

Introduction to SLURM & SLURM batch scripts

Anita Orendt

Assistant Director

Research Consulting & Faculty Engagement

anita.orendt@utah.edu

16 Feb 2017

Overview of Talk

- Basic SLURM commands
- SLURM batch directives
- Accounts and Partitions
- SLURM Environment Variables
- SLURM Batch scripts
- Running an Interactive Batch job
- Where to get more Information

Basic SLURM commands

- sinfo - shows partition/node state
- sbatch <scriptname> - launches a batch script
- squeue - shows all jobs in the queue
 - squeue -u <username> - shows only your jobs
- scancel <jobid> - cancels a job

Tcsh to add to .aliases file:

#SLURM Aliases that provide information in a useful manner for our clusters

```
alias si 'sinfo -o \"%20P %5D %14F %8z %10m %11l %16f %N\"'
```

```
alias si2 'sinfo -o \"%20P %5D %6t %8z %10m %10d %11l %N\"'
```

```
alias sq 'squeue -o \"%8i %12j %4t %10u %20q %20a %10g %20P %10Q %5D %11l %11L %R\"'
```

CENTER FOR HIGH PERFORMANCE COMPUTING

Some Useful Aliases

- Bash to add to .aliases file:

```
alias si="sinfo -o \"\">%20P %5D %14F %8z %10m %10d %11l %16f %N\\""
```

```
alias si2="sinfo -o \"\">%20P %5D %6t %8z %10m %10d %11l %16f %N\\""
```

```
alias sq="squeue -o \"\">%8i %12j %4t %10u %20q %20a %10g %20P %10Q %5D %11l %11L %R\\""
```

- Tcsh to add to .aliases file:

```
alias si 'sinfo -o \"%20P %5D %14F %8z %10m %11l %16f %N\"'
```

```
alias si2 'sinfo -o \"%20P %5D %6t %8z %10m %10d %11l %N\"'
```

```
alias sq 'squeue -o \"%8i %12j %4t %10u %20q %20a %10g %20P %10Q %5D %11l %11L %R\"'
```

SLURM Batch Directives

- #SBATCH --time 1:00:00 ← wall time of a job (or -t)
- #SBATCH --partition=name ← partition to use (or -p)
- #SBATCH --account=name ← account to use (or -A)
- #SBATCH --nodes=2 ← number of nodes (or -N)
- #SBATCH --ntasks 12 ← total number of tasks (or -n)
- #SBATCH --mail-type=FAIL,BEGIN,END ← events on which to send email
- #SBATCH --mail-user=name@example.com ← email address to use
- #SBATCH -o slurm-%j.out-%N ← name for stdout; %j is job#, %N node
- #SBATCH -e slurm-%j.err-%N ← name for stderr; %j is job#, %N node
- #SBATCH --constraint "C20" ← can use features given for nodes (or -C)

Accounts and Partitions

- You need to specify an account and partition to run jobs
- You can see a list of partitions using the `sinfo` command
- For general allocation usage the partition is the cluster name
- If no allocation (or out of allocation) use *clustername*-freecycle for partition
- Your account is typically your PI's name (e.g., if my PI is Baggins, I use the "baggins" account) – there are a few exceptions!
- Private node accounts and partition have the same name – PI last name with cluster abbreviation, e.g., baggins-kp, baggins-em, etc
- Private nodes can be used as a guest using the "owner-guest" account and the cluster-guest partition

SLURM Environment Variables

- Depend on SLURM Batch Directives used
- Can get them for a given set of directives by using the `env` command inside a script (or in a `srun` session).
- Some useful env variables:
 - `$SLURM_JOBID`
 - `$SLURM_SUBMIT_DIR`
 - `$SLURM_NNODES`
 - `$SLURM_NTASKS`

Basic SLURM script flow

1. Set up the #SBATCH directives for the scheduler to request resources for job
2. Set up the working environment, by loading appropriate modules
3. If necessary, add any additional libraries or programs to \$PATH and \$LD_LIBRARY_PATH, or set other environment needs
4. Set up temporary/scratch directories if needed
5. Switch to the working directory
6. Run the program
7. Clean up any temporary files or directories

Basic SLURM script - bash

```
#!/bin/bash
#SBATCH --time=02:00:00
#SBATCH --nodes=1
#SBATCH -o slurmjob-%j
#SBATCH --account=owner-guest
#SBATCH --partition=kingspeak-guest
#Set up whatever package we need to run with
module load somemodule
#set up the temporary directory
SCRDIR=/scratch/general/lustre/$USER/$SLURM_JOBID
mkdir -p $SCRDIR
#Set up the path to the working directory
pp file.input $SCRDIR/.
cd $SCRDIR
#Run the program with our input
myprogram < file.input > file.output
cp file.output $HOME/.
cd $HOME
rm -rf $SCRDIR
```

Basic SLURM script - tcsh

```
#!/bin/tcsh
#SBATCH --time=02:00:00
#SBATCH --nodes=1
#SBATCH -o slurmjob-%j
#SBATCH --account=owner-guest
#SBATCH --partition=kingspeak-guest
#Set up whatever package we need to run with
module load somemodule
#set up the scratch directory
set SCRDIR /scratch/local/$USER/$SLURM_JOBID
mkdir -P $SCRDIR
#move input files into scratch directory
cp file.input $SCRDIR/.
cd $SCRDIR
#Run the program with our input
myprogram < file.input > file.output
cp file.output $HOME/.
cd $HOME
rm -rf $SCRDIR
```

Parallel Execution

- If needed, create the node list:
 - `srun hostname | sort -u > nodefile.$SLURM_JOBID`
 - `srun hostname | sort > nodefile.$SLURM_JOBID`
- MPI installations at CHPC are SLURM aware, so `mpirun` will work without a machinefile (unless you are manipulating the machinefile in your scripts)
- Alternatively, you can use the `srun` command instead, but you need to compile with a more recently compiled MPI
- Mileage may vary, and for different MPI distributions, `srun` or `mpirun` may be preferred (check our slurm page on the CHPC website for more info or email us)

Running interactive batch jobs

- An interactive command is launched through the `srun` command

```
srun --time=1:00:00 --nodes=1 --account=chpc  
      --partition=ember --pty /bin/tcsh -l
```

- Launching an interactive job automatically forwards environment information, including X11 forwarding (
- "`--pty`" must be set to shell preferred for the session (either `/bin/tcsh` or `/bin/bash`)
- `-l` (lower case "L") at the end required

Slurm for use of GPU Nodes

- Ember – 11 GPU nodes
 - Each node with M2090 cards
- Kingspeak – 4 GPU nodes
 - 2 nodes with 4 Tesla K80 cards (8 GPUs)
 - 2 nodes with 8 GeForce TitanX cards
- Use partition and account set to cluster-gpu
- Must get added to account – request via issues@chpc.utah.edu
- Use only if you are making use of the GPU for the calculation
- Most codes do not yet make efficient use of multiple GPUs so we have enabled node sharing
- See <https://www.chpc.utah.edu/documentation/guides/gpus-accelerators.php>

Node Sharing

(currently only on GPU nodes)

- Need to specify number of CPU cores, amount of memory, and number of GPU

Option	Explanation
#SBATCH --gres=gpu:k80:1	request one K80 GPU (others types names are titanx, m2090)
#SBATCH --mem=4G	request 4 GB of RAM
#SBATCH --mem=0	request all memory of the node; use this if you do not want to share the node as this will give you all the memory
#SBATCH --tasks=1	requests 1 core

Slurm Documentation at CHPC

- <https://www.chpc.utah.edu/documentation/software/slurm.php>

Other good documentation sources

- <http://slurm.schedmd.com/documentation.html>
- <http://slurm.schedmd.com/pdfs/summary.pdf>
- <http://www.schedmd.com/slurmdocs/rosetta.pdf>

Getting Help

- CHPC website and wiki
 - www.chpc.utah.edu
 - Getting started guide, cluster usage guides, software manual pages, CHPC policies
- Jira Ticketing System
 - Email: issues@chpc.utah.edu
- Help Desk: 405 INSCC, 581-6440 (9-5 M-F)
- We use chpc-hpc-users@lists.utah.edu for sending messages to users; also have Twitter accounts for announcements -- @CHPCOutages & @CHPCUpdates