Submitting Jobs

Submit a SLURM job file:

```
sbatch job_file.submit
```

Submit a job with additional/overriding SLURM options:

```
sbatch <SBATCH config options> job_file.submit
sbatch --partition=labgroup job_file.submit
```

Submit an interactive job

```
srun --nodes=1 --ntasks=1 --mem=4G --pty /bin/bash
```

- Interactive jobs are great for debugging, development, and troubleshooting with an interactive terminal session on a worker node. If you lose connection or close your laptop, your interactive job may be killed.

- Batch jobs are the preferred method for running workflows on HCC clusters. They allow jobs to run in the background and independently from your terminal.

The config options in a SLURM submit file are prefixed with "#SBATCH", e.g.,:

```
#SBATCH --time=50:00:00
#SBATCH --ntasks=16
```

Using the "SBATCH config options" with sbatch on the command line is great for overriding the config inside the submit file. For example, changing the partition or GPU constraint without changing the submit file.

Job Configuration Options

Config Option	Explanation
Each option has the general syntax followed by an example. In a batch job, these will be after each #SBATCH in the submit file.	More information on how the options are used on HCC clusters is available on the next page.
--time=<HH:MM:SS> --time=4:30:00	Sets the maximum time limit for the job. Example is 4 hours and 30 minutes.
--qos=<qos_name> --qos=short	Defines the QoS or "quality of service" for the job. More information on other side.
--partition=<partition_name> --partition=guest	Defines the partition to use for the job. More details on the other side.
--ntasks=<n> --ntasks=4	Defines the requested total CPU cores.
--ntasks-per-node=<n> --ntasks-per-node=4	Defines the requested CPU cores per node requested.
--nodes=<n> --nodes=2	Defines the number of nodes to request for the job.
--mem=<nX> --mem=4GB	Amount of memory requested per node.
--mem-per-cpu=<nX> --mem-per-cpu=4GB	Amount of memory requested per CPU core.
--job-name=<name_of_job> --job-name=statistic_visual	Name of the job itself. Makes for easier identification in queue.
--gres=<gpu:n> --gres=gpu:1	Amount of GPU's to be requested.
--constraint=<job_constraint> --constraint=gpu_v100	Filter the job to run only on nodes with specific features.

SLURM Job Configuration Explanations

Time – Jobs can run for a maximum of 168 hours or 1 week on HCC clusters.

QoS – HCC Clusters have two primary QoS, normal (default) and short. The 'short' QoS is good for jobs with:

- 6 Hours of runtime or less
- No more than 2 jobs or 16 CPU Cores used per user
- No more than 256 CPU Cores used in 'short' by everyone

Partition – HCC Clusters have four primary partitions:

- batch: Default pool of hardware in a shared use.
- gpu: Shared pool of GPU accelerated nodes.
- guest: Leased nodes not actively running jobs.
- guest_gpu: GPU enabled leased nodes.
- lab_group_name: Hardware owned or leased by your lab.

More details: go.unl.edu/hcc_partitions

Nodes – This is best to set at 1 node unless your software supports working across multiple physical nodes.

Ntasks / CPU Cores - This is best to set at 1 ntask / core unless your software supports working across multiple CPU cores.

Memory - The amount of memory to request depends on your software and data size and can vary a bit. A good starting point is the same or a little more than your laptop or lab computer that used to run the software. Requesting too much can negatively impact your wait time in queue.

GRES - GPUs will only speed up your software execution if the code supports working with a GPU.

Constraint - These are best used when you need a specific type of GPU or GPU memory.

Job Information

View the current queue:

```
squeue
```

View the current queue for your user:

```
squeue -u <username>
```

View status of SLURM job with specific id:

```
squeue -j <job_id>
```

View when a job is estimated to start running:

```
squeue -j <job_id> --start
squeue -u <username> --start
```

View job efficiency and resource utilization after completion:

```
seff <job_id>
```