OAK RIDGE National Laboratory

ORNL Site Report

Slurm User Group 2019

Salt Lake City, Utah

ORNL is managed by UT-Battelle LLC for the US Department of Energy

U.S. DEPARTMENT OF ENERGY

Topics

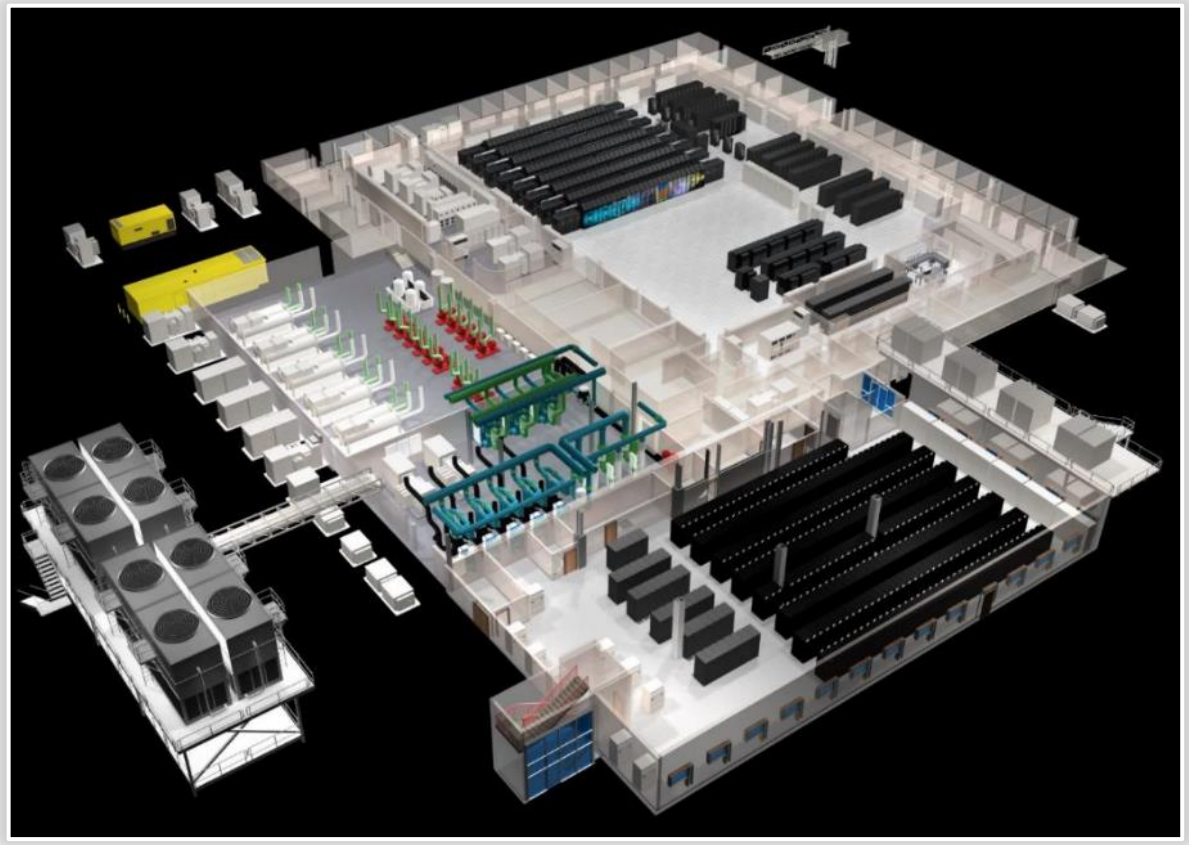
- Overview of ORNL/NCCS/OLCF
- Rhea and DTN Slurm migration
- NOAA/NCRC Slurm Environment
- Air Force Weather
- Slurm on Frontier
- Discussion topics about Slurm

Our vision: Sustain ORNL's leadership and scientific impact in computing and computational sciences

- Provide the world's most powerful open resources for:
  - Scalable computing and simulation
  - Data and analytics at any scale
  - Scalable cyber-secure infrastructure for science
- Follow a well-defined path for maintaining world leadership in these critical areas
- Deliver leading-edge science relevant to missions of DOE and key federal and state agencies
- Build and exploit cross-cutting partnerships
- Attract the brightest talent
- Invest in education and training

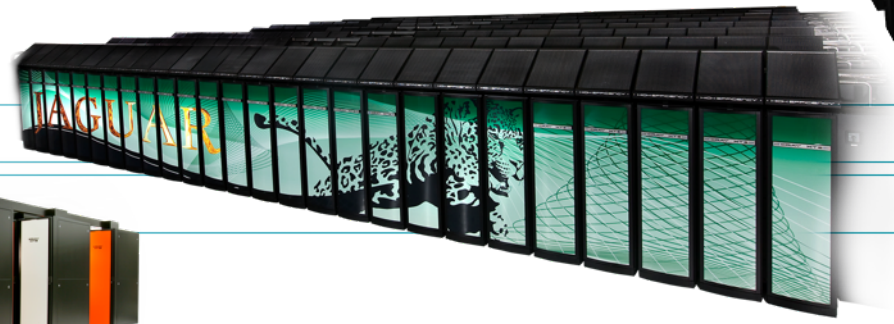
National Center for Computational Sciences
Home to the OLCF, including Summit

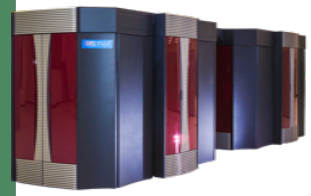
- 65,000 ft^2 of DC space
  - Distributed across 4 controlled-access areas
  - 31,000 ft^2: Very high load bearing (≥625 lb/ft^2)
- 40 MW of high-efficiency highly reliable power
  - Tightly integrated into TVA's 161 kV transmission system
  - Diverse medium-voltage distribution system
  - High-efficiency 3.0/4.0 MVA transformers
- 6,600 tons of chilled water
  - 20 MW cooling capacity at 70 °F
  - 7,700 tons evaporative cooling
- Expansion potential: Power infrastructure to 60 MW, cooling infrastructure to 70 MW



ORNL has systematically delivered a series of leadership-class systems

On scope • On budget • Within schedule





OLCF-1

OLCF-2

OLCF-3

18.5 TF
2004
Cray X1E
Phoenix

25 TF
2005
Cray XT3
Jaguar

54 TF
2006
Cray XT3
Jaguar

62 TF
2007
Cray XT4
Jaguar

263 TF
2008
Cray XT4
Jaguar

1 PF
2008
Cray XT5
Jaguar

2.5 PF
2009
Cray XT5
Jaguar

1000-fold improvement in 8 years

27 PF
2012
Cray XK7
Titan

Titan turned 6 years old in October 2018 and continued to deliver world-class science research in support of our user community until it was decommissioned on August 2nd 2019

We are building on this record of success to enable exascale in 2021

27 PF

2012
Cray XK7
Titan

OLCF-4

200 PF

2018
IBM
Summit

OLCF-5

1.5 EF

2021
Frontier

Adaptive COMPUTING

IBM Spectrum LSF

slurm workload manager

Summit system overview

System performance

- Peak performance of 200 petaflops for M&S; 3.3 exaops (FP16) for data analytics and AI
- Launched in June 2018; ranked #1 on TOP500 list

System elements

- 4608 nodes
- Dual-rail Mellanox EDR InfiniBand network
- 250 PB IBM Spectrum Scale file system transferring data at 2.5 TB/s

Node elements

- 2 IBM POWER9 processors
- 6 NVIDIA Tesla V100 GPUs
- 608 GB of fast memory
- (96 GB HBM2 + 512 GB DDR4)
- 1.6 TB of non-volatile memory

Summit is the fastest supercomputer in the world and has been #1 on the TOP500 list since its launch in June 2018

Rhea and DTN system overview

CPU Nodes

- 512 Compute nodes
- Dual 8-Core Xeon (Sandy Bridge) Processors
- 128 GB Ram

GPU and Large Mem Nodes

- 9 GPU/LargeMem nodes
- Dual 14-Core Xeon Processors
- 1 TB Ram
- 2 NVIDIA K80 GPUs

Cluster elements

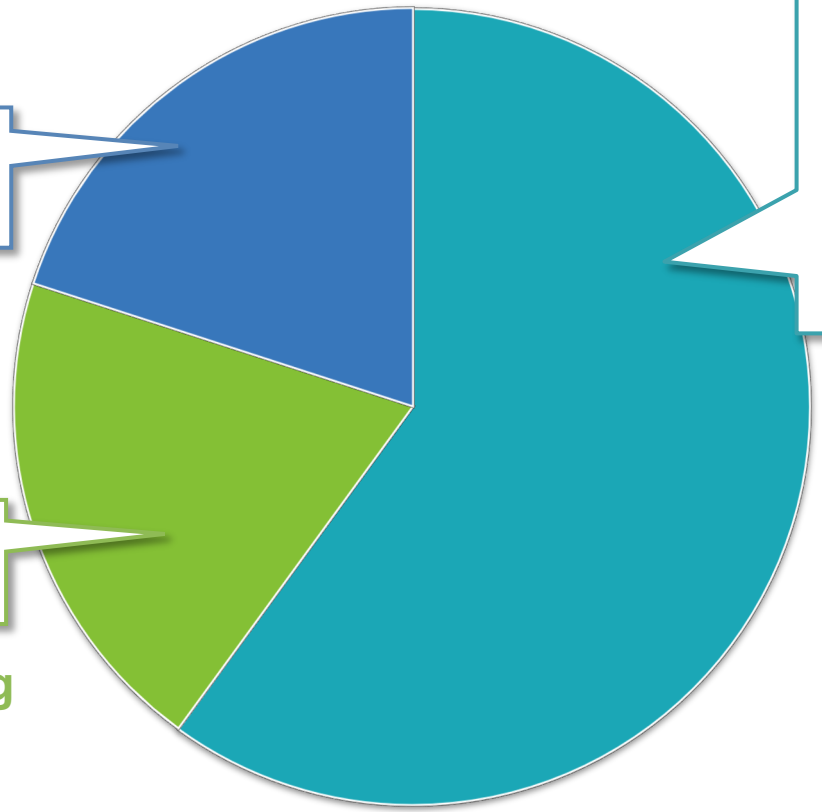
- Mellanox FDR 4x Infiniband (56 Gb/sec/direction)
- Mounts Alpine center-wide GPFS file system
- Slurm resource manager

Rhea is used for analysis, visualization, and postprocessing

Data Transfer Cluster (DTNs) provide interactive and scheduled data transfer services

Primary allocation programs for access to LCF in 2020

Current distribution of allocable hours



20% Director's Discretionary
(Includes LCF strategic programs, ECP)

20% ASCR Leadership Computing Challenge

DOE/SC capability computing

Up to 60% INCITE

Leadership-class computing

- As a DOE Leadership Computing Facility, the OLCF has a mandate that a large portion of Summit's usage come from large, *leadership-class* (aka *capability*, 20+%) jobs

Bin	Min Nodes	Max Nodes	Max Walltime (Hours)	Aging Boost (Days)
1	2,765	4,608	24.0	15
2	922	2,764	24.0	10
3	92	921	12.0	0
4	46	91	6.0	0
5	1	45	2.0	0

Overallocation Policy

- Projects that overrun their allocation are still allowed to run on OLCF systems, although at a reduced priority.

% Of Allocation Used	Priority Reduction
< 100%	0 days
100% to 125%	30 days
> 125%	365 days

No Refunds!

OAK RIDGE National Laboratory

How do we map our policies to priorities in Slurm?

Priority Factors in 18.08

PriorityWeightAge

PriorityWeightFairshare

PriorityWeightJobSize

PriorityWeightPartition

PriorityWeightQOS

PriorityWeightTRES

Priority Items Needed

Time in Queue

Job size by bin

Allocation Type

Overallocation (True/False)

New Priority Options in 19.05

- **PriorityWeightAssoc**
  - slurm.conf options that sets the degree to which the association component contributes to the job's priority
  - Each association can have a priority set in the SlurmDBD
- **Site Factor**
  - Can be set by the admin inside a submit filter or using scontrol
  - Can create a custom plugin that updates the value every PriorityCalcPeriod
  - Not weighted, must provide “raw” values
  - Examples: Priority based on job size *by bin*, XFACTOR

Scheduling Policies

- Normalize all priority factors to “seconds in queue”

```
PriorityType=priority/multifactor
PriorityDecayHalfLife=14-0
PriorityUsageResetPeriod=NONE
PriorityMaxAge=365-0                # 365 Days is max
PriorityWeightAge=31536000          # 365 Days in seconds
PriorityWeightAssoc=86400           # 1 Day in seconds
PriorityWeightQOS=86400             # 1 Day in seconds
PriorityWeightJobSize=0             # What about bins???
PriorityWeightFairshare=0
PriorityWeightPartition=0
PriorityFlags=NO_NORMAL_ASSOC
```

Bin Settings – From job_submit.data

```
bins = {
        bina = {minsize=1, maxsize=16, maxwall=48, priority=0},
        binb = {minsize=17, maxsize=64, maxwall=36, priority=86400},
        binc = {minsize=65, maxsize=512, maxwall=3, priority=172800},
}
```

From job_submit.lua

```
dofile('/etc/slurm/job_submit.data’)
…
function find_bin(job_desc)
    nnodes = job_desc.min_nodes
    for binname, binconf in pairs(bins) do
        if nnodes >= binconf['minsize'] and nnodes <= binconf['maxsize'] then
            return binconf
        end
    end
    return nil
end
…
local mybin = find_bin(job_desc)
job_desc.site_factor = mybin['priority']
```

Note: We actually do caching and error checking

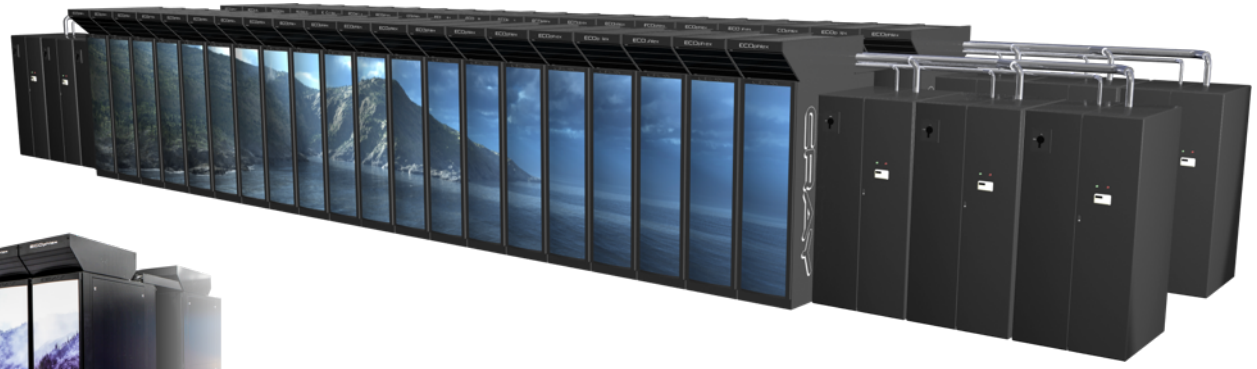
Priority

- Association priority is calculated externally
  - Allocation Type
  - Over-Allocation Penalty
  - “Special” boosts
- Priority cannot be negative, so we use 500 days as our “base”
- Per-job boost *can* be given with a negative nice value

National Climate-Computing Research Center (NCRC)

- Agreement between NOAA and DOE's Oak Ridge National Laboratory for HPC services and climate modeling support
- Strategic Partnership Project, currently in year 9
- 5-year periods. Current IAA effective through FY20
- Within ORNL's National Center for Computational Sciences (NCCS)
- Service provided - DOE-titled equipment
- Secure network enclave; Department of Commerce access policies

Gaea system evolution, 2010–2019



- 2010 C1MS, Cray XT6 — 260 TF — 4.2 PB storage capability
- 2011 C1MS + C2 Cray XE6 — 980 TF
- 2012 C2 + C1, Cray XE6 — 1.1 PF
- 1.1 PF
- 1.1 PF — 6.4 PB storage capability
- 1.1 PF
- 2016 C3, Cray XC40 — 1.77 PF
- 2017 C3 + C4 Cray XC40 — 4.02 PF
- 2017 C3 + C4 Cray XC40 — 4.27 PF
- 2018 C3 + C4 Cray XC40 — 4.78 PF — 38 PB storage capability
- 2019 C3 + C4 Cray XC40 — 5.29 PF

ORNL has successfully lead the effort to procure, configure, and transition to operation sustained petaflops scale computing including infrastructure for storage, and networking in support of the NOAA mission.

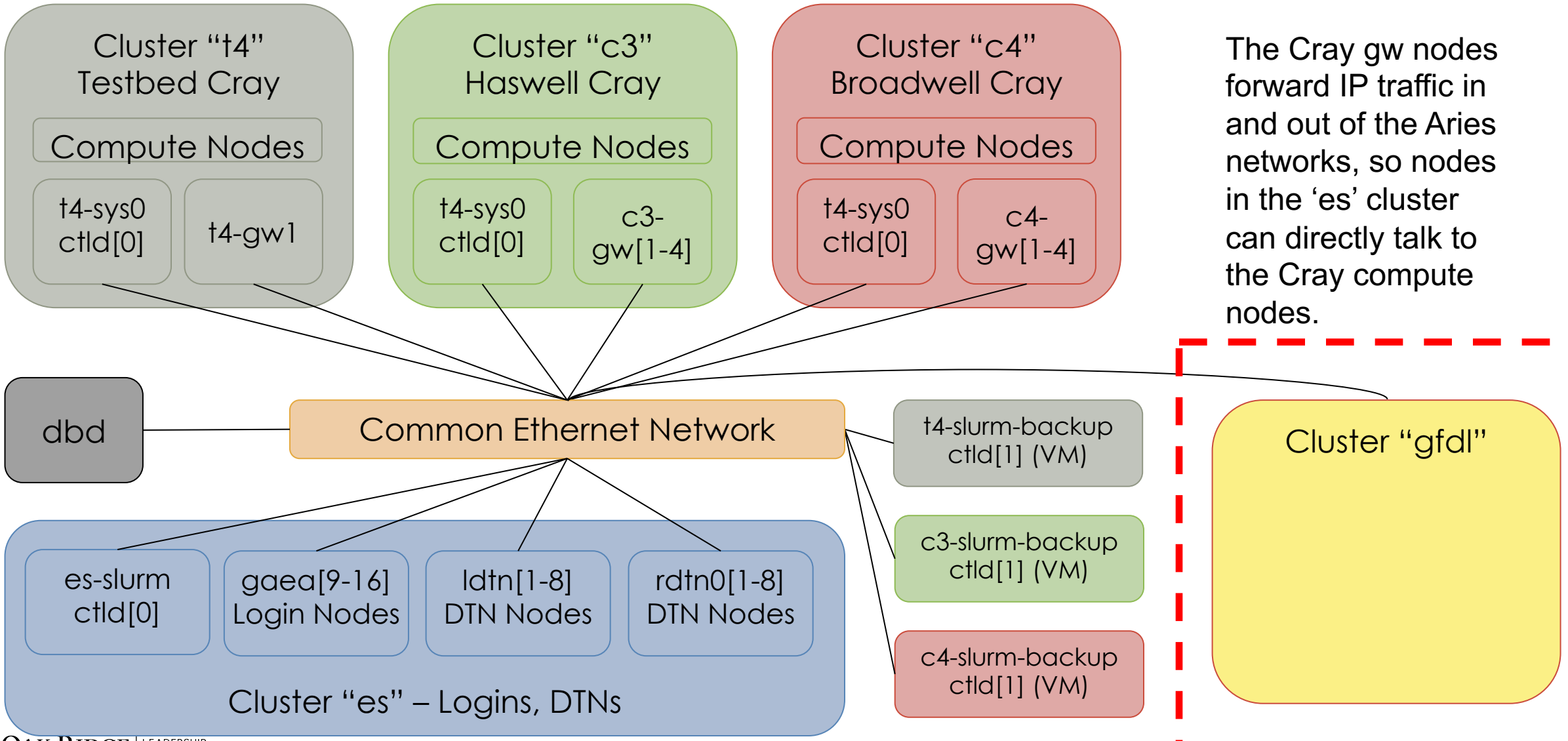
NOAA Workload under Moab

- Single Moab server setup in grid mode
  - One TORQUE server per cluster
- Users submit jobs using *msub*

  -l nodes=num_nodes to request whole nodes

  -l partition=cluster_name to request which cluster to run on
- Jobs migrate to TORQUE "just in time"
- Users check and modify status of jobs using **Moab** commands
  - *showq* to see which jobs are running and what job will run next
  - *canceljob* to delete a job

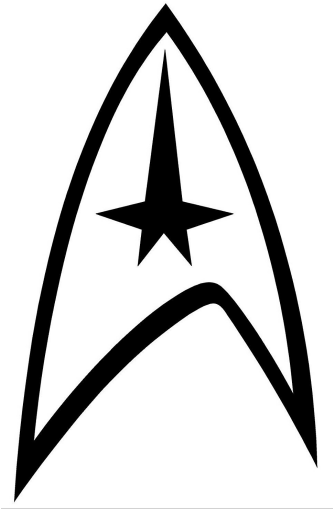
Gaea Slurm Hardware Layout



NOAA Workload under Slurm

- Single SlurmDBD server at ORNL
- One Slurm controller per cluster (multi-cluster)
- Users submit jobs using *sbatch*
  --nodes=num_nodes to request whole nodes
  --cluster=cluster_name to request which cluster to run on
- Jobs live only one one cluster at a time
- Users check and modify status of jobs using Slurm commands
  - *squeue* to see which jobs are running and what job will run next
  - *scontrol* to delete a job
- By default you only see jobs on the local cluster

Federation



- Federation provides
  - Unified Job IDs
  - Unified *squeue* and *sinfo* views
- Still require users to submit to a single cluster
  - Due to different core counts, not limitations of federation
  - Investigating relaxing this requirement

Multi-Cluster without sharing a Slurm Database

- Users at NCRC need to be able to submit post-processing jobs at GFDL
- But multi-cluster requires a shared Slurm Database
- Does it make sense to share a Slurm database between multiple sites?
- Proposed solution: external clusters
- Hack workaround: manually edit the database and carry a local patch to disable deregistering controllers

Changes to NOAA Systems

- Batch nodes are no longer used for running scripts
  - Now DVS+Network nodes
- Home areas are DVS-mounted on compute nodes
- Compute nodes have a route to the rest of the center
- Compute nodes talk to LDAP and resolve users
- Cray Node Health Checker is disabled

Installation and Configuration

- Deploy Slurm as RPMs
- Config file lives in /etc/slurm/slurm.conf
  - Should match everywhere inside a cluster
  - Lives in Puppet
- Puppet (currently) manages *slurm.conf* on the SMW and inside the Cray *configset*
  - Ansible will push *slurm.conf* changes to the compute nodes
- Many changes can be pushed out with `scontrol reconfig`
  - Otherwise use `systemctl restart slurmctld` or `systemctl restart slurmd`

Topology

- Each Dragonfly group is one “switch”
- Jobs that fit inside a switch will attempt to do so
- (With `TopologyParams=dragonfly`) Jobs that don’t fit will try to use as many switches as possible to maximize global bandwidth

```
t4-sys0:~ # cat /etc/slurm/topology.conf
# managed by puppet: modules/slurm/files/etc/slurm/topology.conf/common.erb
SwitchName=s0 Nodes=nid000[08-11]
SwitchName=s1 Nodes=nid000[12-15]
SwitchName=s2 Nodes=nid000[16-19]
SwitchName=s3 Nodes=nid000[20-23]
SwitchName=s4 Nodes=nid000[24-27]
SwitchName=root Switches=s[0-4]
```

PRESS RELEASE

<< Back

View printer-friendly version

CRAY SHASTA SUPERCOMPUTER TO POWER WEATHER FORECASTING FOR THE U.S. AIR FORCE

Strategic Partnership with Oak Ridge National Lab Highlights First Cray Shasta Supercomputer for Operational Weather Forecasting

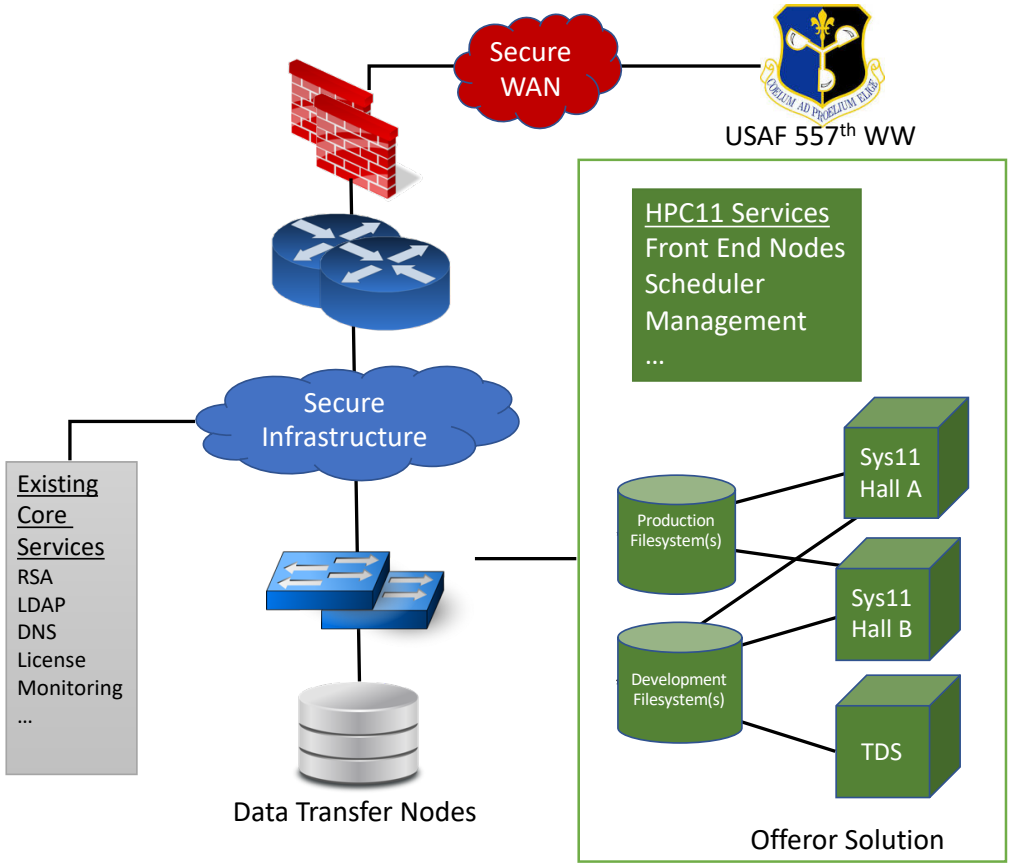
SEATTLE, Aug. 07, 2019 (GLOBE NEWSWIRE) -- Global supercomputer leader Cray Inc. (Nasdaq:CRAY) today announced that the first Cray Shasta™ supercomputing system for operational weather forecasting and meteorology will be acquired by the Air Force Life Cycle Management Center in partnership with Oak Ridge National Laboratory. The powerful high-performance computing capabilities of the new system, named HPC11, will enable higher fidelity weather forecasts for U.S. Air Force and Army operations worldwide. The contract is valued at $25 million.

"We're excited with our Oak Ridge National Laboratory strategic partner's selection of Cray to provide Air Force Weather's next high performance computing system," said Steven Wert, Program Executive Officer Digital, Air Force Life Cycle Management Center at Hanscom Air Force Base in Massachusetts, and a member of the Senior Executive Service. "The system's performance will be a significant increase over the existing HPC capability and will provide Air Force Weather operators with the ability to run the next generation of high-resolution, global and regional models, and satisfy existing and emerging warfighter needs for environmental impacts to operations planning."

Oak Ridge National Laboratory (ORNL) has a history of deploying the world's most powerful supercomputers and through this partnership, will provide supercomputing-as-a-service on the HPC11 Shasta system to the Air Force 557th Weather Wing. The 557th Weather Wing develops and provides comprehensive terrestrial and space weather information to the U.S. Air Force and Army. The new system will feature the revolutionary Cray Slingshot™ interconnect, with features to better support time-critical numerical weather prediction workloads, and will enhance the Air Force's capabilities to [illegible] forecasts and weather threat assessments so that Air Force missions can be

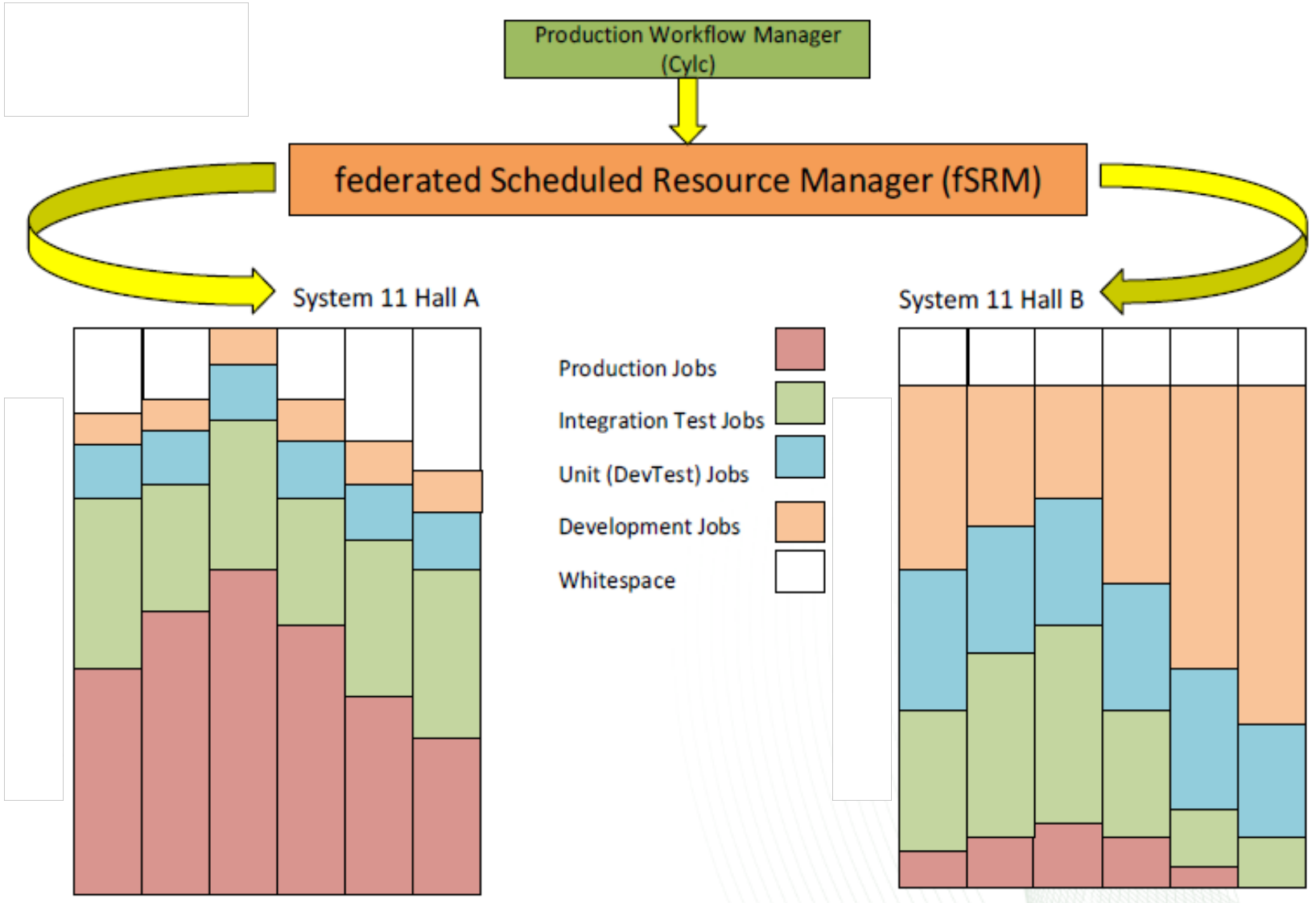
US Air Force: ORNL is supporting the 557th Weather Wing

Highly reliable computing resources that deliver timely and frequent weather products, supporting Air Force mission requirements worldwide

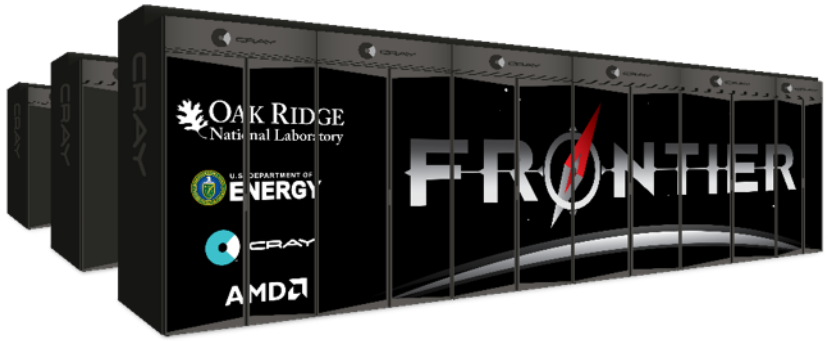


US Air Force Weather Workflow

- Federation protects from single hall failure
- Allow "bursting" across both halls



Frontier Overview



Partnership between ORNL, Cray, and AMD

The Frontier system will be delivered in 2021

Peak Performance greater than 1.5 EF

Composed of more than 100 Cray Shasta cabinets

- Connected by Slingshot™ interconnect with adaptive routing, congestion control, and quality of service

Node Architecture:

- An AMD EPYC™ processor and four Radeon Instinct™ GPU accelerators purpose-built for exascale computing
- Fully connected with high speed AMD Infinity Fabric links
- Coherent memory across the node
- 100 GB/s injection bandwidth
- Near-node NVM storage

Researchers will harness Frontier to advance science in such applications as systems biology, materials science, energy production, additive manufacturing and health data science.

CORAL Requirements for Slurm

- Provide a REST API to accounting data captured within slurmdbd
- Change --exclusive option to salloc/sbatch/srun to assign all GRES in a node to the job
- Add gpu/amd plugin, and work with AMD to develop suitable APIs for GPU control
- Expose additional per-job scheduling details
- Local Storage Management – new burst_buffer plugin
- Add new --ntasks-per-gpu option

CORAL Requirements for Slurm

- Step-level GPU binding/affinity
- Heterogeneous job step launch
- Reservation affinity
- Add acct_gather_interconnect/slingshot plugin
- Retroactive WCKey updates
- AcctGatherEnergy Plugin for Shasta

OAK RIDGE National Laboratory

Discussion

?

ezellma@ornl.gov

Difficulty implementing some limit policies

- "Everyone should only be allowed to run 1 concurrent job in our GPU partition. Except Bob, he can run 5."
  - Partition QOS limit
  - Job QOS limit
  - User association
  - Account association(s), ascending the hierarchy
  - Root/Cluster association
  - Partition limit
- The "Group" Limits are hard to understand

The term “accounting” might be overused

- AccountingStorage means Slurm Database
  - This stores associations, limits, and usage
- AccountingGather plugins collect metrics from the nodes
  - Can be used for Node, Job, Step, and “job profiling”
  - Maybe it would be better to start calling these “metrics”
- JobComp can store accounting data, but might be redundant

What metrics make sense to collect and store?

- Count over the whole job versus time-series data
- Most items we care about have more than "in" and "out"
  - Filesystems have read/write bytes, read/write count, open/close count, stat count, unlink count, etc.
  - Interconnects have in/out bytes, in/out packet count, errors, retransmits, etc.
- Counters versus gauges must be treated differently
- Can you distinguish usage between jobs/steps running concurrently on the same node?

Associations with External Management Systems

- ORNL and NOAA have an "external" canonical source of account information
- How many sites have written script to synchronize an external database to the SlurmDB?
- Anyone use *sacctmgr load*?
- Is there a better way?

Reporting tools

- Are people using *sreport* for all reporting, or you dumping *sacct* data to an external system?
- Does it make sense to query regularly, or should we "stream" by integrating with a job completion plugin?

Collaborating with SchedMD

- Who has written code (feature, bugfix, etc) for Slurm?