

Slurm Fun

Asya Shklyar
June 2021

Editing of the slides is still in progress :)



USC

Advanced Research Computing
Enabling scientific breakthroughs at scale

Aliases!

queue is super-long with all the formatting options

```
queue --format="%.8i %.9P %18j %10u %.8T %.12M %9N"
```

```
alias queue='queue --format="%.8i %.9P %18j %10u %.8T %.12M %9N"'
```

or

```
queue --user=<myusername>  
--Format=jobid,username,account,statecompact,starttime,timelimit,numcpus
```

Add to .bashrc (your profile)

More Aliases

```
sinfo -N --Format=nodehost,cpusstate,cpusload,memory,statecompact
```

```
sacct--jobs=7058,7057  
--format=User,JobID,account,AllocNodes,NodeList,Timelimit,elapsed,ReqMem,MaxRss,ExitCode
```

```
sacct --starttime=<mm.dd.yy>
```

```
sacct --user youruser  
--format=User,JobID,JobName%20,AllocNodes,NodeList
```

Split your stdout and stderr!

With SLURM, the default stdout and stderr are directed to the same file. The file has a .out extension and the filename is dependent on the job ID and array index, if applicable.

The default can be changed in the preamble of the SLURM submit script (the 'top' part where '#SBATCH' is used).

Below is a code fragment showing a redirect of stderr (-e) separately from a redirect of stdout (-o).

The %j will be replaced by the SLURM_JOB_ID.

```
#SBATCH -e slurm-%j.err  
#SBATCH -o slurm-%j.out
```

Please benchmark and profile your code or ask us for help

Makes it much easier to troubleshoot

You could simply start with using “time” command or

https://hpc.nih.gov/training/handouts/paralleljobs_seminar.pdfv

- Don't use multinode for multi-threaded jobs
- Use homogeneous resources sbatch --partition=multinode
--constraint=x2650 --ntasks=64 --ntasks-per-core=1 --time=168:00:00
--exclusive jobscript

Please try not to use GPU nodes if you are only using CPU

They are a precious and rare resource

We would rather hear from you that you need more nodes

We are constantly trying to add more nodes

Changes are coming - we will likely create a separate partition for GPU nodes to prevent users from using them inadvertently

Checkpointing

<https://slurm.schedmd.com/SLUG16/ciemat-cr.pdf>

Is the code integrated with BLCR?

- Yes-> use BLCR
- No
 - Is it parallel?
- Yes -> Use DMTCP
- No -> Use CRIU

Monitoring efficiency - basics - we need your help with this

top

htop

atop

iotop

nettop

10-30 secs and redirect to a file

Might have to use more sophisticated tools like gdb, valgrind,

<https://totalview.io/products/totalview>

<https://developer.nvidia.com/allinea-ddt>