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
  - Four Computational Partitions
  - Software Environment
- Batch Job Submission Workflow
- Useful Commands: scontrol show job, seff, sacct-gacrc, sinfo-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
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/)
Overview

- Two nodes, your "username" is your MyID for both of them:
  1. For batch job workflow, the host to log into is sapelo2.gacrc.uga.edu
  2. For file transfers, the host to log into is xfer.gacrc.uga.edu

- Five Directories:
  1. **Home**: Login landing spot; 200GB quota; Backed-up
  2. **Scratch**: High-speed FS for temp files needed for current jobs; NO quota; NOT backed-up; 30-day purge
  3. **Work**: High-speed FS 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; 200~800GB quota; NOT backed-up

- Five Computational Partitions: batch/batch_30d, highmem_p, gpu_p/gpu_30d_p, inter_p, name_p
## 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 finishes! 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 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>/lscratch</td>
<td>Local Scratch</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! Files generated are persistent</td>
</tr>
</tbody>
</table>
Accessing Directories from Nodes

/home/MyID
/scratch/MyID
/work/abclab
/project/abclab
/non-GACRC storage

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 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.
<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>35</td>
<td>20000</td>
<td>Regular nodes</td>
</tr>
<tr>
<td></td>
<td>highmem_p</td>
<td>7 days</td>
<td>15</td>
<td>20</td>
<td>For high memory jobs</td>
</tr>
<tr>
<td></td>
<td>gpu_p</td>
<td></td>
<td>5</td>
<td>20</td>
<td>For 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>A given user can have up to one job running at a time here, plus one pending, or two pending and none running</td>
</tr>
<tr>
<td></td>
<td>highmem_30d_p</td>
<td>30 days</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>gpu_30d_p</td>
<td></td>
<td></td>
<td></td>
<td></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>RAM(GB) /Node</td>
<td>Max Mem(GB) /Single-node job</td>
<td>Cores /Node</td>
<td>Processor Type</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------</td>
<td>---------------</td>
<td>-------------------------------</td>
<td>-------------</td>
<td>----------------</td>
</tr>
<tr>
<td>batch</td>
<td>40</td>
<td>128</td>
<td>120</td>
<td>48</td>
<td>AMD Opteron</td>
</tr>
<tr>
<td>batch_30d</td>
<td>24</td>
<td>128</td>
<td></td>
<td>64</td>
<td>AMD EPYC</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>128</td>
<td></td>
<td>32</td>
<td>AMD EPYC</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>256</td>
<td>250</td>
<td>48</td>
<td>AMD Opteron</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>192</td>
<td>180</td>
<td>32</td>
<td>Intel Skylake</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>64</td>
<td>58</td>
<td>28</td>
<td>Intel Broadwell</td>
</tr>
<tr>
<td>highmem_p</td>
<td>4</td>
<td>512</td>
<td>500</td>
<td>48</td>
<td>AMD Opteron</td>
</tr>
<tr>
<td>highmem_30d_p</td>
<td>2</td>
<td>512</td>
<td>500</td>
<td>32</td>
<td>AMD EPYC</td>
</tr>
<tr>
<td>gpu_p</td>
<td>1</td>
<td>192</td>
<td>180</td>
<td>32</td>
<td>Intel Xeon Skylake</td>
</tr>
<tr>
<td>gpu_30d_p</td>
<td>1</td>
<td>192</td>
<td>180</td>
<td>32</td>
<td>Intel Xeon Skylake</td>
</tr>
<tr>
<td>name_p</td>
<td>variable</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Software Environment
https://wiki.gacrc.uga.edu/wiki/Software_on_sap2test

• More than 650 software modules are installed (as of Oct. 2020)
• 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
  ➢ ml av: List all available software modules installed on cluster
  ➢ ml: List modules currently loaded
  ➢ ml -moduleName: Remove a module from working environment
• Installation path of 3 kinds of Software on Sap2test:

  ➢ /apps/eb: Software built by EasyBuild, e.g., RepeatModeler/2.0.1-foss-2019b, TensorFlow/2.2.0-fosscuda-2019b-Python-3.7.4

  ➢ /apps/gb: Software built out of EasyBuild (no EB toolchain in software name), e.g., ansys/19.2, matlab/R2019b, CellProfiler/4.0.0rc5_pyenv

  ➢ /apps/singularity-images: Singularity container, e.g., busco-4.0.5.simg, repeatmodeler-2.0.1.simg, trinity-2.8.5.simg

• Once software is loaded, its installation path is stored in EBROOTAPPNAME, where APPNAME needs to be replaced by the actual software name (all capital letters), for example,

  zhuofei@b1-1 ~$ ml TensorFlow/2.2.0-fosscuda-2019b-Python-3.7.4
  zhuofei@b1-1 ~$ echo $EBROOTTENSORFLOW
  /apps/eb/TensorFlow/2.2.0-fosscuda-2019b-Python-3.7.4
Important Tip using Software

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

• Toolchain in included as a part of software name, if the software is built by Easybuild

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

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

• https://wiki.gacrc.uga.edu/wiki/Available_Toolchains_and_Toolchain_Compatibility
Job Submission Workflow

https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sap2test

1. Log on to Login node using MyID and password, and two-factor authentication with Archpass Duo:
   `ssh MyID@sap2test.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`:
   `sbatch ./sub.sh`

8. Check job status:
   `squeue --me` or Cancel a job:
   `scancel jobID`
Step1: Log on to Login node - Mac/Linux using ssh
https://wiki.gacrc.uga.edu/wiki/Connecting_to_the_Slurm_test_cluster

1. Open **Terminal** utility

2. Type command line: `ssh MyID@sap2test.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/](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!

[zhuofei@localhost ~]$ ssh zhuofei@sap2test.gacrc.uga.edu
Password: ➣ Input MyID 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 ➣ Select Duo authentication option!
Success. Logging you in...
Last login: Tue Sep 15 11:22:42 2020 from 128.192.75.65

[zhuofei@ss-sub1 ~]$ ➣ I am on login node ss-sub1!
Step 1 (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/)
Step 1 (Cont.) - Windows using SSH Secure Utilities

Please Note:
Authentication Method needs to be set as Keyboard Interactive in default <profile Setting>
Step 1 (Cont.) - Windows using SSH Secure Utilities

Host Name: sap2test.gacrc.uga.edu
User Name: MyID
Port Number: 22

8. Enter your UGA MyID password and click OK
Step1 (Cont.) - Windows using SSH Secure Utilities

9. Enter “push” and click OK

10. Verify login using Duo, then you will log on!
Step 2: 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

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 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 5:

   Host Name: xfer.gacrc.uga.edu

4. Once you log on, use File Transfer of SSH Secure Utilities, as shown on next page
Step5 (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/
Step 6: Make a job submission script in workDir

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

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

nano is a small and friendly text editor on Linux.

Ctrl-x to save file and quit from nano
To run demo example, please copy 3 files into your current working dir:

```bash
cp /usr/local/training/sub.sh .
cp /usr/local/training/myreads.fq .
cp -r /usr/local/training/index .
```
Step 7: Submit a job from workDir using `sbatch`

https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sap2test#How_to_submit_a_job_to_the_batch_queue

```
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_Sap2test

```
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_Sap2test

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

<table>
<thead>
<tr>
<th>JOBS</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_Sap2test#How_to_cancel_.28delete.29_a_running_or_pending_job

zhuofei@ss-sub1 workDir$
Tue Sep 15 15:08:27 2020
```
$ squeue --me -l
```
<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>32869</td>
<td>batch</td>
<td>testBowt</td>
<td>zhuofei</td>
<td>RUNNING</td>
<td>0:01</td>
<td>1:00:00</td>
<td>1</td>
<td>c5-19</td>
</tr>
<tr>
<td>32868</td>
<td>batch</td>
<td>testBowt</td>
<td>zhuofei</td>
<td>RUNNING</td>
<td>0:05</td>
<td>1:00:00</td>
<td>1</td>
<td>c5-19</td>
</tr>
<tr>
<td>32867</td>
<td>batch</td>
<td>testBowt</td>
<td>zhuofei</td>
<td>RUNNING</td>
<td>0:10</td>
<td>1:00:00</td>
<td>1</td>
<td>c5-19</td>
</tr>
</tbody>
</table>

zhuofei@ss-sub1 workDir$
```
$ scancel 32867
```

zhuofei@ss-sub1 workDir$
```
$ squeue --me -l
```
<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>32867</td>
<td>batch</td>
<td>testBowt</td>
<td>zhuofei</td>
<td>COMPLETET</td>
<td>0:26</td>
<td>1:00:00</td>
<td>1</td>
<td>c5-19</td>
</tr>
<tr>
<td>32869</td>
<td>batch</td>
<td>testBowt</td>
<td>zhuofei</td>
<td>RUNNING</td>
<td>0:19</td>
<td>1:00:00</td>
<td>1</td>
<td>c5-19</td>
</tr>
<tr>
<td>32868</td>
<td>batch</td>
<td>testBowt</td>
<td>zhuofei</td>
<td>RUNNING</td>
<td>0:23</td>
<td>1:00:00</td>
<td>1</td>
<td>c5-19</td>
</tr>
</tbody>
</table>

zhuofei@ss-sub1 workDir$
```
$ squeue --me -l
```
<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>32869</td>
<td>batch</td>
<td>testBowt</td>
<td>zhuofei</td>
<td>RUNNING</td>
<td>0:19</td>
<td>1:00:00</td>
<td>1</td>
<td>c5-19</td>
</tr>
<tr>
<td>32868</td>
<td>batch</td>
<td>testBowt</td>
<td>zhuofei</td>
<td>RUNNING</td>
<td>0:23</td>
<td>1:00:00</td>
<td>1</td>
<td>c5-19</td>
</tr>
</tbody>
</table>
zhuofei@ss-sub1 workDir$ scontrol show job 32870
JobId=32870  JobName=testBowtie2
UserId=zhuofei(1772)  GroupId=gacrc-instruction(21004)  MCS_label=N/A
Priority=1  Nice=0  Account=gacrc-appadmin  QOS=normal
JobState=RUNNING  Reason=None  Dependency=(null)
Requeue=1  Restarts=0  BatchFlag=1  Reboot=0  ExitCode=0:0
RunTime=00:00:21  TimeLimit=01:00:00  TimeMin=N/A
StartTime=2020-09-15T15:30:42  EndTime=2020-09-15T16:30:42  Deadline=N/A
......
Partition=batch  AllocNode:Sid=128.192.75.18:94107
ReqNodeList=(null)  ExcNodeList=(null)
NumNodes=1  NumCPUs=4  NumTasks=1  CPUs/Task=4  ReqB:S:C:T=0:0:*:*:
......
Command=/scratch/zhuofei/workDir/sub.sh
WorkDir=/scratch/zhuofei/workDir
StdErr=/scratch/zhuofei/workDir/testBowtie2_32870.err
StdIn=/dev/null
StdOut=/scratch/zhuofei/workDir/testBowtie2_32870.out
zhuofei@ss-sub1 workDir$ seff 37259

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
zhuofei@ss-sub1 workDir$ sacct-gacr

<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>COMPLETED</td>
<td>00:09:13</td>
<td>00:09:13</td>
<td>01:00:00</td>
<td>0:0/scratch/zhu..</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>275.batch</td>
<td>batch</td>
<td>rc6-10</td>
<td>1 1 1 4Gn</td>
<td>396868K</td>
<td>COMPLETED</td>
<td>00:09:13</td>
<td>00:09:13</td>
<td>0:0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>275.extern</td>
<td>extern</td>
<td>rc6-10</td>
<td>1 1 1 4Gn</td>
<td>142616K</td>
<td>COMPLETED</td>
<td>00:09:13</td>
<td>00:09:13</td>
<td>0:0</td>
<td></td>
<td></td>
<td></td>
<td></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>02:00:00</td>
<td>0:0/scratch/sht..</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>276.batch</td>
<td>batch</td>
<td>c4-23</td>
<td>1 1 1 10Gn</td>
<td>221140K</td>
<td>CANCELLED</td>
<td>00:03:20</td>
<td>00:03:20</td>
<td>0:15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>276.extern</td>
<td>extern</td>
<td>c4-23</td>
<td>1 1 1 10Gn</td>
<td>169800K</td>
<td>COMPLETED</td>
<td>00:03:19</td>
<td>00:03:19</td>
<td>0:0</td>
<td></td>
<td></td>
<td></td>
<td></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>COMPLETED</td>
<td>04:01:12</td>
<td>00:10:03</td>
<td>02:00:00</td>
<td>0:0/scratch/sht..</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>277.batch</td>
<td>batch</td>
<td>c2-11</td>
<td>1 1 12 600Mc</td>
<td>221268K</td>
<td>COMPLETED</td>
<td>02:00:36</td>
<td>00:10:03</td>
<td>0:0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>277.extern</td>
<td>extern</td>
<td>c2-[11-12]</td>
<td>24 600Mc</td>
<td>169800K</td>
<td>COMPLETED</td>
<td>04:01:12</td>
<td>00:10:03</td>
<td>0:0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>277.0</td>
<td>orted</td>
<td>c2-12</td>
<td>1 1 1     600Mc</td>
<td>265640K</td>
<td>COMPLETED</td>
<td>00:00:01</td>
<td>00:00:01</td>
<td>0:0</td>
<td></td>
<td></td>
<td></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>12:00:00</td>
<td>0:0/scratch/sht..</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>278.extern</td>
<td>extern</td>
<td>c2-4</td>
<td>1 1 1 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>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>278.0</td>
<td>bash</td>
<td>c2-4</td>
<td>1 1 1 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>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>JobID</strong></td>
<td>47939</td>
<td></td>
<td></td>
<td></td>
<td></td>
<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></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>JobName</strong></td>
<td>testBowti+</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>User</strong></td>
<td>zhuofei</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Partition</strong></td>
<td>batch</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>NodeList</strong></td>
<td>c1-3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>AllocNodes</strong></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>State</strong></td>
<td>RUNNING</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>CPUPTime</strong></td>
<td>00:00:28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Elapsed</strong></td>
<td>00:00:28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Timelimit</strong></td>
<td>01:00:00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>ExitCode</strong></td>
<td>0:0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>WorkDir</strong></td>
<td>/scratch/zhuofei/workDir_sap2test</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>NTasks</strong></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>NCPUS</strong></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>ReqMem</strong></td>
<td>4Gn</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
zhuofei@ss-sub1 workDir$ sinfo-gacrc

<table>
<thead>
<tr>
<th>PARTITION</th>
<th>NODELIST</th>
<th>STATE</th>
<th>CPUS</th>
<th>MEMORY(MB)</th>
<th>AVAIL_FEATURES</th>
<th>GRES</th>
</tr>
</thead>
<tbody>
<tr>
<td>batch</td>
<td>ra3-19,ra4-12</td>
<td>down*</td>
<td>48</td>
<td>128914</td>
<td>AMD,Opteron,QDR</td>
<td>(null)</td>
</tr>
<tr>
<td>batch</td>
<td>b1-[3-23],d4-[6-10]</td>
<td>drained</td>
<td>32+</td>
<td>128596+</td>
<td>AMD,EPYC,EDR</td>
<td>(null)</td>
</tr>
<tr>
<td>batch</td>
<td>c1-3</td>
<td>drained</td>
<td>28</td>
<td>64205</td>
<td>Intel,Broadwell,EDR</td>
<td>(null)</td>
</tr>
<tr>
<td>batch</td>
<td>ra3-[1-18,20-24],ra4-[2-11,13-14,18-20]</td>
<td>drained</td>
<td>48</td>
<td>128914+</td>
<td>AMD,Opteron,QDR</td>
<td>(null)</td>
</tr>
<tr>
<td>batch</td>
<td>rc6-10</td>
<td>drained</td>
<td>28</td>
<td>64205</td>
<td>Intel,Broadwell,QDR</td>
<td>(null)</td>
</tr>
<tr>
<td>batch</td>
<td>b1-2,d4-5</td>
<td>idle</td>
<td>32+</td>
<td>128596+</td>
<td>AMD,EPYC,EDR</td>
<td>(null)</td>
</tr>
<tr>
<td>batch</td>
<td>c5-19</td>
<td>idle</td>
<td>32</td>
<td>192039</td>
<td>Intel,Skylake,EDR</td>
<td>(null)</td>
</tr>
<tr>
<td>gpu_p</td>
<td>c4-23</td>
<td>drained</td>
<td>32</td>
<td>192038</td>
<td>Intel,Skylake,EDR</td>
<td>gpu:P100:1</td>
</tr>
<tr>
<td>highmem_p</td>
<td>ra4-16</td>
<td>draining</td>
<td>48</td>
<td>128914</td>
<td>AMD,Opteron,QDR</td>
<td>(null)</td>
</tr>
<tr>
<td>highmem_p</td>
<td>d4-[11-12]</td>
<td>drained</td>
<td>32</td>
<td>515743</td>
<td>AMD,EPYC,EDR</td>
<td>(null)</td>
</tr>
<tr>
<td>highmem_p</td>
<td>ra4-[21-24]</td>
<td>drained</td>
<td>48</td>
<td>515986</td>
<td>AMD,Opteron,QDR</td>
<td>(null)</td>
</tr>
<tr>
<td>inter_p</td>
<td>ra4-16</td>
<td>draining</td>
<td>48</td>
<td>128914</td>
<td>AMD,Opteron,QDR</td>
<td>(null)</td>
</tr>
<tr>
<td>inter_p</td>
<td>ra4-17</td>
<td>drained</td>
<td>48</td>
<td>257938</td>
<td>AMD,Opteron,QDR</td>
<td>(null)</td>
</tr>
</tbody>
</table>
```
Obtain Job Details

https://wiki.gacrc.uga.edu/wiki/Monitoring_Jobs_on_Sap2test#How_to_see_detailed_information_about_a_given_job

Option 1: `scontrol show job JobID` for details of a running job

Option 2: `seff` for details of computing resource usage of a finished job

Option 3: `sacct --gacrc` for details of computing resource usage of a running or finished job

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

```
#SBATCH --mail-user=username@uga.edu
#SBATCH --mail-type=END,FAIL
```
## Interactive jobs

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

<table>
<thead>
<tr>
<th>Description</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>qlogin</td>
<td><code>srn --pty -p inter_p --mem=2G --nodes=1 --ntasks-per-node=1 --time=8:00:00 --job-name=qlogin bash</code></td>
</tr>
<tr>
<td>xqlogin</td>
<td><code>srn --pty --x11 -p inter_p --mem=2G --nodes=1 --ntasks-per-node=1 --time=8:00:00 --job-name=xqlogin bash</code></td>
</tr>
</tbody>
</table>

[https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sap2test#How_to_open_an_interactive_session](https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sap2test#How_to_open_an_interactive_session)
GACRC Wiki http://wiki.gacrc.uga.edu

Documentation on Sap2test: https://wiki.gacrc.uga.edu/wiki/Systems#Slurm_Test_Cluster_.28Sap2test.29
Connection: https://wiki.gacrc.uga.edu/wiki/Connecting_to_the_Slurm_test_cluster
Migrating to Slurm: https://wiki.gacrc.uga.edu/wiki/Migrating_from_Torque_to_Slurm
Software: https://wiki.gacrc.uga.edu/wiki/Software_on_sap2test
Running Jobs: https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sap2test
Monitoring Jobs: https://wiki.gacrc.uga.edu/wiki/Monitoring_Jobs_on_Sap2test
Sample scripts: https://wiki.gacrc.uga.edu/wiki/Sample_batch_job_submission_scripts_on_the_Slurm_test_cluster
Transfer File: https://wiki.gacrc.uga.edu/wiki/Transferring_Files
Linux Command: https://wiki.gacrc.uga.edu/wiki/Command_List
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.

My Recent Requests

- home directory is not fully provisioned: ss57215
- GACRC Sapele2 New Lab/Use Account Request 2018-11-14_preTraining
- GACRC Sapele2 Cluster New Lab/Use Account Request 2018-11-05_preTraining
- provision 5 user accounts for usgahedesk group
- GACRC Sapele2 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
### 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>
</tr>
</thead>
<tbody>
<tr>
<td>Submit a batch job to queue</td>
</tr>
<tr>
<td>Delete a job from queue</td>
</tr>
<tr>
<td>Cancel all your job(s)</td>
</tr>
<tr>
<td>Cancel all your pending job(s)</td>
</tr>
<tr>
<td>Cancel your job(s) by job name</td>
</tr>
<tr>
<td>Cancel an element (index) of an array job (jobID)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Slurm Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>sbatch sub.sh</td>
</tr>
<tr>
<td>scancel &lt;jobID&gt;</td>
</tr>
<tr>
<td>scancel -u &lt;username&gt;</td>
</tr>
<tr>
<td>scancel -t PENDING -u &lt;username&gt;</td>
</tr>
<tr>
<td>scancel --name &lt;jobname&gt;</td>
</tr>
<tr>
<td>scancel &lt;jobID&gt;_&lt;index&gt;</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><code>squeue</code> or <code>squeue -l</code></td>
</tr>
<tr>
<td>Job status of all your jobs</td>
<td><code>squeue --me</code> or <code>squeue --me -l</code></td>
</tr>
<tr>
<td>Job status of a job</td>
<td><code>squeue -j &lt;jobID&gt;</code></td>
</tr>
<tr>
<td>Job status of job(s) from a user</td>
<td><code>squeue -u &lt;username&gt;</code></td>
</tr>
<tr>
<td>Job status with details</td>
<td><code>scontrol show job &lt;job ID&gt;</code> or <code>scontrol show job -dd &lt;job ID&gt;</code></td>
</tr>
<tr>
<td>Job’s resource usage</td>
<td><code>sacct-gacrc</code></td>
</tr>
<tr>
<td>View job batch script</td>
<td><code>scontrol write batch_script &lt;job ID&gt; [filename]</code></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_the_Slurm_test_cluster

```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
```
#! /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**

```bash
#!/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
```

---

50
Slurm headers for running a Threaded (OpenMP) job

```bash
#!/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
```

- Job name (testOpenMP)
- Queue name (batch)
- Run all processes on a single node
- Run in a single task on a single node
- Number of CPU cores per task (12)
- Job memory limit (10 GB)
- Do not load any users' explicit environment variables
- Time limit hrs:min:sec or days-hours:minutes:seconds
- Standard output and error log, e.g., testOpenMP_1234.log
- Mail events (BEGIN, END, FAIL, ALL)
- Where to send mail

Sets the number of threads to use for OpenMP parallel regions.
Slurm headers for running a MPI job

#!/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 $SLURM_SUBMIT_DIR # Change directory to job submission directory
export OMP_NUM_THREADS=$SLURM_CPUS_PER_TASK # Sets the number of threads to use for OpenMP parallel regions
ml foss/2019b # Load toolchain module
mpirun -n 16 ./myProgram.x # 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-foss cuda-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 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 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.
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/