AI Cluster - Slurm

Cluster is up and running now. People involved in the cluster creation should go check it out.

The nodes aren't locked down so anyone with a CS account can login so feel free to get some students to check it out. You are responsible for communicating with anyone you bring on though for now. Any communication should be done via this email chain or in the #ai-cluster Discord channel.

TODO

Since I'm still working on it, I don't guarantee any uptime yet. Mainly I need to make sure TRES tracking is working like we want. This will involve restarting slurmd and slurmctld which will kill running jobs.

- ~~generate report of storage usage~~
- ~~groups (Slurm 'Accounts') created for PI's.~~
    - ~~e.g. ericj_group: ericj, user1, user1, etc~~
- ~~grab QOS data from somewhere (gsheet or some kind of DB)~~
- ~~Properly deploy sync script~~
    - ~~Systemd unit~~
    - ~~main loop~~
- ~~research on slurm plugin to force GRES selection on job submit. Might be able to use:~~
    - ~~SallocDefaultCommand~~
    - ~~Otherwise look for 'AccountingStorageTRES' and 'JobSubmitPlugins' and /etc/slurm-llnl/job_submit.lua ⇐ used to force user to specify '-gres'.~~
    - ~~jobs that do not specify a specific gpu type (e.g. gpu:rtx8000 or gpu:rtx2080ti) could be counted against either one but not specifically the you actually used.~~
    - ~~From 'AccountingStorageTRES' in slurm.conf: "Given a configuration of "AccountingStorageTRES=gres/gpu:tesla,gres/gpu:volta" Then "gres/gpu:tesla" and "gres/gpu:volta" will track jobs that explicitly request those GPU types. If a job requests GPUs, but does not explicitly specify the GPU type, then its resource allocation will be accounted for as either "gres/gpu:tesla" or "gres/gpu:volta", although the accounting may not match the actual GPU type allocated to the job and the GPUs allocated to the job could be heterogeneous. In an environment containing various GPU types, use of a job_submit plugin may be desired in order to force jobs to explicitly specify some GPU type."~~
- ~~ganglia for Slurm:~~ http://ai-mgmt2.ai.cs.uchicago.edu
    - figure why summary view is no longer a thing.
- ~~update 'coolgpus'. Lose VTs when this is running.~~
    - ~~coolgpus: sets fan speeds of all gpus in system.~~
    - ~~Goal is to statically set fan speeds to 80%. The only way to do this is with fake Xservers... but that means you lose all the VTs. Is this a compromise I'm willing to make? It is.~~
- home directory
    - setup backups for home dirs
    - default quota
    - home directory usage report

Demo

kauffman3 is my test CS account.

```
$ ssh kauffman3@fe.ai.cs.uchicago.edu
```

I've created a couple scripts that run some of the Slurm commands but with more useful output. cs-sinfo and cs-squeue being the only two right now.

```
kauffman3@fe01:~$ cs-sinfo
NODELIST     NODES  PARTITION  STATE  CPUS  S:C:T   MEMORY  TMP_DISK WEIGHT
AVAIL_FEATURES                  REASON  GRES
a[001-006]  6      geforce*   idle   64    2:16:2  190000  0        1
'turing,geforce,rtx2080ti,11g'  none    gpu:rtx2080ti:4
a[007-008]  2      quadro     idle   64    2:16:2  383000  0        1
'turing,quadro,rtx8000,48g'     none    gpu:rtx8000:4
```

```
kauffman3@fe01:~$ cs-squeue
JOBID   PARTITION   USER           NAME                        NODELIST
TRES_PER_NSTATE     TIME
```

List the device number of the devices I've requested from Slurm. # These numbers map to /dev/nvidia?

```
kauffman3@fe01:~$ cat ./show_cuda_devices.sh
#!/bin/bash
hostname
echo $CUDA_VISIBLE_DEVICES
```

Give me all four GPUs on systems 1-6

```
kauffman3@fe01:~$ srun -p geforce --gres=gpu:4 -w a[001-006]
./show_cuda_devices.sh
a001
0,1,2,3
a002
0,1,2,3
a006
0,1,2,3
a005
0,1,2,3
a004
0,1,2,3
a003
0,1,2,3
```

give me all GPUs on systems 7-8 # these are the Quadro RTX 8000s

```
kauffman3@fe01:~$ srun -p quadro --gres=gpu:4 -w a[007-008]
./show_cuda_devices.sh
a008
0,1,2,3
a007
0,1,2,3
```

Fairshare

Check out the fairshare values

```
kauffman3@fe01:~$ sshare --long --accounts=kauffman3,kauffman4 --
users=kauffman3,kauffman4
             Account        User  RawShares  NormShares     RawUsage NormUsage
EffectvUsage  FairShare     LevelFS GrpTRESMins     TRESRunMins
------------------- ---------- ---------- ----------- ---------- ---------
-- ----------- ---------- --------- ---------------------------- ------
-----------------------
kauffman3                                  1    0.000094        428 1.000000
1.000000               0.000094 cpu=475,mem=2807810,energy=0,+
 kauffman3             kauffman3           1    1.000000        428 1.000000
1.000000    0.000094    1.000000 cpu=475,mem=2807810,energy=0,+
kauffman4                                  1    0.000094          0 0.000000
0.000000                    inf cpu=0,mem=0,energy=0,node=0,b+
 kauffman4             kauffman4           1    1.000000          0 0.000000
0.000000    1.000000         inf cpu=0,mem=0,energy=0,node=0,b+
```

We are using the FairTree (fairshare algorithm). This is the default in Slurm these days and from what I can tell probably better suits our needs. It is no big deal to change to classic fairshare.

As the system exists now. One Account per User.

```
 Account: kauffman
   Member: kauffman
 User: kauffman
```

We will probably assign fairshare points to accounts, not users.

Storage

```
/net/scratch:
   Create  yourself a directory /net/scratch/$USER. Use it for whatever you
want.
```

```
/net/projects:
  Lives on the home directory server.
```

```
  Idea would be to create a dataset with a quota for people to use.
  Normal LDAP groups that you are used to and available everywhere else
would control access to these directories.
  e.g. jonaslab, sandlab
```

Currently there is no quota on home directories. This is set per user per dataset.

I was able to get homes and scratch each connected via 2x 25G. Both are SSD only so the storage should be FAST.

Each compute node (nodes with gpus) has a zfs mirror mounted at /local I set compression to lz4 by default. Usually this has a performance gain as less data is read and written to disk with a small overhead in CPU usage. As of right now there is no mechanism to clean up /local. At some point I'll probably put a find command in cron that deletes files older than 90 days or so.

Asked Questions

> Do we have a max job runtime?

Yes. 4 hours. This is done per partition.

```
PartitionName=geforce Nodes=a[001-006] Default=YES DefMemPerCPU=2900
MaxTime=04:00:00 State=UP Shared
=YES
PartitionName=quadro  Nodes=a[007-008] Default=NO DefMemPerCPU=5900
MaxTime=04:00:00 State=UP Shared=
YES
```

You can take a look at all the values we set here:

```
fe0[1,2]$ cat /etc/slurm-llnl/slurm.conf
```

The man page: https://slurm.schedmd.com/slurm.conf.html

From:
https://howto.cs.uchicago.edu/ - **How do I?**

Permanent link:
https://howto.cs.uchicago.edu/techstaff:aicluster?rev=1605115561

Last update: **2020/11/11 11:26**