Quick Guide for the Slurm Cluster Manager

Introduction:

One of the main purposes of the Aries Cluster is to accommodate especially long-running programs. Users who run long jobs (which take hours or days to run) will need to run these jobs through the Slurm Scheduler. Slurm provides a method for handling these jobs on a first-come first-served basis. In this manner, all jobs will run more efficiently and finish faster since each is allowed to have all system resources for the duration of its run. **All Slurm Jobs must be launched from the ariessrv.**

The Three Most Common Commands:

The basic commands provided by Slurm for starting and stopping jobs and for manipulating jobs in queues are shown below. For complete manual of Slurm commands, type "man <cmd>" (e.g. "man sbatch").

sbatch - the basic command for running jobs (adding a run script to the job queue)

squeue - show currently running jobs/queued jobs

sinfo - gives the current status of all nodes

How to Run a Batch Job

The Slurm Scheduler will not accept an R program or a C program directly. It is designed instead to accept a shell script — a .sh file — which itself runs the commands necessary to launch your program. Once your script is ready, you can submit it to Slurm with the sbatch command:

```
sbatch your_script.sh
```

When this command is issued, you will be given a job number, we will call it **XX** in this example, which is used for tracking and manipulating your job. Standard out and standard error from the job will be copied to files in your current directory named *slurm-XX.out* for your reference.

To Run a Job on Several Nodes

If your job requires more than one node in order to run, the number of nodes can be specified when issuing the "sbatch --nodes=" command. In the following example, the test1.sh script requires 4 nodes in order to run. Issue this command to request 4 nodes for this job.

```
sbatch --nodes=4 ~/test1.sh
```

Once this command is issued, the system will verify that there are 4 machines ready for use. If there are, it will allocate 4 machines for this job and this job will run. If 4 machines are not yet available, the job will be put in the queue.

A user can issue the “squeue” command to track the status of the job. A job show a status of “PD” (Pending) or “R” (Running). Once completed, the job will disappear from the queue.

Note: The text of test1.sh is shown for your reference in the Appendix.

Appendix

I - Further Details Regarding the Basic Commands:

About sbatch

The sbatch command submits a sequence of commands to the batch server along with the parameters specifying job resource requirements. The parameters may be provided on the command line, from within the job script, or a combination of both. To facilitate optimal scheduling, you should specify as many resources as possible.

The syntax for the sbatch command is:

```
sbatch [ option(s) ] [ script-file ]
```

option	Default	Action
`--time=HH:MM:SS`		The length of time your job will need to run. Your job will end after the allocated walltime has expired whether it is finished or not, so choose this value carefully. Appropriate walltime should be chosen in order to prevent programs that either run out of control, or who never exit, from consuming all system resources. **If this option is not specified, the system will use default walltime settings.**
`--ntasks=n`	1	Specifies the number of threads/processes for the job. **Use n <= 4 * number of nodes to match the hardware on the cluster – otherwise, your job WILL NOT RUN**.
`--nodes=n`	1	Specifies the number of nodes required.

Examples:
`sbatch --nodes=4 --ntasks=4 --time=1:00:00 your_script.sh`
(requests four nodes and 4 processes, and a maximum runtime of 1 hour)

About squeue

The squeue command monitors the status of all jobs currently submitted to Slurm on the aries cluster.

Example:

squeue show all jobs

About scancel

A queued job may be removed from a queue or a running job may be killed using the scancel command.

Example:

```
scancel 1234
```
(where '1234' is the job ID. The job ID can be obtained with the squeue command.)

II - The test1.sh script

This script is designed to run with Slurm and requires multiple nodes to operate. The script, when run through Slurm, will examine each of the nodes present and return the name of each available node to a file named cluster_nodes.

```
#!/bin/bash

HOSTS=.hosts-job$SLURM_JOB_ID
HOSTFILE=cluster_nodes

srun hostname -f > $HOSTS
sort $HOSTS | uniq -c | awk '{print $2 ":" $1}' >> $HOSTFILE

rm $HOSTS
```