 zhuofei$ pwd
  /scratch/zhuofei
  ← this is my scratch space!
  ```

- Use `ls` command to take a look in `/scratch/MyID`

  ```
  zhuofei@ss-sub1 zhuofei$ ls
  user_test
  ```
Step 3 - 4: Create and cd to a working subdirectory

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

```
zhuofei@ss-sub1 zhuofei$ mkdir workDir
zhuofei@ss-sub1 zhuofei$ ls
user_test  workDir
```

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

```
zhuofei@ss-sub1 zhuofei$ cd workDir
zhuofei@ss-sub1 workDir$ pwd
/scratch/zhuofei/workDir
zhuofei@ss-sub1 workDir$ ls
← it is empty!
```
Step 5: Transfer data from local computer to workDir - Mac/Linux

https://wiki.gacrc.uga.edu/wiki/Transferring_Files#Using_scp

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/ .
```
Step5 (Cont.) - Windows using WinSCP
https://wiki.gacrc.uga.edu/wiki/Transferring_Files#Using_WinSCP

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

2. Use WinSCP on local computer
   - WinSCP can be downloaded from https://winscp.net/eng/index.php
   - Default installation procedure is simple

Step5 (Cont.) - Windows using WinSCP

https://wiki.gacrc.uga.edu/wiki/Transferring_Files#Using_WinSCP
Step 5 (Cont.) - Windows using WinSCP

https://wiki.gacrc.uga.edu/wiki/Transferring_Files#Using_WinSCP

DUO option
Step5 (Cont.) - Windows using WinSCP

https://wiki.gacrc.uga.edu/wiki/Transferring_Files#Using_WinSCP

Change paths on your local computer and transfer node

Drag to transfer files or folders
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 15-16)
  ✓ Windows: use PuTTY to log in MyID@xfer.gacrc.uga.edu (page 17-19)

• 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/
Step 6: Make a job submission script in workDir

https://wiki.gacrc.uga.edu/wiki/Sample_batch_job_submission_scripts_on_Sapelo2

```bash
$ nano sub.sh
```

Nano is a simple text editor on Linux. You are welcome to use other editors like `vim` or `emacs`.

Ctrl-x to save file and quit from nano.
To run the demo example, please copy these files into your working dir:

cp -r /usr/local/training/Sapelo2/* .
Step 7: Submit a job from workDir using `sbatch`.

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

```
zhuofei@ss-sub1 workDir$ pwd
/scratch/zhuofei/workDir
zhuofei@ss-sub1 workDir$ ls
index myreads.fq sub.sh
zhuofei@ss-sub1 workDir$ sbatch sub.sh
Submitted batch job 32860
```

`sub.sh` is a 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 `squeue --me`

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

```
zhuofei@ss-sub1 workDir$ squeue --me
JOBID PARTITION   NAME      USER ST  TIME NODES NODELIST(REASON)
32862 batch testBowt zhuofei PD 0:00  1 (None)
32861 batch testBowt zhuofei R  0:05  1 c5-19
32860 batch testBowt zhuofei R  4:37  1 c5-19
32859 batch testBowt zhuofei CD 9:29  1 b1-2
```

- **R**: job is Running
- **PD**: job is Pending, waiting for resources to become available
- **CD**: job is Completed and is not longer running
- **CA**: job is Canceled and is not longer running
- **F**: job is Failed (crashed) on the node and is not longer running

Note: “TIME” is the elapsed wall-clock time of your job running on cluster, instead of the CPU time.
Step 8: Check job status using `squeue --me -l`

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

---

```bash
zhuofei@ss-sub1 workDir$ squeue --me -l
Tue Sep 15 15:00:51 2020

<table>
<thead>
<tr>
<th>JOBID</th>
<th>PARTITION</th>
<th>NAME</th>
<th>USER</th>
<th>STATE</th>
<th>TIME</th>
<th>TIME_LIMI</th>
<th>NODES</th>
<th>NODELIST (REASON)</th>
</tr>
</thead>
<tbody>
<tr>
<td>32866</td>
<td>batch</td>
<td>testBowt</td>
<td>zhuofei</td>
<td>RUNNING</td>
<td>0:14</td>
<td>1:00:00</td>
<td>1</td>
<td>rc6-10</td>
</tr>
<tr>
<td>32865</td>
<td>batch</td>
<td>testBowt</td>
<td>zhuofei</td>
<td>RUNNING</td>
<td>0:30</td>
<td>1:00:00</td>
<td>1</td>
<td>rc6-10</td>
</tr>
<tr>
<td>32864</td>
<td>batch</td>
<td>testBowt</td>
<td>zhuofei</td>
<td>RUNNING</td>
<td>0:33</td>
<td>1:00:00</td>
<td>1</td>
<td>rc6-10</td>
</tr>
</tbody>
</table>
```
Step 8 (Cont.): Cancel job using `scancel`

https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sapelo2#How_to_cancel_.28delete.29_a_running_or_pending_job

zhuofei@ss-sub1 workDir$ `squeue --me -l`

```
Tue Sep 15 15:08:27 2020

+-----------------+-----------------+----------+-----------------+----------+----------+----------+
| JOBID | PARTITION | NAME      | USER   | STATE   | TIME     | TIME_LIMI | NODES | NODELIST(REASON) |
+-------+-----------+-----------+--------+---------+----------+-----------+-------+--------------------+
| 32869 | batch     | testBowt  | zhuofei| RUNNING | 0:01     | 1:00:00   | 1 c5-19 |
| 32868 | batch     | testBowt  | zhuofei| RUNNING | 0:05     | 1:00:00   | 1 c5-19 |
| 32867 | batch     | testBowt  | zhuofei| RUNNING | 0:10     | 1:00:00   | 1 c5-19 |
+-------+-----------+-----------+--------+---------+----------+-----------+-------+--------------------+
```

zhuofei@ss-sub1 workDir$

```
squeue --me -l
```

```
Tue Sep 15 15:08:45 2020

+-----------------+-----------------+----------+-----------------+----------+----------+----------+
| JOBID | PARTITION | NAME      | USER   | STATE   | TIME     | TIME_LIMI | NODES | NODELIST(REASON) |
+-------+-----------+-----------+--------+---------+----------+-----------+-------+--------------------+
| 32867 | batch     | testBowt  | zhuofei| COMPLETED | 0:26     | 1:00:00   | 1 c5-19 |
| 32869 | batch     | testBowt  | zhuofei| RUNNING | 0:19     | 1:00:00   | 1 c5-19 |
| 32868 | batch     | testBowt  | zhuofei| RUNNING | 0:23     | 1:00:00   | 1 c5-19 |
+-------+-----------+-----------+--------+---------+----------+-----------+-------+--------------------+
```

zhuofei@ss-sub1 workDir$

```
scancel 32867
```

zhuofei@ss-sub1 workDir$

```
squeue --me -l
```

```
Tue Sep 15 15:08:50 2020

+-----------------+-----------------+----------+-----------------+----------+----------+----------+
| JOBID | PARTITION | NAME      | USER   | STATE   | TIME     | TIME_LIMI | NODES | NODELIST(REASON) |
+-------+-----------+-----------+--------+---------+----------+-----------+-------+--------------------+
| 32869 | batch     | testBowt  | zhuofei| RUNNING | 0:19     | 1:00:00   | 1 c5-19 |
| 32868 | batch     | testBowt  | zhuofei| RUNNING | 0:23     | 1:00:00   | 1 c5-19 |
+-------+-----------+-----------+--------+---------+----------+-----------+-------+--------------------+
```
### sacct

<table>
<thead>
<tr>
<th>JobID</th>
<th>JobName</th>
<th>Partition</th>
<th>Account</th>
<th>AllocCPUS</th>
<th>State</th>
<th>ExitCode</th>
</tr>
</thead>
<tbody>
<tr>
<td>3326893</td>
<td>testBowti</td>
<td>batch gacrc-ins+</td>
<td>1</td>
<td>RUNNING</td>
<td>0:0</td>
<td></td>
</tr>
<tr>
<td>3326893.bat+</td>
<td>batch gacrc-ins+</td>
<td>1</td>
<td>RUNNING</td>
<td>0:0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3326893.ext+</td>
<td>extern gacrc-ins+</td>
<td>1</td>
<td>RUNNING</td>
<td>0:0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### sacct -X

<table>
<thead>
<tr>
<th>JobID</th>
<th>JobName</th>
<th>Partition</th>
<th>Account</th>
<th>AllocCPUS</th>
<th>State</th>
<th>ExitCode</th>
</tr>
</thead>
<tbody>
<tr>
<td>3326893</td>
<td>testBowti+</td>
<td>batch gacrc-ins+</td>
<td>1</td>
<td>RUNNING</td>
<td>0:0</td>
<td></td>
</tr>
</tbody>
</table>

### sacct -X --format jobid,state

<table>
<thead>
<tr>
<th>JobID</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>3326893</td>
<td>RUNNING</td>
</tr>
</tbody>
</table>

---

**6/23/2021**

GACRC SAPELO2 CLUSTER NEW USER TRAINING WORKSHOP

---
<table>
<thead>
<tr>
<th>JobID</th>
<th>JobName</th>
<th>User</th>
<th>Partition</th>
<th>NodeList</th>
<th>AllocNodes</th>
<th>NTasks</th>
<th>NCPUS</th>
<th>ReqMem</th>
<th>MaxVMSize</th>
<th>State</th>
<th>CPUTime</th>
<th>Elapsed</th>
<th>Timelimit</th>
<th>ExitCode</th>
<th>WorkDir</th>
</tr>
</thead>
<tbody>
<tr>
<td>275</td>
<td>bowti+</td>
<td>zhuofei</td>
<td>batch</td>
<td>rc6-10</td>
<td>1</td>
<td>1</td>
<td>4Gn</td>
<td>COMPLETE</td>
<td>00:09:13</td>
<td>00:09:13</td>
<td>0:0</td>
<td>/scratch/zhu..</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>275.</td>
<td>batch</td>
<td></td>
<td></td>
<td>rc6-10</td>
<td>1</td>
<td>1</td>
<td>4Gn</td>
<td>COMPLETE</td>
<td>00:09:13</td>
<td>00:09:13</td>
<td>0:0</td>
<td>/scratch/zhu..</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>275.</td>
<td>extern</td>
<td></td>
<td></td>
<td>rc6-10</td>
<td>1</td>
<td>1</td>
<td>4Gn</td>
<td>COMPLETE</td>
<td>00:09:13</td>
<td>00:09:13</td>
<td>0:0</td>
<td>/scratch/zhu..</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>276</td>
<td>amberjob</td>
<td>shtsai</td>
<td>gpu_p</td>
<td>c4-23</td>
<td>1</td>
<td>1</td>
<td>10Gn</td>
<td>CANCELLED</td>
<td>00:03:19</td>
<td>00:03:19</td>
<td>0:15</td>
<td>/scratch/sht..</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>276.</td>
<td>batch</td>
<td></td>
<td></td>
<td>c4-23</td>
<td>1</td>
<td>1</td>
<td>10Gn</td>
<td>CANCELLED</td>
<td>00:03:20</td>
<td>00:03:20</td>
<td>0:15</td>
<td>/scratch/sht..</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>276.</td>
<td>extern</td>
<td></td>
<td></td>
<td>c4-23</td>
<td>1</td>
<td>1</td>
<td>10Gn</td>
<td>COMPLETE</td>
<td>00:03:19</td>
<td>00:03:19</td>
<td>0:0</td>
<td>/scratch/sht..</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>277</td>
<td>mpitest</td>
<td>shtsai</td>
<td>batch</td>
<td>c2-[11-12]</td>
<td>2</td>
<td>24</td>
<td>600Mc</td>
<td>COMPLETE</td>
<td>04:01:12</td>
<td>00:10:03</td>
<td>0:0</td>
<td>/scratch/sht..</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>277.</td>
<td>batch</td>
<td></td>
<td></td>
<td>c2-11</td>
<td>1</td>
<td>1</td>
<td>12</td>
<td>600Mc</td>
<td>221268K</td>
<td>02:00:36</td>
<td>0:0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>277.</td>
<td>extern</td>
<td></td>
<td></td>
<td>c2-[11-12]</td>
<td>2</td>
<td>2</td>
<td>24</td>
<td>600Mc</td>
<td>169800K</td>
<td>04:01:12</td>
<td>0:0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>277.0</td>
<td>orted</td>
<td></td>
<td></td>
<td>c2-12</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>600Mc</td>
<td>265640K</td>
<td>00:00:01</td>
<td>0:0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>278</td>
<td>bash</td>
<td>shtsai</td>
<td>inter_p</td>
<td>c2-4</td>
<td>1</td>
<td>1</td>
<td>2Gn</td>
<td>RUNNING</td>
<td>00:13:37</td>
<td>00:13:37</td>
<td>0:0</td>
<td>/scratch/sht..</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>278.</td>
<td>extern</td>
<td></td>
<td></td>
<td>c2-4</td>
<td>1</td>
<td>1</td>
<td>2Gn</td>
<td>RUNNING</td>
<td>00:13:37</td>
<td>00:13:37</td>
<td>0:0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>278.0</td>
<td>bash</td>
<td></td>
<td></td>
<td>c2-4</td>
<td>1</td>
<td>1</td>
<td>2Gn</td>
<td>RUNNING</td>
<td>00:13:37</td>
<td>00:13:37</td>
<td>0:0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Job Information

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>JobID</td>
<td>47939</td>
</tr>
<tr>
<td>JobName</td>
<td>testBowti+</td>
</tr>
<tr>
<td>User</td>
<td>zhuofei</td>
</tr>
<tr>
<td>Partition</td>
<td>batch</td>
</tr>
<tr>
<td>NodeList</td>
<td>c1-3</td>
</tr>
<tr>
<td>AllocNodes</td>
<td>1</td>
</tr>
<tr>
<td>State</td>
<td>RUNNING</td>
</tr>
<tr>
<td>CPUPTime</td>
<td>00:00:28</td>
</tr>
<tr>
<td>Elapsed</td>
<td>00:00:28</td>
</tr>
<tr>
<td>Timelimit</td>
<td>01:00:00</td>
</tr>
<tr>
<td>ExitCode</td>
<td>0:0</td>
</tr>
<tr>
<td>WorkDir</td>
<td>/scratch/zhuofei/workDir_sapelo2</td>
</tr>
<tr>
<td>NTasks</td>
<td>1</td>
</tr>
<tr>
<td>NCPUS</td>
<td>1</td>
</tr>
<tr>
<td>ReqMem</td>
<td>4Gn</td>
</tr>
</tbody>
</table>
Job ID: 37259
Cluster: tc2
User/Group: zhuofei/gacrc-instruction
State: COMPLETED (exit code 0)
Cores: 1
CPU Utilized: 00:09:45
CPU Efficiency: 99.66% of 00:09:47 core-walltime
Job Wall-clock time: 00:09:47
Memory Utilized: 197.34 MB
Memory Efficiency: 4.82% of 4.00 GB
Obtain Job Details

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

Option 1: `squeue --me` for details of a pending or running job

Option 2: `sacct` or `sacct-gacrc(-v)` for details of computing resource usage of a running or finished job

Option 3: `seff` for details of computing resource usage of a finished job

Option 4: Email notification from finished jobs (completed, canceled, or crashed), if using:

```
#SBATCH --mail-user=username@uga.edu
#SBATCH --mail-type=ALL
```
### Interactive jobs

[https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sapelo2#How_to_open_an_interactive_session](https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sapelo2#How_to_open_an_interactive_session)


<table>
<thead>
<tr>
<th>Description</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start an interactive session</td>
<td><code>qlogin</code></td>
</tr>
<tr>
<td>Start an interactive session with X forwarding</td>
<td><code>xqlogin</code></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><code>qlogin</code></th>
<th><code>srun --pty -p inter_p --mem=2G --nodes=1 --ntasks-per-node=1 --time=12:00:00 --job-name=qlogin bash -l</code></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>xqlogin</code></td>
<td><code>srun --pty --x11 -p inter_p --mem=2G --nodes=1 --ntasks-per-node=1 --time=12:00:00 --job-name=xqlogin bash -l</code></td>
</tr>
</tbody>
</table>
GACRC Wiki [http://wiki.gacrc.uga.edu](http://wiki.gacrc.uga.edu)
Kaltura channel [https://kaltura.uga.edu/channel/GACRC/176125031](https://kaltura.uga.edu/channel/GACRC/176125031)

System: [https://wiki.gacrc.uga.edu/wiki/Systems#Sapelo2](https://wiki.gacrc.uga.edu/wiki/Systems#Sapelo2)
Connection: [https://wiki.gacrc.uga.edu/wiki/Connecting#Connecting_to_Sapelo2](https://wiki.gacrc.uga.edu/wiki/Connecting#Connecting_to_Sapelo2)
Software: [https://wiki.gacrc.uga.edu/wiki/Software_on_Sapelo2](https://wiki.gacrc.uga.edu/wiki/Software_on_Sapelo2)
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)
Sample scripts: [https://wiki.gacrc.uga.edu/wiki/Sample_batch_job_submission_scripts_on_Sapelo2](https://wiki.gacrc.uga.edu/wiki/Sample_batch_job_submission_scripts_on_Sapelo2)
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)
GACRC Help and Support
https://wiki.gacrc.uga.edu/wiki/Getting_Help

➢ 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

When you ask GACRC to test or troubleshoot your jobs, Please make sure of the correctness of your datasets being used!
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.
Click to request
# Slurm job states

<table>
<thead>
<tr>
<th>Code</th>
<th>State</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>Running</td>
<td>Job is running on compute node(s)</td>
</tr>
<tr>
<td>PD</td>
<td>Pending</td>
<td>Job is waiting for compute node(s)</td>
</tr>
<tr>
<td>CD</td>
<td>Completed</td>
<td>Job completed</td>
</tr>
<tr>
<td>CG</td>
<td>Completing</td>
<td>Job is completing</td>
</tr>
<tr>
<td>CA</td>
<td>Canceled</td>
<td>Job was canceled</td>
</tr>
<tr>
<td>F</td>
<td>Failed</td>
<td>Job terminated with non-zero exit code</td>
</tr>
<tr>
<td>NF</td>
<td>Node Fail</td>
<td>Job terminated due to failure of node(s)</td>
</tr>
</tbody>
</table>
### Commands for submitting and canceling jobs

<table>
<thead>
<tr>
<th>Description</th>
<th>Slurm Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Submit a batch job to queue</td>
<td><code>sbatch sub.sh</code></td>
</tr>
<tr>
<td>Delete a job from queue</td>
<td><code>scancel &lt;jobID&gt;</code></td>
</tr>
<tr>
<td>Cancel all your job(s)</td>
<td><code>scancel -u &lt;username&gt;</code></td>
</tr>
<tr>
<td>Cancel all your pending job(s)</td>
<td><code>scancel -t PENDING -u &lt;username&gt;</code></td>
</tr>
<tr>
<td>Cancel your job(s) by job name</td>
<td><code>scancel --name &lt;jobname&gt;</code></td>
</tr>
<tr>
<td>Cancel an element (index) of an array job (jobID)</td>
<td><code>scancel &lt;jobID&gt;_&lt;index&gt;</code></td>
</tr>
</tbody>
</table>
## Commands for monitoring jobs

<table>
<thead>
<tr>
<th>Description</th>
<th>Slurm Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command</td>
<td>squeue   or squeue -l</td>
</tr>
<tr>
<td>Job status of all your jobs</td>
<td>squeue --me or squeue --me -l</td>
</tr>
<tr>
<td>Job status of a job</td>
<td>squeue -j &lt;jobID&gt;</td>
</tr>
<tr>
<td>Job status of job(s) from a user</td>
<td>squeue -u &lt;username&gt;</td>
</tr>
<tr>
<td>Job status with details</td>
<td>scontrol show job &lt;job ID&gt;</td>
</tr>
<tr>
<td></td>
<td>scontrol show job -dd &lt;job ID&gt;</td>
</tr>
<tr>
<td>Job’s resource usage</td>
<td>sacct-gacrc</td>
</tr>
<tr>
<td>View job batch script</td>
<td>scontrol write batch_script &lt;job ID&gt; [filename]</td>
</tr>
</tbody>
</table>
Slurm headers for running a **Serial (single-core) job**

https://wiki.gacrc.uga.edu/wiki/Sample_batch_job_submission_scripts_on_Sapelo2

```bash
#!/bin/bash

#SBATCH --job-name=testBowtie2  # Job name (testBowtie2)
#SBATCH --partition=batch       # Queue name (batch)
#SBATCH --ntasks=1              # Run in a single task using one CPU core on a single node
#SBATCH --mem=4G               # Job memory limit (4 GB)
#SBATCH --time=1:00:00          # Time limit hrs:min:sec or days-hours:minutes:seconds
#SBATCH --export=NONE          # Do not load any users’ explicit environment variables
#SBATCH --output=%x_%j.out     # Standard output log, e.g., testBowtie2_1234.out
#SBATCH --error=%x_%j.err      # Standard error log, e.g., testBowtie2_1234.err
#SBATCH --mail-type=END,FAIL   # Mail events (BEGIN, END, FAIL, ALL)
#SBATCH --mail-user=username@uga.edu  # Where to send mail

cd $SLURM_SUBMIT_DIR          # Change directory to job submission directory
ml Bowtie2/2.4.1-GCC-8.3.0    # Load software module and run bowtie2 below
bowtie2 -x ./index/lambda_virus -U ./myreads.fq -S output.sam
```
Slurm headers for running a Threaded job

```bash
#!/bin/bash
#SBATCH --job-name=testBowtie2  # Job name (testBowtie2)
#SBATCH --partition=batch        # Queue name (batch)
#SBATCH --nodes=1                # Run all processes on a single node
#SBATCH --ntasks=1               # Run in a single task on a single node
#SBATCH --cpus-per-task=8       # Number of CPU cores per task (8)
#SBATCH --mem=10G                # Job memory limit (10 GB)
#SBATCH --time=1:00:00           # Time limit hrs:min:sec or days-hours:minutes:seconds
#SBATCH --export=NONE            # Do not load any users’ explicit environment variables
#SBATCH --output=%x_%j.out       # Standard output log, e.g., testBowtie2_1234.out
#SBATCH --error=%x_%j.err        # Standard error log, e.g., testBowtie2_1234.err
#SBATCH --mail-type=END,FAIL     # Mail events (BEGIN, END, FAIL, ALL)
#SBATCH --mail-user=username@uga.edu  # Where to send mail

cd $SLURM_SUBMIT_DIR
ml Bowtie2/2.4.1-GCC-8.3.0
bowtie2 -p 8 -x ./index/lambda_virus -U ./myreads.fq -S output.sam
```
Slurm headers for running an Array job

```
#!/bin/bash
#SBATCH --job-name=testBowtie2Array  # Job name (testBowtie2Array)
#SBATCH --partition=batch            # Queue name (batch)
#SBATCH --ntasks=1                   # Run in a single task using one CPU core on a single node
#SBATCH --mem=4G                     # Job memory limit (4 GB)
#SBATCH --time=1:00:00               # Time limit hrs:min:sec or days-hours:minutes:seconds
#SBATCH --export=NONE                # Do not load any users’ explicit environment variables
#SBATCH --output=%x_%j.out           # Standard output log, e.g., testBowtie2Array_1234.out
#SBATCH --error=%x_%j.err            # Standard error log, e.g., testBowtie2Array_1234.err
#SBATCH --array=0-9                  # Array element range from 0 to 9, i.e. 10 element jobs

cd $SLURM_SUBMIT_DIR
ml Bowtie2/2.4.1-GCC-8.3.0           # Original data is split into 10 pieces and run in each element job
bowtie2  -x ./index/lambda_virus -U ./myreads_$SLURM_ARRAY_TASK_ID.fq \  
-S output_$SLURM_ARRAY_TASK_ID.sam
```
#!/bin/bash
#SBATCH --job-name=testOpenMP
#SBATCH --partition=batch
#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=12
#SBATCH --mem=10G
#SBATCH --export=NONE
#SBATCH --time=24:00:00
#SBATCH --output=%x_%j.log
#SBATCH --mail-type=END,FAIL
#SBATCH --mail-user=username@uga.edu
cd $SLURM_SUBMIT_DIR
export OMP_NUM_THREADS=$SLURM_CPUS_PER_TASK
ml foss/2019b
./myProgram.x
Slurm headers for running an MPI job

```bash
#!/bin/bash

#SBATCH --job-name=testMPI           # Job name (tesMPI)
#SBATCH --partition=batch            # Queue name (batch)
#SBATCH --nodes=2                    # Run on two nodes
#SBATCH --ntasks-per-node=16         # How many tasks on each node; Number of tasks=32=MPI ranks
#SBATCH --cpus-per-task=1            # Number of CPU cores per task; 16 CPU cores per node
#SBATCH --mem-per-cpu=500M           # Memory per allocated CPU; 8GB (500MB*16) memory per node
#SBATCH --time=24:00:00              # Time limit hrs:min:sec or days-hours:minutes:seconds
#SBATCH --export=NONE                # Do not load any users’ explicit environment variables
#SBATCH --output=%x_%j.log            # Standard output and error log, e.g., testMPI_1234.log
#SBATCH --mail-type=END,FAIL         # Mail events (BEGIN, END, FAIL, ALL)
#SBATCH --mail-user=username@uga.edu # Where to send mail

cd $SLURM_SUBMIT_DIR
ml foss/2019b                       # Load toolchain module
mpirun -n 32 ./myProgram.x          # Run your program binary compiled with OpenMPI with 32 ranks
```
Slurm headers for running a Hybrid MPI/OpenMP job

```bash
#!/bin/bash

#SBATCH --job-name=testHybrid  # Job name (testHybrid)
#SBATCH --partition=batch       # Queue name (batch)
#SBATCH --nodes=2               # Run on two nodes
#SBATCH --ntasks-per-node=8     # How many tasks on each node; Number of tasks=16=MPI ranks
#SBATCH --cpus-per-task=4       # Number of CPU cores per task; 32 CPU cores per node
#SBATCH --mem-per-cpu=500M      # Memory per allocated CPU; 16GB (500MB*32) memory per node
#SBATCH --time=24:00:00         # Time limit hrs:min:sec or days-hours:minutes:seconds
#SBATCH --export=NONE           # Do not load any users’ explicit environment variables
#SBATCH --output=%x_%j.log       # Standard output and error log
#SBATCH --mail-type=END,FAIL    # Mail events (BEGIN, END, FAIL, ALL)
#SBATCH --mail-user=username@uga.edu  # Where to send mail

cd $SLURM_SUBMIT_DIR
export OMP_NUM_THREADS=$SLURM_CPUS_PER_TASK       # Sets the number of threads to use for OpenMP parallel regions
ml foss/2019b
mpirun -n 16 ./myProgram.x

# Change directory to job submission directory
# Load toolchain module
# Run your program binary compiled with OpenMPI with 16 ranks
```
Slurm headers for running a GPU job

```bash
#!/bin/bash

#SBATCH --job-name=amber # Job name
#SBATCH --partition=gpu_p # Partition (queue) name, i.e., gpu_p
#SBATCH --gres=gpu:1 # Requests one GPU device; --gres=gpu:P100:1, --gres=gpu:K40:1
#SBATCH --ntasks=1 # Run a single task
#SBATCH --cpus-per-task=2 # Number of CPU cores per task
#SBATCH --mem=40gb # Job memory request
#SBATCH --time=10:00:00 # Time limit hrs:min:sec
#SBATCH --export=NONE # Do not load any users’ explicit environment variables
#SBATCH --output=%x_%j.out # Standard output and error log, e.g., amber_1234.out
#SBATCH --mail-type=END,FAIL # Mail events (BEGIN, END, FAIL, ALL)
#SBATCH --mail-user=username@uga.edu # Where to send mail

cd $SLURM_SUBMIT_DIR
ml Amber/18-fosscuda-2018b-AmberTools-18-patchlevel-10-8
mpiexec $AMBERHOME/bin/pmemd.cuda -O -i ./prod.in -o prod_c4-23.out -p ./dimerFBP_GOL.prmtop -c ./restart.rst -r prod.rst -x prod.mdcrd
```
#!/bin/bash

#SBATCH --job-name=test_sortmerna       # Job name
#SBATCH --partition=batch               # Partition (queue) name
#SBATCH --ntasks=1                      # Run on a single CPU
#SBATCH --mem=8gb                       # Job memory request
#SBATCH --time=02:00:00                  # Time limit hrs:min:sec
#SBATCH --export=NONE                    # Do not load any users’ explicit environment variables
#SBATCH --output=sortmerna.%j.out        # Standard output log, e.g., sortmerna.1234.out
#SBATCH --error=sortmerna.%j.err         # Standard error log, e.g., sortmerna.1234.err
#SBATCH --cpus-per-task=4               # Number of CPU cores per task
#SBATCH --mail-type=END,FAIL            # Mail events (NONE, BEGIN, END, FAIL, ALL)
#SBATCH --mail-user=username@uga.edu    # Where to send mail

cd $SLURM_SUBMIT_DIR

singularity exec /apps/singularity-images/sortmerna-3.0.3.simg sortmerna --threads 4 --ref db.fasta,db.idx --reads file.fa \   
   --aligned base_name_output
Guideline Tips

• Do NOT use Login node to run CPU/memory intensive tasks directly ➔ submit jobs to compute nodes!
• 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 finishes or exits from node
• Do NOT park data permanently in Project ➔ download data to your local drive
• NO large memory job running on batch partition ➔ use highmem_p
• NO small memory job running on highmem_p partition ➔ use batch
• In general, number of threads you want to run with a parallel job = number of cores requested
• When you archive data using tar on /scratch, please do not use z option (compression option). After you achieved data by tar, you can use gzip to compress it.
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! ✅
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/