

Basic Commands

The ARC managed HPC clusters use a batch manager called Slurm. Extensive documentation for Slurm can be found at <http://slurm.schedmd.com/slurm.html>

Some of the most basic and useful Slurm commands are:

```
sinfo  
squeue  
sbatch  
scancel
```

To see the full documentation for any of these commands (e.g. sinfo), type:

```
man sinfo
```

You can run the command `sinfo` to see the available queues and compute resources.

All jobs must be run through the batch manager. Any jobs found running on the compute nodes outside of the queueing system will be killed.

Once you have a job submitted, you can check the status with the command:

```
squeue
```

To kill a submitted job, type:

```
scancel JOBID
```

Where the JOBID can be obtained from the squeue command.

Resource Requests

Resource requests using Slurm are the most important part of your job submission.

You will only get the resources you ask for, including number of cores, memory, and number of GPUs.

For example, if you only request one CPU core, but your job spawns four threads, all of these threads will be constrained to a single core.

Prior to submitting your job, it is extremely important you know exactly what resources you need, and make the analogous request.

Below are the most relevant sbatch or batch script flags to use for nodes/CPU/RAM/GPU/walltime requests.

Number of nodes:

-N NNODES

where NNODES is the number of nodes requested

Number of cores per node:

-n NCPUS

where NCPUS is the number of cores per node requested

Amount of memory in MB:

--mem MB_PER_NODE

where MB_PER_NODE is the amount of memory required per node. Default units are in MB. This can be modified using [KIMIGIT] to ask for KBIMBIGBITB of RAM. The following would request 64 GB of RAM per node for the job:

--mem 64G

Number of GPUs:

--gres=gpu:NGPU

where NGPU is the number of GPUs requested

Type of GPU:

-C GPU_TYPE

where GPU_TYPE is the NVIDIA model number (e.g. K20, K40, K80)

Walltime:

-t HH:MM:SS

where HH:MM:SS (hours:minutes:seconds) is the amount of wall time your job will be allocated

Batch Job Submissions

Multiple jobs, or a single job script that performs many functions, through the use of a batch script.

Instead of passing each resource flag, the requests would go at the top of the script preceded by #SBATCH.

```
#!/bin/bash
#SBATCH -N 1
#SBATCH -n 4
#SBATCH -p short
#SBATCH -o gompbot.out
#SBATCH -t 1:00:00
```

```
srun -l gompbot.x
srun -l /bin/pwd
```

The script above contained in a file job.sh can be submitted to the queue using the sbatch command:

```
sbatch job.sh
```

Important: Options supplied on the command line to the sbatch command will override any options specified within the script.

Matlab Example

```
#!/bin/bash
#SBATCH --output=simple.out
#SBATCH --error=simple.err
#SBATCH -N 1
#SBATCH -n 4
#SBATCH -p long
#SBATCH -t 168:00:00
#SBATCH -C E5-2695
```

```
module load matlab/R2016a
```

```
matlab -nojvm -nodisplay -nosplash < simple.m
```

Could replace matlab line above with any mpiexec or mpirun command. For example:

```
mpirun -n 4 cp2k.popt -i input.inp -o output.log
```

More Advanced Slurm Commands

Show Running Jobs

squeue

squeue -u USERNAME

Show Detailed Running Job Output (using output formatting flag and values)

squeue -o "%8i %12P %15j %10u %4t %10M %6D %18R %6C %8b %f"

Show State of Nodes

sinfo

Show Detailed State of Nodes (using output formatting flag and values)

sinfo -o "%20N %6c %8m %16f %8G %10T %10O"

Cancel Jobs

scancel

cancel all of your jobs:

scancel -u USERNAME

cancel all jobs of your jobs on a partition:

scancel -p PARTITION

Show State and Resource Request of Job

scontrol show job JOBID

Show Priority of Queued Jobs

sprio

Show Detailed Priority of Queued Jobs (using output formatting flag and values)

sprio -lo "%10Y %10u %10i %10A %10F %10J %10P %T"

Show Fairshare Usage

sshare