AI Cluster - Slurm

Cluster is up and running now. Anyone with a CS account who wishes to test it out should do so.

Feedback is requested:

#ai-cluster Discord channel or email Phil Kauffman (kauffman@cs dot uchicago dot edu).

Knowledge of how to use Slurm already is preferred at this stage of testing.

The information from the older cluster mostly applies and I suggest you read that documentation: https://howto.cs.uchicago.edu/techstaff:slurm

Demo

kauffman3 is my CS test 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
```

Storage

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

```
/net/projects: (Please ignore this for now)
  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.

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. You are expected to write your code to accommodate for this.

```
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
```