Using the Sapelo2 Cluster at the GACRC

Cluster New User Training Workshop

Georgia Advanced Computing Resource Center (GACRC)

EITS/University of Georgia

Zhuofei Hou zhuofei@uga.edu

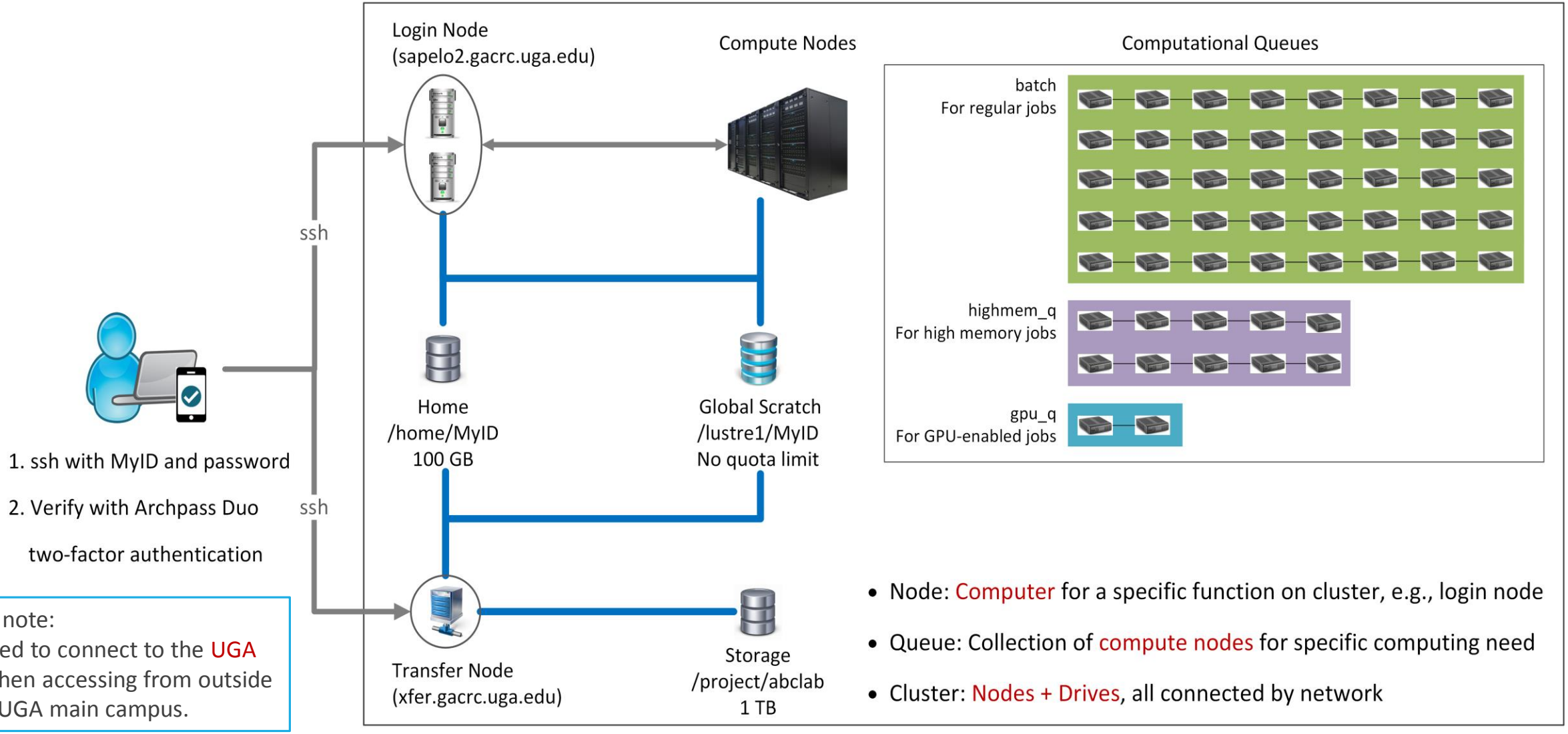
Outline

- GACRC
- Sapelo2 Cluster
  - Diagram
  - Overview
  - Four Directories
  - Four Computational Queues
  - Software Environment
- Job Submission Workflow
- GACRC Wiki and Support

GACRC

- We are a high-performance-computing (HPC) center at UGA
- We provide to the UGA research and education community an advanced computing environment:
  - HPC computing and networking infrastructure located at the Boyd Data Center
  - Comprehensive collection of scientific, engineering and business applications
  - Consulting and training services
- http://wiki.gacrc.uga.edu (GACRC Wiki)
- https://wiki.gacrc.uga.edu/wiki/Getting_Help (GACRC Support)
- http://gacrc.uga.edu (GACRC Web)

Sapelo2 Cluster



Please note:
You need to connect to the UGA VPN when accessing from outside of the UGA main campus.

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

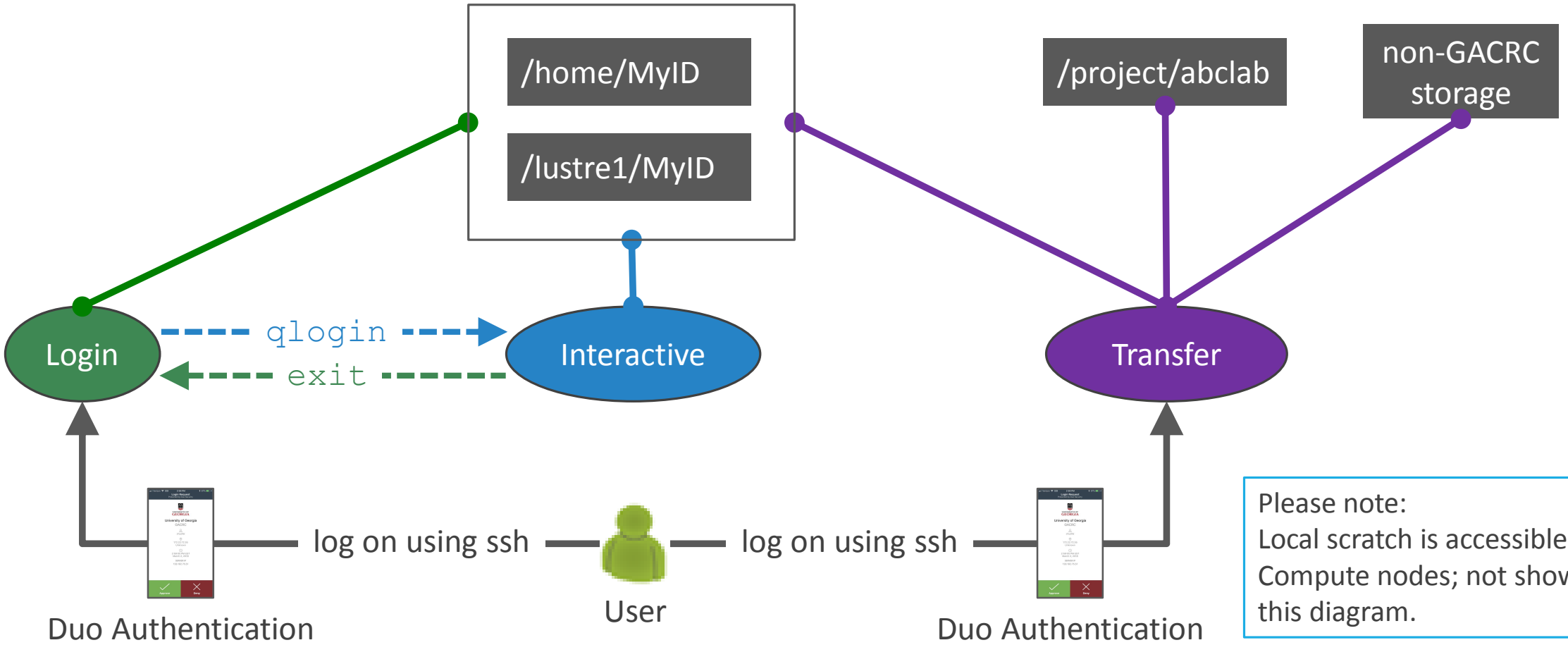
Sapelo2: A Linux HPC cluster (64-bit Centos 7)

- 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
- Four Directories:
  1. Home: Landing spot; 100GB quota; Backed-up
  2. Global Scratch: High-performance job working space; NO quota; NOT backed-up
  3. Local Scratch: Local storage on compute node; 200GB; NOT backed-up
  4. Storage: Temporary data parking; 1TB quota; Backed-up (**ONLY accessible from Transfer node!**)
- Four Computational Queues: batch, highmem_q, gpu_q, groupBuyin_q

Four Directories

4 Directories	Role	Quota	Accessible from	Intended Use	Backed-up	Notes
/home/MyID	Home	100GB	Login Transfer	Static data: 1. Scripts, source codes 2. Local software	Yes	
/lustre1/MyID	Global Scratch	No Limit		Current job data: data being read/written by running jobs	No	**User to clean up!** *Subject to deletion in 30 days
/project/abclab	Storage	1TB (initial)	Transfer	Temporary data parking: non-current active data	Yes	Group sharing possible
/lscratch	Local Scratch	~200GB	Compute	Jobs with heavy disk I/O	No	1. **User to clean up** when job exits from the node! 2. **Persistent data**

Accessing Directories from Nodes



Please note:
Local scratch is accessible from Compute nodes; not shown in this diagram.

4 Computational Queues

Queue	Node Feature	Total Nodes	RAM(GB) /Node	Max RAM(GB) /Single-node Job	Cores /Node	Processor Type	GPU Cards /Node	InfiniBand
batch	Intel	30	64	**62**	**28**	Intel Xeon	N/A	Yes
		42	192	**188**	**32**	Intel Xeon (Skylake)		
	AMD	90	128	**125**	**48**	AMD Opteron		
highmem_q	Intel/AMD	4/1	1024	**997**	**28**	Intel Xeon		
	AMD/Intel	4/1	512	**503**	**48**	AMD Opteron		
gpu_q	GPU	2	128	**125**	**16**	Intel Xeon	8 NVIDIA K40	
		2	96/80	**92/76**	**12**		7 NVIDIA K20	
		4	192	**188**	**32**	Intel Xeon (Skylake)	1 NVDIA P100	
groupBuyin_q	variable							

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/2.7.14-foss-2016b
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
   - `ml spider pattern` : Search module names matching a pattern (case-insensitive)

Job Submission Workflow

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

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 global scratch : `cd /lustre1/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`

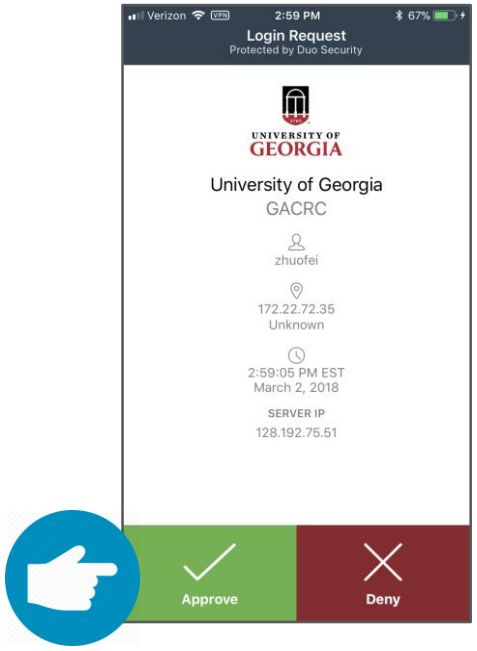
Step1: Log on to Login node - Mac/Linux using ssh

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

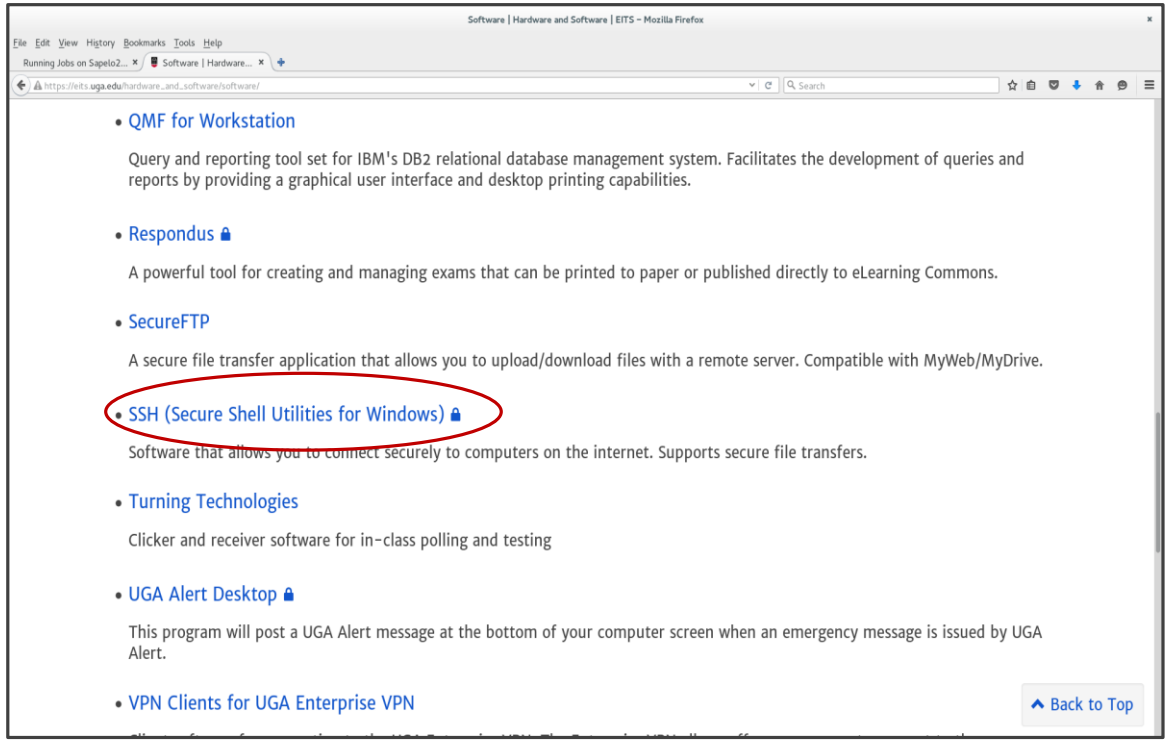
Use Terminal utility on Mac or Linux!

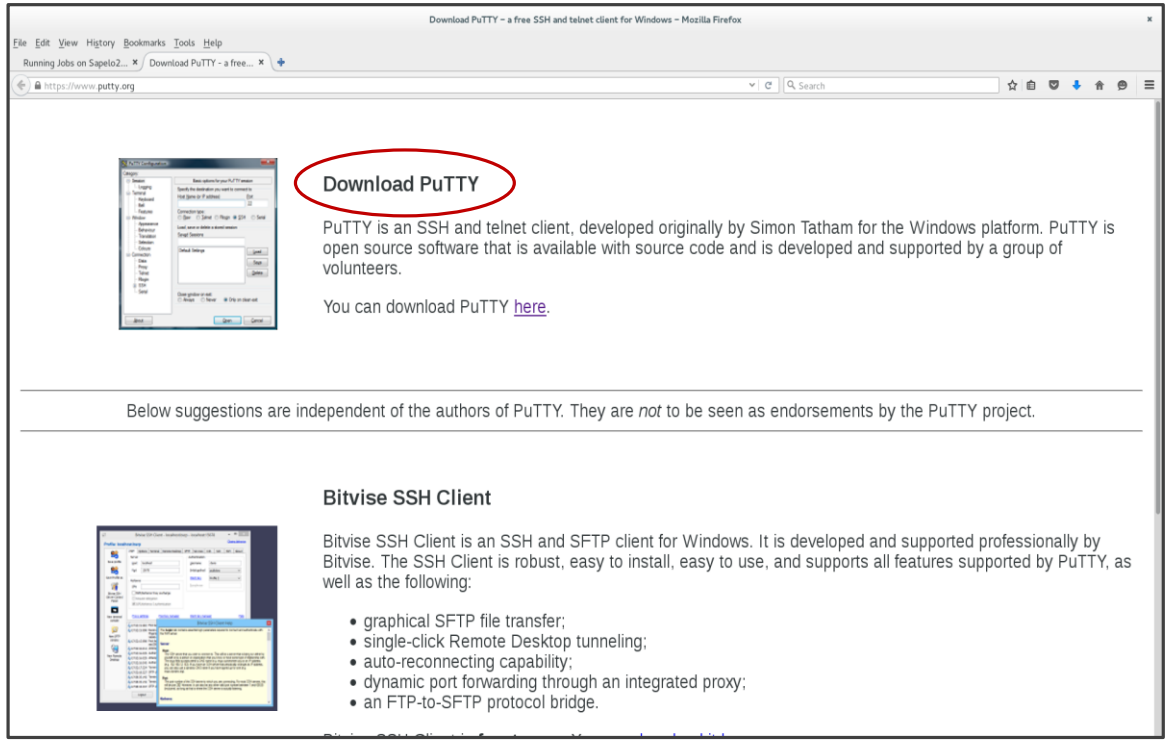



```
zhuofeihou — zhuofei@sapelo2-sub2:~ — ssh — 80×22
zhuofeihou@s172-20-19-h149:~$
zhuofeihou@s172-20-19-h149:~$
zhuofeihou@s172-20-19-h149:~$ssh zhuofei@sapelo2.gacrc.uga.edu   ← Log on
Password:                                                        ← Input MyID password!
Duo two-factor login for zhuofei

Enter a passcode or select one of the following options:

 1. Duo Push to XXX-XXX-5758
 2. Phone call to XXX-XXX-1925
 3. Phone call to XXX-XXX-5758
 4. SMS passcodes to XXX-XXX-5758

Passcode or option (1-4): 1                                      ← Select Duo login option!

Pushed a login request to your device...
Success. Logging you in...
zhuofei@sapelo2-sub2 ~$
zhuofei@sapelo2-sub2 ~$                                          ← I am on sapelo2-sub2!
```

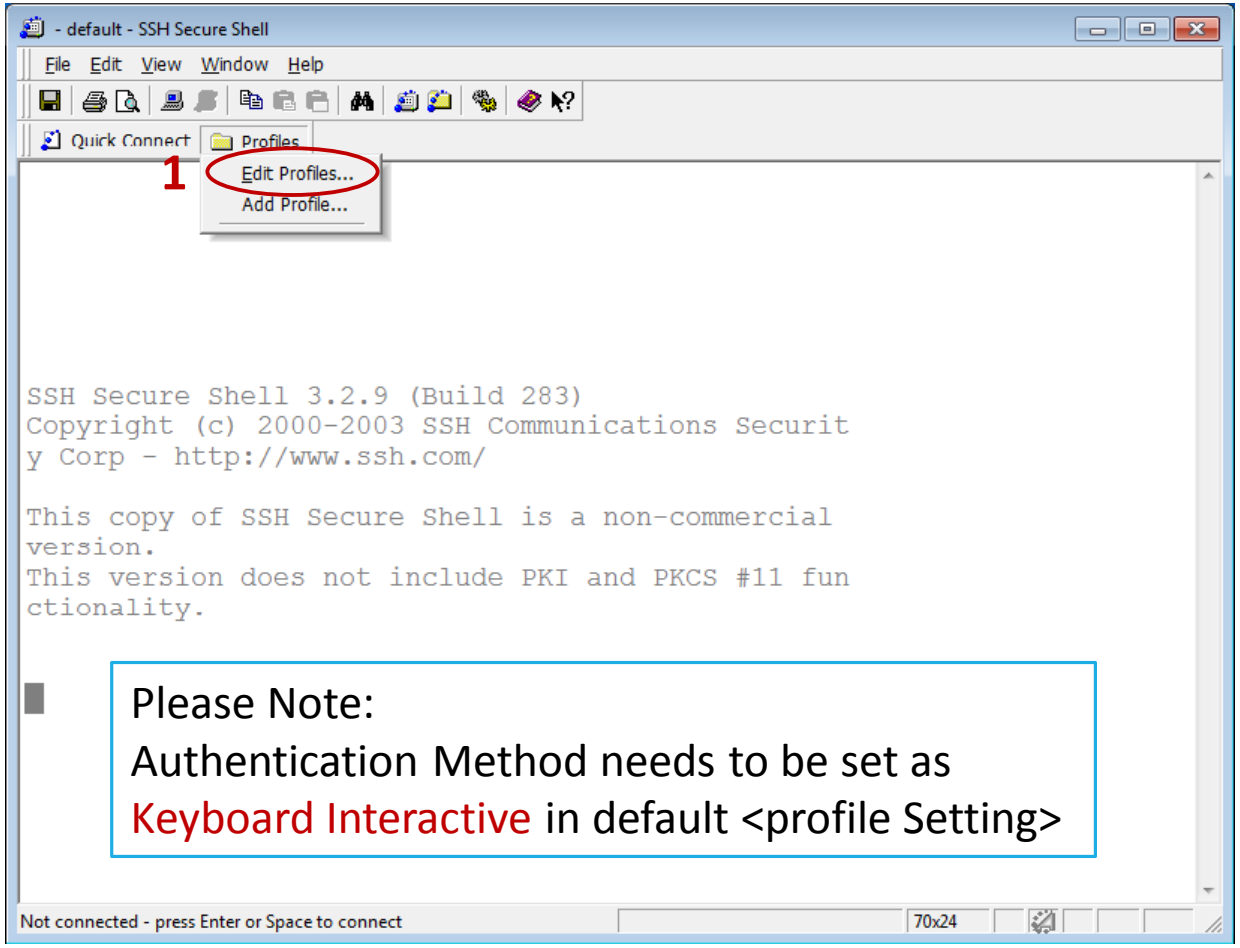
Step1 (Cont.) - Windows using SSH Secure Utilities

1. Download and install SSH Secure Utilities: http://eits.uga.edu/hardware_and_software/software/
2. You can use PuTTY as an alternative: https://www.putty.org/

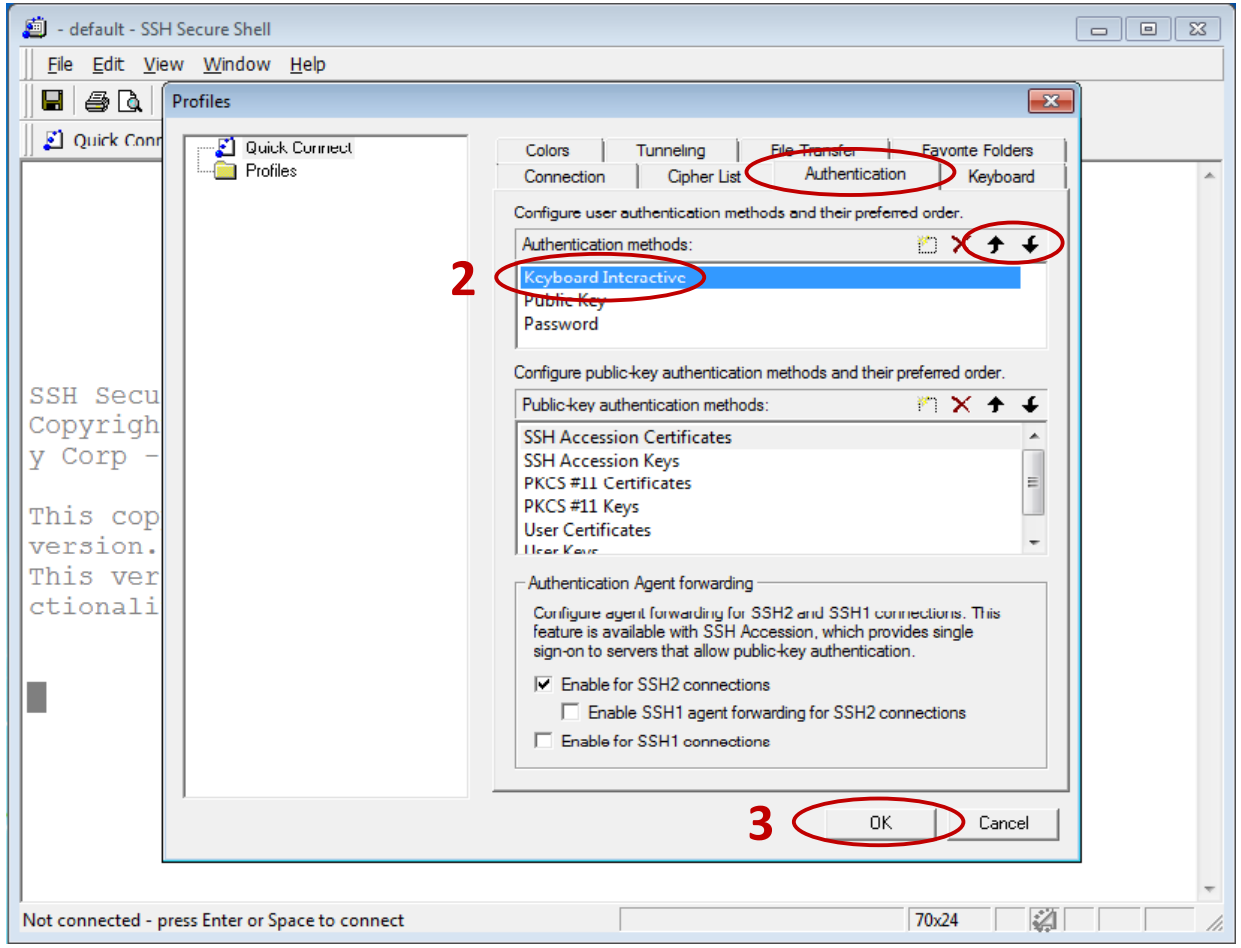




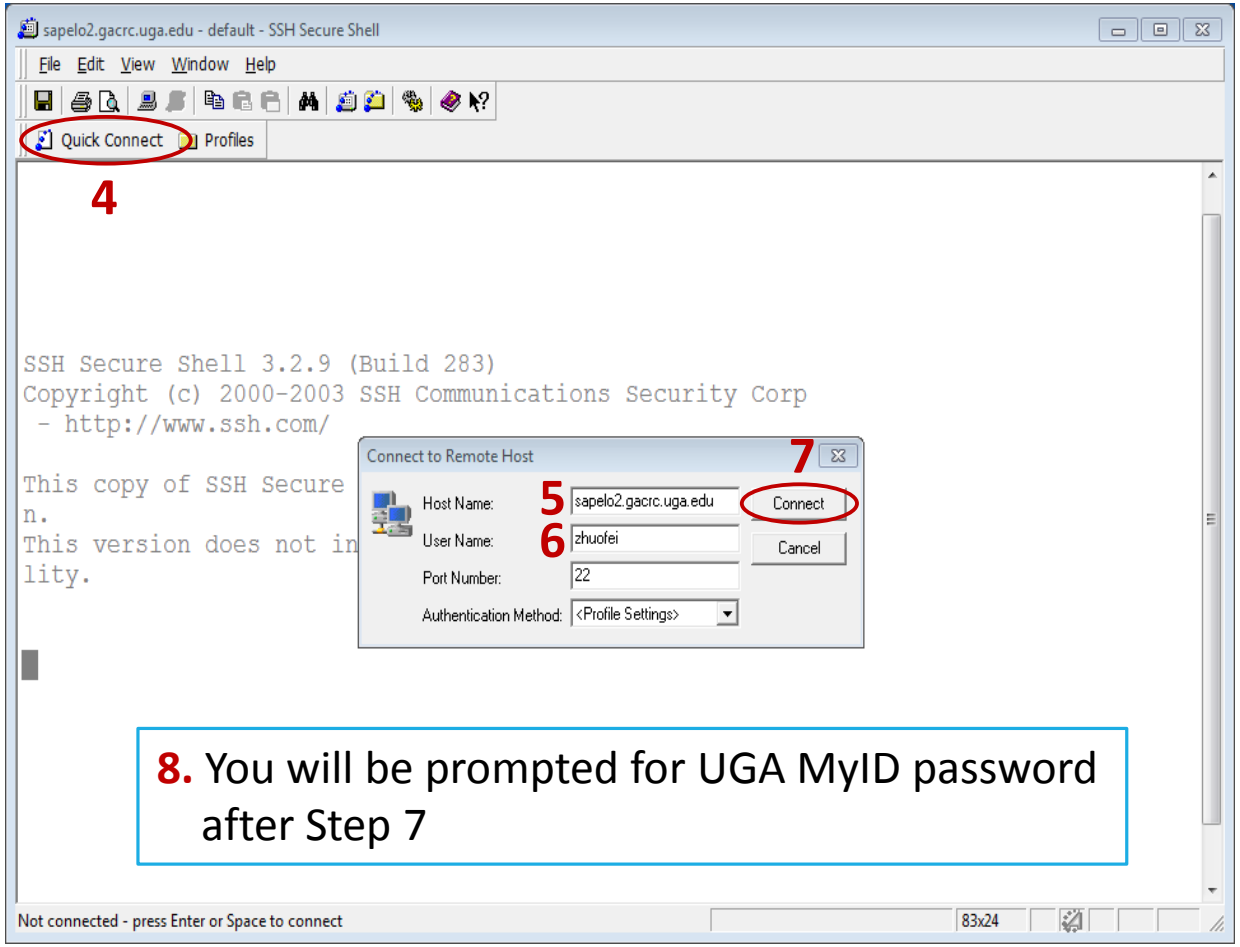
Step1 (Cont.) - Windows using SSH Secure Utilities



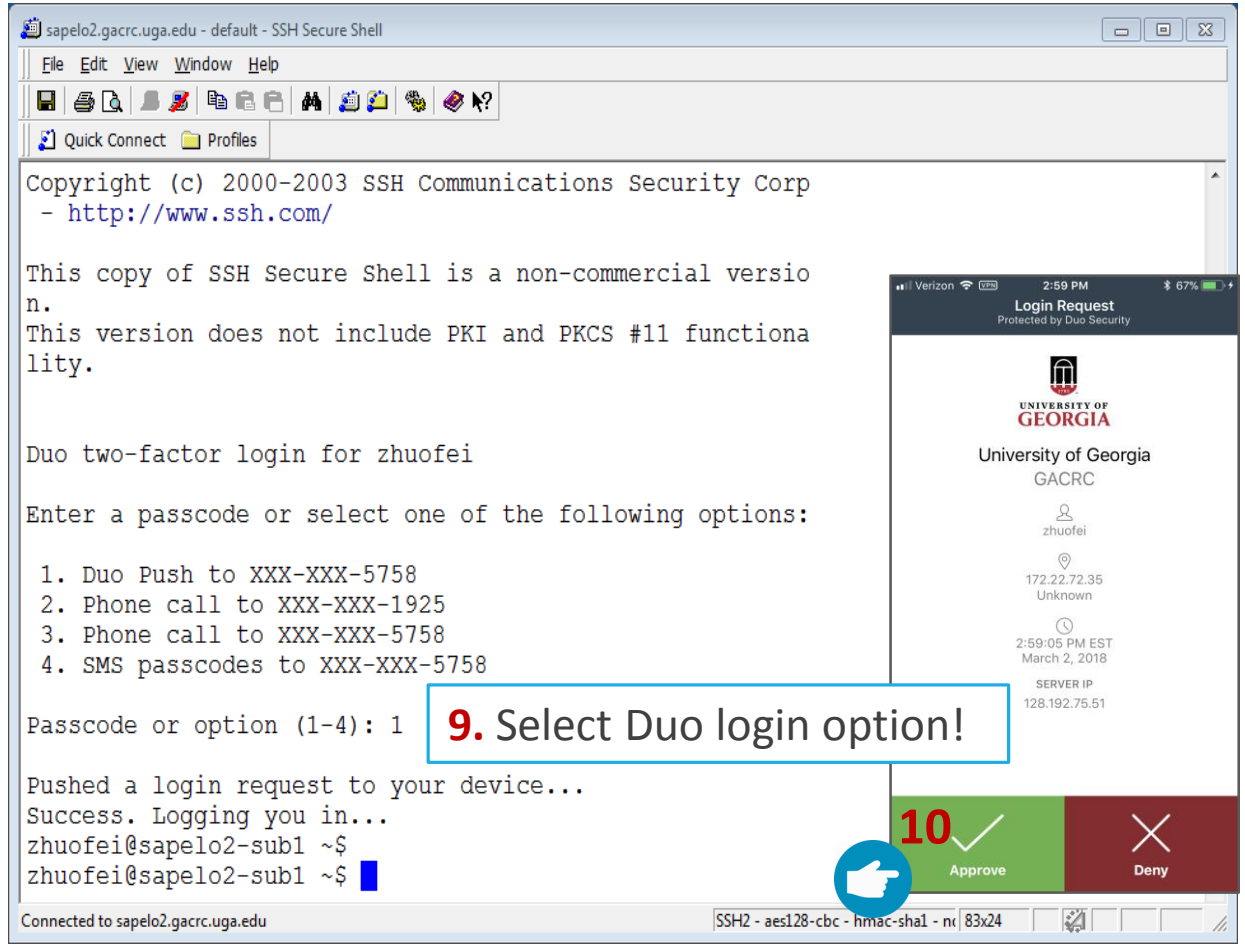
Please Note:
Authentication Method needs to be set as
Keyboard Interactive in default <profile Setting>



Step1 (Cont.) - Windows using SSH Secure Utilities



8. You will be prompted for UGA MyID password after Step 7



9. Select Duo login option!

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 /lustre1/MyID

```
zhuofei@sapelo2-sub2 /$ cd /lustre1/zhuofei
zhuofei@sapelo2-sub2 zhuofei$ pwd
/lustre1/zhuofei                       ← this is my global scratch folder!
```

- Use `ls` command to take a look in /lustre1/MyID

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

Step3 - 4: Create and cd to a working subdirectory

- Use `mkdir` command to make a subdirectory in /lustre1/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 /lustre1/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:/lustre1/zhuofei/workDir/
```

```
scp -r ./folder/ zhuofei@xfer.gacrc.uga.edu:/lustre1/zhuofei/workDir/
```

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

```
scp zhuofei@xfer.gacrc.uga.edu:/lustre1/zhuofei/workDir/file .
```

```
scp -r zhuofei@xfer.gacrc.uga.edu:/lustre1/zhuofei/workDir/folder/ .
```

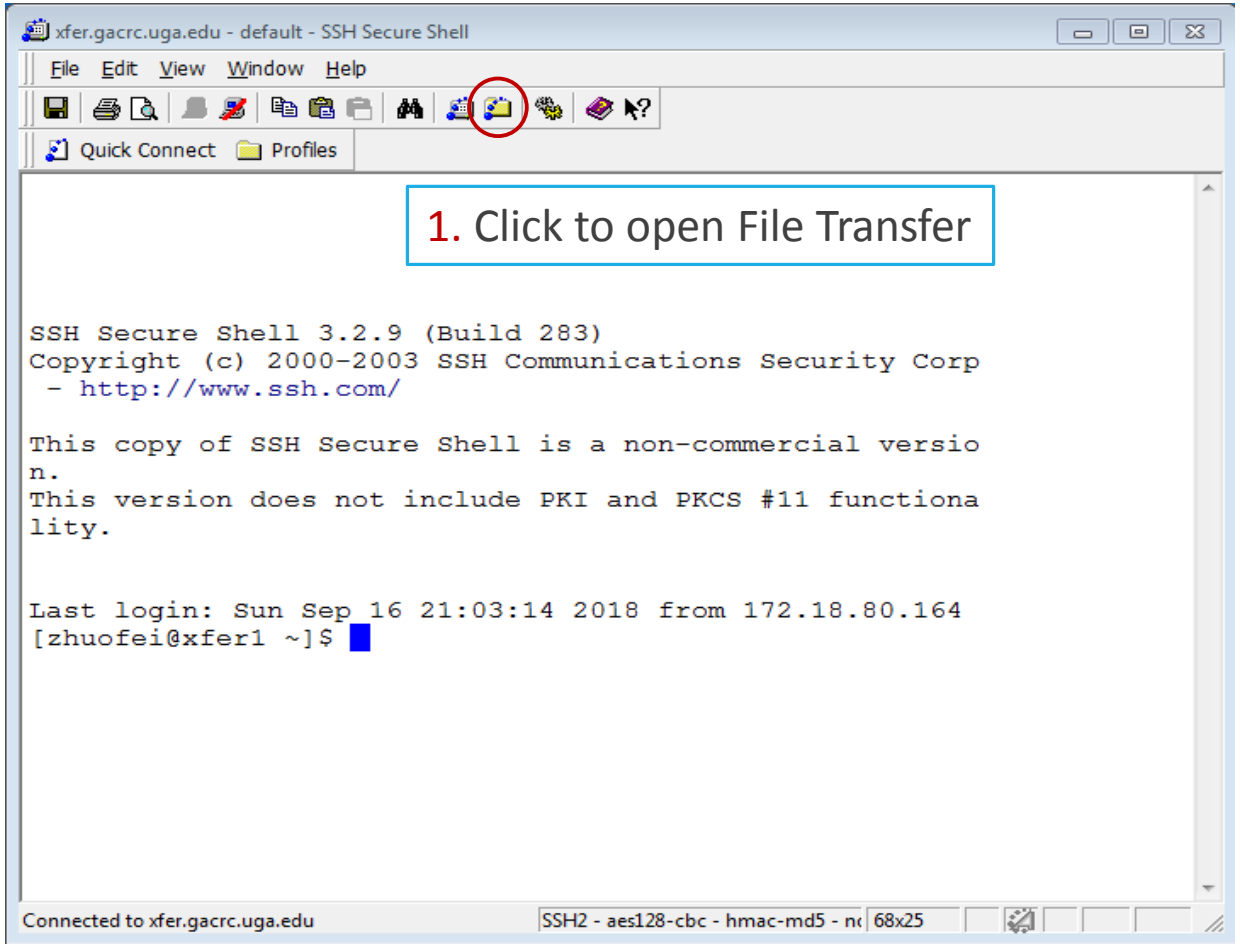
Step5 (Cont.) - Windows using SSH Secure Utilities

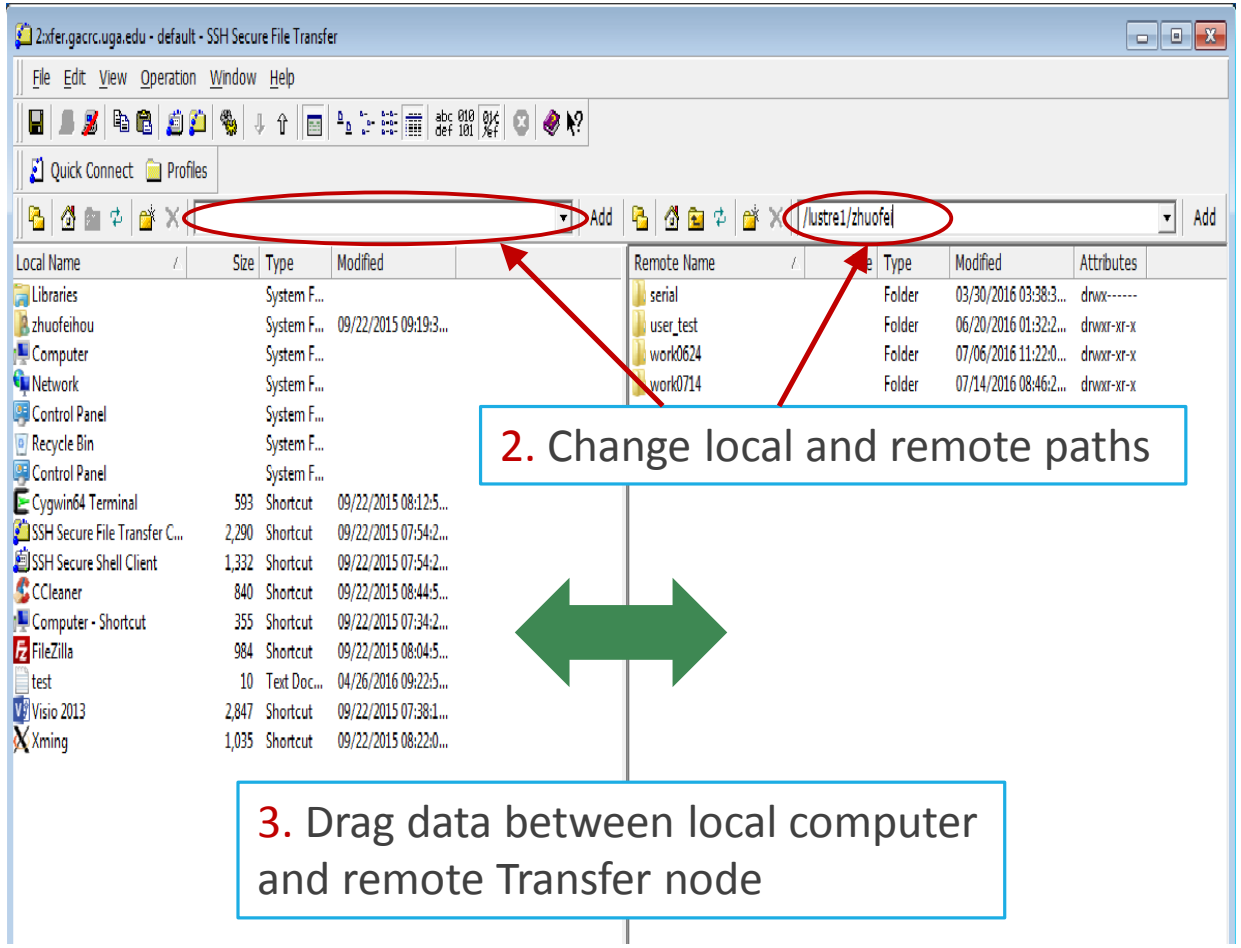
1. You need to connect to cluster's Transfer node (xfer.gacrc.uga.edu)
2. Use SSH File Transfer on local computer (alternative FileZilla or WinSCP)
3. Steps 1-8 are the same as steps on page 14-15, except for Host Name in step 5:

   Host Name: xfer.gacrc.uga.edu

4. Step 9-10 are not required for logging on Transfer node, as of 2018-09-18
5. Once you log on, use File Transfer of SSH Secure Utilities, as shown on next page

Step5 (Cont.) - Windows using SSH Secure Utilities





Step5 (Cont.): Transfer data on cluster to workDir

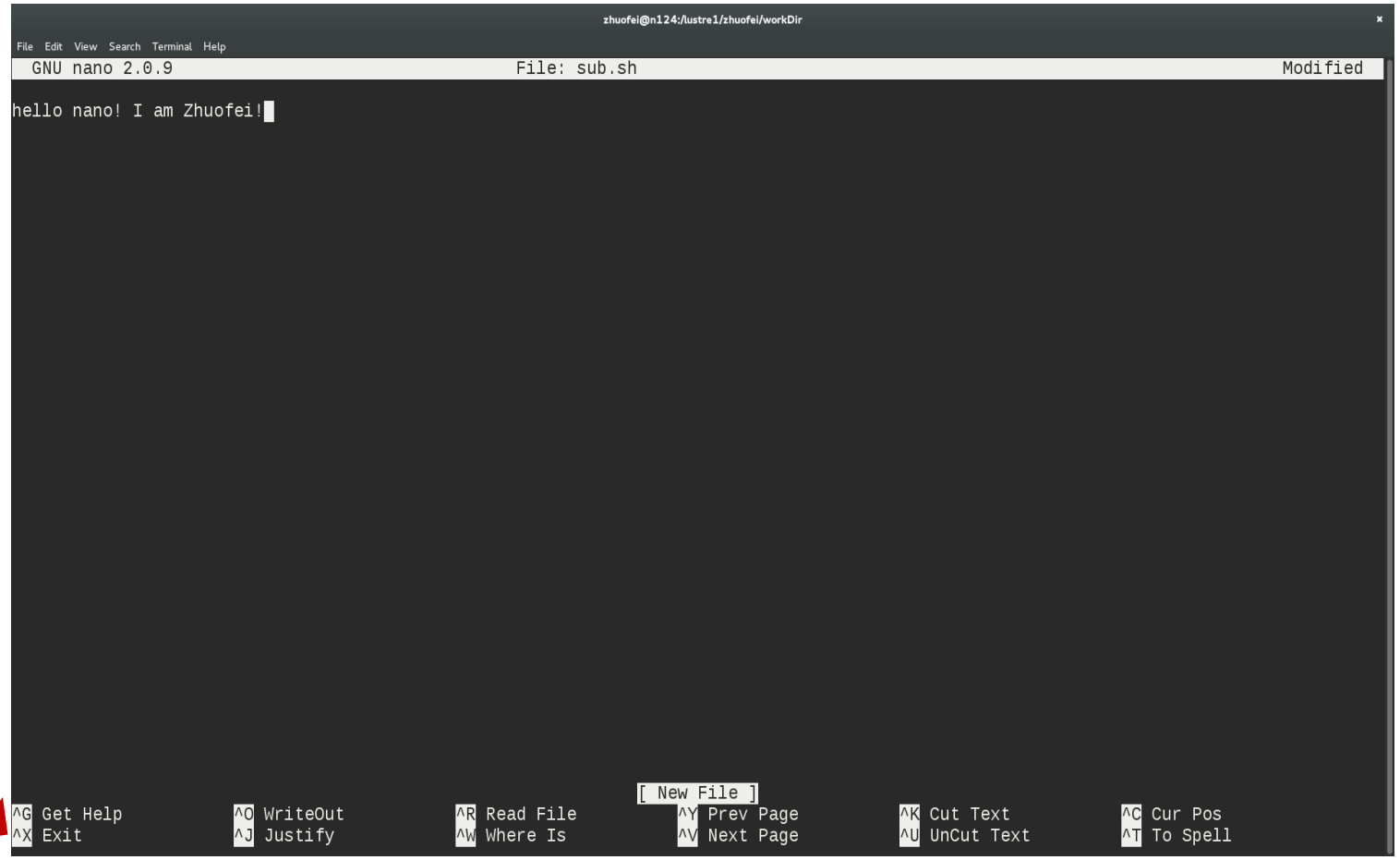
- Log on to Transfer node (xfer.gacrc.uga.edu)
  - ✓ Mac/Linux: ssh MyID@xfer.gacrc.uga.edu (page 12)
  - ✓ Windows: use SSH Secure Utilities (page 19)
- Landing folder: /home/MyID (Home)
- Transfer data between folders on cluster using cp, mv
- Directories you can access using full path:
  1. /home/MyID
  2. /lustre1/MyID
  3. /project/abclab
- 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.



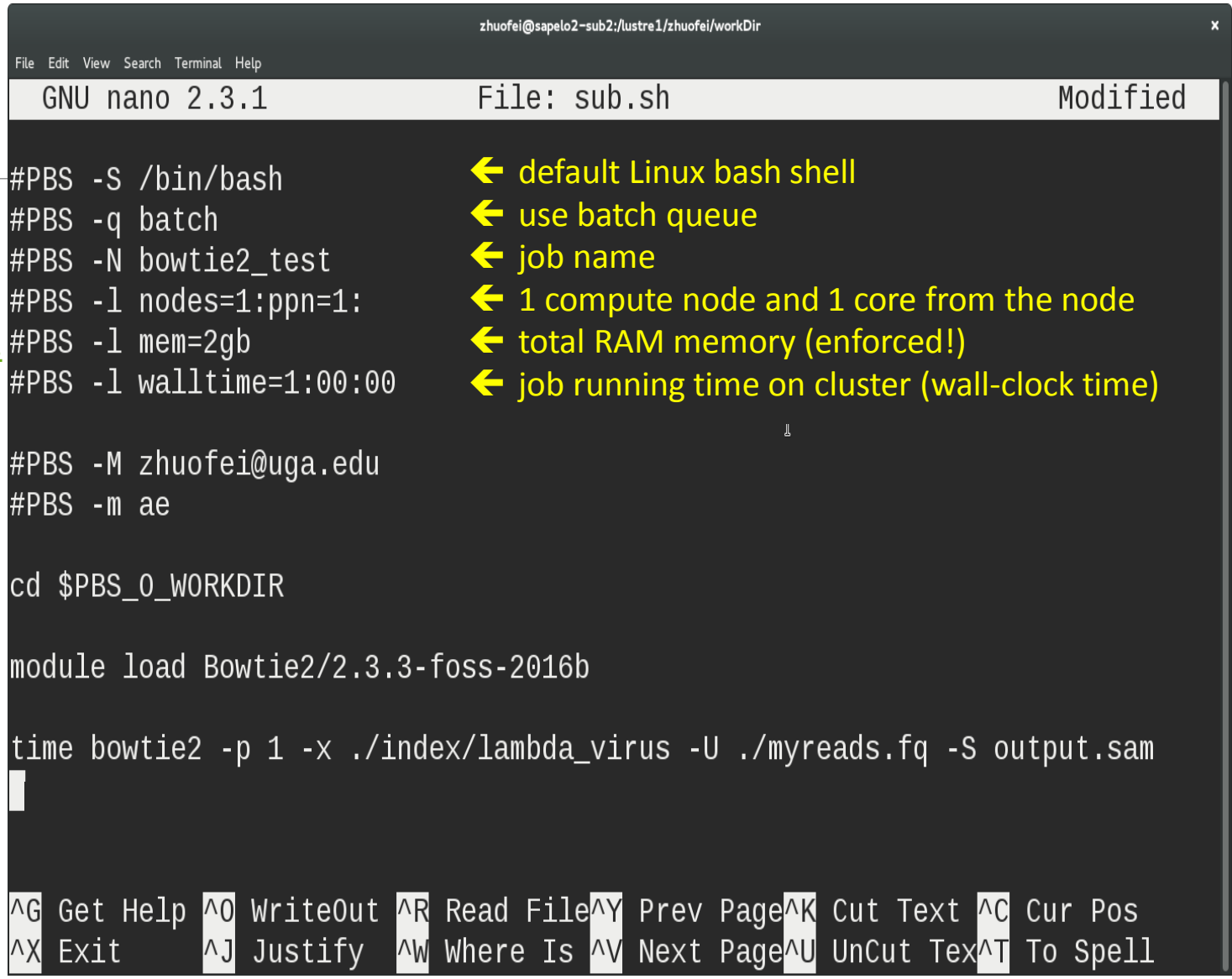
Ctrl-x to save file and quit from nano

Step6 (Cont.)

1. Sample script on GACRC Wiki Software page: 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  .
```



Step7: 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
/lustre1/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 ml load
3. run any Linux commands you want to run
4. run the software

Step8: Check job status using qstat_me

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

```
zhuofei@sapelo2-sub2 workDir$ qstat_me
Job ID               Name             User      Time Use S Queue
-------------------- ---------------- --------- -------- - -----
11943.sapelo2        bowtie2_test     zhuofei   00:06:40 C batch
11944.sapelo2        bowtie2_test     zhuofei   00:05:17 R batch
11946.sapelo2        bowtie2_test     zhuofei   00:12:51 R batch
11947.sapelo2        bowtie2_test     zhuofei          0 R batch
11948.sapelo2        bowtie2_test     zhuofei          0 Q batch
```

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

Step8 (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
Job ID                    Name             User            Time Use S Queue
------------------------- ---------------- --------------- -------- - -----
11943.sapelo2             bowtie2_test     zhuofei         00:06:40 C batch
11944.sapelo2             bowtie2_test     zhuofei         00:05:17 R batch
11946.sapelo2             bowtie2_test     zhuofei         00:12:51 R batch
11947.sapelo2             bowtie2_test     zhuofei         00:00:09 C batch
11948.sapelo2             bowtie2_test     zhuofei                0 Q batch
```

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:
                                                                  Req'd   Req'd       Elap
Job ID          Username  Queue   Jobname        SessID  NDS  TSK  Memory  Time      S  Time
--------------- --------  -----  ------------  ------  ---  ---  ------  --------  -  --------
12175.sapelo2 zhuofei    batch  bowtie2_test  132442  1    1    2gb     01:00:00  R  00:23:44
   n238/0  ← job is running on node238/CPU0
12176.sapelo2 zhuofei    batch  bowtie2_test  67226   1    1    2gb     01:00:00  R  00:20:44
   n237/0
12177.sapelo2 zhuofei    batch  bowtie2_test  119643  1    1    2gb     01:00:00  R  00:05:44
```

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

Step8 (Cont.): Check all Jobs on cluster using qstat

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

```
zhuofei@sapelo2-sub2 workDir$ qstat
Job ID                    Name             User            Time Use S Queue
------------------------- ---------------- --------------- -------- - -----
11267.sapelo2              L80-500          jx57780          164:32:5 R batch
11269.sapelo2              L80-502          jx57780          164:55:5 C batch
11270.sapelo2              L80-503          jx57780          165:38:5 C batch
11607.sapelo2              canu             gd98309          3414:46: R bergman_q
11726.sapelo2              ..._3_constoptTS sm39091          3157:30: R wheeler_q
11729.sapelo2              ..._2_constoptTS sm39091          2731:29: R wheeler_q
11790.sapelo2              sp2_run19b_dye   castelao         4412:52: C batch
11804.sapelo2              ...e-4_Nnoise=64 cotter           98:26:20 R batch
11806.sapelo2              ...e-4_Nnoise=64 cotter           98:14:22 R batch
11987.sapelo2              ..._th_W18-T5-L4 qbcg             08:02:40 C batch
11989.sapelo2              matlabjob        zhyw86                  0 C batch
11990.sapelo2              ..._1_constoptTS sm39091          445:42:1 R wheeler_q
11991.sapelo2              ..._1_constoptTS sm39091          444:51:4 R wheeler_q
11992.sapelo2              ..._cl_W18-T3-D1 qbcg             03:04:21 C batch
.
.
```

Note: qstat command will give you a long list of all jobs from all users on cluster!

Step8 (Cont.): Check all Jobs on cluster using qstat

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

```
zhuofei@sapelo2-sub2 workDir$ qstat
Job ID                    Name             User            Time Use S Queue
------------------------- ---------------- --------------- -------- - -----
11267.sapelo2              L80-500           jx57780          164:32:5 R batch
11269.sapelo2              L80-502           jx57780          164:55:5 C batch
11270.sapelo2              L80-503           jx57780          165:38:5 C batch
11607.sapelo2              canu              gd98309          3414:46: R bergman_q
11726.sapelo2              ..._3_constoptTS  sm39091          3157:30: R wheeler_q
11729.sapelo2              ..._2_constoptTS  sm39091          2731:29: R wheeler_q
11790.sapelo2              sp2_run19b_dye    castelao         4412:52: C batch
11804.sapelo2              ...e-4_Nnoise=64  cotter           98:26:20 R batch
11806.sapelo2              ...e-4_Nnoise=64  cotter           98:14:22 R batch
11987.sapelo2              ..._th_W18-T5-L4  qbcg             08:02:40 C batch
11989.sapelo2              matlabjob         zhyw86                  0 C batch
11990.sapelo2              ..._1_constoptTS  sm39091          445:42:1 R wheeler_q
11991.sapelo2              ..._1_constoptTS  sm39091          444:51:4 R wheeler_q
11992.sapelo2              ..._cl_W18-T3-D1  qbcg             03:04:21 C batch
.
.
```

Note: qstat command will give you a long list of all jobs from all users on cluster!

Workflow Diagram

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

Windows user:

Login

2. `cd /lustre1/MyID`

3. `mkdir ./workDir`
4. `cd ./workDir`

6. `nano ./sub.sh`

```
#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……
```

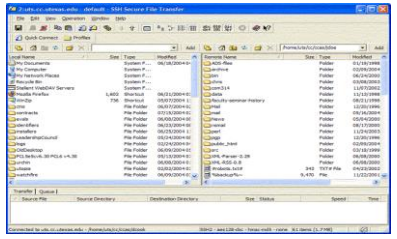
7. `$ qsub sub.sh`

8. `$ qstat_me or qdel JobID`

lustre1

5. Linux/Mac user:
`scp file MyID@xfer.gacrc.uga.edu:/lustre1/MyID/workDir`

Windows user:



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

Running Jobs: https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_Sapelo2

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

Job Submission Queue: https://wiki.gacrc.uga.edu/wiki/Job_Submission_Queues

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

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

User Account Request: https://wiki.gacrc.uga.edu/wiki/User_Accounts

GACRC 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

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

GACRC's HelpDesk

The Georgia Advanced Computing Resource Center
A Collaborative Partnership between the *Vice President for Research* and the *Vice President for Information Technology*

Request Support

* indicates Required fields.

Your Name * First & Last

MyID * myid

E-mail * you@example.edu

Phone Number 411-555-1212

Brief Description

Request Details * (less than 2,000 characters)

Cluster ☐ sapelo2 ☐ sapelo ☐ teach ☐ other

Review

* For questions on cluster or software, please include the command/script used, working path and working nodes (interactive / queue name) if applicable.
* For software installation, please specify software name, version and include link to the software if applicable.
* Please review your message on the next page and then click the Submit button.

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

```
#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] …
```

➔ 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)

➔ Run blastn with 4 threads (-num_threads 4)

*Example 3: MPI job script running RAxML using 2 full nodes

```
#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

cd $PBS_O_WORKDIR

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

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

➔ Number of nodes (2), number of cores (28)
Total cores requested = 2 × 28 = 56
We suggest, Number of MPI Processes (50) ≤ Number of cores requested (56)

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

➔ Run raxmlHPC-MPI-AVX with 50 MPI processes (–np 50), default 56

Guideline Tips

- Do NOT use Login node to run CPU/memory intensive jobs directly ➔ submit jobs to queue!
- Do NOT use Login Node to upload/download large data to/from cluster ➔ use Transfer node!
- Do NOT use home dir for storing large job data ➔ use global scratch /lustre1/MyID
- Do NOT park data on global or local scratch ➔ clean up when job finished or exits from node
- NO large memory job running on batch or jlm_q ➔ use highmem_q
- NO small memory job running on highmem_q ➔ use batch or jlm_q
- As a general rule, threads # = cores # requested

Practical Tips

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

All files are in ONE single dir!

Files are organized in subdirs!

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/