HPC User Group of Orlando

Introduction to SLURM

R. Paul Wiegand
Institute for Simulation & Training
University of Central Florida
wiegand@ist.ucf.edu

October 2015

1 Brief Overview of SLURM

2 Status & Information Commands

3 Managing SLURM Jobs

4 Wrap-up Q&A

What is SLURM?

- Open-source workload manager designed for Linux clusters
- Both resource manager and scheduler, and so replaces Torque & Moab
- Designed to be portable, scalable, fault-tolerant, and *simple*
- System becoming widely adopted by HPCs around the country

Why is the ARCC Moving to SLURM?

- Moab costs a lot of money
- SLURM is easier to configure and maintain than Torque/Moab
- We've had a lot of problems with Torque/Moab, historically
- SSERCA is standardizing on SLURM
- Developer and administrator communities are active on SLURM support lists

What is Going to Change for STOKES Users?

- All basic scheduler commands will be different
- Submit script directives & syntax must be changed

  (Yes, you will have to rewrite your submit scripts)
- We will take this opportunity to revise queue names & constraints

Useful Commands For Gathering Information

SLURM	Torque/Moab
`sinfo`	`qstat -q`
`squeue`	`qstat -u <username>`
`sinfo`	`qstat -q`
`scontrol show job`	`checkjob`

Listing Information about Partitions/Queues (`sinfo`)

- What we call a "queue", SLURM calls a *partition*
- We can get a list of the resources available under different partitions using the `sinfo` command
- Akin to Torque's `qstat -q` command

```
$ sinfo

PARTITION          AVAIL  TIMELIMIT  NODES  STATE NODELIST
OneNode.FourDays      up 4-00:00:00      6   idle evc[1,5-9]
TwoNodes.TwoDays*     up 2-00:00:00      6   idle evc[1,5-9]
FourNodes.OneDay      up 1-00:00:00      6   idle evc[1,5-9]
```

Listing Information about Running Jobs (squeue)

- We can get information about running jobs using `squeue`
- Akin to Torque's `qstat -u <username>`

```
$ squeue
     JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
       537 OneNode.F SocNetPD pwiegand  R       0:15      1 evc7
```

Detailing Information about a Job (`scontrol`)

- We can show detailed information about a particular job using `scontrol show job`
- Akin to Torque's `checkjob <job-id>`

```
$ scontrol show job 537

JobId=537 JobName=SocNetPDraw
   UserId=pwiegand(5031) GroupId=pwiegand(5118)
   Priority=1087 Nice=0 Account=arcc QOS=normal
   JobState=RUNNING Reason=None Dependency=(null)
   Requeue=1 Restarts=0 BatchFlag=1 Reboot=0 ExitCode=0:0
   RunTime=00:04:11 TimeLimit=01:00:00 TimeMin=N/A
   SubmitTime=2015-10-26T22:18:11 EligibleTime=2015-10-26T22:18:11
   StartTime=2015-10-26T22:18:12 EndTime=2015-10-26T23:18:12
   ...
```

Detailing Information about a Usage (`sshare`)

- Show current usage using `sshare`

```
$ sshare -l

 Account     User Raw Shares Norm Shares   Raw Usage  Norm Usage Effectv Usage  FairShare  GrpCPUMins
-------- -------- ---------- ----------- ----------- ----------- ------------- ---------- ----------- ----
root                          1.000000      342006                  1.000000   0.500000
 arcc                     1    0.333333      341785    0.999354      0.999354   0.125168      600000
  arcc   pwiegand          1    0.066667      340246    0.994854      0.995754   0.000032
```

Other Kinds of Reporting (`sreport`)

- Run various reports over different periods (including usage reports) using `sreport`

```
$ sreport -a cluster AccountUtilizationByUser  start=100115 Tree

--------------------------------------------------------------------------------
Cluster/Account/User Utilization 2015-10-01T00:00:00 - 2015-10-26T23:59:59 (2246400 secs)
Time reported in CPU Minutes
--------------------------------------------------------------------------------
  Cluster              Account     Login     Proper Name       Used     Energy
--------- -------------------- --------- --------------- ---------- ----------
   newton root                                              5698          0
   newton  root                    root            root          3          0
   newton  arcc                                             5695          0
   newton   arcc                  nlucas                      26          0
   newton   arcc                 pwiegand                   5669          0
```

Gathering Information about Account Activity (`sacct`)

- Can get a variety of information about your account, including recent activity via `sacct`

```
$ sacct
       JobID    JobName  Partition    Account  AllocCPUS      State ExitCode
------------ ---------- ---------- ---------- ---------- ---------- --------
538            hostname TwoNodes.+       arcc          2  COMPLETED      0:0
539            hostname TwoNodes.+       arcc          1  COMPLETED      0:0
540            hostname TwoNodes.+       arcc          4  COMPLETED      0:0
541            hostname TwoNodes.+       arcc          6  COMPLETED      0:0
542               hello TwoNodes.+       arcc          4  COMPLETED      0:0
543               hello FourNodes+       arcc          6  COMPLETED      0:0
544          PaulSlurm+ TwoNodes.+       arcc          4  COMPLETED      0:0
544.batch         batch                   arcc          4  COMPLETED      0:0
544.0             orted                   arcc          1  COMPLETED      0:0
545          PaulSlurm+ TwoNodes.+       arcc          4  COMPLETED      0:0
545.batch         batch                   arcc          4  COMPLETED      0:0
545.0             orted                   arcc          1  COMPLETED      0:0
546                bash TwoNodes.+       arcc          4  COMPLETED      0:0
546.0             orted                   arcc          2  COMPLETED      0:0
547          SocNetPDr+ OneNode.F+       arcc          1 CANCELLED+      0:0
547.batch         batch                   arcc          1  CANCELLED     0:15
```

Useful Commands For Managing Jobs

SLURM	Torque/Moab
`srun`	`qsub` (sort of)
`sbatch`	`qsub`
`salloc`	N/A
`scancel`	`qdel`
`scontrol`	N/A

Running Commands Directly (`srun`)

- To simply run a command or program via the workload manager, use `srun`

```
$ srun /usr/bin/hostname
evc7
```

- You can also use `srun` to run the same program in parallel

```
$ srun --nodes=2 --ntasks-per-node=3 hostname
evc5
evc6
evc5
evc6
evc5
evc6
```

Running MPI Programs Directly (`srun`)

- If the MPI library was built correctly, `srun` understands MPI directly:

```
$ srun --partition=FourNodes.OneDay  --nodes=3 --ntasks-per-node=2 ./hello
Process 5 on evc7 out of 6,  :: FOO_VAR=(null)
Process 4 on evc7 out of 6,  :: FOO_VAR=(null)
Process 0 on evc5 out of 6,  :: FOO_VAR=(null)
Process 3 on evc6 out of 6,  :: FOO_VAR=(null)
Process 1 on evc5 out of 6,  :: FOO_VAR=(null)
Process 2 on evc6 out of 6,  :: FOO_VAR=(null)
```

Submitting a Batch Script (`sbatch`)

- But you can (and should) write a submit script just as you do for Torque/Moab
- To submit a script to the scheduler, use `sbatch`

```
$ sbatch simple-mpi-ex.slurm
Submitted batch job 545

$ squeue
     JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
       545 TwoNodes. PaulSlur pwiegand PD       0:00      2 (None)
```

SLURM Submit Script

```
#!/bin/bash
#SBATCH --account=arcc
#SBATCH --partition FourNodes.OneDay
#SBATCH --nodes=2
#SBATCH --ntasks-per-node=2
#SBATCH --time=00:10:00
#SBATCH --error=rpwslurm-%J.err
#SBATCH --output=rpwslurm-%J.out
#SBATCH --job-name=PaulSlurmMPIJob
#SBATCH --mail-type=FAIL
#SBATCH --mail-type=BEGIN
#SBATCH --mail-type=END
#SBATCH --mail-user rpwiegand@gmail.com

# Load modules
echo "Slurm nodes: $SLURM_JOB_NODELIST"
module load openmpi-1.8.6-ic-2015.3.187-slurm-14.11.7
mpirun ./hello
```

Allocating Resources Interactively (`salloc`)

- Can also just request resources, then use them interactively
- `salloc` will request resources and give you a new shell

```
$ salloc --nodes=2 --ntasks-per-node=1
salloc: Granted job allocation 546
$ squeue
       JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
         546 TwoNodes.     bash pwiegand  R       0:05      2 evc[5-6]
$ ssh evc5
Last login: Wed Sep 30 12:15:47 2015 from euser3
$ module load openmpi/openmpi-1.8.6-ic-2015.3.187-slurm-14.11.7
$ mpicc -o hello  hello-world.c
$ mpirun ./hello
Process 0 on evc5 out of 2,  :: FOO_VAR=(null)
Process 1 on evc6 out of 2,  :: FOO_VAR=(null)
$ exit
$ exit
exit
salloc: Relinquishing job allocation 546
```

Canceling Jobs (`scancel`)

- Can easily cancel a job using `scancel`

```
$ sbatch wc-test.slurm
Submitted batch job 547

$ squeue
      JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
        547 OneNode.F SocNetPD pwiegand  R       0:03      1 evc7

$ scancel 547

$ squeue
      JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
```

The Wonders of `scontrol`

- Can do a lot of things with `scontrol`
- For instance *hold* a job:

```
$ scontrol hold 548
```

- Or *release* a held job:

```
$ scontrol release 548
```

- Send a message to stderr input of the job:

```
$ scontrol notify 548 "Read this cool message"
```

- Requeue a running, suspended, or finished batch job:

```
$ scontrol requeue 548
```

Requesting Particular Generic Resources

Can request specific resources:

```
$ salloc --nodes=2 --ntasks-per-node=2  --gres=gpu:2
salloc: Granted job allocation 550
$ squeue
      JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
        550 TwoNodes.     bash pwiegand  R       0:03      2 evc[1,9]
$ exit
exit
salloc: Relinquishing job allocation 550
$ salloc --nodes=1 --ntasks-per-node=1  --gres=phi:1
salloc: Granted job allocation 551
$ squeue
      JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
        551 TwoNodes.     bash pwiegand  R       0:02      1 evc5
$ exit
exit
salloc: Relinquishing job allocation 551
```

Thank You

Thank you for coming!

R. Paul Wiegand
Institute for Simulation & Training
University of Central Florida
wiegand@ist.ucf.edu