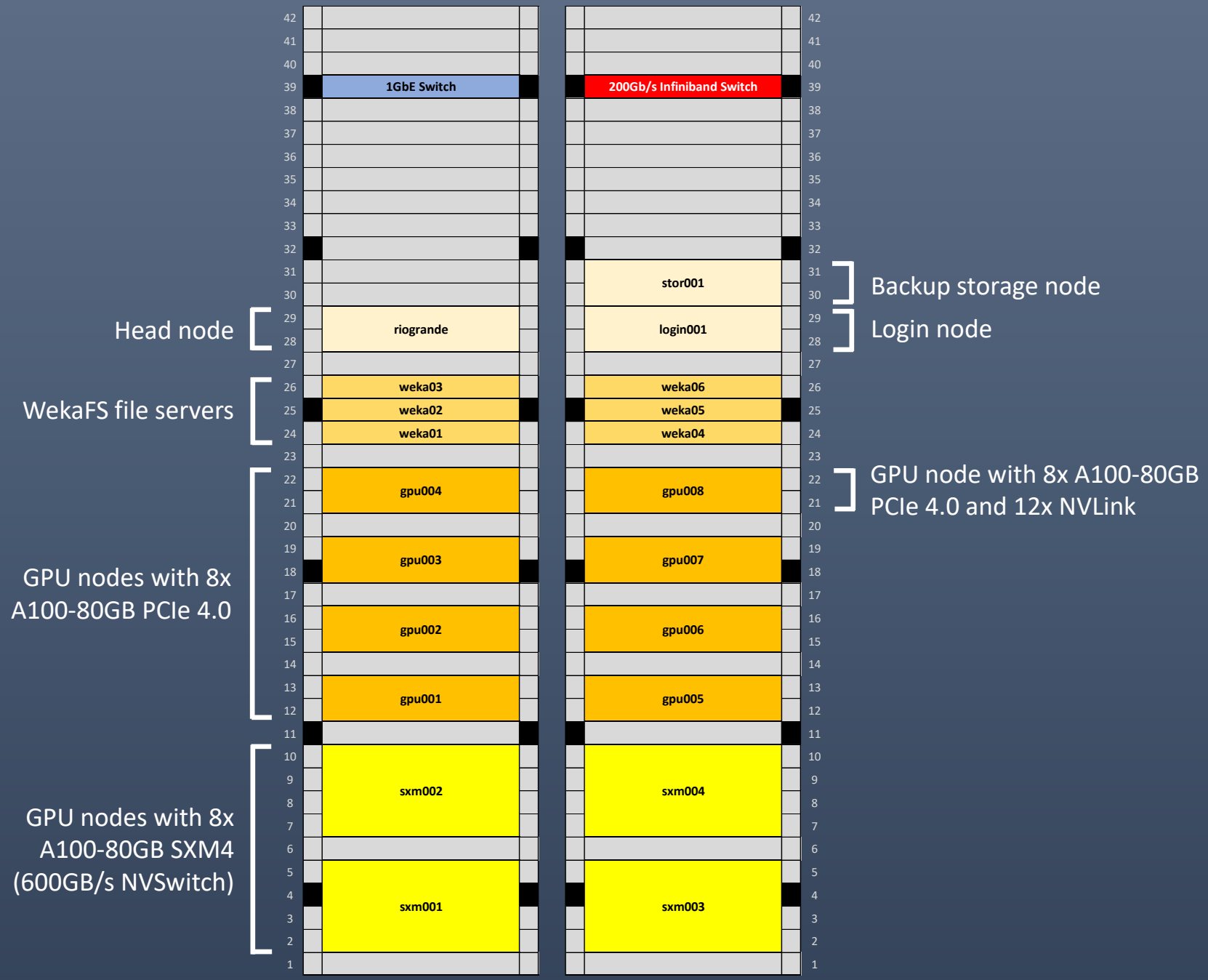
GPU HPC Cluster

The University of Texas
Rio Grande Valley

Bart Willems

Cluster Nodes



Cluster Nodes

2x Intel Xeon Gold 6330 (28 cores, 2.0GHz)

256GB DDR4 memory

2x 960GB SSD

OS mirrored with linux mdadm

12x 12TB SAS HDD

RAID-6 with cache protection

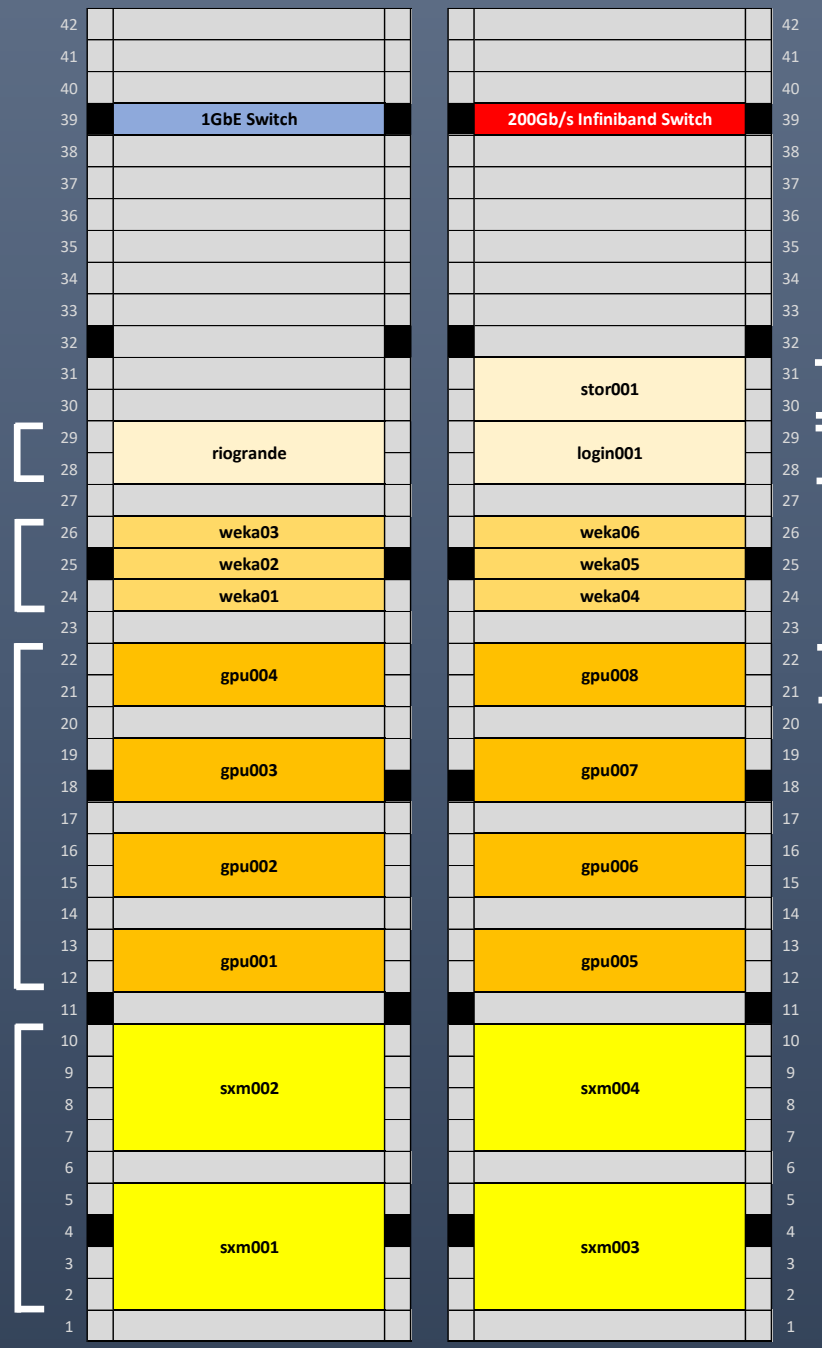
120TB usable (/home)

2x 10GbE SFP+

Campus network

Bonded for redundancy

200Gb/s Infiniband



Head node

WekaFS file servers

GPU nodes with 8x A100-80GB PCIe 4.0

GPU nodes with 8x A100-80GB SXM4 (600GB/s NVSwitch)

Backup storage node

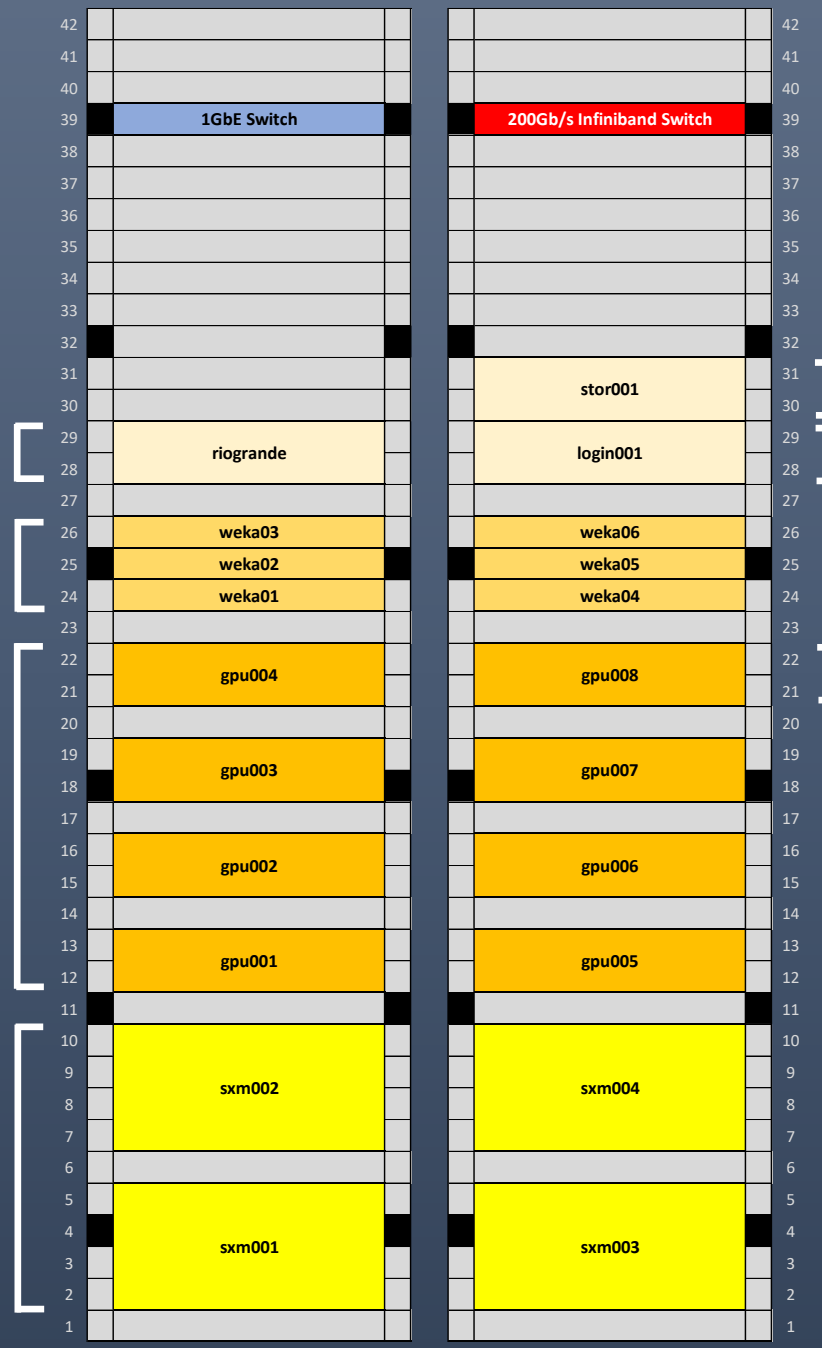
Login node

GPU node with 8x A100-80GB PCIe 4.0 and 12x NVLink

Same as head node

Same as head node minus storage

Cluster Nodes



Backup storage node

Login node

Head node

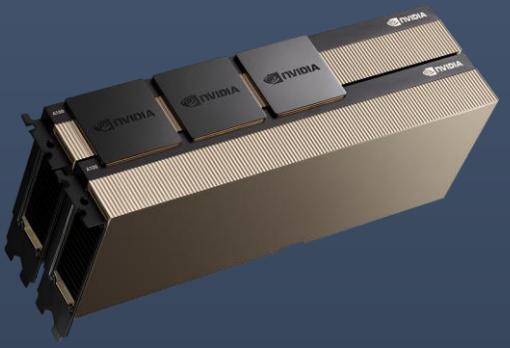
WekaFS file servers

GPU node with 8x A100-80GB PCIe 4.0 and 12x NVLink

2x Intel Xeon Gold 6330 (28 cores, 2.0GHz)
1TB DDR4 memory
3.84TB NVMe SSD
200Gb/s Infiniband

GPU nodes with 8x A100-80GB PCIe 4.0

NVLink bridges connect pairs of GPUs



2x Intel Xeon Gold 6330 (28 cores, 2.0GHz)
1TB DDR4 memory
3.84TB NVMe SSD
200Gb/s Infiniband

GPU nodes with 8x A100-80GB SXM4 (600GB/s NVSwitch)

Cluster Nodes

Parallel file system

/weka/scratch

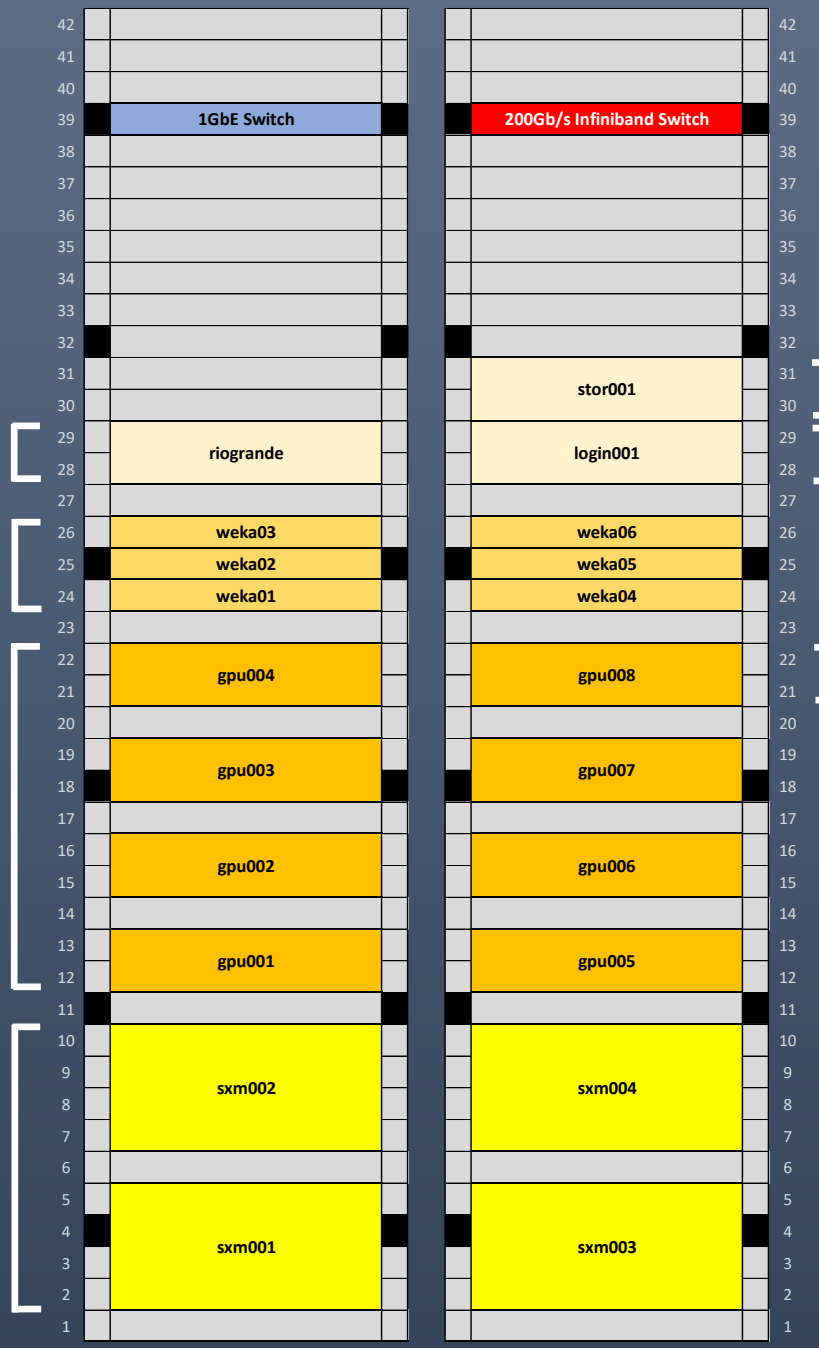
92TB usable

4M Read IOPS

630K Write IOPS

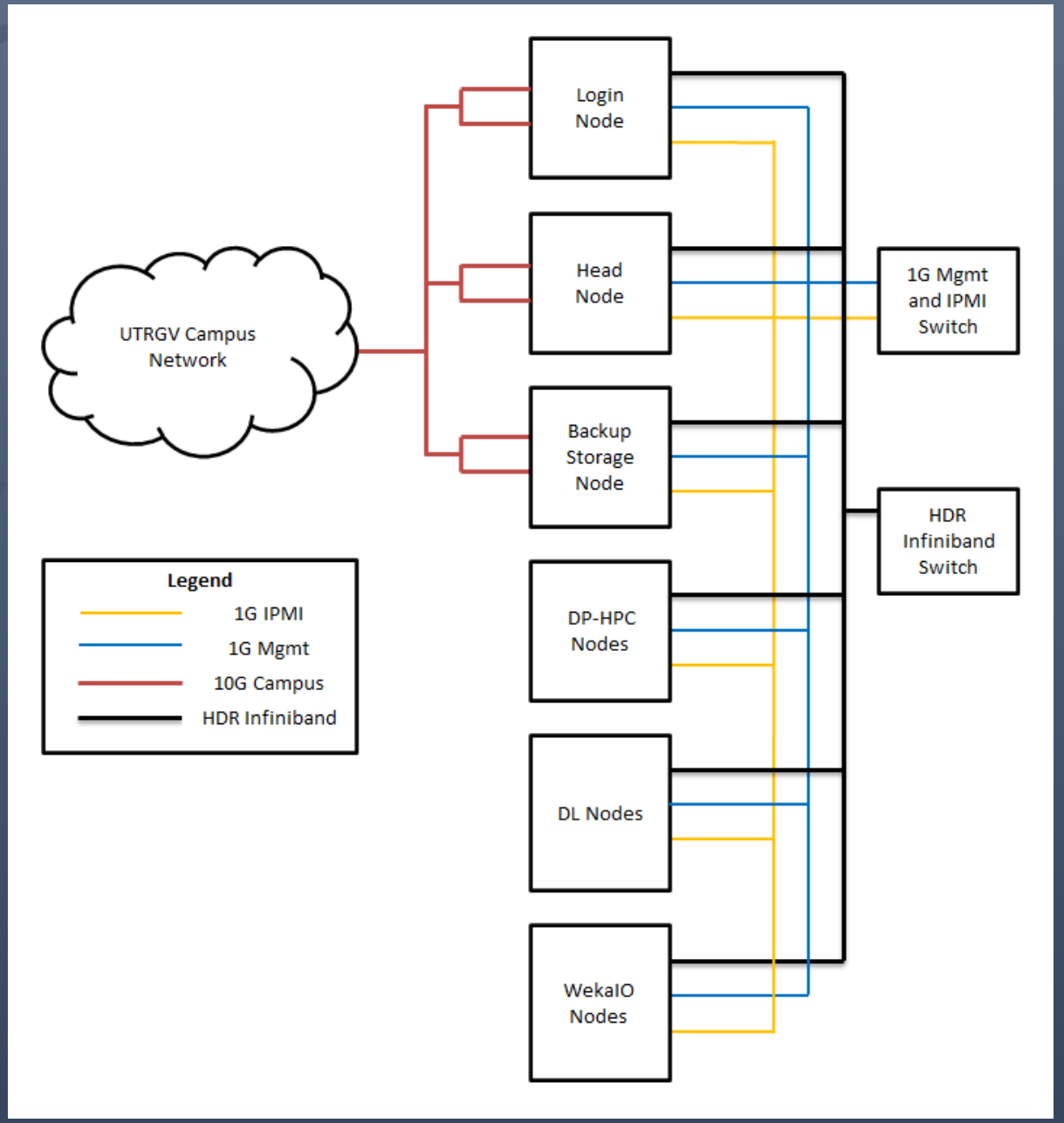
67GB/s Read BW

33GB/s Write BW



Networks

- In-band management
  - 1GbE
  - Linux CLI, ssh, SLURM, …
- Out-of-band management (IPMI)
  - 1GbE
  - Remote power on/off, system event log, …
- HPC communication fabric
  - 200Gb/s Infiniband
  - MPI and data transfer



Cluster Software

Rocky Linux 8.6	Atipa Phoenix Cluster Management 2.2	Mellanox OFED 5.6	Environment Modules 4.5.2	SLURM 21.08.8
gcc-toolset-11 (gcc 11.2.1)	OpenMPI 4.1.3	Intel OneAPI 2022.1.2	ganglia 3.7.2	CUDA 11.6
cuDNN 8	pytorch 1.10.2	tensorflow 2.6.2	caffe2	

Data Storage: /home

Head Node (riogrande)

- 12x 12TB SAS HDD
- RAID-6 (10+2)
- 120TB usable
  - /home
  - NFS export to all nodes
  - Synced to stor001 every night

Backup Storage Node (stor001)

- 12x 12TB SAS HDD
- RAID-6 (10+2)
- 120TB usable
  - /backup
  - /etc/cron.daily/backupUserDirs

```
[root@stor001 ~]# ls -l /backup/
total 0
drwxr-xr-x 3 root root 19 Jun 15  2020 home
[root@stor001 ~]# ls -l /backup/home/
total 4
drwx------ 31 atipa atipa 4096 Jul 25 09:56 atipa
```

Power On Sequence

1. Switches

   Come on as soon as power is plugged in

2. Head node (riogrande)

   Power button

3. WekaFS nodes (weka01…06)

   # ipmipower –u ADMIN2 –p UTRGV@6002900 –h weka0[1-6]-ipmi --on

   (note the double "-" before "on")

4. Start Weka file system (Tuesday afternoon training session)

5. Login and backup storage nodes (login001, stor001)

   # ipmipower –u ADMIN2 –p UTRGV@6002900 –h login001-ipmi --on

   # ipmipower –u ADMIN2 –p UTRGV@6002900 –h stor001-ipmi --on

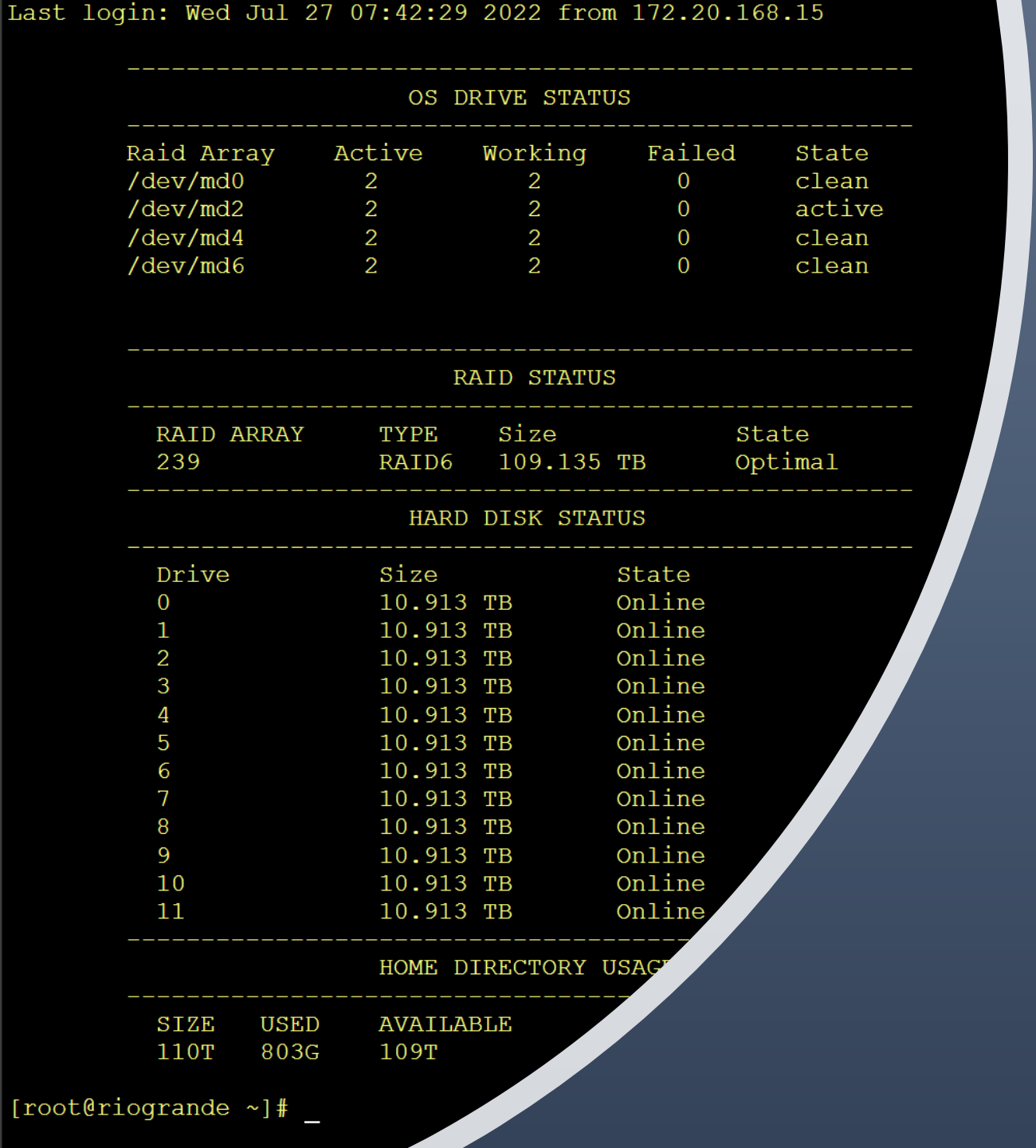
6. GPU nodes (gpu001…008, sxm001…004)

   # ipmipower –u ADMIN2 –p UTRGV@6002900 –h gpu00[1-8]-ipmi --on

   # ipmipower –u ADMIN2 –p UTRGV@6002900 –h sxm00[1-4]-ipmi --on

Since the head node is powered on before the Weka nodes, WekaFS needs to be mounted manually on riogrande after the file system is started:

mount /weka/scratch



Logging into the Head Node

ssh root@riogrande

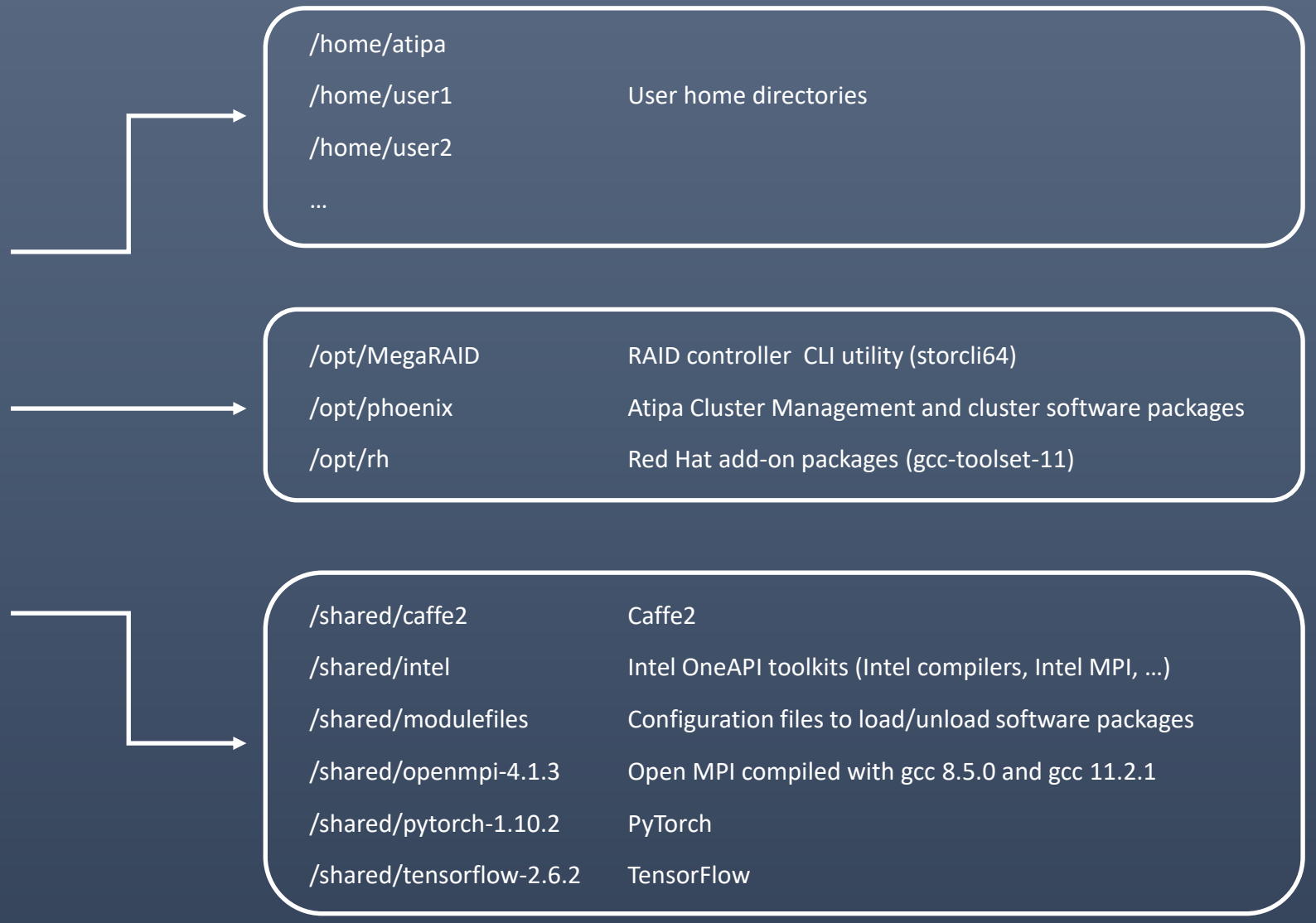
Storage summary:

- *checkSoftwareRAID* (linux mdadm RAID-1)
- *checkRAID (RAID controller RAID-6)*

Note: regular users should log into login001 to compile codes and launch jobs

Directory Structure

/bin
/boot
/dev
/etc
/home
/lib
/lib64
/opt
/proc
/root
/sbin
/shared
/sys
/tftpboot
/tmp
/usr
/var



WARNING: /opt/phoenix/sbin contains scripts used to initialize the cluster nodes. Running these scripts from the Linux command line can break the cluster.

/shared is exported via NFS to all nodes

Use /shared to install 3rd party software that needs to be available on all nodes

Software packages with a "configure" script often have a "--prefix" option to specify the installation directory

Checking Hardware Health: clusterQC

```
[root@riogrande ~]# clusterQC
Class    | Property          | Actual Value    | Expected Value  | Result
=============================================================================
bios     | date              | 10/28/2021      | 10/28/2021      | >>> Good
bios     | revision          | F09             | F09             | >>> Good
part     | boot              | 1040852         | 1040852         | >>> Good
part     | root              | 818667972       | 818667972       | >>> Good
part     | var               | 67009572        | 67009572        | >>> Good
part     | tmp               | 33504276        | 33504276        | >>> Good
part     | home              | 117181521920    | 117181521920    | >>> Good
cpu      | cores             | 56              | 56              | >>> Good
cpu      | version           | Intel(R) Xeon(R) Gold 6330 CPU @ 2.00GHz| Intel(R) Xeon(R) Gold 6330 CPU @ 2.00GHz| >>> Good
mem      | memtotal          | 263724788       | 263724788       | >>> Good
mem      | swaptotal         | 33554424        | 33554424        | >>> Good
hdd      | number            | 3               | 3               | >>> Good
nfs      | exports           | [/opt/phoenix-2.2.0, /shared, /home]| [/opt/phoenix-2.2.0, /shared, /home]| >>> Good
ib       | firmware_version  | 20.33.1048      | 20.33.1048      | >>> Good
ib       | rate              | 200             | 200             | >>> Good
ib       | state             | Active          | Active          | >>> Good
ib       | physical_state    | LinkUp          | LinkUp          | >>> Good
ipmi     | firmware          | 13.03           | 13.03           | >>> Good
sensors  | CPU0              | OK              | OK              | >>> Good
sensors  | CPU1              | OK              | OK              | >>> Good
sensors  | INLET             | OK              | OK              | >>> Good
sensors  | FAN1A             | OK              | OK              | >>> Good
sensors  | FAN2A             | OK              | OK              | >>> Good
sensors  | FAN3A             | OK              | OK              | >>> Good
sensors  | FAN4A             | OK              | OK              | >>> Good
kernel   | version           | 4.18.0-372.9.1.el8.x86_64| 4.18.0-372.9.1.el8.x86_64| >>> Good
Done!

[root@riogrande ~]# clusterQC | grep -i bad
[root@riogrande ~]# _
```

Creating User Accounts: cuseradd

 # cuseradd <username>

Create a new Linux user account on the head node, generate ssh keys, and propagate the user account info to all other nodes

```
[root@riogrande ~]# cuseradd bart

The username given is available.
Creating new user: bart

setting password...
New password:
Retype new password:
Changing password for user bart.
passwd: all authentication tokens updated successfully.
generating ssh key...
```

Executing Commands on all Nodes

fornodes

fornodes <command>

Execute <command> in parallel on all nodes in /opt/phoenix/etc/phoenix/phoenix.nodes (default: sxm001…004, gpu001…008)

Example: *fornodes uptime*

fornodes -s <command>

Execute <command> sequentially on all nodes in /opt/phoenix/etc/phoenix/phoenix.nodes

Example: *fornodes –s uptime*

fornodes –b <basename> -i <range> <command>

Execute <command> on range of nodes with base name

Example: *fornodes –b gpu –i 1-4 uptime*

fornodes –b sxm –i 1-4 “clusterQC | grep –i bad”

Copying a File to all Nodes

acp

acp <file> <destination_dir>

Copy <file> in parallel to all nodes in /opt/phoenix/etc/phoenix/phoenix.nodes (default: sxm001…004, gpu001…008)

Example: *acp /tmp/test.txt /tmp/*

acp "<file1> <file2> …" <destination_dir>

Copy <file1>, <file2>, … in parallel to all nodes in /opt/phoenix/etc/phoenix/phoenix.nodes (default: sxm001…004, gpu001…008)

Example: *acp "/tmp/test.txt /tmp/test2.txt" /tmp/*

acp –b <basename> -i <range> <file> <dest_dir>

Copy <file> to range of nodes with base name

Example: *acp –b gpu –i 1-4 /tmp/test.txt /tmp/*

WARNING: Be careful copying configuration files to all nodes, e.g. copying /etc/fstab to all nodes will render nodes unbootable!

Syncing Configuration Files to all Nodes

syncConfig
syncPasswds

syncConfig

Copy files in /opt/phoenix/etc/phoenix/syncFiles.xml from head to all nodes in /opt/phoenix/etc/phoenix/phoenix.nodes (default: sxm001…004, gpu001…008)

Example: *syncConfig*

Should only be necessary if files are added to syncFiles.xml or files in are syncFiles.xml edited

syncPasswds

Copy /etc/passwd, /etc/group, /etc/shadow from head to all nodes in /opt/phoenix/etc/phoenix/phoenix.nodes (default: sxm001…004, gpu001…008)

Example: *syncPasswds*

Should only be necessary if a user account was created without *cuseradd*

Options:

-n: Copy to specific node, e.g. *syncConfig –n gpu004*

-b, -i: Copy to range of nodes with base name

Checking OS Disk Health in Head, Login, Storage Nodes

2x 960GB SSD mirrored with Linux software RAID

cat /proc/mdstat

```
[root@riogrande ~]# cat /proc/mdstat
Personalities : [raid1]
md6 : active raid1 sdb5[0] sdc5[1]
      33520640 blocks super 1.2 [2/2] [UU]
      bitmap: 0/1 pages [0KB], 65536KB chunk

md4 : active raid1 sdb3[0] sdc3[1]
      67042304 blocks super 1.2 [2/2] [UU]
      bitmap: 1/1 pages [4KB], 65536KB chunk

md2 : active raid1 sdc6[1] sdb6[0]
      819067904 blocks super 1.2 [2/2] [UU]
      bitmap: 2/7 pages [8KB], 65536KB chunk

md0 : active raid1 sdc2[1] sdb2[0]
      1046528 blocks super 1.2 [2/2] [UU]
      bitmap: 0/1 pages [0KB], 65536KB chunk

unused devices: <none>
```

checkSoftwareRAID

```
[root@riogrande ~]# checkSoftwareRAID

        ------------------------------------------------------
                          OS DRIVE STATUS
        ------------------------------------------------------
        Raid Array     Active     Working     Failed     State
        /dev/md0          2           2          0       clean
        /dev/md2          2           2          0       clean
        /dev/md4          2           2          0       clean
        /dev/md6          2           2          0       clean
```

Checking RAID Disk Health in Head/Storage Nodes

12x 12TB HDD hardware RAID-6

checkRAID

```
[root@riogrande ~]# checkRAID

        ------------------------------------------------------
                            RAID STATUS
        ------------------------------------------------------
          RAID ARRAY     TYPE    Size            State
          239            RAID6   109.135 TB      Optimal
        ------------------------------------------------------
                          HARD DISK STATUS
        ------------------------------------------------------
          Drive          Size            State
          0              10.913 TB       Online
          1              10.913 TB       Online
          2              10.913 TB       Online
          3              10.913 TB       Online
          4              10.913 TB       Online
          5              10.913 TB       Online
          6              10.913 TB       Online
          7              10.913 TB       Online
          8              10.913 TB       Online
          9              10.913 TB       Online
          10             10.913 TB       Online
          11             10.913 TB       Online
        ------------------------------------------------------
                        HOME DIRECTORY USAGE
        ------------------------------------------------------
          SIZE   USED    AVAILABLE       PERCENT USED
          110T   793G    109T            1%
```

/opt/MegaRAID/storcli/storcli64 /c0 show

/opt/MegaRAID/storcli/storcli64 /c0 show all

Checking the Infiniband Subnet Manager

```
[root@riogrande ~]# systemctl status opensmd
* opensmd.service - LSB: Activates/Deactivates InfiniBand Subnet Manager
   Loaded: loaded (/etc/rc.d/init.d/opensmd; generated)
   Active: active (running) since Wed 2022-07-27 11:52:27 CDT; 1 day 7h ago
     Docs: man:systemd-sysv-generator(8)
  Process: 3154 ExecStart=/etc/rc.d/init.d/opensmd start (code=exited, status=0/SUCCESS)
 Main PID: 3216 (opensm)
    Tasks: 184 (limit: 1648042)
   Memory: 76.3M
   CGroup: /system.slice/opensmd.service
           |-3216 /usr/sbin/opensm --daemon
           `-3219 osm_crashd

Jul 27 11:52:26 riogrande systemd[1]: Starting LSB: Activates/Deactivates InfiniBand Subnet Manager...
Jul 27 11:52:26 riogrande OpenSM[3216]: /var/log/opensm.log log file opened
Jul 27 11:52:26 riogrande OpenSM[3216]: OpenSM 5.11.0.MLNX20220418.fd3d650
Jul 27 11:52:26 riogrande OpenSM[3216]: Entering DISCOVERING state
Jul 27 11:52:26 riogrande OpenSM[3216]: SM port is down
Jul 27 11:52:27 riogrande opensmd[3154]: Starting IB Subnet Manager.[  OK  ]
Jul 27 11:52:27 riogrande systemd[1]: Started LSB: Activates/Deactivates InfiniBand Subnet Manager.
Jul 27 11:52:36 riogrande OpenSM[3216]: SM port is up
Jul 27 11:52:36 riogrande OpenSM[3216]: Entering MASTER state
```

```
[root@riogrande ~]# sminfo
sminfo: sm lid 1 sm guid 0xe8ebd3030008483e, activity count 37974 priority 14 state 3 SMINFO_MASTER
```

The Infiniband fabric needs at least one instance of opensmd (subnet manager) to be running on the cluster

Master SM: riogrande

Standby SM: stor001

 # systemctl status opensmd

 # sminfo

Checking Infiniband Host Card Status

ibstat

```
[root@riogrande ~]# ibstat
CA 'mlx5_0'
        CA type: MT4123
        Number of ports: 1
        Firmware version: 20.33.1048
        Hardware version: 0
        Node GUID: 0xe8ebd3030008483e
        System image GUID: 0xe8ebd3030008483e
        Port 1:
                State: Active
                Physical state: LinkUp
                Rate: 200
                Base lid: 1
                LMC: 0
                SM lid: 1
                Capability mask: 0xa651e84a
                Port GUID: 0xe8ebd3030008483e
                Link layer: InfiniBand
```

Check port rate on all GPU nodes:

 # fornodes “ibstat | grep Rate”

Checking the Infiniband Fabric

ibdiagnet

Sweep the fabric and collect information from all Infiniband devices (switches and host cards)

Example: *ibdiagnet*

ibdiagnet --get_cable_info

iblinkinfo

Report info on all links in the fabric

Example: *iblinkinfo*

NVIDIA System Management Interface

nvidia-smi
nvidia-smi –q
nvidia-smi –q –i 0 (GPU 0)

```
[root@gpu001 ~]# nvidia-smi
Thu Jul 28 19:50:32 2022
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 510.47.03    Driver Version: 510.47.03    CUDA Version: 11.6     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  NVIDIA A100 80G...  On   | 00000000:1B:00.0 Off |                    0 |
| N/A   36C    P0    43W / 300W |      0MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   1  NVIDIA A100 80G...  On   | 00000000:1C:00.0 Off |                    0 |
| N/A   35C    P0    42W / 300W |      0MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   2  NVIDIA A100 80G...  On   | 00000000:4F:00.0 Off |                    0 |
| N/A   36C    P0    44W / 300W |      0MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   3  NVIDIA A100 80G...  On   | 00000000:50:00.0 Off |                    0 |
| N/A   36C    P0    43W / 300W |      0MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   4  NVIDIA A100 80G...  On   | 00000000:9C:00.0 Off |                    0 |
| N/A   36C    P0    42W / 300W |      0MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   5  NVIDIA A100 80G...  On   | 00000000:9D:00.0 Off |                    0 |
| N/A   36C    P0    45W / 300W |      0MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   6  NVIDIA A100 80G...  On   | 00000000:CE:00.0 Off |                    0 |
| N/A   36C    P0    43W / 300W |      0MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   7  NVIDIA A100 80G...  On   | 00000000:CF:00.0 Off |                    0 |
| N/A   36C    P0    42W / 300W |      0MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|  No running processes found                                                 |
+-----------------------------------------------------------------------------+
```

Reprovisioning a Node

reinstall

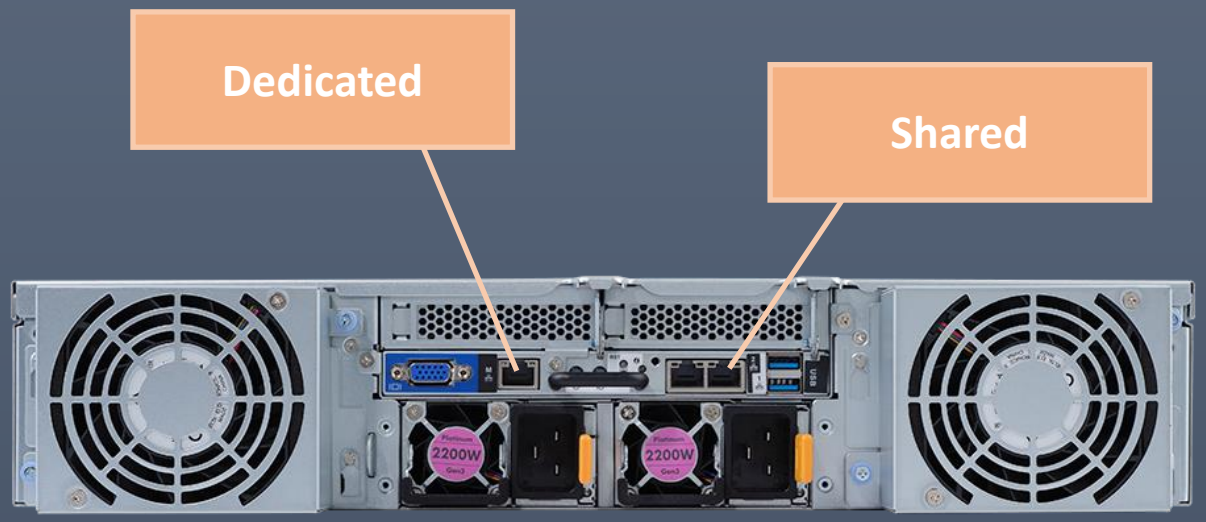
reinstall –f Force reinstall without prompting for confimation

Example:

```
[root@gpu001 ~]# reinstall
Reinstalling the operating system will cause all local data on the node to be lost.
Are you sure you want to continue?
```

The *reinstall* command wipes out the OS disk, reboots the node, and loads the installation image over the network (PXE Boot)

Intelligent Platform Management Interface (IPMI)



- Standardized hardware management interface
- Dedicated hardware chip on motherboard
  - Baseboard Management Controller (BMC)
  - Always powered on even when system is off
  - Shared or dedicated NIC
- Most commonly use to
  - Remotely power server on/off
  - Remotely check server hardware

Displaying IPMI Sensor Information

parseSensors
ipmi-sensors
ipmitool sdr

In-Band (run locally on each host)

parseSensors

Display summary of main temperature and fan sensors

```
[root@riogrande ~]# parseSensors
CPU0     Temp    44.00 C 'OK'
CPU1     Temp    45.00 C 'OK'
INLET    Temp    27.00 C 'OK'
FAN1A    Speed   8700.00 RPM 'OK'
FAN2A    Speed   8700.00 RPM 'OK'
FAN3A    Speed   8700.00 RPM 'OK'
FAN4A    Speed   8700.00 RPM 'OK'
```

ipmi-sensors

ipmitool sdr

Display full IPMI sensor information

Out-of-Band (run from the head node)

ipmitool –U <user> –P <passwd> –H <host> sdr

Example:

ipmitool -U ADMIN2 -P UTRGV@6002900 -H gpu001-ipmi sdr

Displaying IPMI Event Log Information

ipmitool sel
ipmitool sel clear

In-Band (run locally on each host)

ipmitool sel list

Display the IPMI event log

```
[root@stor001 ~]# ipmitool sel list
   1 | 07/31/2022 | 13:47:17 | Event Logging Disabled #0xe1 | Log area reset/cleared | Asserted
   2 | 07/31/2022 | 13:47:25 | Power Unit #0xff | Power off/down | Asserted
   3 | 07/31/2022 | 13:48:29 | System Event | OEM System boot event | Asserted
   4 | 07/31/2022 | 13:48:35 | System Event | OEM System boot event | Asserted
   5 | 07/31/2022 | 13:49:59 | OS Boot #0x22 | boot completed - device not specified | Asserted
```

ipmitool sel clear

Clear the IPMI event log

```
[root@stor001 ~]# ipmitool sel clear
Clearing SEL.  Please allow a few seconds to erase.
[root@stor001 ~]# ipmitool sel list
   1 | 07/31/2022 | 13:52:25 | Event Logging Disabled #0xe1 | Log area reset/cleared | Asserted
```

Out-of-Band (run from the head node)

ipmitool –U <user> –P <passwd> –H <host> sel list

Example:

ipmitool -U ADMIN2 -P UTRGV@6002900 -H gpu001-ipmi sel list

Remotely Powering Servers On/Off

ipmipower

Out-of-Band (run from the head node)

ipmipower –u <user> –p <passwd> –h <host> -s

Display power status of the host

ipmipower –u <user> –p <passwd> –h <host> --on

Power on the host

ipmipower –u <user> –p <passwd> –h <host> --off

Power off the host

```
[root@riogrande ~]# ipmipower -u ADMIN2 -p UTRGV@6002900 -h gpu001-ipmi -s
gpu001-ipmi: off
[root@riogrande ~]# ipmipower -u ADMIN2 -p UTRGV@6002900 -h gpu001-ipmi --on
gpu001-ipmi: ok
[root@riogrande ~]# ipmipower -u ADMIN2 -p UTRGV@6002900 -h gpu001-ipmi -s
gpu001-ipmi: on
```

Other Useful IPMI Commands

ipmitool bmc reset cold
ipmitool

In-Band (run locally on each host)

ipmitool bmc reset cold

Reboot the Baseboard Management Controller (IPMI will be unresponsive for ~2min)

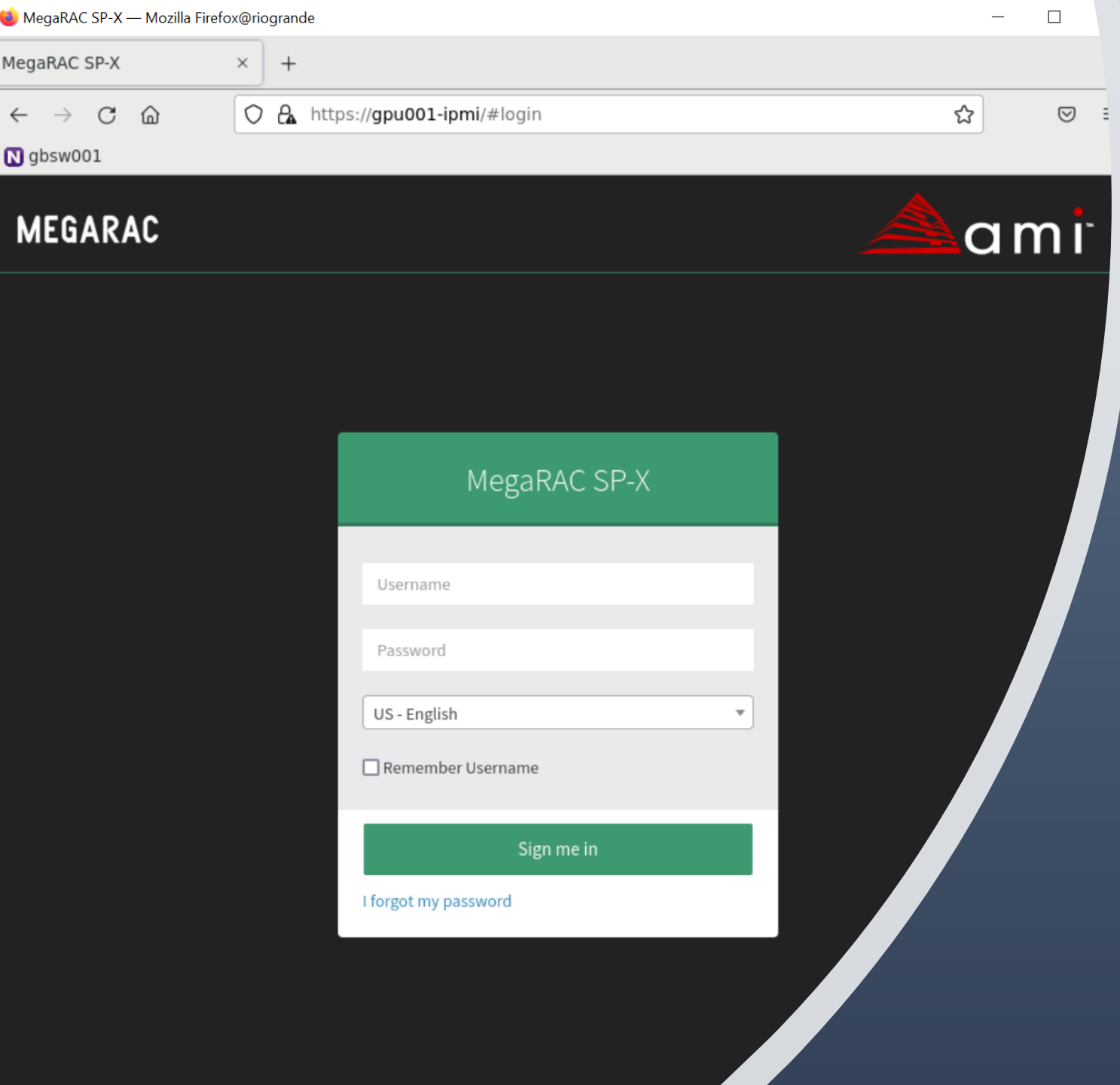
ipmitool lan print

Display BMC network information

```
[root@riogrande ~]# ipmitool lan print
Set in Progress         : Set Complete
Auth Type Support       : NONE MD2 MD5 PASSWORD OEM
Auth Type Enable        : Callback : MD5
                        : User     : MD5
                        : Operator : MD5
                        : Admin    : MD5
                        : OEM      : MD5
IP Address Source       : Static Address
IP Address              : 192.168.122.1
Subnet Mask             : 255.255.255.0
MAC Address             : d8:5e:d3:68:25:44
```

Out-of-Band (run from the head node)

ipmitool –U <user> –P <passwd> –H <host> bmc reset cold



IPMI WebGUI

Launch firefox on the head node:

https://<host>-ipmi/

gbsw001

MegaRAC SP-X

13.03.09
Jun 23 2022 07:38:25 UTC
Host Offline

Quick Links..

- Dashboard
- Sensor
- System Inventory
- FRU Information
- Logs & Reports
- Settings
- Remote Control
- Image Redirection
- Power Control
- Maintenance
- Sign out

2022-07-31 14:23:28 (UTC+00:00 GMT) | US - English | BIOS | Sync | Refresh | ADMIN2

Dashboard Control Panel

Home > Dashboard

1 d 0 hrs
Power-On Hours

1
Access Logs
More info

25
Pending Deassertions
More info

Today (10) Details

30 days (33) Details

Sensor Monitoring

All sensors are good now!

Currently recovered

MegaRAC SP-X

13.03.09
Jun 23 2022 07:38:25 UTC
Host Offline

Quick Links..

- Dashboard
- Sensor
- System Inventory
- FRU Information
- Logs & Reports
  - IPMI Event Log
  - Audit Log
  - Video Log
- Settings
- Remote Control
- Image Redirection
- Power Control

2022-07-31 14:24:25 (UTC+00:00 GMT) | US - English | BIOS | Sync | Refresh | ADMIN2

Event Log All sensor event logs

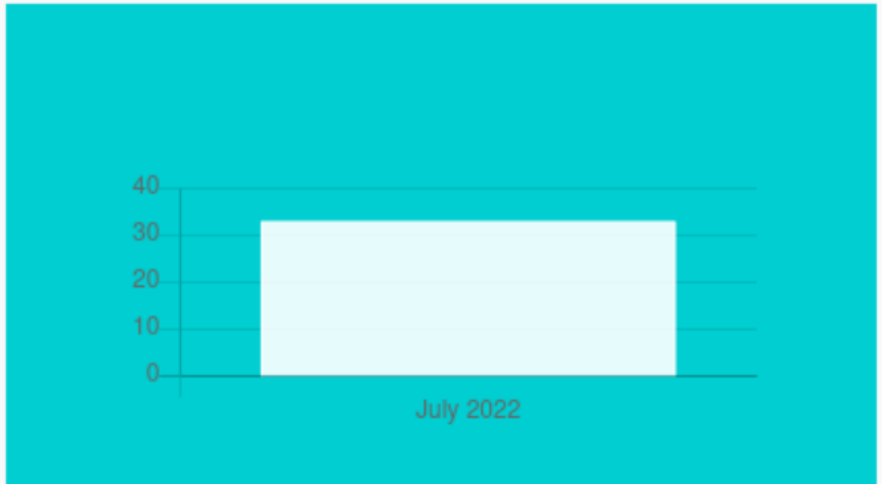
Home > Event Log

Filter by Date 1970-01-01 00:00 - 2022-07-31 14:24 Filter by type All Events

All Sensors Page 1

Clear Event Logs | Download Event Logs

Event Log: 33 (0 ~ 20)



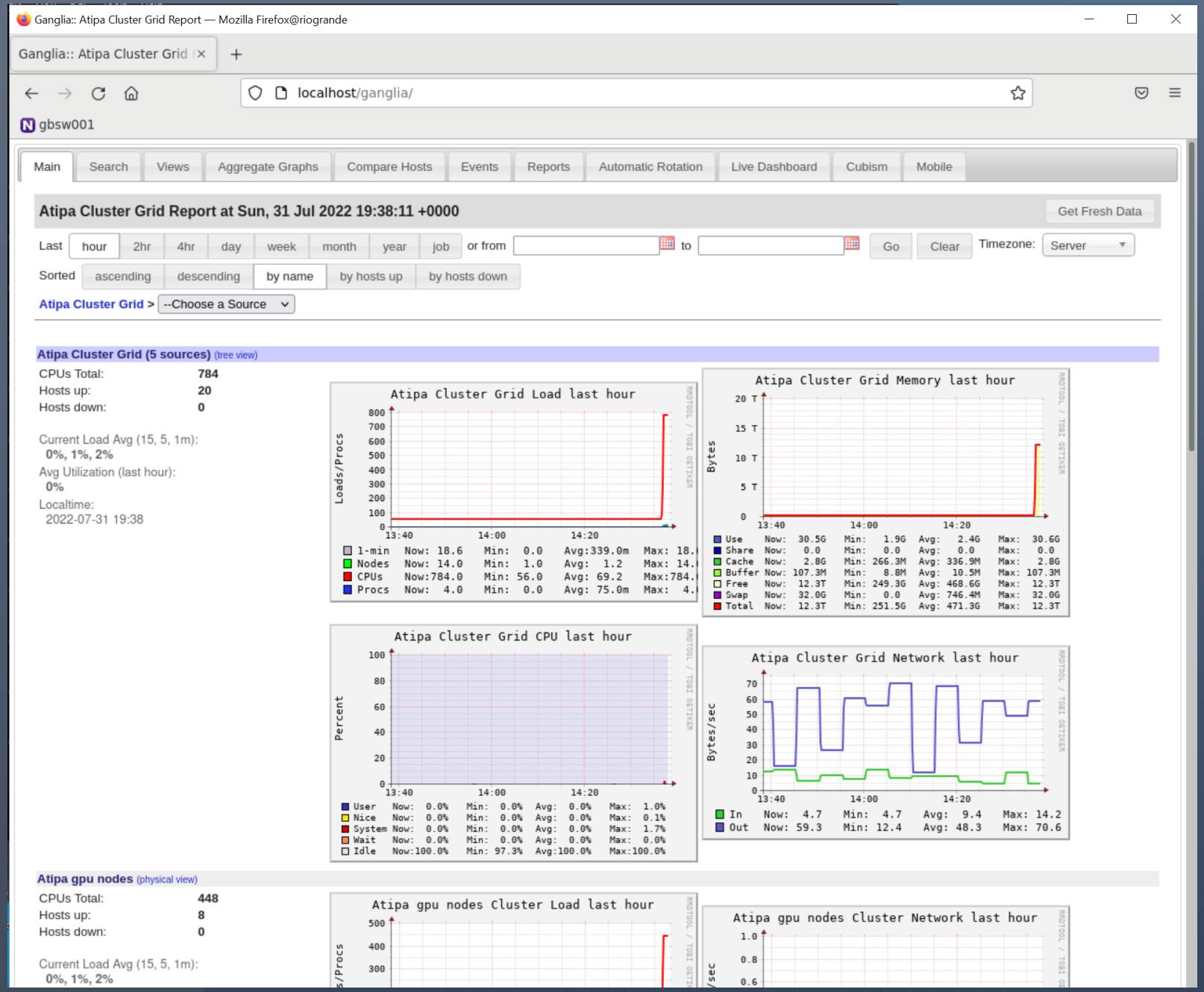
July 2022

ID: 33 Unknown sensor of type power_unit logged a BMC Event : Power Off / Power Down was asserted — 15 minutes ago

ID: 32 PS2_Status sensor of type power_supply logged a BMC Event : Presence detected was asserted — 15 minutes ago

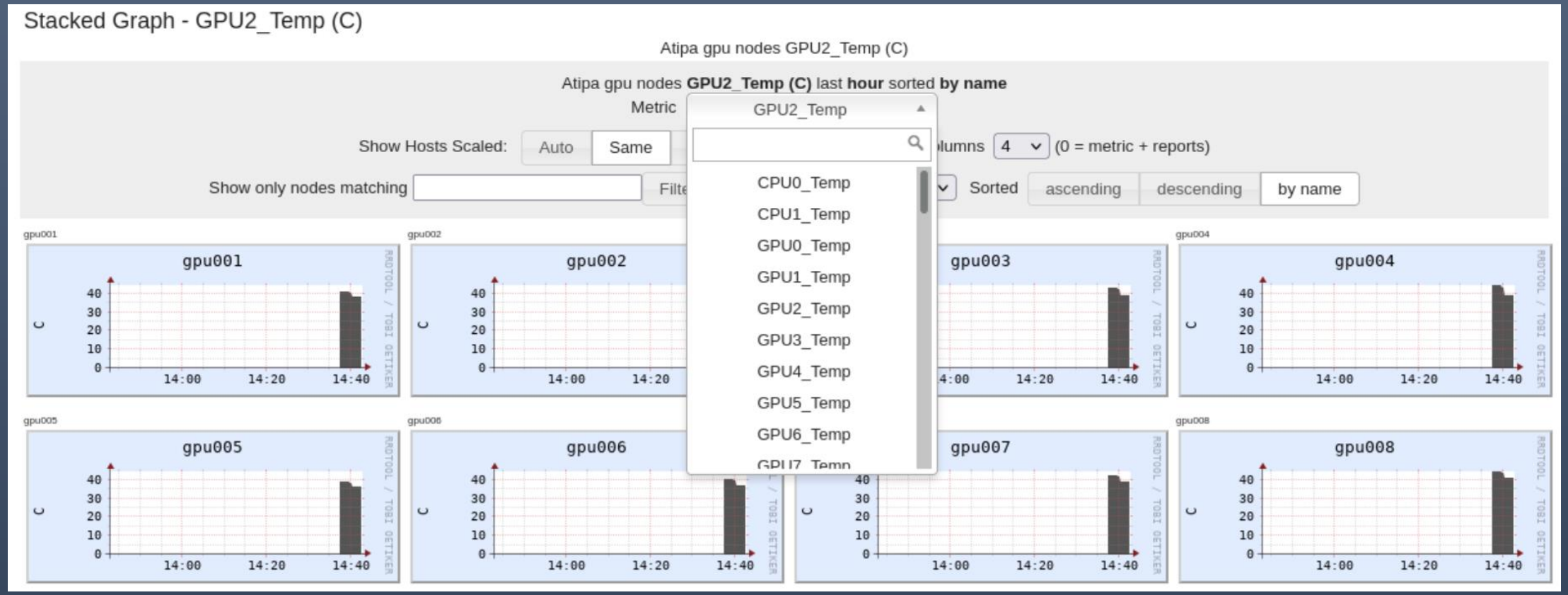
Ganglia: Cluster Health WebGUI

http://localhost/ganglia



Ganglia: Cluster Health WebGUI

http://localhost/ganglia



Installing 3rd Party Software

/shared is accessible by all compute nodes and does not conflict with stock OS packages

Example:

[root@riogrande ~]# gcc –v

…

gcc version 11.2.1

[root@riogrande ~]# cd /opt/phoenix/install/contrib/src

[root@riogrande ~]# wget https://download.open-mpi.org/release/open-mpi/v4.1/openmpi-4.1.3.tar.gz

[root@riogrande src]# tar xvfz openmpi-4.1.3.tar.gz

[root@riogrande src]# cd openmpi-4.1.3

[root@riogrande openmpi-4.1.3]# ./configure --prefix=/shared/openmpi-4.1.3/gcc-11.2.1 --enable-mpirun-prefix-by-default --enable-static --without-verbs --with-ucx --with-knem=/opt/knem-1.1.4.90mlnx1 --with-cuda CC=gcc CXX=g++ FC=gfortran

[root@riogrande openmpi-4.1.3]# make -j 32 all

[root@riogrande openmpi-4.1.3]# make install

[root@riogrande openmpi-4.1.3]# cp config.log /shared/openmpi-4.1.3/gcc-11.2.1/

```
[root@riogrande ~]# module avail
---------------------------------------- /shared/modulefiles ----------------------------------------
gcc/11.2.1  openmpi/4.1.3/gcc/8.5.0  openmpi/4.1.3/gcc/11.2.1

---------------------------------- /shared/intel/oneapi/modulefiles ----------------------------------
advisor/2022.0.0         debugger/2021.5.0        init_opencl/2022.0.2          mkl32/2022.0.2
advisor/latest           debugger/latest          init_opencl/latest            mkl32/latest
ccl/2021.5.1             dev-utilities/2021.5.2   inspector/2022.0.0            mpi/2021.5.1
ccl/latest               dev-utilities/latest     inspector/latest              mpi/latest
clck/2021.5.0            dnnl-cpu-gomp/latest     intel_ipp_ia32/2021.5.2       oclfpga/2022.0.2
clck/latest              dnnl-cpu-iomp/latest     intel_ipp_ia32/latest         oclfpga/latest
compiler-rt/2022.0.2     dnnl-cpu-tbb/latest      intel_ipp_intel64/2021.5.2    tbb/2021.5.1
compiler-rt/latest       dnnl/latest              intel_ipp_intel64/latest      tbb/latest
compiler-rt32/2022.0.2   dpct/2022.0.0            intel_ippcp_ia32/2021.5.1     tbb32/2021.5.1
compiler-rt32/latest     dpct/latest              intel_ippcp_ia32/latest       tbb32/latest
compiler/2022.0.2        dpl/2021.6.0             intel_ippcp_intel64/2021.5.1  vpl/2022.0.0
compiler/latest          dpl/latest               intel_ippcp_intel64/latest    vpl/latest
compiler32/2022.0.2      icc/2022.0.2             itac/2021.5.0                 vtune/2022.0.0
compiler32/latest        icc/latest               itac/latest                   vtune/latest
dal/2021.5.3             icc32/2022.0.2           mkl/2022.0.2
dal/latest               icc32/latest             mkl/latest
```

Environment Modules

- Add/remove locations of binaries ($PATH) and libraries ($LD_LIBRARY_PATH) to the user environment by loading/unloading modules
- Module definitions are located in */shared/modulefiles*
- Add preferred default modules to $HOME/.bash_profile

Loading and Unloading Modules

```
[root@riogrande ~]# module list
Currently Loaded Modulefiles:
 1) gcc/11.2.1
[root@riogrande ~]# which mpirun
/usr/bin/which: no mpirun in (/opt/rh/gcc-toolset-11/root/usr/bin:/usr/share/Modules/bin:/usr/local/sb
in:/usr/local/bin:/usr/sbin:/usr/bin:/usr/local/cuda/bin:/opt/phoenix-2.2.0/sbin:/opt/phoenix-2.2.0/bi
n:/root/bin)
[root@riogrande ~]# module load openmpi/4.1.3/gcc/11.2.1
[root@riogrande ~]# module list
Currently Loaded Modulefiles:
 1) gcc/11.2.1   2) openmpi/4.1.3/gcc/11.2.1
[root@riogrande ~]# which mpirun
/shared/openmpi-4.1.3/gcc-11.2.1/bin/mpirun
[root@riogrande ~]# module purge
[root@riogrande ~]# module list
No Modulefiles Currently Loaded.
```

```
[root@riogrande ~]# cat /home/atipa/.bash_profile
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
        . ~/.bashrc
fi

# User specific environment and startup programs
module load gcc/11.2.1
```

module list

Display currently loaded modules

module load <module>

Load <module>

module unload <module>

Unload <module>

module purge

Unload all currently loaded modules

Installing RPM Packages

- Rocky Linux packages

  yum install <package>

  Example: *yum install ipmitool*

- 3rd Party RPM packages

  Download to /opt/phoenix/install/contrib/RPMS/

  Update repo *setupRepos –u –r contrib yum*

  Clear cache *yum clean all*

  Install *yum install <package>*

  Do not run setupRepos on the Rocky Linux repositories (BaseOS and AppStream)!

- When installing RPMs on the GPU nodes, they should also be added to the node provisioning system (contact us)

Installing Deep Learning Frameworks

Step 1: Install in $HOME as a non-privileged user

Step 2: If multiple users need the same framework, repeat the installation as root in */shared*

Use python “virtual environments”

Each virtual environment has its own Python binary (which matches the version of the binary that was used to create this environment) and can have its own independent set of installed Python packages in its site directories.

https://docs.python.org/3/library/venv.html

Example: Installing TensorFlow

```
[root@riogrande ~]# cd /shared
[root@riogrande shared]# mkdir tensorflow-2.6.2
[root@riogrande shared]# cd tensorflow-2.6.2
[root@riogrande tensorflow-2.6.2]# python3 -m venv tf_env
[root@riogrande tensorflow-2.6.2]# source tf_env/bin/activate
(tf_env) [root@riogrande tensorflow-2.6.2]# pip install --upgrade pip
(tf_env) [root@riogrande tensorflow-2.6.2]# pip install --upgrade tensorflow-gpu
(tf_env) [root@riogrande tensorflow-2.6.2]# pip show tensorflow-gpu
Name: tensorflow-gpu
Version: 2.6.2
…
(tf_env) [root@riogrande tensorflow-2.6.2]# python -c 'import tensorflow as tf; print(tf.__version__)'
2.6.2
(tf_env) [root@riogrande tensorflow-2.6.2]# deactivate
[root@riogrande tensorflow-2.6.2]#
```

Create virtual environment tf_env

Activate virtual environment

Update Python Package Manager

Install TensorFlow

SLURM

Open-source, highly scalable resource managemer and job scheduler

https://slurm.schedmd.com/tutorials.html

https://lists.schedmd.com/cgi-bin/mailman/listinfo/slurm-users

SLURM Services

slurmctld

Runs on head node (riogrande)
Controller accepts jobs from users and allocates resources to jobs

systemctl status slurmctld.service

slurmd

Runs on compute nodes and login node
Monitors tasks, accepts work from slurmctld, starts/stops tasks

systemctl status slurmd.service

Log files: */var/log/slurm*

SLURM Configuration File

/etc/slurm/slurm.conf

Needs to be the same on ALL nodes

Restart service to activate any changes

systemctl restart slurmctld.service

systemctl restart slurmd.service

OR *fornodes "systemctl restart slurmd.service"*

login001 is only used to submit jobs

2 cores reserved for WekaFS

```
# cat /etc/slurm/slurm.conf
…
NodeName=gpu00[1-8] CPUs=56 RealMemory=1024000 Sockets=4 CoresPerSocket=14 ThreadsPerCore=1 Gres=gpu:8 State=UNKNOWN CpuSpecList=0,1
NodeName=sxm00[1-4] CPUs=56 RealMemory=1024000 Sockets=4 CoresPerSocket=14 ThreadsPerCore=1 Gres=gpu:8 State=UNKNOWN CpuSpecList=0,1
NodeName=login001 CPUs=56 RealMemory=257000 Sockets=4 CoresPerSocket=14 ThreadsPerCore=1 CpuSpecList=0,1

PartitionName=gpuq Nodes=gpu00[1-8] Default=YES MaxTime=INFINITE State=UP
PartitionName=gpunvq Nodes=gpu008 Default=NO MaxTime=INFINITE State=UP
PartitionName=sxmq Nodes=sxm00[1-4] Default=NO MaxTime=INFINITE State=UP
```

3 job queues

SLURM Administrator Commands I

sinfo

View partition and node information

```
[root@riogrande ~]# sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
gpuq*        up   infinite      8   idle gpu[001-008]
gpunvq       up   infinite      1   idle gpu008
sxmq         up   infinite      4   idle sxm[001-004]
```

scontrol update nodename=<nodename> state=draining reason="…"

Change node state to draining

```
[root@riogrande ~]# scontrol update nodename=gpu00[2-3] state=drain reason="Memory error"
[root@riogrande ~]# sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
gpuq*        up   infinite      2  drain gpu[002-003]
gpuq*        up   infinite      6   idle gpu[001,004-008]
gpunvq       up   infinite      1   idle gpu008
sxmq         up   infinite      4   idle sxm[001-004]
[root@riogrande ~]# sinfo -R
REASON               USER      TIMESTAMP           NODELIST
Memory error         root      2022-08-01T15:53:20 gpu[002-003]
```

scontrol update nodename=<nodename> state=resume

Change node state to resume

SLURM Administrator Commands II

scontrol show node <NodeName>

Change node state to draining

```
[root@riogrande ~]# scontrol show node gpu001
NodeName=gpu001 Arch=x86_64 CoresPerSocket=14
   CPUAlloc=0 CPUTot=56 CPULoad=0.00
   AvailableFeatures=(null)
   ActiveFeatures=(null)
   Gres=gpu:8
   NodeAddr=gpu001 NodeHostName=gpu001 Version=21.08.8
   OS=Linux 4.18.0-372.9.1.el8.x86_64 #1 SMP Tue May 10 14:48:47 UTC 2022
   RealMemory=1024000 AllocMem=0 FreeMem=1029050 Sockets=4 Boards=1
   CoreSpecCount=2 CPUSpecList=0-1
   State=IDLE ThreadsPerCore=1 TmpDisk=0 Weight=1 Owner=N/A MCS_label=N/A
   Partitions=gpuq
   BootTime=2022-08-10T09:04:23 SlurmdStartTime=2022-08-10T09:07:09
   LastBusyTime=2022-08-10T09:07:09
   CfgTRES=cpu=56,mem=1000G,billing=56
   AllocTRES=
   CapWatts=n/a
   CurrentWatts=0 AveWatts=0
   ExtSensorsJoules=n/s ExtSensorsWatts=0 ExtSensorsTemp=n/s
```

SLURM User Commands

sbatch <ScriptName>

Submit a batch script to Slurm

squeue

Display job info

scancel <JobID>

Cancel job

scontrol show job <JobID>

Display detailed job state

```
[atipa@riogrande mpihello]$ squeue
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
                18      gpuq mpihello    atipa PD       0:00      4 (Resources)
                19      gpuq mpihello    atipa PD       0:00      4 (Priority)
                20      gpuq mpihello    atipa PD       0:00      4 (Priority)
                21      gpuq mpihello    atipa PD       0:00      4 (Priority)
                22      gpuq mpihello    atipa PD       0:00      4 (Priority)
                23      gpuq mpihello    atipa PD       0:00      4 (Priority)
                16      gpuq mpihello    atipa  R       0:09      4 gpu[001-004]
                17      gpuq mpihello    atipa  R       0:05      4 gpu[005-008]
[atipa@riogrande mpihello]$ scontrol show job 18
JobId=18 JobName=mpihello
   UserId=atipa(1000) GroupId=atipa(1000) MCS_label=N/A
   Priority=4294901755 Nice=0 Account=(null) QOS=(null)
   JobState=PENDING Reason=Resources Dependency=(null)
   Requeue=1 Restarts=0 BatchFlag=1 Reboot=0 ExitCode=0:0
   RunTime=00:00:00 TimeLimit=00:10:00 TimeMin=N/A
   SubmitTime=2022-08-04T07:42:05 EligibleTime=2022-08-04T07:42:05
   AccrueTime=2022-08-04T07:42:05
   StartTime=Unknown EndTime=Unknown Deadline=N/A
   SuspendTime=None SecsPreSuspend=0 LastSchedEval=2022-08-04T07:42:10 Scheduler=Main
   Partition=gpuq AllocNode:Sid=riogrande:496548
   ReqNodeList=(null) ExcNodeList=(null)
   NodeList=(null)
   NumNodes=4-4 NumCPUs=216 NumTasks=216 CPUs/Task=1 ReqB:S:C:T=0:0:*:*
   TRES=cpu=216,mem=21600M,node=4,billing=216
   Socks/Node=* NtasksPerN:B:S:C=54:0:*:* CoreSpec=*
   MinCPUsNode=54 MinMemoryCPU=100M MinTmpDiskNode=0
   Features=(null) DelayBoot=00:00:00
   OverSubscribe=OK Contiguous=0 Licenses=(null) Network=(null)
   Command=/home/atipa/mpihello/slurm_submit_gpuq-openmpi.sh
   WorkDir=/home/atipa/mpihello
   StdErr=/home/atipa/mpihello/mpihello.out.18
   StdIn=/dev/null
   StdOut=/home/atipa/mpihello/mpihello.out.18
   Power=
```

Submitting a job to SLURM I

```
[atipa@riogrande ~] cd /home/atipa/mpihello
[atipa@riogrande mpihello]$ cat slurm_submit_gpuq-openmpi.sh
#!/bin/bash
#
#SBATCH --job-name=mpihello
#SBATCH --output=mpihello.out.%j
#SBTACH --error=mpihello.err.%j
#
#SBATCH -N 4 --ntasks-per-node=54
#SBATCH --time=10:00
#SBATCH -p gpuq

module purge
module load openmpi/4.1.3/gcc/8.5.0
module list

mpirun mpihello
```

JobID

4 Nodes, 54 Cores per Node

Run-time limit, hours:minutes:seconds

Job queue/partition (A100 PCIe nodes)

Job commands

Submitting a job to SLURM I

```
[atipa@riogrande mpihello]$ sbatch slurm_submit_gpuq-openmpi.sh
Submitted batch job 15
[atipa@riogrande mpihello]$ squeue
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
                15      gpuq mpihello    atipa  R       0:02      4 gpu[004-007]
```

JobID used for "scontrol show job"

Job is running

Job is using 4 nodes:
gpu004
gpu005
gpu006
gpu007

Submitting a job to SLURM II

[atipa@riogrande ~] cd /home/atipa/mpihello

[atipa@riogrande mpihello]$ cat slurm_submit_gpuq-keras.sh

(Note: keras is not part of TensorFlow)

```
#!/bin/bash

#SBATCH --job-name=keras_resnet_cifar
#SBATCH --output=keras_resnet_cifar.out.%j
#SBTACH --error=keras_resnet_cifar.err.%j

#SBATCH -N 1 --ntasks-per-node=1
#SBATCH --gres=gpu:4
#SBATCH -p gpuq
#SBATCH --time=1:00:00
```

← 1 Node, 1 Core

← 4 GPUs

← Job queue/partition (A100 PCIe nodes)

← Run-time limit, hours:minutes:seconds

```
echo "CUDA_VISIBLE_DEVICES: $CUDA_VISIBLE_DEVICES"
source /shared/tensorflow-2.6.2/tf_env/bin/activate
cd /home/atipa/keras-examples/TensorFlow2-tutorial/01-basic-image-classification
python3 ./resnet_cifar.py
deactivate
```

← Job commands

Submitting a job to SLURM II

```
[atipa@riogrande mpihello]$ sbatch slurm_submit_gpuq-keras.sh
Submitted batch job 25
[atipa@riogrande mpihello]$ squeue
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
                25      gpuq keras_re    atipa  R       0:02      1 gpu001
[atipa@riogrande mpihello]$ scontrol show job 25
JobId=25 JobName=keras_resnet_cifar
   UserId=atipa(1000) GroupId=atipa(1000) MCS_label=N/A
   Priority=4294901748 Nice=0 Account=(null) QOS=(null)
   JobState=RUNNING Reason=None Dependency=(null)
   Requeue=1 Restarts=0 BatchFlag=1 Reboot=0 ExitCode=0:0
   RunTime=00:01:20 TimeLimit=01:00:00 TimeMin=N/A
   SubmitTime=2022-08-10T12:04:08 EligibleTime=2022-08-10T12:04:08
   AccrueTime=2022-08-10T12:04:08
   StartTime=2022-08-10T12:04:08 EndTime=2022-08-10T13:04:08 Deadline=N/A
   SuspendTime=None SecsPreSuspend=0 LastSchedEval=2022-08-10T12:04:08 Scheduler=Main
   Partition=gpuq AllocNode:Sid=riogrande:1816784
   ReqNodeList=(null) ExcNodeList=(null)
   NodeList=gpu001
   BatchHost=gpu001
   NumNodes=1 NumCPUs=1 NumTasks=1 CPUs/Task=1 ReqB:S:C:T=0:0:*:*
   TRES=cpu=1,node=1,billing=1
   Socks/Node=* NtasksPerN:B:S:C=1:0:*:* CoreSpec=*
   MinCPUsNode=1 MinMemoryNode=0 MinTmpDiskNode=0
   Features=(null) DelayBoot=00:00:00
   OverSubscribe=OK Contiguous=0 Licenses=(null) Network=(null)
   Command=/home/atipa/mpihello/slurm_submit_gpuq-keras.sh
   WorkDir=/home/atipa/mpihello
   StdErr=/home/atipa/mpihello/keras_resnet_cifar.out.25
   StdIn=/dev/null
   StdOut=/home/atipa/mpihello/keras_resnet_cifar.out.25
   Power=
   TresPerNode=gres:gpu:4
```

```
[atipa@gpu001 ~]$ nvidia-smi
Wed Aug 10 12:04:20 2022
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 510.47.03    Driver Version: 510.47.03    CUDA Version: 11.6     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  NVIDIA A100 80G...  On   | 00000000:1B:00.0 Off |                    0 |
| N/A   41C    P0    69W / 300W |  79535MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   1  NVIDIA A100 80G...  On   | 00000000:1C:00.0 Off |                    0 |
| N/A   39C    P0    63W / 300W |  79535MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   2  NVIDIA A100 80G...  On   | 00000000:4F:00.0 Off |                    0 |
| N/A   40C    P0    67W / 300W |  79535MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   3  NVIDIA A100 80G...  On   | 00000000:50:00.0 Off |                    0 |
| N/A   40C    P0    66W / 300W |  79535MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   4  NVIDIA A100 80G...  On   | 00000000:9C:00.0 Off |                    0 |
| N/A   40C    P0    42W / 300W |      0MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   5  NVIDIA A100 80G...  On   | 00000000:9D:00.0 Off |                    0 |
| N/A   38C    P0    45W / 300W |      0MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   6  NVIDIA A100 80G...  On   | 00000000:CE:00.0 Off |                    0 |
| N/A   41C    P0    44W / 300W |      0MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   7  NVIDIA A100 80G...  On   | 00000000:CF:00.0 Off |                    0 |
| N/A   39C    P0    43W / 300W |      0MiB / 81920MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|    0   N/A  N/A     25218      C   python3                         79531MiB |
|    1   N/A  N/A     25218      C   python3                         79531MiB |
|    2   N/A  N/A     25218      C   python3                         79531MiB |
|    3   N/A  N/A     25218      C   python3                         79531MiB |
+-----------------------------------------------------------------------------+
```

Best Practices I

- Check storage status daily (riogrande, stor001, login001)
  - OS disks: *checkSoftwareRAID* or "*cat /proc/mdstat*"
  - /home and /backup: *checkRAID* or "/opt/MegaRAID/storcli/*storcli64 /c0 show*"
- Install 3rd party software in */shared* when possible
  - Include version numbers in the directory structure
- Install new applications as an unprivileged user first
- Be careful adding external yum repositories (e.g. epel) to the cluster
  - Download RPMs locally instead
- If it's not broken, don't fix it (updates can wreak havoc)
  - Kernel updates will require Infiniband driver updates
  - Infiniband driver updates require WekaFS updates
  - …

Best Practices II

- Document all changes
- Backup configuration files before making changes

  Create a time-stamped backup of a file: *backupFile <file>*

  Example:

  *[root@riogrande ~]# ls -l /etc/slurm/slurm.conf**
  -rw-r--r-- 1 root root 3515 Jul 27 12:14 /etc/slurm/slurm.conf
  -rw-r--r-- 1 root root 3062 May 11 11:40 /etc/slurm/slurm.conf.example

  [root@riogrande ~]# backupFile /etc/slurm/slurm.conf
  File /etc/slurm/slurm.conf was backed up to /etc/slurm/slurm.conf.07282022-15:41:13

  *[root@riogrande ~]# ls -l /etc/slurm/slurm.conf**
  -rw-r--r-- 1 root root 3515 Jul 27 12:14 /etc/slurm/slurm.conf
  -rw-r--r-- 1 root root 3515 Jul 27 12:14 /etc/slurm/slurm.conf.07282022-15:41:13
  -rw-r--r-- 1 root root 3062 May 11 11:40 /etc/slurm/slurm.conf.example

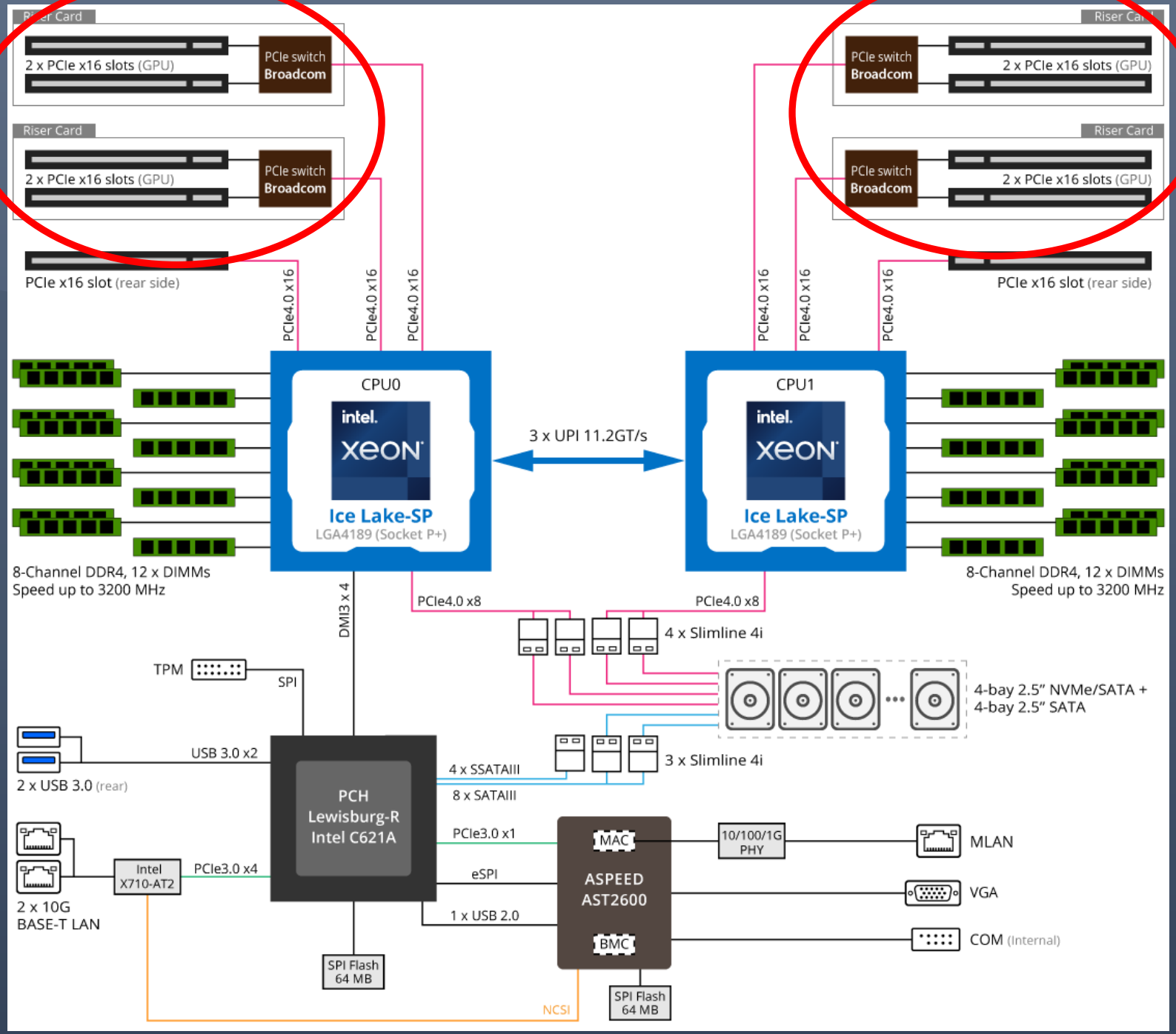
- If possible, reboot after changes

Thank you!

- Support: support@atipa.com
- Sales Order: 6002900

GPU Topology

nvidia-smi topo -m



GPU Topology

nvidia-smi topo -m

```
[root@gpu001 ~]# nvidia-smi topo -m
        GPU0    GPU1    GPU2    GPU3    GPU4    GPU5    GPU6    GPU7    mlx5_0  CPU Affinity    NUMA Affinity
GPU0     X      PIX     SYS     SYS     SYS     SYS     SYS     SYS     SYS     0-6     0
GPU1    PIX      X      SYS     SYS     SYS     SYS     SYS     SYS     SYS     0-6     0
GPU2    SYS     SYS      X      PIX     SYS     SYS     SYS     SYS     NODE    7-13    1
GPU3    SYS     SYS     PIX      X      SYS     SYS     SYS     SYS     NODE    7-13    1
GPU4    SYS     SYS     SYS     SYS      X      PIX     SYS     SYS     SYS     14-20   2
GPU5    SYS     SYS     SYS     SYS     PIX      X      SYS     SYS     SYS     14-20   2
GPU6    SYS     SYS     SYS     SYS     SYS     SYS      X      PIX     SYS     21-27   3
GPU7    SYS     SYS     SYS     SYS     SYS     SYS     PIX      X      SYS     21-27   3
mlx5_0  SYS     SYS     NODE    NODE    SYS     SYS     SYS     SYS      X

Legend:

  X    = Self
  SYS  = Connection traversing PCIe as well as the SMP interconnect between NUMA nodes (e.g., QPI/UPI)
  NODE = Connection traversing PCIe as well as the interconnect between PCIe Host Bridges within a NUMA node
  PHB  = Connection traversing PCIe as well as a PCIe Host Bridge (typically the CPU)
  PXB  = Connection traversing multiple PCIe bridges (without traversing the PCIe Host Bridge)
  PIX  = Connection traversing at most a single PCIe bridge
  NV#  = Connection traversing a bonded set of # NVLinks
```

Without NVLink GPUs communicate via PCIe (32GB/s)

```
[root@gpu008 ~]# nvidia-smi topo -m
        GPU0    GPU1    GPU2    GPU3    GPU4    GPU5    GPU6    GPU7    mlx5_0  CPU Affinity    NUMA Affinity
GPU0     X      NV12    SYS     SYS     SYS     SYS     SYS     SYS     SYS     0-6     0
GPU1    NV12     X      SYS     SYS     SYS     SYS     SYS     SYS     SYS     0-6     0
GPU2    SYS     SYS      X      NV12    SYS     SYS     SYS     SYS     NODE    7-13    1
GPU3    SYS     SYS     NV12     X      SYS     SYS     SYS     SYS     NODE    7-13    1
GPU4    SYS     SYS     SYS     SYS      X      NV12    SYS     SYS     SYS     14-20   2
GPU5    SYS     SYS     SYS     SYS     NV12     X      SYS     SYS     SYS     14-20   2
GPU6    SYS     SYS     SYS     SYS     SYS     SYS      X      NV12    SYS     21-27   3
GPU7    SYS     SYS     SYS     SYS     SYS     SYS     NV12     X      SYS     21-27   3
mlx5_0  SYS     SYS     NODE    NODE    SYS     SYS     SYS     SYS      X

Legend:

  X    = Self
  SYS  = Connection traversing PCIe as well as the SMP interconnect between NUMA nodes (e.g., QPI/UPI)
  NODE = Connection traversing PCIe as well as the interconnect between PCIe Host Bridges within a NUMA node
  PHB  = Connection traversing PCIe as well as a PCIe Host Bridge (typically the CPU)
  PXB  = Connection traversing multiple PCIe bridges (without traversing the PCIe Host Bridge)
  PIX  = Connection traversing at most a single PCIe bridge
  NV#  = Connection traversing a bonded set of # NVLinks
```

With NVLink bridges GPUs in a pair communicate “directly” (600GB/s)

```
[root@sxm001 ~]# nvidia-smi topo -m
        GPU0    GPU1    GPU2    GPU3    GPU4    GPU5    GPU6    GPU7    mlx5_0  CPU Affinity    NUMA Affinity
GPU0     X      NV12    NV12    NV12    NV12    NV12    NV12    NV12    NODE    0-6     0
GPU1    NV12     X      NV12    NV12    NV12    NV12    NV12    NV12    NODE    0-6     0
GPU2    NV12    NV12     X      NV12    NV12    NV12    NV12    NV12    PXB     0-6     0
GPU3    NV12    NV12    NV12     X      NV12    NV12    NV12    NV12    PXB     0-6     0
GPU4    NV12    NV12    NV12    NV12     X      NV12    NV12    NV12    SYS     14-20   2
GPU5    NV12    NV12    NV12    NV12    NV12     X      NV12    NV12    SYS     14-20   2
GPU6    NV12    NV12    NV12    NV12    NV12    NV12     X      NV12    SYS     21-27   3
GPU7    NV12    NV12    NV12    NV12    NV12    NV12    NV12     X      SYS     21-27   3
mlx5_0  NODE    NODE    PXB     PXB     SYS     SYS     SYS     SYS      X

Legend:

  X    = Self
  SYS  = Connection traversing PCIe as well as the SMP interconnect between NUMA nodes (e.g., QPI/UPI)
  NODE = Connection traversing PCIe as well as the interconnect between PCIe Host Bridges within a NUMA node
  PHB  = Connection traversing PCIe as well as a PCIe Host Bridge (typically the CPU)
  PXB  = Connection traversing multiple PCIe bridges (without traversing the PCIe Host Bridge)
  PIX  = Connection traversing at most a single PCIe bridge
  NV#  = Connection traversing a bonded set of # NVLinks
```

With NVSwitch all GPUs communicate “directly“ (600GB/s)