



IBS574 – P45 Grace Crum Rollins Building

**Running batch jobs using
sun grid engine (SGE)**

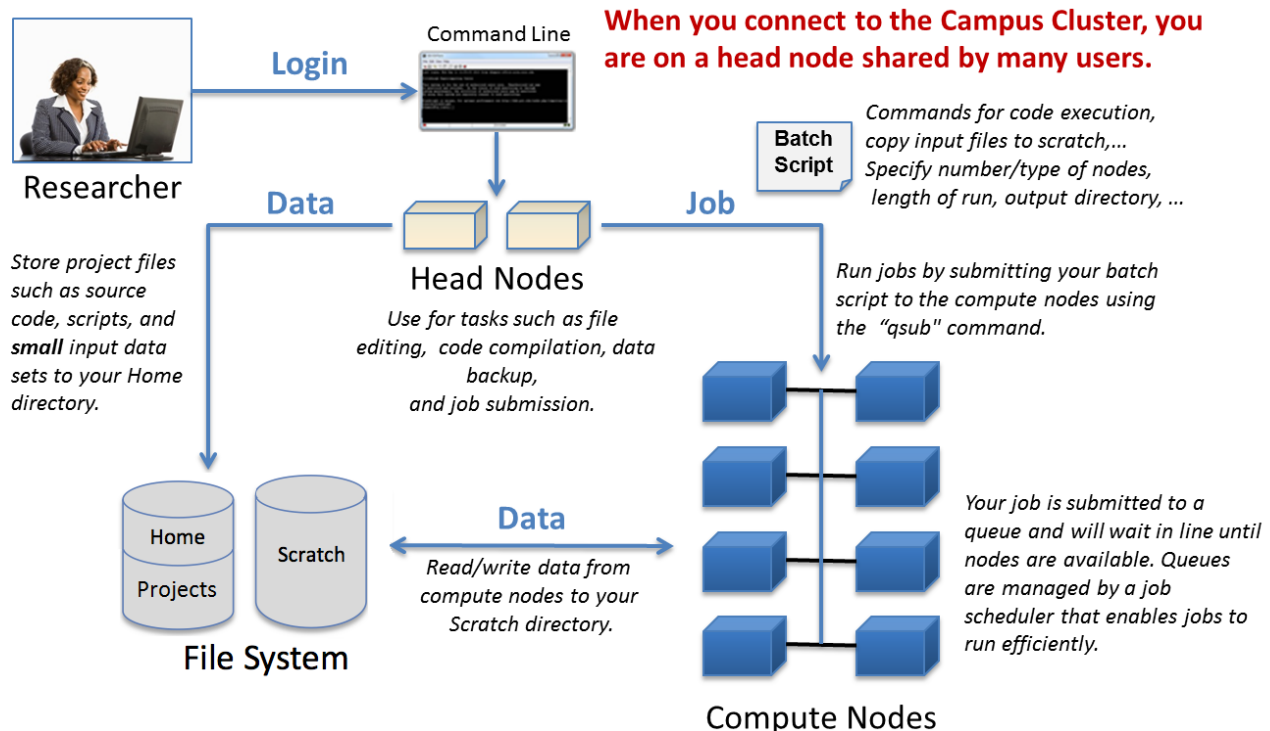
Ashok Dinasarapu Ph.D

**Zwick Group, Dept. of Human Genetics
Emory University, Atlanta**

Login at HGCC

ssh user_name@hgcc.genetics.emory.edu

Campus Cluster Usage Overview



<https://campuscluster.illinois.edu>

```
\_ / / \_ | | _ _ _ _ _ _ _ _ | | _ _ | | | | /
_ | / _ / _ |
\ \ / / / \ | / _ / _ \ | ' _ ` _ \ / _ \ | _ / _ \ | | _ | |
| _ | | | | |
\ \ V \ / _ / | ( _ | ( _ ) | | | | | | _ / | | | ( _ ) | | _ |
| _ | | | _ | | _
\ \ / \ \ | _ \ \ \ \ / | | | | | \ \ | \ \ \ / | |
| _ \ \ \ | \ \ \ \ \ |
```


HGCC is a private system subject to Emory IT policies
(<http://policies.emory.edu>). Access and use of HGCC indicates your acceptance of
these policies.


```
*****
*****
***** Next Scheduled Maintenance Window: Tue, Apr. 4, 2017, 7
AM - 12 PM *****
*****
*****
```

```
node01 - node02      62 cores/node      496 GB/node
node03 - node06      30 cores/node      240 GB/node
node07 - node10       6 cores/node       48 GB/node
```

```
b.q batch jobs      node01 - node10
i.q interactive jobs node07 - node10
```

```
=====
=====
```

```
hgcc:node00: [~] %
```

Organize Your Project

```
mkdir -p project/{data,logs,script,out}
```

```
~/project/data
```

```
~/project/logs
```

```
~/project/script
```

```
~/project/out
```

```
cd project/data
```

wget <https://github.com/CGATOxford/UMI-tools/releases/download/v0.2.3/example.fastq.gz>

Explore Cluster Environment

```
lscpu
```

```
lscpu | egrep 'Thread|Core|Socket|^CPU\('
```

2 sockets;

8 cores/socket;

1 threads/core;

i.e total 16 threads

You need to be careful of the term CPU as it means different things in different contexts.

Queue

- The cluster resources (the nodes) are grouped into Queues.
- Each queue is associated with a number of slots.
- One computational process runs in each slot.

```
qconf -sql  
qconf -sq b.q
```

Queue

```
qconf -sq b.q
```

```
slots 248, [node01.local=60], [node02.local=60],  
         [node03.local=28], [node04.local=28],  
         [node05.local=28], [node06.local=28],  
         [node07.local=4],  [node08.local=4],  
         [node09.local=4],  [node10.local=4]
```

```
h_vmem      8g
```

```
h_rt        864000
```

```
pe_list     smp
```

Module Loading

- List of installed modules

module avail

- Load a module

module load FastQC/0.11.4

- List of loaded modules

module list

- Unload a module

module unload FastQC/0.11.4

Bash Script (fastq.sh)

```
#!/bin/sh
### set name for the job
#$ -N my_fastqc
### requested 'b.q' queue
#$ -q b.q
### max time required
#$ -l h_rt=1:00:00
### ask for more CPU (slots) using -pe option
#$ -pe smp 2
### execute the job for the current working directory
#$ -cwd
### merge standard error with standard output
#$ -j y
### email notification of job, start and end etc.,
#$ -m abe
#$ -M xyz@emory.edu
### project directory
PROJ_DIR=$HOME/project
### fastqc is a java program for NextGen Seq Read quality check
module load FastQC/0.11.4
fastqc ${PROJ_DIR}/data/example.fastq.gz -o ${PROJ_DIR}/out
module unload FastQC/0.11.4
```

Job Submission

- Check currently running your jobs

```
qstat
```

- Check currently running all jobs

```
qstat -u '*'
```

- Submit your job from [~/project/logs](#)

```
qsub ~/project/script/fastq.sh
```

Job Status Check

- Check currently running your jobs

```
qstat
```

- Check currently running all jobs

```
qstat -u '*'
```

- Kill your running job

```
qdel here_your_job_id
```



**Practice
Makes
Perfect**