(Re)Introduction to High-Performance Computing (HPC)
HPC cluster structure
HPC cluster components

**Nodes:** Individual computers in the cluster

**Cores (threads):** individual processing units available within each CPU of each Node

- e.g. a “Node” with eight “quad”-core CPUs = 32 cores for that node.

**Shared disk:** storage that can be shared (and accessed) by all nodes
High-Performance Computing

For 3 samples

Serial

Multithreaded & Serial

Multithreaded and Parallel
Introduction to High Performance Computing and O2 for New Users

HMS Research Computing

(Slides courtesy of Kris Holton at HMS-RC)
O2 Tech Specs

- 11000 cores
- 32, 28, or 20 cores per node
- 256-160GB RAM (memory) per node (8-9GB/core)
- 756GB RAM (high memory nodes)
- 40 GPUs (V100, K80, M40)
- CentOS 7 Linux
- SLURM job scheduler
2-Factor Authentication

- For logins using WiFi networks other than HMS Private/Harvard Secure
- Easiest: download Duo app to phone
- Similar to the setup for Harvard Key logins
- Setup details at: https://wiki.rc.hms.harvard.edu:8443/display/O2/Two+Factor+Authentication+on+O2
The Job Scheduler, SLURM

Choosing the proper resources for your job with the appropriate Slurm arguments
Submitting Jobs

In an “interactive session”, programs can be run directly, however your computer will have to remain connected to the cluster for the duration of this run.

```
msk8@login-1:~$ srun --pty -p interactive -t 0-8:00
            -mem 6G /bin/bash
```

```
msk8@compute-a:~$ bowtie2 -c 4 hg19 file1_1.fq
```
Submitting Jobs

What if you wanted to run the program, close your computer and come back later to check on it?

A script with the required commands can be submitted to O2 (SLURM) using the `sbatch` command.

```bash
mfk8@login-1:~$ sbatch mybowtiejob.sh

OR

mfk8@compute-a:~$ sbatch mybowtiejob.sh
```
Creating a job submission script

```bash
#!/bin/sh

#SBATCH -p short
#SBATCH -t 0-03:00  #aim for 125% over
#SBATCH -c 4        #number of cores/threads, match with what is used in script
#SBATCH --mem=8G
#SBATCH -o %j.out
#SBATCH -e %j.err
#SBATCH -J bowtie2_run1  #job name
#SBATCH --mail-type=ALL  #notify about job completion or errors
#SBATCH --mail-user=mk8@med.harvard.edu

module load gcc/6.2.0
module load bowtie2/2.2.9

bowtie -c 4 hg19 file1_1.fq
```

Save script as myJobScript.run and run it as follows from login or compute node:

```
$ sbatch myJobScript.run
```
## Partitions -p

<table>
<thead>
<tr>
<th>Partition</th>
<th>Priority</th>
<th>Max Runtime</th>
<th>Max Cores</th>
<th>Limits</th>
</tr>
</thead>
<tbody>
<tr>
<td>short</td>
<td>12</td>
<td>12 hours</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>medium</td>
<td>6</td>
<td>5 days</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>long</td>
<td>4</td>
<td>30 days</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>interactive</td>
<td>14</td>
<td>12 hours</td>
<td>20</td>
<td>2 job limit</td>
</tr>
<tr>
<td>priority</td>
<td>14</td>
<td>30 days</td>
<td>20</td>
<td>2 job limit</td>
</tr>
<tr>
<td>mpi</td>
<td>12</td>
<td>5 days</td>
<td>640</td>
<td>20 core min</td>
</tr>
<tr>
<td>highmem</td>
<td>12</td>
<td>5 days</td>
<td>8</td>
<td>750G</td>
</tr>
<tr>
<td>gpu</td>
<td></td>
<td>120 GPU hours</td>
<td>20 cpu</td>
<td></td>
</tr>
<tr>
<td>transfer</td>
<td></td>
<td>5 days</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>
Job Priority

- Dynamically assigned
- Factors contributing: Age, Fairshare, Partition, QOS, Nice
- Fairshare: 0-1 scale
Managing jobs and getting information about submitted/running jobs
Job Monitoring

- $ 02squeue #HMS wrapper
- $ squeue -u eCommons

Detailed job info:
- $ scontrol show jobid <jobid>

Completed job statistics:
- $ 02sacct #HMS wrapper
Cancelling Jobs

- $ scancel <jobid>
- $ scancel --name JOBNAME
Filesystems and storage
Filesystems and storage
O2 Primary Storage

**O2 Cluster**
- 11000+ cores
- SLURM batch system

**/home**
- /home/user_id
- quota: 100GB per user
- Backup: extra copy & snapshots:
  - daily to 14 days, weekly up to 60 days

**/n/data1, /n/data2, /n/groups**
- /n/data1/institution/dept/lab/your_dir
- quota: expandable
- Backup: extra copy & snapshots:
  - daily to 14 days, weekly up to 60 days
Temporary “Scratch” storage

- `/n/scratch2`
- For data only needed temporarily during analyses.
- Each account can use up to 10 TB and 1 million files/directories

- **Lustre** --> a high-performance parallel file system running on DDN Storage.
- More than 1 PB of total shared disk space.
- No backups! Files are automatically deleted after unaccessed for 30 days, to save space.
- More info at: [http://hmsrc.me/O2docs](http://hmsrc.me/O2docs)
Checking storage usage

- For /n/data1, /n/data2, /n/groups
  - $ quota
  - Breaks down per user, directory

- For /n/scratch2
  - $ lfs quota -h /n/scratch2
  - 1 million files/folders, 10TB limit
For more direction

- [http://hmsrc.me/O2docs](http://hmsrc.me/O2docs)
- [http://rc.hms.harvard.edu](http://rc.hms.harvard.edu)
- RC Office Hours: Wed 1-3p Gordon Hall 500
- [rchelp@hms.harvard.edu](mailto:rchelp@hms.harvard.edu)