Introduction to GACRC Computing Facility - Sapelo2 Cluster

Georgia Advanced Computing Resource Center (GACRC)

EITS/University of Georgia

Zhuofei Hou zhuofei@uga.edu

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

Support: https://uga.teamdynamix.com/TDClient/Requests/ServiceCatalog?CategoryID=11593

Web Site: http://gacrc.uga.edu

Outline

What is Sapelo2 Cluster

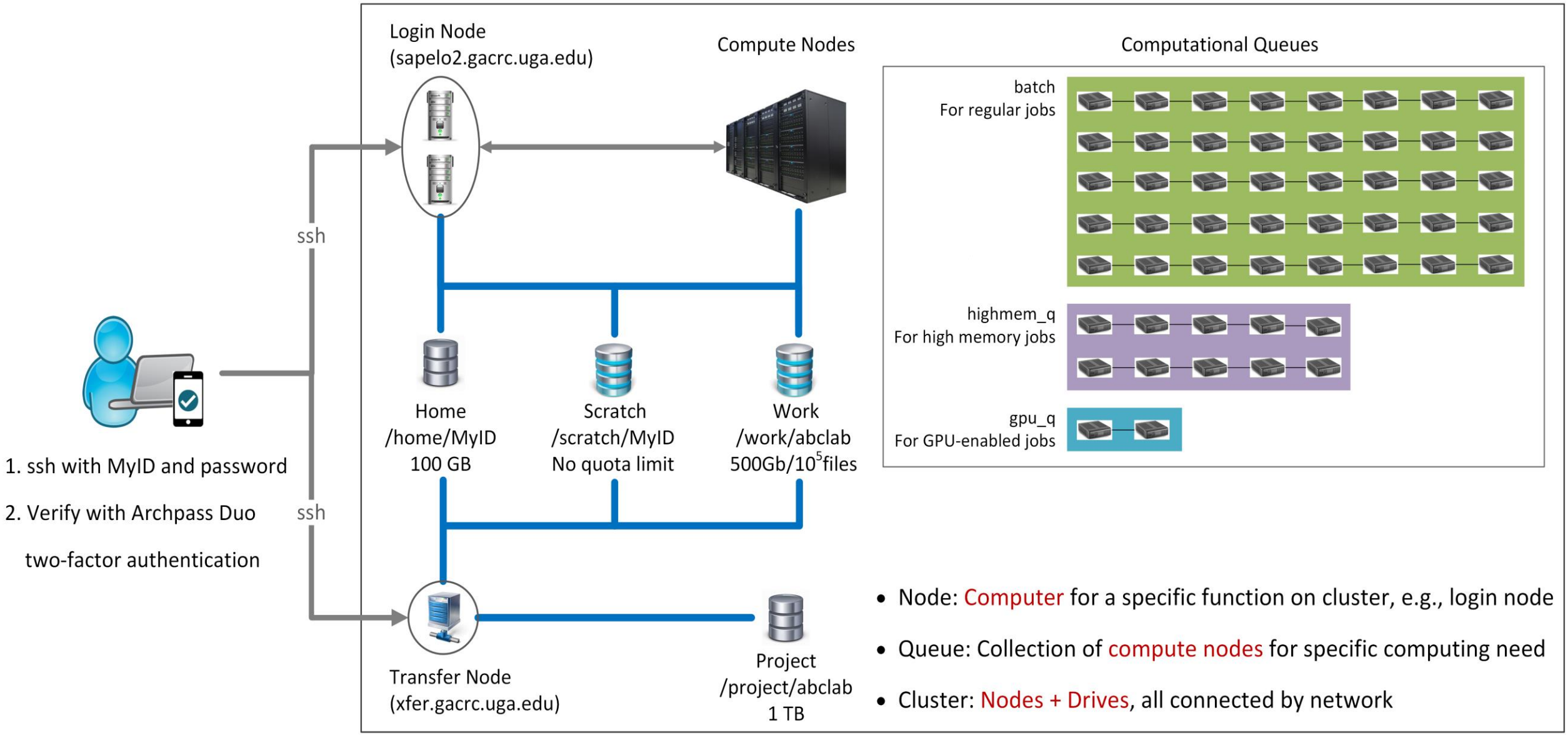
- Diagram and Overview
- Storage Environment
- Computing Resources
- Software Environment

Work on Sapelo2 Cluster

- Job Submission Workflow
- How to Know Job Details
- How to Know Node Details
- qlogin Command
- Code Compilation

GACRC Links

Appendix



Please Note: You need to connect to the UGA network using VPN when accessing from outside of the **UGA main campus**.
UGA VPN: https://eits.uga.edu/access_and_security/infosec/tools/vpn/

Overview https://wiki.gacrc.uga.edu/wiki/Systems#Sapelo2

- Two Nodes:
  1. Login node for batch job workflow: MyID@sapelo2.gacrc.uga.edu
  2. Transfer node for data transferring: MyID@xfer.gacrc.uga.edu
- Five Directories:
  1. Home: Login landing spot; 100GB quota; Backed-up
  2. Scratch: High-speed storage for temp files needed for current jobs; NO quota; NOT backed-up
  3. Work: High-speed storage for input files needed for repeated jobs; per group quota of 500GB and max 100,000 single files; NOT backed-up
  4. Project: Temporary data parking; per group quota of 1TB; Backed-up (ONLY accessible from Transfer node!)
  5. Local Scratch: Local storage on each individual compute node; 200GB quota; NOT backed-up
- Four Computational Queues: batch, highmem_q, gpu_q, groupBuyin_q

Storage Environment https://wiki.gacrc.uga.edu/wiki/Disk_Storage

Directory	Name	Quota	Accessible from	Intended Use	Backed-up	Important Notes
/home/MyID	Home	100GB	Login Transfer Compute	Static data, e.g. 1. Scripts, source codes 2. Local software	Yes	Not for storing data of your jobs!
/scratch/MyID	Scratch	No Limit		Temporary files needed for current running jobs	No	Clean up when your job finished! Subject to "30-day purge" policy
/work/abclab	Work	500GB 10^5 files		Input files needed for repeated jobs	No	Clean up any old data! Group sharing is possible
/project/abclab	Project	1TB (initial)	Transfer	Temporary data parking	Yes	Group sharing is possible
/lscratch	Local Scratch	200GB	Compute	Jobs with heavy disk I/O operations	No	Clean up when job exits from node! Data are persistent

More about scratch file system "30-day purge" policy

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

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

Measures circumventing this policy will be monitored and actively discouraged.

- You have a list of those purgeable files located at /usr/local/var/lustre_stats/$USER.over30d.files.lst
- You are suggested to copy files from /scratch to /project or outside of GACRC
- You should first move all unnecessary files and folders to /scratch/trash/$USER
- The fastest way to save your old files is to copy them to /project area, using the fpsync utility on xfer.gacrc.uga.edu
- If you want to first create a tar archive of your /scratch area, DO NOT compress the archive when creating the archive

Queue	Total Nodes	RAM(GB) /Node	Max Mem(GB) /Single-node job	Cores /Node	Processor Type	GPU Cards /Node	IB
batch	42	192	184	32	Intel Xeon Skylake	N/A	Yes
	32	64	58	28	Intel Xeon Broadwell		
	106	128	120	48	AMD Opteron		
	18 (+46)			32	AMD EPYC		
highmem_q	5 (+4)	1024	990	28 48 (64)	Intel Xeon Broadwell (4) AMD Opteron (1) AMD EPYC (+4)		
	15 (+10)	512	502	32, 48	Intel Xeon Nehalem (1) AMD Opteron (6) AMD EPYC (8+10)		
gpu_q	4	192	184	32	Intel Xeon Skylake	1 NVDIA P100	
	2	128	120	16	Intel Xeon	8 NVIDIA K40m	
	4	96	90	12		7 NVIDIA K20Xm	
groupBuyin_q	variable						

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

1. Software names are long and have a EasyBuild toolchain name associated to it
2. Complete module name: Name/Version-toolchain, e.g., Python/3.6.4-foss-2018a
3. Software names are case-sensitive!
   - `module avail` : List all available software modules installed on cluster
   - `module load moduleName` : Load a module into your working environment
   - `module list` : List modules currently loaded
   - `module unload moduleName` : Remove a module from working environment
   - `module spider pattern` : Search module names using 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 your scratch space: `cd /scratch/MyID`
3. Create a working subdirectory for a job : `mkdir ./workDir`
4. Change directory to workDir : `cd ./workDir`
5. Transfer data from local computer to workDir : use `scp` or SSH File Transfer to connect Transfer node

   Transfer data on cluster to workDir : log on to Transfer node and then use `cp` or `mv`
6. Make a job submission script in workDir : `nano ./sub.sh`
7. Submit a job from workDir : `qsub ./sub.sh` (refer to slide 20-22 for example job scripts)
8. Check job status : `qstat_me` or Cancel a job : `qdel JobID`

How to Know Job Details

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

Option 1: **`qstat -n1rt -u MyID`** for node info of running jobs (array jobs)

Option 2: **`qstat -f JobID`** for details of running jobs or finished jobs within 24 hours

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

if using:

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

How to Know Node Details

Option 1: mdiag -v –n | grep [pattern] | …

mdiag -v -n | grep batch | grep AMD
mdiag -v -n | grep batch | grep Intel
mdiag -v -n | grep highmem_q
mdiag -v -n | grep grpBuyin_q

Option 2: from login node, ssh to a compute node and run a command there

ssh n72 'lscpu‘
ssh n222 'free -g‘
ssh n237 "ps aux | grep '^MyID'"

qlogin Commands

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

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

Code Compilation – Compiler Suite and Compiler toolchain

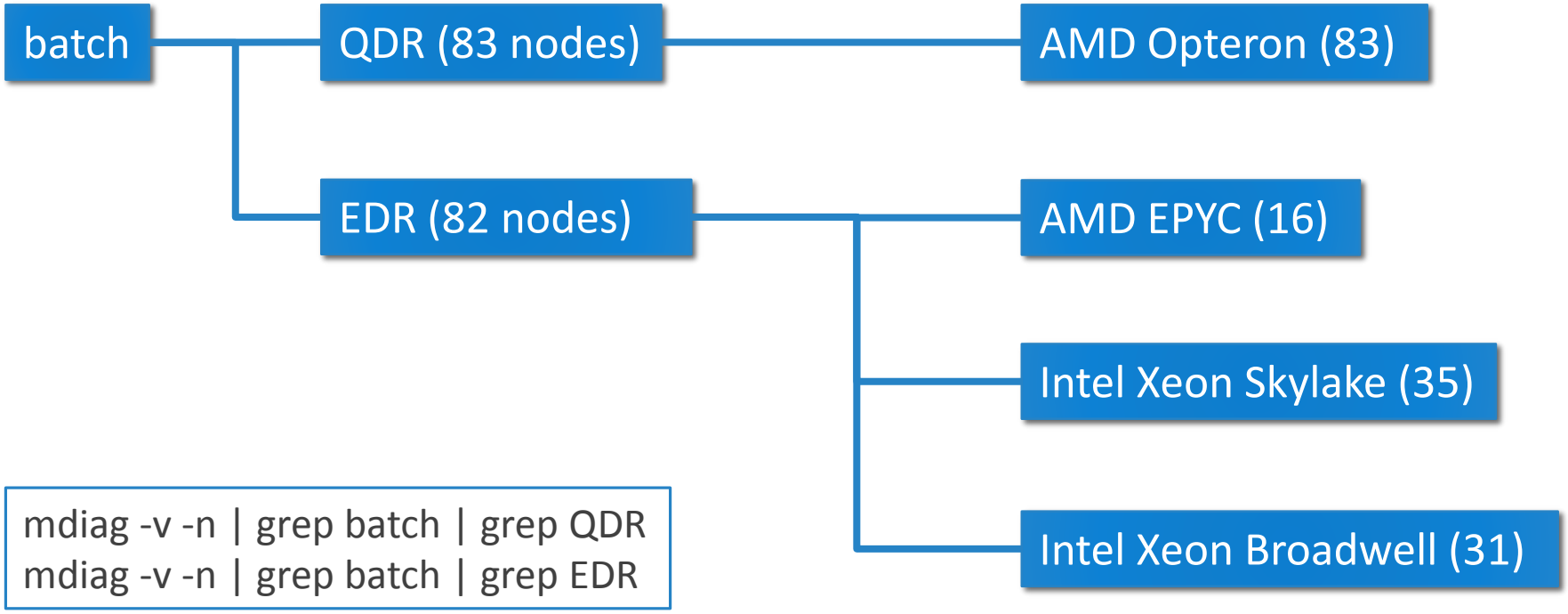
https://wiki.gacrc.uga.edu/wiki/Code_Compilation_on_Sapelo2

GCC/8.3.0-2.32	→ **GNU 8.3.0-2.32 compiler suite**
PGI/18.10-GCC-6.4.0-2.28	→ PGI 18.10 compiler suite
iccifort/2018.1.163-GCC-6.4.0-2.28	→ Intel 18.0.1.163 compiler suite
foss/2016b	→ GCC 5.4.0, OpenMPI 1.10.3, OpenBLAS 0.2.18, FFTW 3.3.4, ScaLAPACK 2.0.2
foss/2018a	→ GCC 6.4.0, OpenMPI 2.1.2, OpenBLAS 0.2.20, FFTW 3.3.7, ScaLAPACK 2.0.2
foss/2018b	→ GCC/7.3.0, OpenMPI 3.1.1, OpenBLAS 0.3.1, FFTW 3.3.8, ScaLAPACK 2.0.2
fosscuda/2018b	→ foss/2018b with CUDA 9.2.88
gmvolf/2016b	→ GCC 5.4.0, MVAPICH2 2.2, OpenBLAS 0.2.18, FFTW 3.3.4, ScaLAPACK 2.0.2
iomkl/2018a	→ Intel 2018.1.163 compiler suite, OpenMPI 2.1.2, MKL 2018.1.163
imvmkl/2018a	→ Intel 2018.1.163 compiler suite, MVAPICH2 2.2, MKL 2018.1.163

Code Compilation – Considerations for MPI

https://en.wikipedia.org/wiki/InfiniBand

Sapelo2 nodes are communicating via InfiniteBand (IB): QDR and EDR




```
mdiag -v -n | grep batch | grep QDR
mdiag -v -n | grep batch | grep EDR
```

Code Compilation – Considerations for MPI

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

When compile MPI codes using MVAPICH2, you will need to differentiate EDR from QDR;

But it is not necessary for OpenMPI

module spider MVAPICH2

QDR Versions:	EDR Versions:
MVAPICH2/2.2-GCC-5.4.0-2.26	MVAPICH2/2.3-GCC-5.4.0-2.26-EDR
MVAPICH2/2.2-GCC-6.4.0-2.28	MVAPICH2/2.3-GCC-6.4.0-2.28-EDR
MVAPICH2/2.2-iccifort-2013_sp1.0.080	MVAPICH2/2.3-iccifort-2013_sp1.0.080-EDR
MVAPICH2/2.2-iccifort-2015.2.164-GCC-4.8.5	MVAPICH2/2.3-iccifort-2015.2.164-GCC-4.8.5-EDR
MVAPICH2/2.2-iccifort-2018.1.163-GCC-6.4.0-2.28	MVAPICH2/2.3-iccifort-2018.1.163-GCC-6.4.0-2.28-EDR

Code Compilation – Considerations for MPI

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

Memory request for MPI job: using pmem, instead of mem, in your job script

For example:

#PBS -l pmem=2gb

pmem: maximum memory for each process

GACRC Buy-In Program

Cost of hardware + one-time service costs (IB, Ethernet port charges, etc.)

GACRC takes responsibility of systems administration for 5 years

Special queue accesses group buy-in nodes

Designate who can access your buy-in nodes. For example, we have 9 research groups that are sharing 22 compute nodes.

Does not stop you from submitting jobs to general-purpose queues

Matching funds are available – see next slide

GACRC Buy-In Program – Matching Funds

Each fiscal year, Tim Chester makes $100k available as matching funds

Conditions are:

First-come, first-served

Buy 1 or more compute node(s), get 1 matched, up to $10k

If node cost is >$10k, buy 2 or more nodes and get $10k discount

Matched nodes incur one-time service costs

Here's a very recent example ->

Georgia Advanced Computing Resource Center
UNIVERSITY OF GEORGIA

	#	Unit Cost	Total cost
1P Compute Node - 32 Cores 256GB RAM	3	$7,491.20	$22,473.60
Dell PowerEdge R6415 Server			
single AMD EPYC 7551P 2.00GHz, 32C/64T, 64M Cache			
eight 32GB RDIMM 2666MT/s Dual Rank			
240GB SSD SATA Read Intensive MLC 6Gbps 2.5in Hot-plug Drive			
960GB SSD SATA Read Intensive 6Gbps 512n 2.5in Hot-plug			
Mellanox ConnectX-4 dual-port Port 10/25 GbE SFP+ PCIE			
Mellanox ConnectX-5 Single-Port EDR Infiniband network adapter			
5-Year Warranty			
Cable, InfiniBand EDR, QSFP/QSFP, Passive Copper, 3m, Mellanox	3	$139.46	$418.38
Adaptive Moab Suite BE - 2 sockets, cotermed to 12/17/2020	3	$213.76	$641.28
VPIT Matching Funds			-$7,844.42
EITS One-Time Service Costs			**$2,334.09**
		TOTAL:	**$18,022.93**

	#	$/unit	costs
1GigE port	3	$72.92	$218.76
Mgmt. Port	3	$72.92	$218.76
IB Port	3	$532.19	$1,596.57
Rack Space	3	$100.00	$300.00
			$2,334.09

GACRC Links

Main Page: http://wiki.gacrc.uga.edu

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

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

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

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

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/