SLURM

Simple Linux Utility for Resource Management

John H. Osorio Ríos

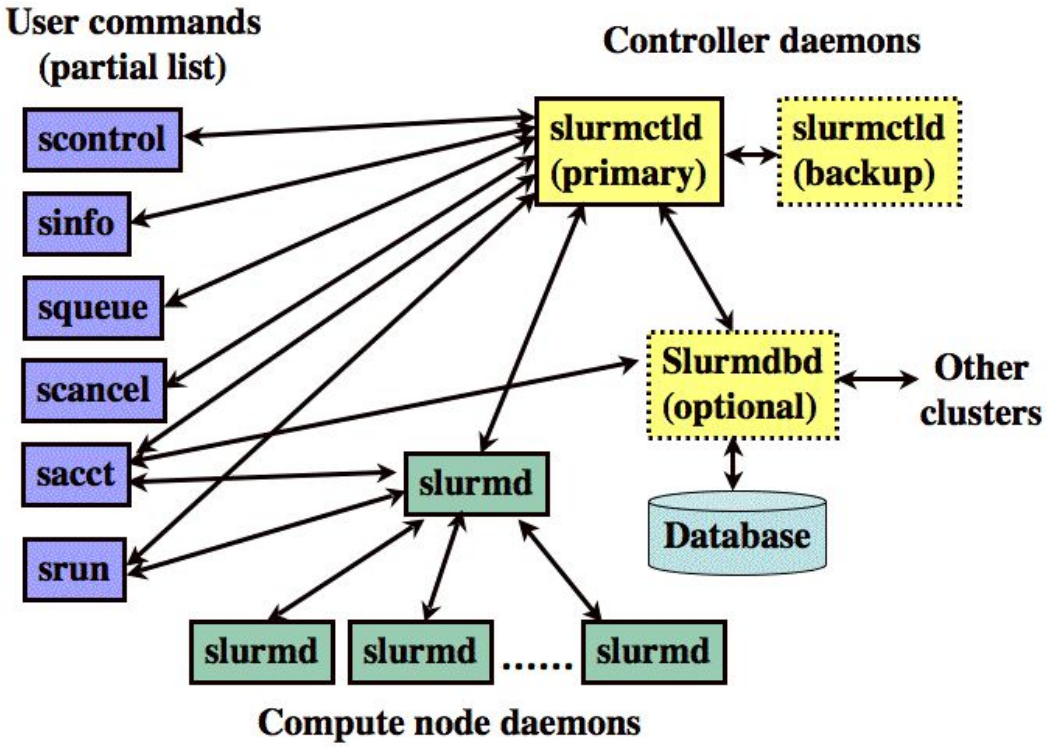
Overview (1/2)

slurm
workload manager

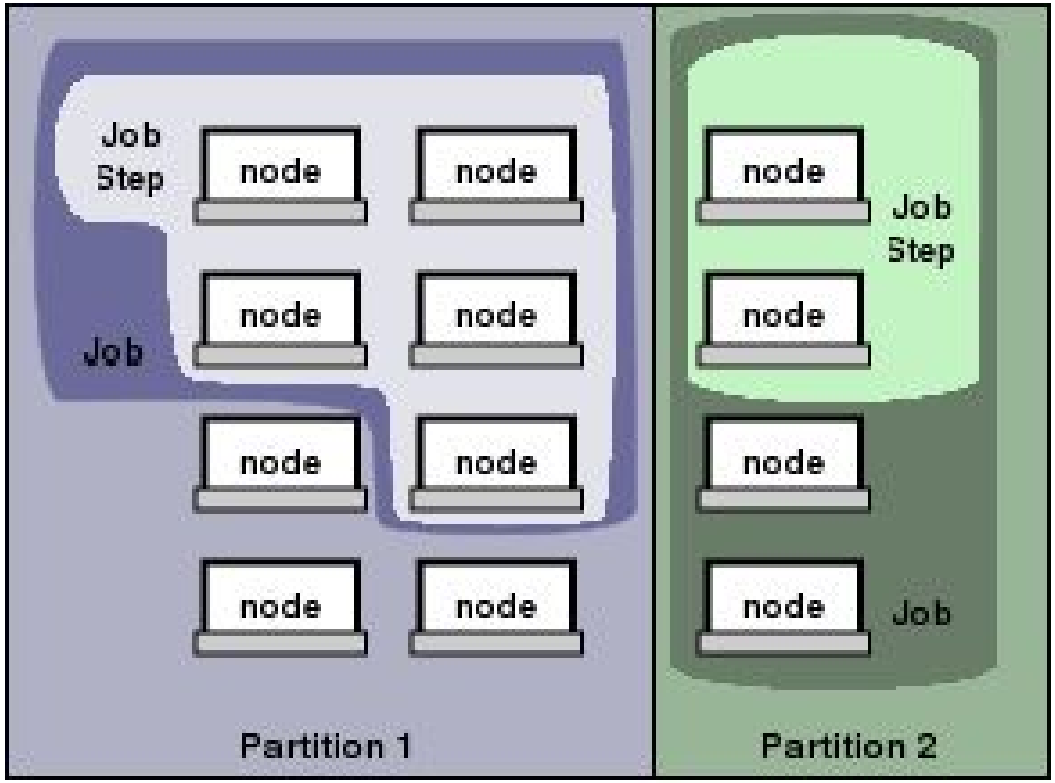
Overview (2/2)

- Slurm is an open source, fault-tolerant, and highly scalable cluster management and job scheduling system for large and small Linux clusters

Architecture (1/2)



Architecture (2/2)

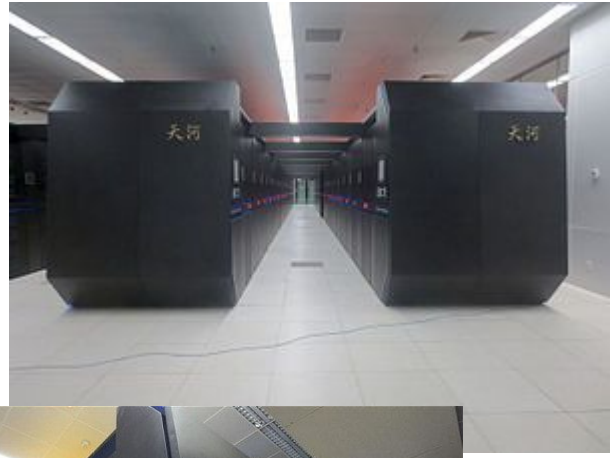


Configurability (1/1)

```
# slurm.conf file generated by configurator easy.html.
# Put this file on all nodes of your cluster.
# See the slurm.conf man page for more information.
#
ControlMachine=masterNode
ControlAddr=192.168.26.114
#
#MailProg=/bin/mail
MpiDefault=none
#MpiParams=ports=#-#
ProctrackType=proctrack/pgid
ReturnToService=1
SlurmctldPidFile=/var/run/slurm-llnl/slurmctld.pid
#SlurmctldPort=6817
SlurmdPidFile=/var/run/slurm-llnl/slurmd.pid
#SlurmdPort=6818
SlurmdSpoolDir=/var/lib/slurm-llnl/slurmd
SlurmUser=slurm
#SlurmdUser=root
StateSaveLocation=/var/lib/slurm-llnl/slurmctld
SwitchType=switch/none
TaskPlugin=task/none
#
#
# TIMERS
#KillWait=30
#MinJobAge=300
#SlurmctldTimeout=120
#SlurmdTimeout=300
#
#
# SCHEDULING
FastSchedule=1
SchedulerType=sched/backfill
#SchedulerPort=7321
SelectType=select/linear
#
#
# LOGGING AND ACCOUNTING
AccountingStorageType=accounting_storage/none
ClusterName=cluster
#JobAcctGatherFrequency=30
JobAcctGatherType=jobacct_gather/none
#SlurmctldDebug=3
SlurmctldLogFile=/var/log/slurm-llnl/slurmctld.log
#SlurmdDebug=3
SlurmdLogFile=/var/log/slurm-llnl/slurmd.log
#
#
# COMPUTE NODES
GresTypes=gpu
NodeName=node[01-06] CPUs=8 RealMemory=30000
Sockets=1 CoresPerSocket=4 ThreadsPerCore=2
Gres=gpu:1 State=UNKNOWN
PartitionName=compute Nodes=node[01-06]
Default=YES MaxTime=INFINITE State=UP
```

Contributors (1/1)

Tianhe-2 (1/1)



SLURM Examples (1/5)

```
master@masterNode:~$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
compute*     up   infinite      6   idle node[01-06]
```

```
mpiu@masterNode:~/hpccourse/slurmexamples$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
compute*     up   infinite      1  alloc node01
compute*     up   infinite      5   idle node[02-06]
```

SLURM Examples (2/5)

```
mpiu@masterNode:~/hpccourse/slurmexamples$ squeue
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
```

```
mpiu@masterNode:~/hpccourse/slurmexamples$ squeue
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
              2582   compute slurmexa     mpiu  R       0:14      1 node01
```

SLURM Examples (3/5)

```
mpiu@masterNode:~/hpccourse/slurmexamples$ srun -N6 hostname
node02
node05
node06
node04
node03
node01
```

```
mpiu@masterNode:~/hpccourse/slurmexamples$ srun -N6 hostname | sort
node01
node02
node03
node04
node05
node06
```

SLURM Examples (4/5)

```
mpiu@masterNode:~/hpccourse/slurmexamples$ srun -N2 bash
hostname
node02
node01
```

```
master@masterNode:~$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
compute*     up   infinite      2  alloc node[01-02]
compute*     up   infinite      4   idle node[03-06]
```

SLURM Examples (5/5)

```
mpiu@masterNode:~$ srun --pty --mem 500  -t 0-1:00 /bin/bash
mpiu@node01:~$
```

```
master@masterNode:~$ squeue -l
Wed Feb 15 10:51:32 2017
             JOBID PARTITION     NAME     USER    STATE       TIME TIME_LIMI  NODES NODELIST(REASON)
              2591   compute     bash     mpiu  RUNNING       0:27   1:00:00      1 node01
```

SBATCH Examples (1/2)

```
1  #!/bin/bash
2  #
3  #SBATCH --job-name=omp_hello_world
4  #SBATCH --output=res_omp_hello_world.out
5  #SBATCH --ntasks=1
6  #SBATCH --cpus-per-task=8
7  #SBATCH --time=10:00
8  #SBATCH --mem-per-cpu=100
9
10 export OMP_NUM_THREADS=$SLURM_CPUS_PER_TASK
11
12 ./omp_hello
```

SBATCH Examples (2/2)

```
mpiu@masterNode:~/hpccourse/openmexamples/omp_hello$ gcc -o omp_hello omp_hello.c -fopenmp
mpiu@masterNode:~/hpccourse/openmexamples/omp_hello$ sbatch omp_hello.sh
Submitted batch job 2599
```

```
mpiu@masterNode:~/hpccourse/openmexamples/omp_hello$ ls
omp_hello  omp_hello.c  omp_hello.sh  res_omp_hello_world.out
```

```
mpiu@masterNode:~/hpccourse/openmexamples/omp_hello$ cat res_omp_hello_world.out
Hello World from thread = 0
Number of threads = 2
Hello World from thread = 1
```

TODO (1/1)

- Run the matrix multiplication made in OpenMP using SLURM
- Check the professor Github Repo and run the OpenMP examples.
- Remember to measure the execution time of the programs.

Bibliography (1/1)

- https://computing.llnl.gov/tutorials/openMP/
- https://slurm.schedmd.com/
- https://slurm.schedmd.com/tutorials.html
- https://slurm.schedmd.com/publications.html

THANKS

john@sirius.utp.edu.co