Introduction to HPC Using Sapelo Cluster at GACRC

Georgia Advanced Computing Resource Center
EITS/University of Georgia
Zhuofei Hou, zhuofei@uga.edu

Outline

- GACRC
- Sapelo Cluster at GACRC
- Sapelo Workflows
- Work with Sapelo

GACRC

Who Are We?

- **G**eorgia **A**dvanced **C**omputing **R**esource **C**enter
- Collaboration between the Office of Vice President for Research (**OVPR**) and the Office of the Vice President for Information Technology (**OVPIT**)
- Guided by a faculty advisory committee (GACRC-AC)

Why Are We Here?

- To provide computing hardware and network infrastructure in support of high-performance computing (**HPC**) at UGA

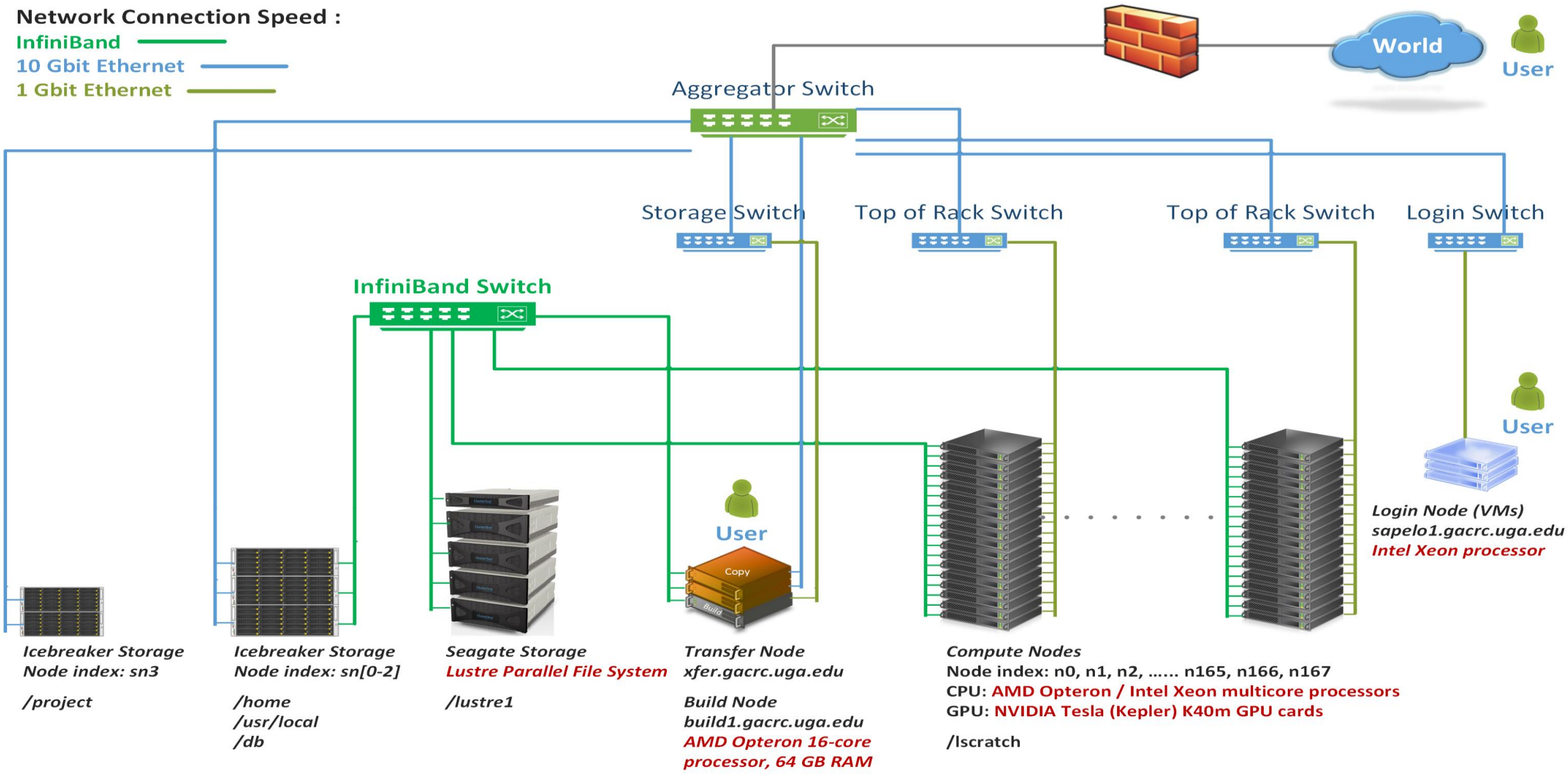
Where Are We?

- http://gacrc.uga.edu (Web) http://wiki.gacrc.uga.edu (Wiki)
- http://gacrc.uga.edu/help/ (Web Help)
- https://wiki.gacrc.uga.edu/wiki/Getting_Help (Wiki Help)

Sapelo Cluster at GACRC

- Cluster Structural Diagram
- Cluster Overview
- Computing Resources
- Storage Environment

The New GACRC Linux HPC Cluster Structural Diagram



Cluster Overview

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

- OS: 64-bit CentOS Linux 6.5
- You can log on to:

  Login (username@sapelo1.gacrc.uga.edu) : edit script, submit batch job

  Transfer (username@xfer.gacrc.uga.edu) : transfer, compress, package data

  Build (username@build1.gacrc.uga.edu) : compile, test

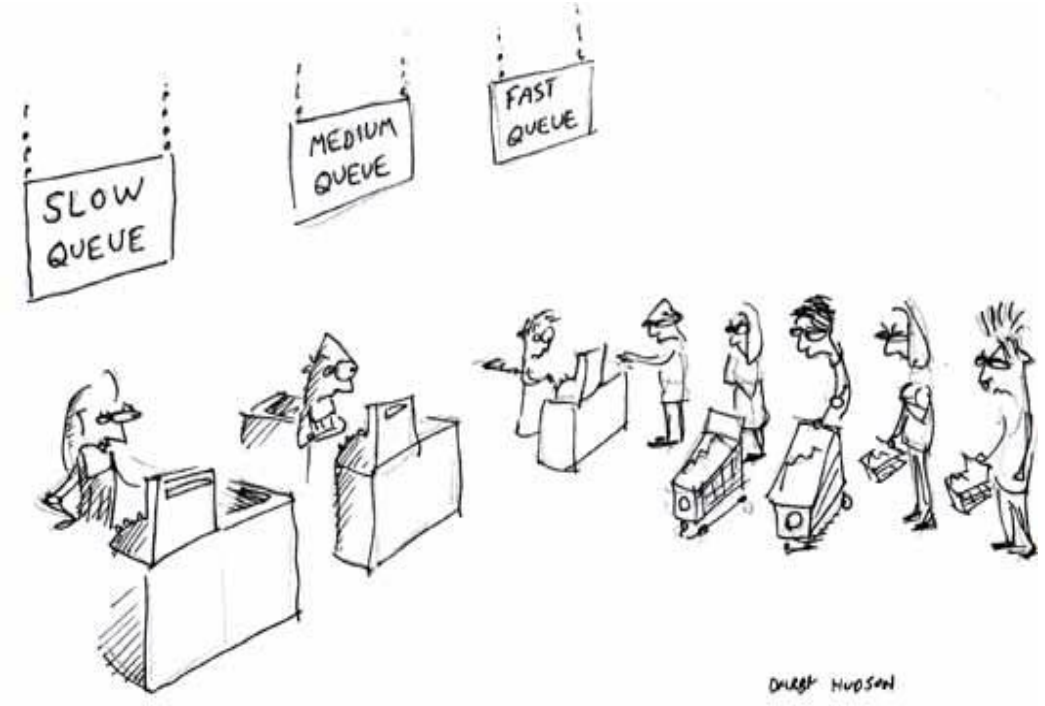
- Login → (qlogin) Interactive Node : run interactive job, edit script, submit batch job
- Internodal Communication: **InfiniBand network**

  compute nodes ⇔ compute nodes ⇔ storage systems, e.g., /home and /lustre1

Cluster Overview

- Batch-queueing System:
  - Jobs can be started (submitted), monitored, and controlled
  - Determine which compute node is the best place to run a job
  - Determine appropriate execution priority for a job to run

- On Sapelo:
  - Torque Resource Manager + Moab Workload Manager
  - Queueing commands: `qsub`, `qstat`, `qdel` `showjobs`, `showq`

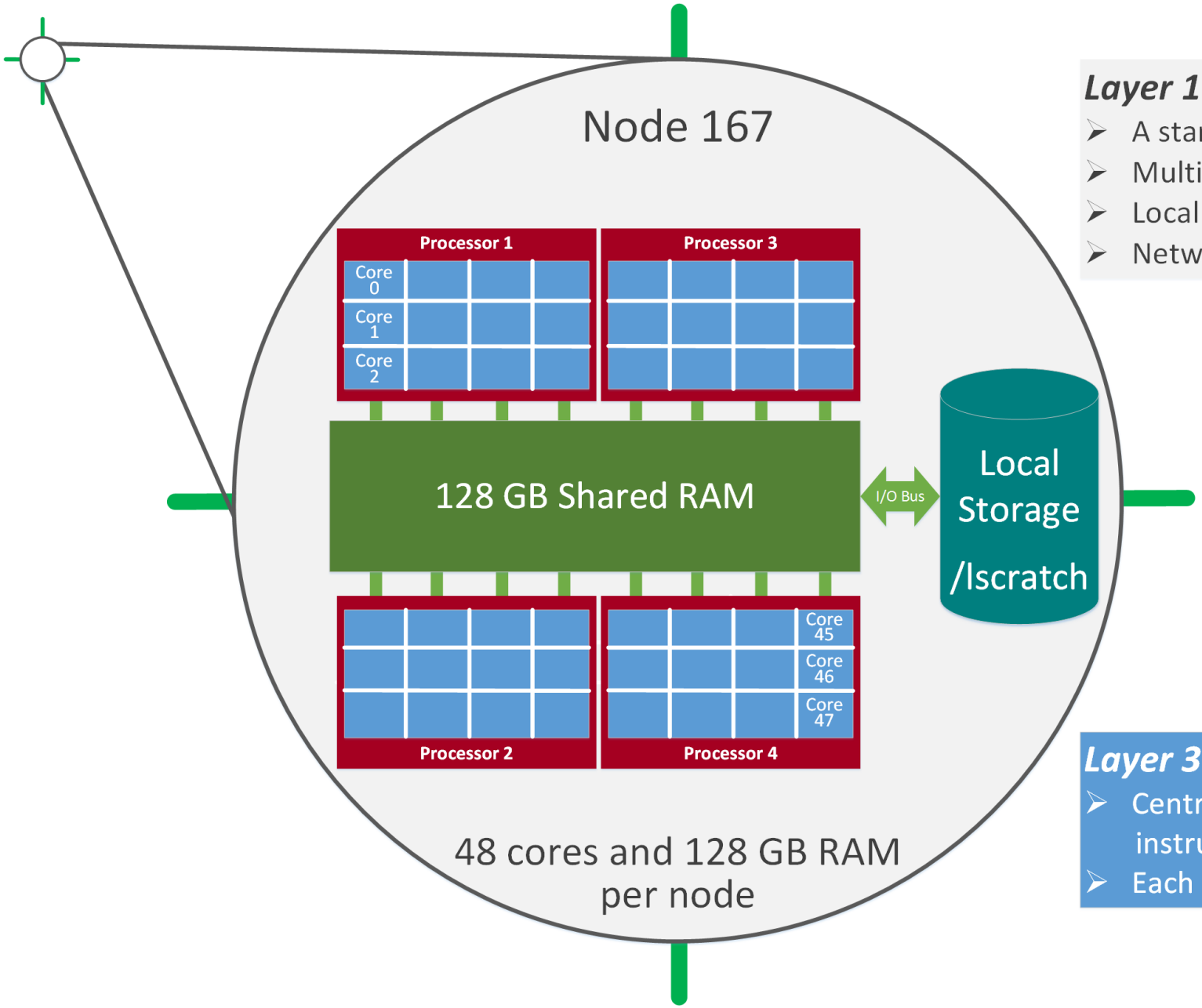


Computing Resources

Queue	Node Feature	Total	Processor	Cores /Node	RAM (GB) /Node	Max RAM (GB) /Single-node Job	GPU	GPU Cards /Node	InfiniBand
batch	AMD	112	AMD Opteron	48	128	126	N/A	N/A	Yes
		4			256	252			
	HIGHMEM	7	AMD Opteron	48	512 (6)	504	N/A	N/A	Yes
					1024 (1)	997			
	GPU	2	Intel Xeon	16	128	126	NVIDIA K40m	8	Yes
	abcnode (buy-in)	2	AMD Opteron	48	256	252	N/A	N/A	Yes

Home: /home/username: 100GB

Global scratch: /lustre1/username: NO quota limit, auto-moved to /project if no modification in 30 days!



Layer 1: Node

- A standalone "computer in a box"
- Multiple processors, e.g. 4, sharing memory
- Local disk storage, network interface, etc.
- Networked into a cluster

Layer 2: Processor

- A single computing component
- Multicore processor, e.g. 12 cores

Layer 3: Core

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

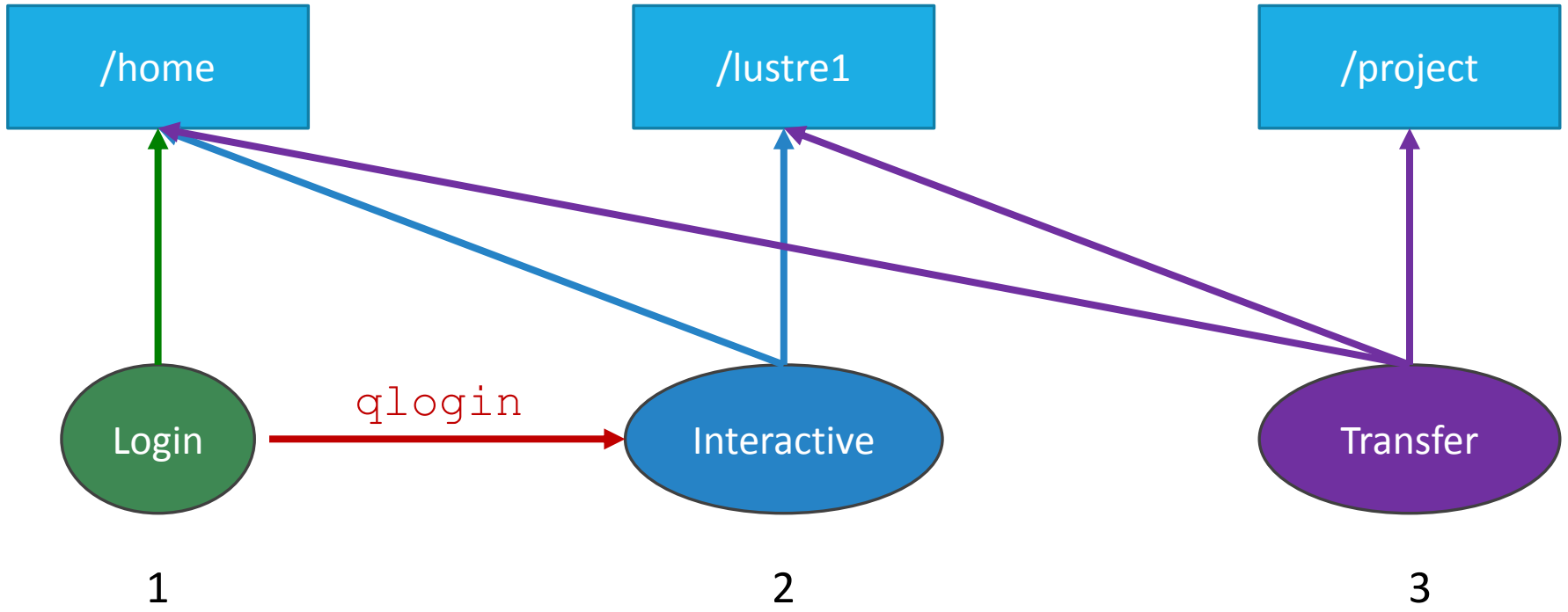
Storage Environment

4 Filesystems	Role	Quota	Accessible from	Intended Use	Notes
/home/username	Home	100GB	sapelo1.gacrc.uga.edu (Login) Interactive nodes (Interactive) xfer.gacrc.uga.edu (Transfer) build1.gacrc.uga.edu (Build) compute nodes (Compute)	Highly static data being used frequently	Snapshots
/lustre1/username	Global Scratch	No Limit	Interactive nodes (Interactive) xfer.gacrc.uga.edu (Transfer) compute nodes (Compute)	Temporarily storing large data being used by jobs	Auto-moved to /project if 30 days no modification
/lscratch/username	Local Scratch	250GB	Individual compute node	Jobs with heavy disk I/O	User to clean up
/project/abclab	Storage	Variable	xfer.gacrc.uga.edu (Transfer)	Long-term data storage	Group sharing possible

Note: /usr/local/apps : Software installation directory
/db : Bioinformatics database installation directory

Storage Environment

Accessing Rule of 123:



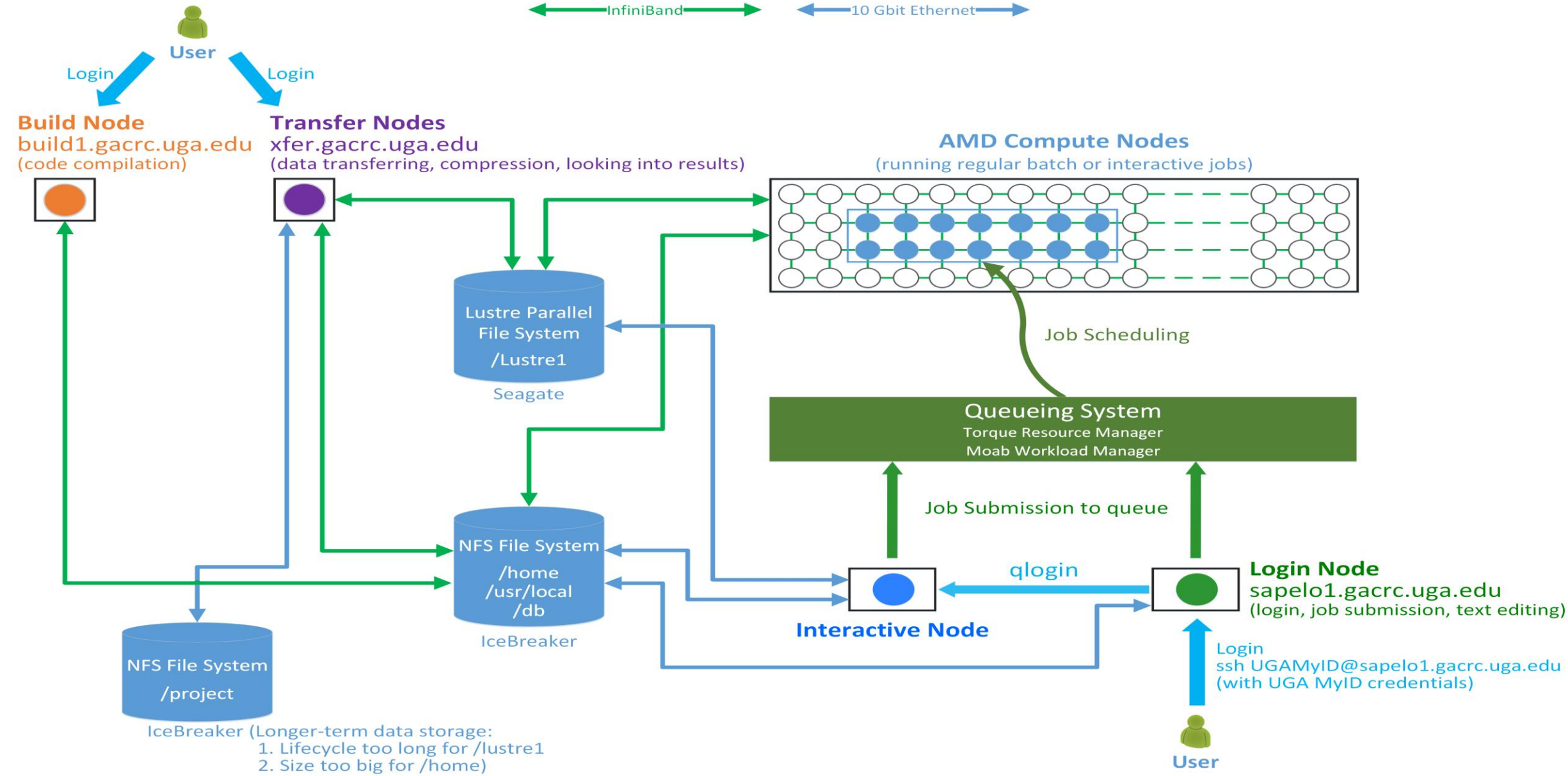
Storage Environment

7 Main Functions	On/From-Node	Related Filesystem
Login Landing	Login or Transfer or Build	/home/username (Home) (Always!)
Batch Job Submitting	Login	/home/username (Home)
	Interactive	/lustre1/username (Scratch) (Suggested!) /home/username (Home)
Interactive Job Running	Interactive	/lustre1/username (Scratch) /home/username (Home)
Data Transferring, Archiving , and Compressing	Transfer	/lustre1/username (Scratch) /home/username (Home)
Job Data Temporarily Storing	Compute	/lscratch/username (Local Scratch) /lustre1/username (Scratch)
Long-term Data Storing	Transfer or Copy	/project/abclab
Code Compilation	Build	/home/username (Home)

Sapelo Workflows

Next Page

The New GACRC Linux HPC Cluster Operational Diagram



Workflow – Global Scratch as Job Working Space

1. Linux/Mac user:

```
ssh userID@sapelo1.gacrc.uga.edu
```

Windows user:

Login

2. `qlogin`

Interactive

3. `cd /lustre1/userID`

4. `mkdir ./workDir`

5. `cd ./workDir`

lustre1

7. `nano ./sub.sh`

```
#PBS -S /bin/bash
#PBS -q batch
#PBS -N testBlast
#PBS -l nodes=1:ppn=1:AMD
#PBS -l mem=20gb
#PBS -l walltime=48:00:00
cd $PBS_O_WORKDIR
module load ncbiblast+/2.2.29
blastn [options] > outputfile
```

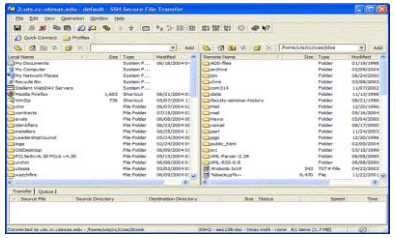
8.

```
$ qsub sub.sh
```

6. Linux/Mac user:

```
scp input userID@xfer.gacrc.uga.edu:/lustre1/userID/workDir
```

Windows user:



Note: inputDate could be files or data folder (`scp -r`)

Useful PBS headers:

```
#PBS mem=200gb : use 256GB AMD nodes)
ppn=24 : request 24 cores for 24 threads
```

Workflow – Global Scratch as Job Working Space

1. Log on to Sapelo Login node: `ssh userID@sapelo1.gacrc.uga.edu`
2. From Sapelo Login node, transfer to Interactive node by issuing the command: `qlogin`
3. From Interactive node: Change directory to global scratch: `cd /lustre1/userID`
4. Create a working subdirectory on global scratch: `mkdir ./workDir`
5. Change directory to `workDir`: `cd ./workDir`
6. Transfer data to `workDir` using `scp` or SSH Client File Transfer (with `tar` or `gzip`)
7. Make a Sapelo job submission script: `nano ./sub.sh`
8. Submit job: `qsub ./sub.sh`

Useful PBS headers: `#PBS -l mem=200gb` : use 256GB high-RAM AMD nodes

`#PBS -l nodes=1:ppn=24:AMD` : request 24 cores for 24 threads, max 48!

Workflow – Home as Job Working Space

1. Linux/Mac user:

```
ssh userID@sapelo1.gacrc.uga.edu
```

Windows user:

Login

2. `mkdir ./workDir`

3. `cd ./workDir`

Home

5. `nano ./sub.sh`

```
#PBS -S /bin/bash
#PBS -q batch
#PBS -N testBlast
#PBS -l nodes=1:ppn=1:AMD
#PBS -l mem=20gb
#PBS -l walltime=48:00:00
cd $PBS_O_WORKDIR
module load ncbiblast+/2.2.29
time blastn [options] > outputfile
```

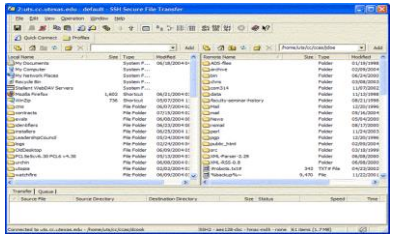
6.

```
$ qsub sub.sh
```

4. Linux/Mac user:

```
scp input userID@xfer.gacrc.uga.edu:~/workDir
```

Windows user:



Note: inputDate could be files or data folder (`scp -r`)

Useful PBS headers:

```
#PBS mem=200gb : use 256GB AMD nodes)
ppn=24 : request 24 cores for 24 threads
```

Workflow – Home as Job Working Space

1. Log on to Sapelo Login node: `ssh userID@sapelo1.gacrc.uga.edu`
2. Create a working subdirectory in home: `mkdir ./workDir`
3. Change directory to `workDir`: `cd ./workDir`
4. Transfer data to `workDir` using `scp` or SSH Client File Transfer (with `tar` or `gzip`)
5. Make a Sapelo job submission script: `nano ./sub.sh`
6. Submit job: `qsub ./sub.sh`

Useful PBS headers: `#PBS -l mem=200gb` : use 256GB high-RAM AMD nodes

`#PBS -l nodes=1:ppn=24:AMD` : request 24 cores for 24 threads, max 48!

Work on Sapelo

Before we start:

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

GACRC Support: https://wiki.gacrc.uga.edu/wiki/Getting_Help

Work on Sapelo

To submit a ticket to us?

- **Job Troubleshooting:**

  Please tell us details of your question or problem, including but not limited to:

  - ✓ Your user name
  - ✓ Your job ID
  - ✓ Your working directory
  - ✓ The queue name and command you used to submit the job

- **Software Installation:**

  - ✓ Specific name and version of the software
  - ✓ Download website
  - ✓ Supporting package information if have

Note:
It's USER's responsibility to make sure the correctness of datasets being used by jobs!

Work on Sapelo – Important Notice

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

What you do may affect others on the cluster!

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

Work on Sapelo

- User Account
- Connect and Login
- Transfer Files Using Transfer Node
- Software Packages
- Run Interactive Jobs
- Run Batch Jobs
  - ✓ Submit Batch Jobs
  - ✓ ***Serial***, ***Threaded***, and ***MPI*** Batch Job Submission Scripts
  - ✓ Check Job Status and Cancel a Job
  - ✓ Check Computing Resources Used by a Job

User Account

- User Account: UGAMyID@sapelo1.gacrc.uga.edu
  A valid official UGA MyID is a MUST to create a user account! 

- To get a user account:

1. Computing Lab Registration: http://help.gacrc.uga.edu/labAcct.php (for PI of a new group)
2. User Account Request: http://help.gacrc.uga.edu/userAcct.php (for PI of an existing group)
3. New User Training: http://gacrc.uga.edu/help/training/
4. Welcome letter with whole package of information about your Sapelo user account

Connect and Login

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

```
ssh zhuofei@sapelo1.gacrc.uga.edu
```

or

```
ssh -X zhuofei@sapelo1.gacrc.uga.edu
```

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

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

- Logging in: You will be prompted for your UGA MyID password:

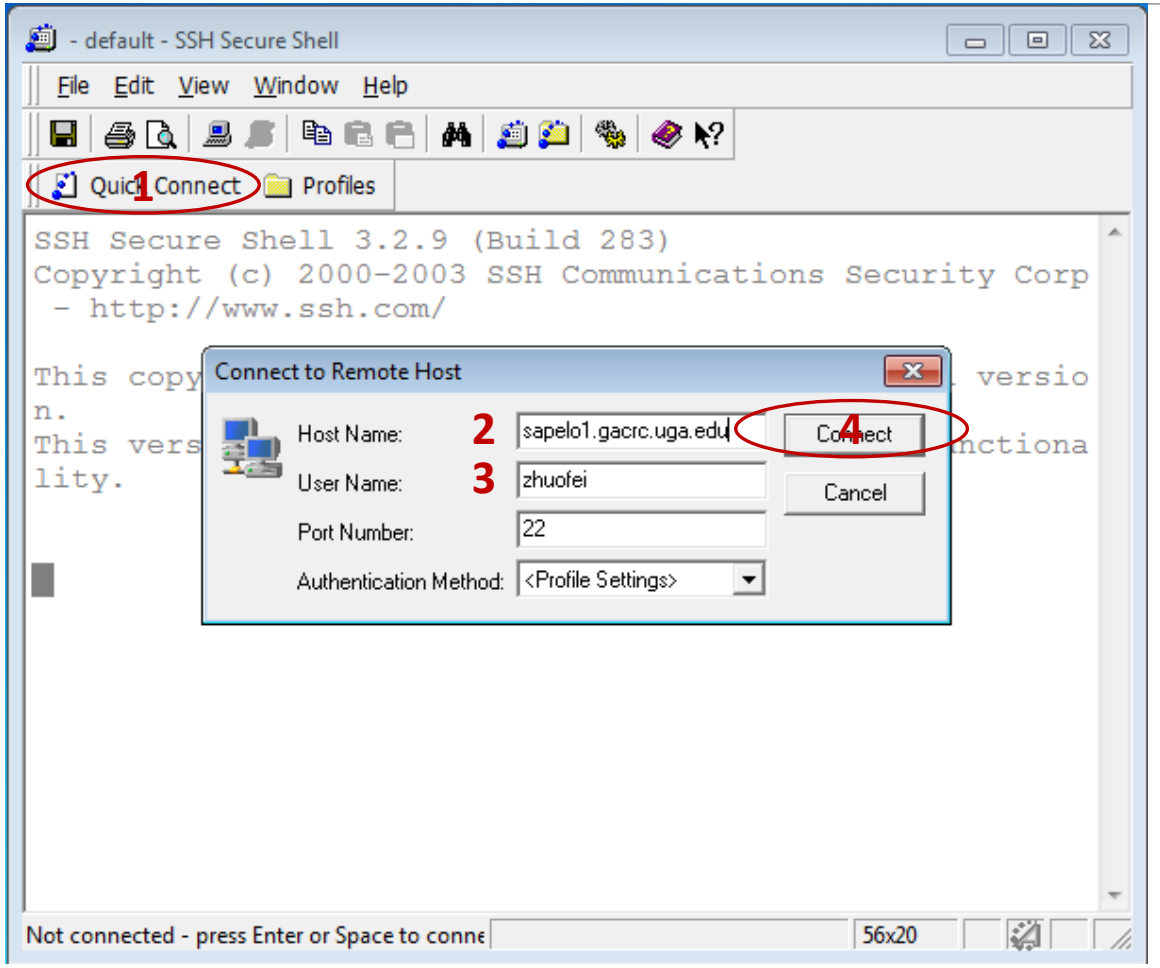
```
zhuofei@sapelo1.gacrc.uga.edu's password:
```

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

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

```
[zhuofei@75-104 ~]$ exit
```

Connect and Login



1. To download:

http://eits.uga.edu/hardware_and_software/software/

with your UGA MyID and password

2. After connection is built, working environment is Linux, same as Linux/Mac users'

Transfer Files Using Transfer Node xfer.gacrc.uga.edu

- ✓ `ssh username@xfer.gacrc.uga.edu` with your UGA MyID password
- ✓ Landing directory: `/home/username` (Sapelo home)
- ✓ Move data into/out of Sapelo (`scp`, `sftp`, `rsync`, SSH Secure Shell File Transfer, FileZilla)
- ✓ Compress or package data on Sapelo (`tar`, `gzip`)
- ✓ Transfer data between Sapelo and zcluster (`cp`, `mv`)
- ✓ Filesystems you can access:
  - /home/username/ : Sapelo home (landing spot)
  - /lustre1/username/ : Sapelo global scratch
  - /panfs/pstor.storage/home/abclab/username/ : zcluster home
  - /escratch4/username/ : zcluster scratch
  - /project/abclab/ : long-term active data storage
- ✓ Most file systems on Transfer are *auto-mounted* upon *the first time full-path access*, e.g., `cd /project/abclab/`

Transfer Files Using Transfer Node xfer.gacrc.uga.edu

User's local  

 Transfer (xfer.gacrc.uga.edu)

- On Linux, Mac or cygwin on Windows : `scp (-r) [Source] [Target]`

E.g. 1: working on local machine, from Local ➔ Sapelo global scratch

```
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 machine, from Sapelo global scratch ➔ Local

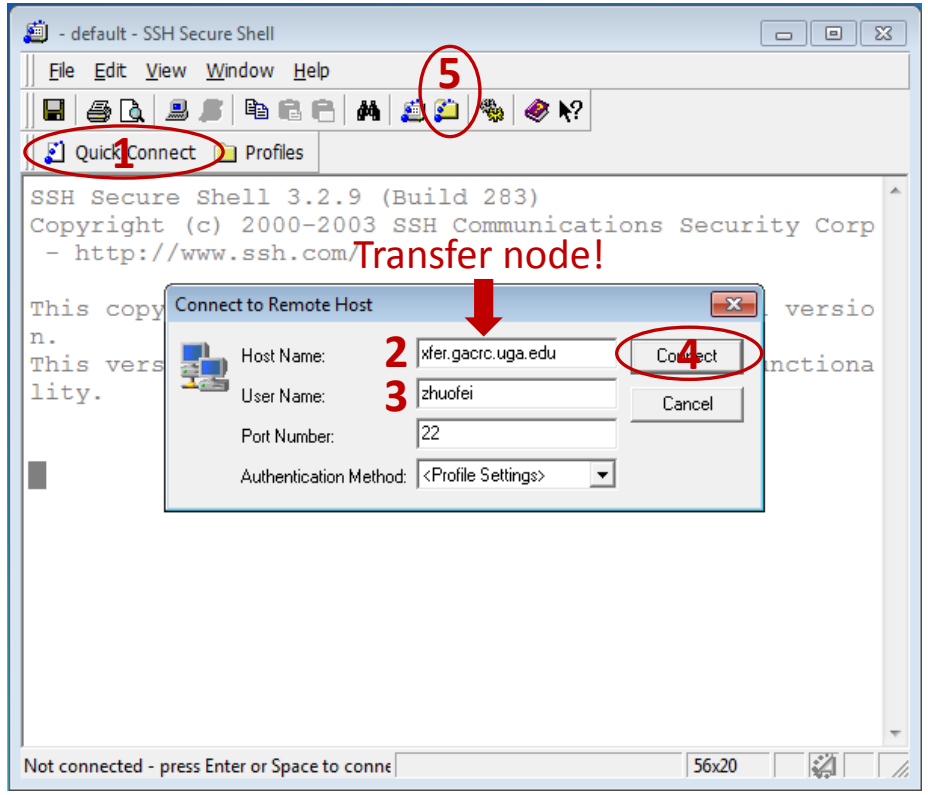
```
scp zhuofei@xfer.gacrc.uga.edu:/lustre1/zhuofei/workDir/file ./
```

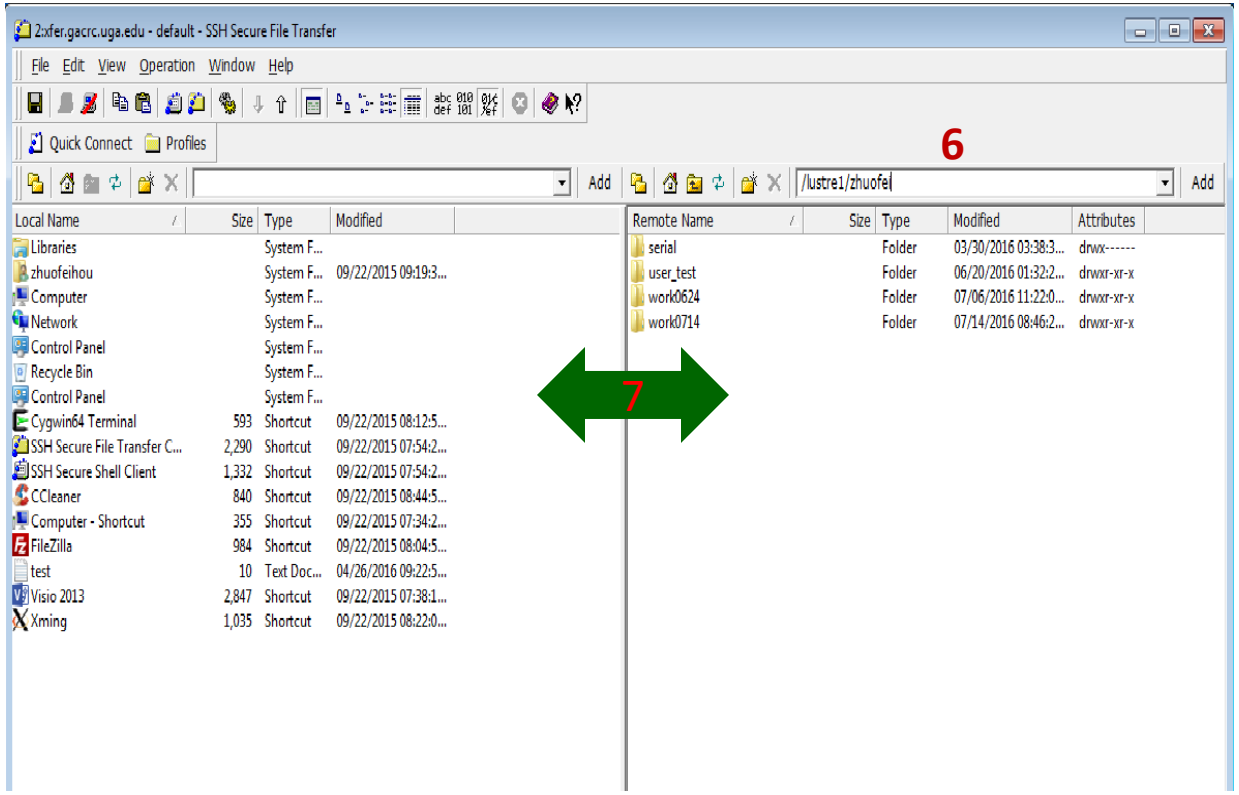
```
scp -r zhuofei@xfer.gacrc.uga.edu:/lustre1/zhuofei/workDir/folder/ ./
```

- On Window: SSH Secure Shell File Transfer, FileZilla, WinSCP (next page)

Transfer Files Using Transfer Node xfer.gacrc.uga.edu

User's local  SSH client  Transfer (xfer.gacrc.uga.edu)





Software Packages

- The cluster uses environment modules to define the various paths for software packages
- Current number of modules installed is ~300 and expanding daily!
- `module avail` to list all available modules (centralized installation):

```
[zhuofei@75-104 ~]$ module avail

------------------------------------------------------------------------ /usr/local/modulefiles ------------------------------------------------------------------------
  Core/StdEnv                        exabayes/1.4.1                java/jdk1.8.0_20                openmpi/1.6.5/gcc/4.4.7         rsem/latest
  Data/cache/moduleT.new             examl/3.0.11                  java/latest              (D)    openmpi/1.6.5/pgi/14.9          rsem/1.2.20              (D)
  Data/cache/moduleT          (D)    expat/latest                  lammps/5Sep14                   openmpi/1.8.3/gcc/4.4.7         samtools/latest
  Data/system.txt                    expat/2.0.1            (D)    lammps/16Aug13           (D)    openmpi/1.8.3/gcc/4.7.4         samtools/0.1.19
  R/3.1.2                            fastqc/latest                 moab/7.2.10                     openmpi/1.8.3/gcc/4.8.0  (D)    samtools/1.1
  bedops/latest                      fastqc/0.11.3          (D)    moab/8.1.1               (D)    openmpi/1.8.3/intel/14.0        samtools/1.2             (D)
  bedops/2.4.14               (D)    gcc/4.7.4                     moabs/1.3.2                     openmpi/1.8.3/intel/15.0.2 (D)  scripture/latest
  boost/1.47.0/gcc447                gcc/4.8.0              (D)    mvapich2/2.0.0/gcc/4.4.7        openmpi/1.8.3/pgi/14.9          scripture/03202015       (D)
  boost/1.57.0/gcc447                gmap-gsnap/latest             mvapich2/2.0.0/pgi/14.9         orca/3.0.3                      sparsehash/latest
  boost/1.57.0_thread/gcc447         gmap-gsnap/2014-12-24  (D)    ncbiblast+/2.2.29               perl/latest                     sparsehash/2.0.2         (D)
  bowtie/latest                      gnuplot/5.0.0                 netcdf/3.6.3/gcc/4.4.7          perl/5.20.1                     tophat/latest
  bowtie/1.1.1                (D)    gsl/1.16/gcc/4.4.7            netcdf/3.6.3/intel/14.0         perl/5.20.2              (D)    tophat/2.0.13            (D)
  bowtie2/latest                     hdf5/1.8.14/gcc/4.4.7         netcdf/3.6.3/intel/15.0.2 (D)   pgi/14.9                        trinity/latest
  bowtie2/2.2.4               (D)    hdf5/1.8.14/intel/15.0.2      netcdf/4.1.3/gcc/4.4.7          pgi/14.10                (D)    trinity/r20140717
  cuda/5.0.35/gcc/4.4.7              hdf5/1.8.14/pgi/14.9          netcdf/4.1.3/intel/15.0.2       python/2.7.8-ucs4               trinity/2.0.6            (D)
  cuda/6.5.14/gcc/4.4.7              imb/3.2                       netcdf/4.1.3/pgi/14.10          python/2.7.8                    zlib/gcc447/1.2.8
  cufflinks/latest                   intel/14.0                    netcdf/4.3.2/gcc/4.4.7          python/3.4.3             (D)
  cufflinks/2.2.1             (D)    intel/15.0.2           (D)    netcdf/4.3.2/pgi/14.9           raxml/8.1.20
```

Software Packages

- `module list` to list which modules currently loaded:

```
[zhuofei@75-104 ~]$ module list

Currently Loaded Modules:
  1) StdEnv   2) moab/7.2.10
```

- `module load` to load the needed modules:

```
[zhuofei@75-104 ~]$ module load ncbiblast+/2.2.29
[zhuofei@75-104 ~]$ module load python/2.7.8
[zhuofei@75-104 ~]$ module load R/3.1.2
[zhuofei@75-104 ~]$ module list

Currently Loaded Modules:
  1) StdEnv   2) moab/7.2.10   3) ncbiblast+/2.2.29   4) python/2.7.8   5) R/3.1.2
```

- `module unload` to remove the specific module:

```
[zhuofei@75-104 ~]$ module unload R/3.1.2
[zhuofei@75-104 ~]$ module list

Currently Loaded Modules:
  1) StdEnv   2) moab/7.2.10   3) ncbiblast+/2.2.29   4) python/2.7.8
```

Run Interactive Jobs

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

```
[zhuofei@75-104 ~]$ qlogin
qsub: waiting for job 1058157.pbs.scm to start
qsub: job 1058157.pbs.scm ready
[zhuofei@n14 ~]
[zhuofei@n14 ~]$ module load R/3.2.3
[zhuofei@n14 ~]$ R

R version 3.2.3 (2015-12-10) -- "Wooden Christmas-Tree"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

...
[Previously saved workspace restored]

>
```

← Now I am on n14, which is an interactive node

- When you are done, remember to `exit` the session

Run Batch Jobs

- Components you need to run a job:
  - Software already loaded. If not, used `module load`
  - Job submission script to run the software, and specify computing resources:
    - ✓ Number of nodes and cores
    - ✓ Amount of memory
    - ✓ Type of nodes
    - ✓ Maximum wallclock time, etc.

- Common commands you need:
  - qsub, qdel
  - qstat -f, showjobs, showq

Submit Batch Jobs

```
[zhuofei@n15 workDir]$ pwd          ← n15: interactive node
/lustre1/zhuofei/workDir            ← /lustre1/zhuofei/: global scratch
[zhuofei@n15 workDir]$
[zhuofei@n15 workDir]$ qsub sub.sh
1165617.pbs.scm
```

qsub is to submit a job

sub.sh is your **job submission script** specifying:

- ✓ Number of nodes and cores
- ✓ Amount of memory
- ✓ Type of nodes
- ✓ Maximum wallclock time, etc.

Example 1: Serial job script *sub.sh* running NCBI Blast +

#PBS -S /bin/bash	➔ Linux shell (bash)
#PBS -q batch	➔ Queue name (batch)
#PBS -N testBlast	➔ Name of the job (testBlast)
#PBS -l nodes=1:ppn=1:AMD	➔ Number of nodes (1), number of cores/node (1), node type (AMD)
#PBS -l mem=20gb	➔ Maximum amount of physical memory (20 GB) used by the job
#PBS -l walltime=48:00:00	➔ Maximum wall clock time (48 hours) for the job, default 6 minutes
cd $PBS_O_WORKDIR	➔ Use the directory from which the job is submitted as the working directory
module load ncbiblast+/2.2.29	➔ Load the module of ncbiblast+, version 2.2.29
time blastn [options] >outputfile 2>&1	➔ Run blastn with 'time' command to measure the amount of time it takes to run the application

Example 1: Serial job script *sub.sh* running NCBI Blast +

```
#PBS -S /bin/bash
#PBS -q batch
#PBS -N testBlast
#PBS -l nodes=1:ppn=1:AMD
#PBS -l mem=200gb
#PBS -l walltime=48:00:00

cd $PBS_O_WORKDIR


module load ncbiblast+/2.2.29

time blastn [options] >outputfile 2>&1
```

➔ Job will be dispatched to run on AMD 256GB node

Example 2: Threaded job script *sub.sh* running NCBI Blast+

```
#PBS -S /bin/bash
#PBS -q batch
#PBS -N testBlast
#PBS -l nodes=1:ppn=24:AMD
#PBS -l mem=200gb
#PBS -l walltime=480:00:00

#PBS -M jSmith@uga.edu
#PBS -m ae
#PBS -j oe

cd $PBS_O_WORKDIR

module load ncbiblast+/2.2.29

time blastn -num_threads 24 [options] >outputfile 2>&1
```

➔ Number of nodes (1), number of cores/node (24), node type (AMD)
Number of threads (24) = Number of cores requested (24)

➔ Email address to receive a notification for computing resources
➔ Send email notification when job aborts (a) or terminates (e)
➔ Standard error file (testBlast.e1234) will be merged into standard out file (testBlast.o1234)

➔ Run blastn with 24 threads (-num_threads 24)

Example 3: MPI job script *sub.sh* running RAxML

```
#PBS -S /bin/bash
#PBS -q batch
#PBS -N testRAxML
#PBS -l nodes=2:ppn=48:AMD
#PBS -l walltime=480:00:00
#PBS -l mem=20gb

cd $PBS_O_WORKDIR

module load raxml/8.1.20
module load intel/15.0.2
module load openmpi/1.8.3/intel/15.0.2

mpirun –np 50 raxmlHPC-MPI-AVX [options] >outputfile 2>&1
```

➔ Number of nodes (2), number of cores/node (48), node type (AMD)
Total cores requested = 2 × 48 = 96
We suggest, Number of MPI Processes (50) ≤ Number of cores requested (96)

➔ To run raxmlHPC-MPI-AVX, MPI version using OpenMPI 1.8.3/Intel 15.0.2

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

```
#PBS -S /bin/bash
#PBS -q batch
#PBS -N testRAxML
#PBS -l nodes=2:ppn=27:AMD
#PBS -l walltime=480:00:00
#PBS -l mem=20g

cd $PBS_O_WORKDIR

# Context Sharing
CONTEXTS=$(/usr/local/bin/set_contexts.sh $PBS_NUM_PPN)
if [[ "$?" -eq "0" ]] ; then
  export PSM_SHAREDCONTEXTS_MAX=$CONTEXTS
fi

module load raxml/8.1.20
module load intel/15.0.2
module load openmpi/1.8.3/intel/15.0.2

mpirun –np 50 raxmlHPC-MPI-AVX [options] >outputfile 2>&1
```

➔ ppn number (27) fewer than 48 MUST be a multiplier of 3!

New lines copied from GACRC Wiki

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

Check Job Status (`qstat`) and Cancel a Job (`qdel`)

```
[jSmith@75-104 MPIs]$ qstat
Job ID                  Name             User            Time Use S Queue
------------------------------------------------------------------------
481929.pbs              testJob1         jSmith          900:58:0 C batch
481931.pbs              testJob2         jSmith          04:00:03 R batch
481934.pbs              testJob3         jSmith                 0 Q batch
```

Job status:
R : job is running
C : job completed (or crashed) and is not longer running. Jobs stay in this state for 1h
Q : job is pending, waiting for resources to become available

```
[jsmith@75-104 MPIs]$ qdel 481934
[jSmith@75-104 MPIs]$ qstat
Job ID                  Name             User            Time Use S Queue
------------------------------------------------------------------------
481929.pbs              testJob1         jSmith          900:58:0 C batch
481931.pbs              testJob2         jSmith          04:00:03 R batch
481934.pbs              testJob3         jSmith                 0 C batch   ← Stay on list 1 hr
```

Check Computing Resources Used by a Job

Option 1: **`qstat -f JobID`** for ***running jobs*** or ***finished jobs in 1 hour***

Option 2: **`showjobs JobID`** for ***finished jobs over 1 hour, but ≤ 7 days***

Option 3: Email notification from ***finished jobs (completed, canceled, or crashed)***,

if using:

```
#PBS -M jSmith@uga.edu
#PBS -m ae
```

`qstat -f JobID` for running jobs or finished jobs in 1 hour

```
[zhuofei@75-104 MPIs]$ qstat -f 699847
Job Id: 699847.pbs.scm
    Job_Name = testJob
    Job_Owner = zhuofei@uga-2f0f976.scm
    resources_used.cput = 00:11:55
    resources_used.energy_used = 0
    resources_used.mem = 411572kb
    resources_used.vmem = 6548528kb
    resources_used.walltime = 07:01:36
    job_state = C
    queue = batch
     .
    Error_Path = uga-2f0f976.scm:/home/zhuofei/MPIs/testJob.e699847
    exec_host = n165/0-23
    Output_Path = uga-2f0f976.scm:/home/zhuofei/MPIs/testJob.o699847
     .
    Resource_List.mem = 5gb
    Resource_List.nodect = 1
    Resource_List.nodes = 1:ppn=24:AMD
    Resource_List.walltime = 10:00:00
     .
    Variable_List = PBS_O_QUEUE=batch,PBS_O_HOME=/home/zhuofei, …… ,
                    PBS_O_WORKDIR=/home/zhuofei/MPIs,
```

`showjobs JobID` for finished jobs over 1 hour, but ≤ 7 days

```
[zhuofei@75-104 MPIs]$ showjobs 699847
Job Id             : 699847.pbs.scm
Job Name           : testJob
Output File        : uga-2f0f976.scm:/home/zhuofei/MPIs/testJob.o699847
Error File         : uga-2f0f976.scm:/home/zhuofei/MPIs/testJob.e699847
Working Directory  : /home/zhuofei/MPIs
Home Directory     : /home/zhuofei
Submit Arguments   : sub.sh
User Name          : zhuofei
Group Name         : rccstaff
Queue Name         : batch
Wallclock Limit    : 10:00:00
Wallclock Duration: 07:01:36
CPUTime            : 00:11:55
Memory Used        : 401.9Mb
Memory Limit       : 5gb
vmem Used          : 6.2Gb
Submit Time        : Wed Nov  4 12:02:22 2015
Start Time         : Wed Nov  4 12:03:41 2015
End Time           : Wed Nov  4 12:04:45 2015
Exit Code          : 0
Master Host        : n165
```

Email notification from finished jobs

```
PBS Job Id: 700009.pbs.scm
Job Name:   testJob
Exec host:  n1/4-27
Execution terminated
Exit_status=0
resources_used.cput=00:05:12
resources_used.energy_used=0
resources_used.mem=410984kb
resources_used.vmem=6548516kb
resources_used.walltime=04:00:59
Error_Path: uga-
2f0f976.scm:/home/zhuofei/MPIs/testJob.o700009
Output_Path: uga-
2f0f976.scm:/home/zhuofei/MPIs/testJob.o700009
```

```
PBS Job Id: 700097.pbs.scm
Job Name:   testJob
Exec host:  n1/4-27
Execution terminated
Exit_status=271
resources_used.cput=00:11:22
resources_used.energy_used=0
resources_used.mem=412304kb
resources_used.vmem=6548524kb
resources_used.walltime=05:00:41
Error_Path: uga-
2f0f976.scm:/home/zhuofei/MPIs/testJob.o700097
Output_Path: uga-
2f0f976.scm:/home/zhuofei/MPIs/testJob.o700097
```

Check Queue Status (*showq*)

```
[zhuofei@75-104 MPIs]$ showq
active jobs------------------------
JOBID              USERNAME      STATE PROCS   REMAINING            STARTTIME

481914                brant    Running     1    20:46:21  Fri Jun 12 11:32:23
481915                brant    Running     1    20:48:56  Fri Jun 12 11:34:58
481567               becton    Running   288  2:04:15:48  Wed Jun 10 15:01:50
481857                 kkim    Running    48  9:18:21:41  Fri Jun 12 09:07:43
481859                 kkim    Running    48  9:18:42:21  Fri Jun 12 09:28:23
.
108 active jobs       5141 of 5740 processors in use by local jobs (89.56%)
                      121 of 122 nodes active      (99.18%)
eligible jobs----------------------
481821               joykai       Idle    48 50:00:00:00  Thu Jun 11 13:41:20
481813               joykai       Idle    48 50:00:00:00  Thu Jun 11 13:41:19
481811               joykai       Idle    48 50:00:00:00  Thu Jun 11 13:41:19
.
50 eligible jobs

blocked jobs-----------------------
JOBID              USERNAME      STATE PROCS     WCLIMIT            QUEUETIME

0 blocked jobs
Total jobs:  158
```

Thank You!