
SLURMFS

Resource Manager File System for SLURM

Steven Senator
sts@lanl.gov

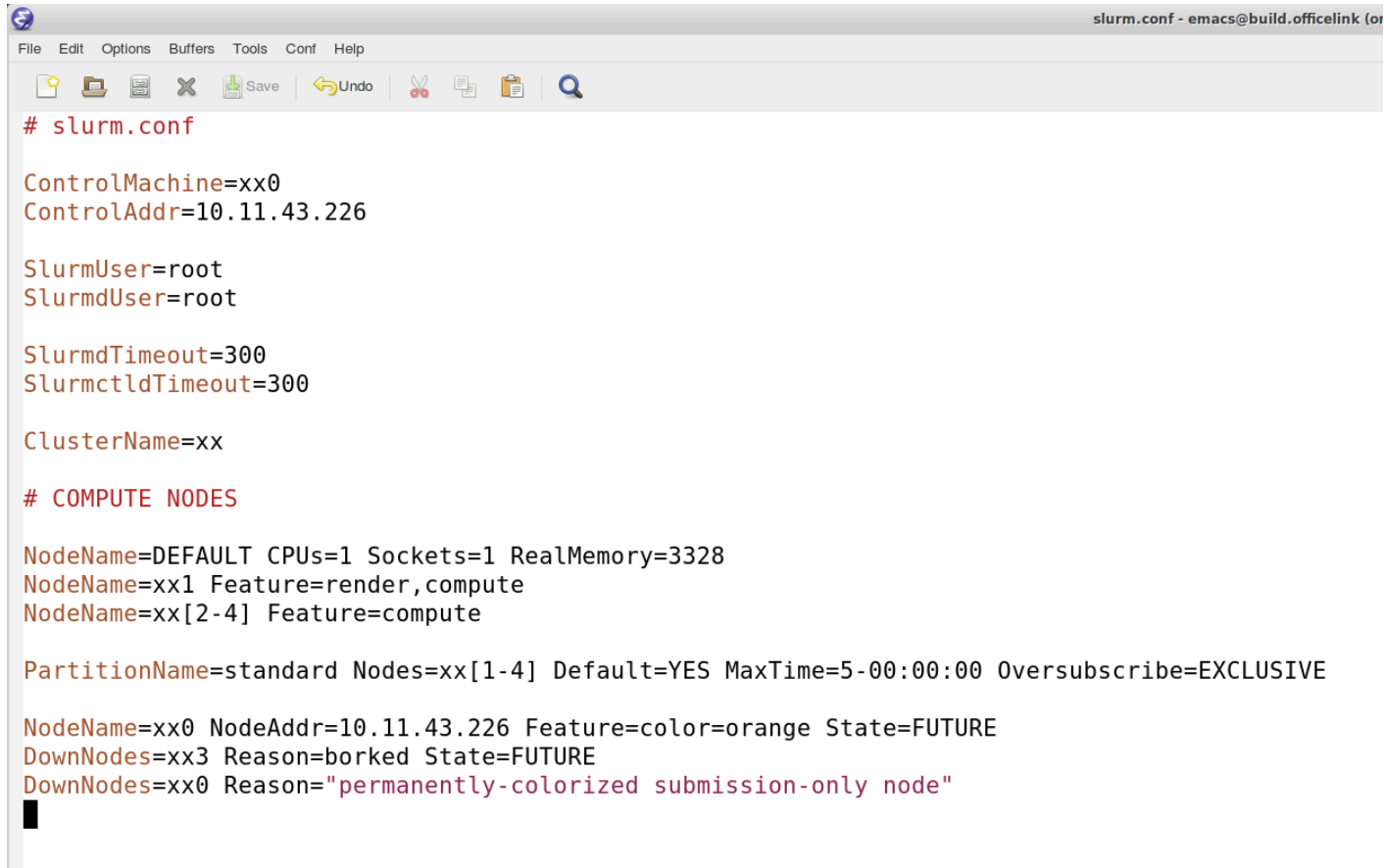
SLUG 2017
Berkeley, CA

Slurmfs is:

- a file system view into slurm state
 - using the published slurm api, only
- a file system interface to effect slurm state change
- an interface rationalizer for diverse tools and user communities
- an enabler for other tools

Slurmfs is not:

- authoritative for slurm state
 - slurm itself is
- a plugin or filter
 - although it may be used as a means to connect to them



```
# slurm.conf

ControlMachine=xx0
ControlAddr=10.11.43.226

SlurmUser=root
SlurmdUser=root

SlurmdTimeout=300
SlurmctldTimeout=300

ClusterName=xx

# COMPUTE NODES

NodeName=DEFAULT CPUs=1 Sockets=1 RealMemory=3328
NodeName=xx1 Feature=render,compute
NodeName=xx[2-4] Feature=compute

PartitionName=standard Nodes=xx[1-4] Default=YES MaxTime=5-00:00:00 Oversubscribe=EXCLUSIVE

NodeName=xx0 NodeAddr=10.11.43.226 Feature=color=orange State=FUTURE
DownNodes=xx3 Reason=borked State=FUTURE
DownNodes=xx0 Reason="permanently-colored submission-only node"
```

```
123% scontrol show node xx0
NodeName=xx0 CoresPerSocket=1
  CPUAlloc=0 CPUErr=0 CPUTot=1 CPULoad=N/A
  AvailableFeatures=color=orange
  ActiveFeatures=color=orange
  Gres=(null)
  NodeAddr=10.11.43.226 NodeHostName=xx0
  RealMemory=11855 AllocMem=0 FreeMem=N/A Sockets=1 Boards=1
  State=DOWN* ThreadsPerCore=1 TmpDisk=8705 Weight=1 Owner=N/A MCS_label=N/A
  BootTime=None SlurmdStartTime=None
  CfgTRES=cpu=1,mem=11855M
  AllocTRES=
  CapWatts=n/a
  CurrentWatts=0 LowestJoules=0 ConsumedJoules=0
  ExtSensorsJoules=n/s ExtSensorsWatts=0 ExtSensorsTemp=n/s
  Reason=permanently-colored submission-only node [root@2017-09-08T14:21:28]
```

5

slurmfs

```

123% scontrol show node xx0
NodeName=xx0 CoresPerSocket=1
CPUAllo
Availab
ActiveF
Gres=(n
NodeAdd
RealMem
State=D
BootTim
CfgTRES
AllocTR
CapWatt
Current
ExtSens
Reason=
-r--r----. 1 root root 6 Sep 19 18:08 real_memory
-r--r----. 1 root root 2 Sep 19 18:08 sockets
-r--r----. 1 root root 2 Sep 19 18:08 threads
136%
136% cat fs/partitions/standard/nodes/xx0/attributes/real_memory
11855
137% █

```

6

slurmfs

```
xx-build ~/lanl/sw/slurm/slurmfs/SOURCES [Linux 3.10.0-514.26.2.el7.x86_64]
File Edit View Search Terminal Help
138% squeue
JOBID PARTITION NAME USER ST TIME NODES NODELIST(REASON)
202024 standard holdjob sts PD 0:00 1 (JobHeldUser)
139% █
```

```

xx-build ~/lanl/sw/slurm/slurmfs/SOURCES [Linux 3.10.0-514.26.2.el7.x86_64]
File Edit View Search Terminal Help
141% squeue
      JOBID PARTITION   NAME   USER ST   TIME   NODES NODELIST(REASON)
      202024 standard  holdjob sts PD   0:00     1 (JobHeldUser)
142% ls -l fs/jobs
total 2
dr-xr-x--T. 5 root root 344 Sep 19 18:08 202024
143% ls -l fs/jobs/202024
total 1
dr-xr-x---. 2 root root 344 Sep 19 18:08 jobsteps
144% █

```



```
145% tree fs
fs
├── attributes
│   ├── ClusterName
│   ├── ControlMachine
│   ├── slurm_api_version
│   ├── SlurmctldTimeout
│   ├── SlurmdTimeout
│   ├── SlurmdUser
│   ├── slurmd_user_name
│   ├── SlurmUser
│   └── slurm_user_name
├── jobs
│   └── 202024
│       └── jobsteps
├── partitions
│   └── standard
│       ├── attributes
│       ├── flags
│       ├── name
│       ├── state_up
│       ├── total_cpus
│       └── total_nodes
│       └── nodes
│           └── <node>
│               ├── attributes
│               ├── allocjob
│               ├── arch
│               ├── cores
│               ├── cpus
│               ├── name
│               ├── node_hostname
│               ├── node_state
│               ├── os
│               ├── real_memory
│               ├── sockets
│               └── threads
```

9

slurmfs

What broke when porting from slurm 2.3.3 to 17.02:

- individual nodes have limited view
- artifact: <node> directory
- control mechanisms:
 - `chmod +x jobs/<jobid>`
 - `chmod -x partitions/<partitionname>`
 - `echo "drain" > <nodename>/control/state`
 - `echo "exclusive" > partitions/<partitionname>/control/oversubscribe`
- libcollect is not retrieving all attributes
 - type system needs extension and update
- backing store & dependent functions
 - history tail / remounts
- selinux context

10

slurmfs

What's next:

- Job
 - attributes: complete scontrol show job parity
 - priority / queue position
- new* features:
 - Views of:
 - reservations
 - sacctmgr datums: accounts, qos, associations, users
 - scheduling attributes
 - slurmdb as backing store
- examples and prototypes:
 - prolog mounts slurmfs on a node; epilog unmounts it
 - a non-slurm-aware client script can get/set state

Questions, to the audience and community of interest:

- Associations
 - most useful fs structure? ...for which use cases?
 - sub-dir containing sym-links to entities
 - multiple groupings, to enable convenient queries
 - ordered by?
- Job Priorities
 - within hierarchy
 - by job state sub-dir?
 - by partition?
- Other entities
 - Gres & Tres, plugins, power
 - If, where & how could this be represented?
- Federation

12

slurmfs

What's next:

- sharing of the code with interested parties
- testing at scale
- contribution to the community