OSiRIS

Distributed Ceph and Software Defined Networking for Multi-Institutional Research

Benjeman Meekhof
University of Michigan
Advanced Research Computing – Technology Services
October 6, 2016

- Project and participants overview
- Structural overview and site details
- Orchestration, monitoring and visualization
- Networking, NMAL, SDN
- Latency and Ceph - our experiments
- AAA infrastructure
- First science domains: ATLAS and Physical Ocean Modeling

OSiRIS Summary

We proposed to design and deploy MI-OSiRIS (Multi-Institutional Open Storage Research Infrastructure) as a pilot project to evaluate a software-defined storage infrastructure for our primary Michigan research universities.

Our goal is to provide transparent, high-performance access to the same storage infrastructure from well-connected locations on any of our campuses.

By providing a single data infrastructure that supports computational access "in-place" we can meet many of the data-intensive and collaboration challenges faced by our research communities and enable them to easily undertake research collaborations beyond the border of their own universities.

OSiRIS Team

OSiRIS is composed of scientists, computer engineers and technicians, network and storage researchers and information science professionals from **University of Michigan**, **Michigan State University**, **Wayne State University**, and **Indiana University** (focusing on SDN and net-topology)

We have a wide-range of science stakeholders who have data collaboration and data analysis challenges to address within, between and beyond our campuses:

High-energy physics, High-Resolution Ocean Modeling, Degenerative Diseases, Biostatics and Bioinformatics, Population Studies, Genomics, Statistical Genetics and Aquatic Bio-Geochemistry

Multi Institutional Data Challenges

Scientists working with large amounts of data face many obstacles in conducting their research

Typically the workflow needed to get data to where they can process it becomes a substantial burden

The problem intensifies when adding in collaboration across their institution or especially beyond their institution

Institutions have sometimes responded to this challenge by constructing specialized and expensive infrastructures to support specific science domain needs

OSiRIS Features

Scientists get customized, optimized data interfaces for their multi-institutional data needs

Network topology and perfSONAR-based monitoring components ensure the distributed system can optimize its use of the network for performance and resiliency

Ceph provides seamless rebalancing and expansion of the storage

A single, scalable infrastructure is much easier to build and maintain

Allows universities to reduce cost via economies-of–scale while better meeting the research needs of their campus

Eliminates isolated science data silos on campus:

- Data sharing, archiving, security and life-cycle management are feasible to implement and maintain with a single distributed service.
- Data infrastructure view for each research domain can be optimized for performance and resiliency.

Project Challenges

Deploying and managing a fault tolerant multi-site infrastructure

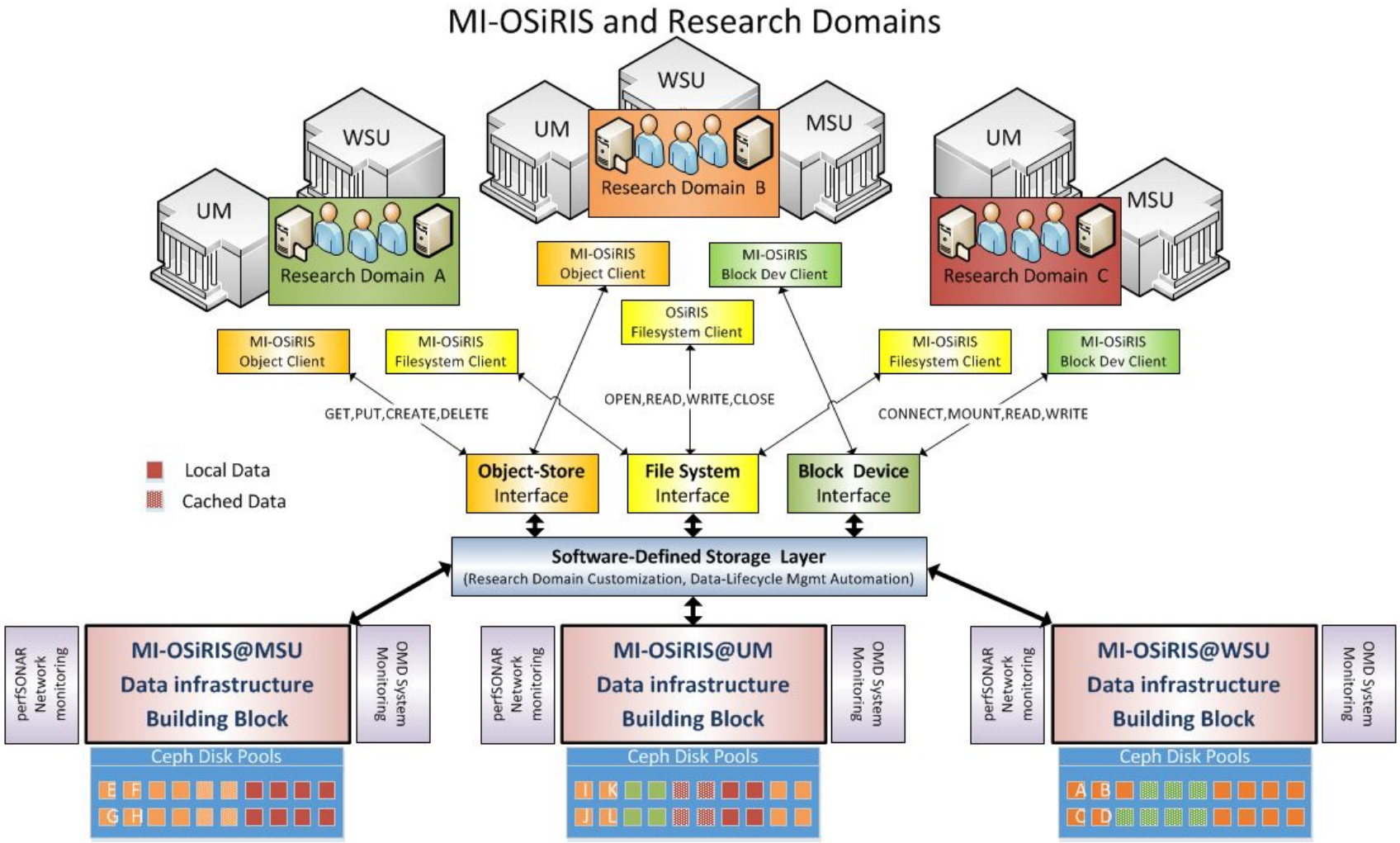
Resource management and optimization to maintain a sufficient quality of service for all stake-holders

Enabling the gathering and use of metadata to support data lifecycle management

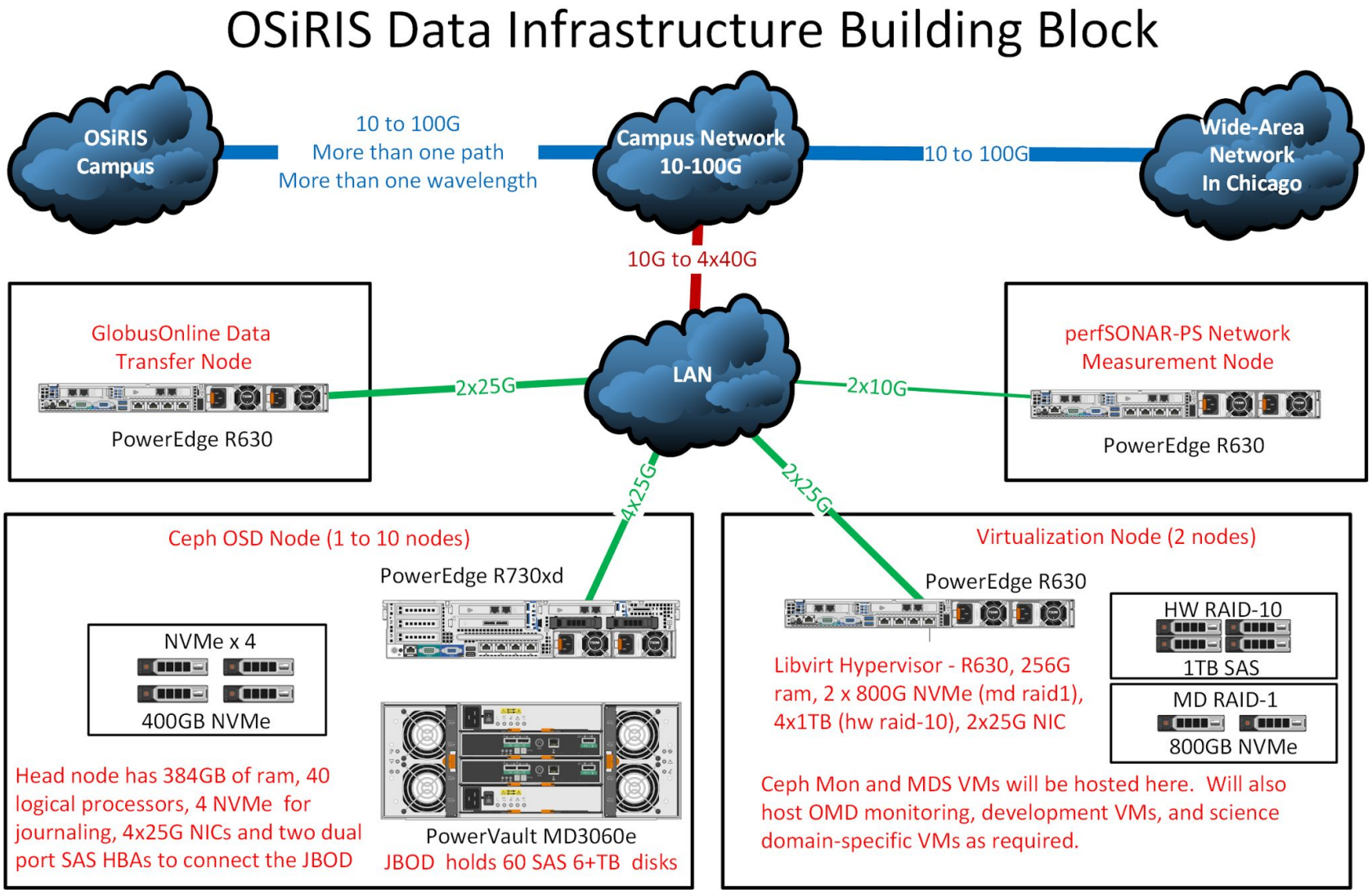
Research domain customization using CEPH API and/or additional services

Authorization which integrates with existing campus systems

Logical View



Site View


OSiRIS Data Infrastructure Building Block
OSiRIS Campus
10 to 100G
More than one path
More than one wavelength
Campus Network 10-100G
10 to 100G
Wide-Area Network In Chicago
10G to 4x40G
GlobusOnline Data Transfer Node
PowerEdge R630
2x25G
LAN
2x10G
perfSONAR-PS Network Measurement Node
PowerEdge R630
4x25G
2x25G
Ceph OSD Node (1 to 10 nodes)
PowerEdge R730xd
NVMe x 4
400GB NVMe
PowerVault MD3060e
JBOD holds 60 SAS 6+TB disks
Head node has 384GB of ram, 40 logical processors, 4 NVMe for journaling, 4x25G NICs and two dual port SAS HBAs to connect the JBOD
Virtualization Node (2 nodes)
PowerEdge R630
Libvirt Hypervisor - R630, 256G ram, 2 x 800G NVMe (md raid1), 4x1TB (hw raid-10), 2x25G NIC
HW RAID-10
1TB SAS
MD RAID-1
800GB NVMe
Ceph Mon and MDS VMs will be hosted here. Will also host OMD monitoring, development VMs, and science domain-specific VMs as required.

Ceph in OSiRIS

Ceph gives us a robust open source platform to host our multi-institutional science data

- Self-healing and self-managing
- Multiple data interfaces
- Rapid development supported by RedHat

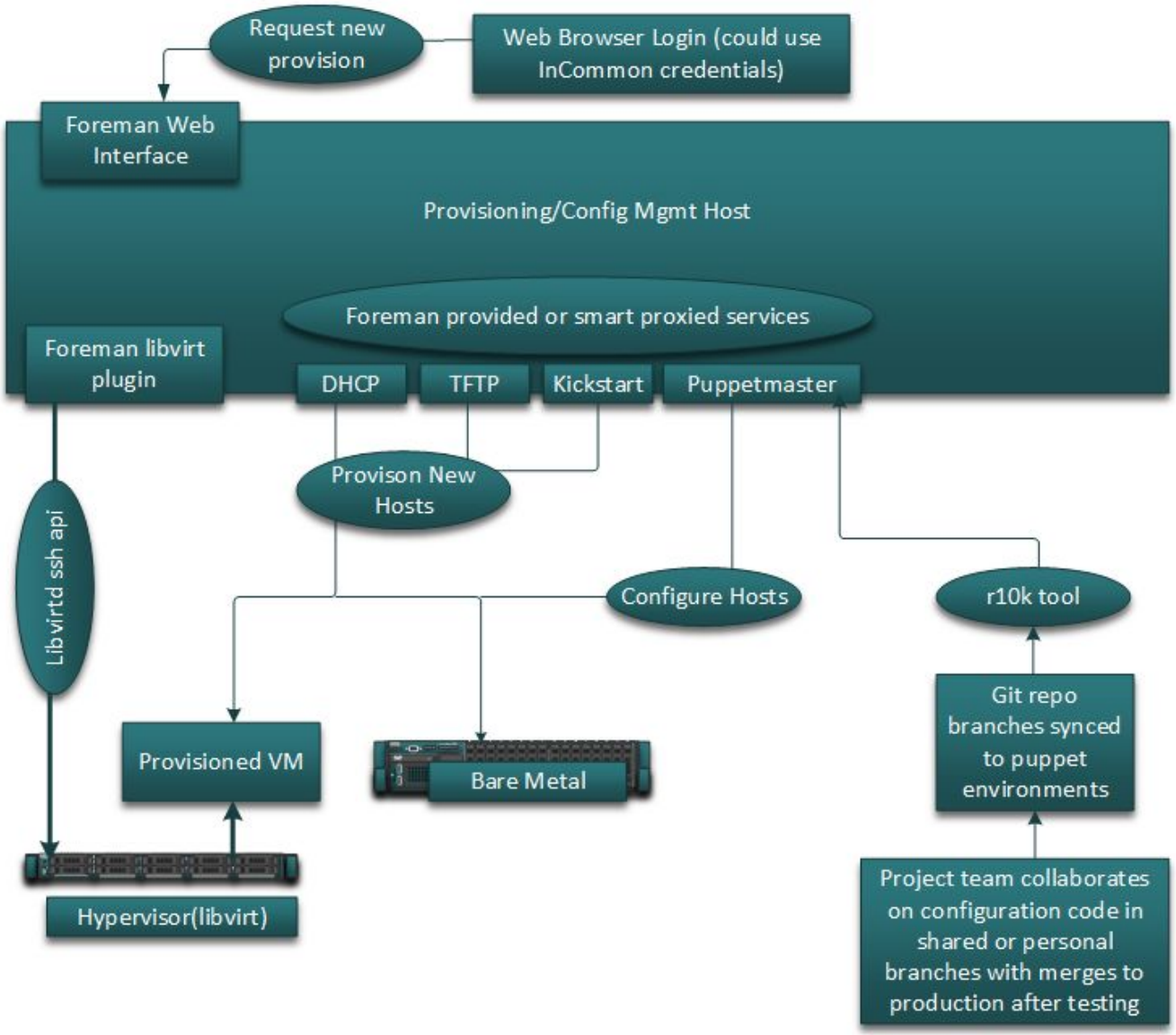
Able to tune components to best meet specific needs

Software defined storage gives us more options for data lifecycle management automation

Sophisticated allocation mapping (CRUSH) to isolate, customize, optimize by science use case

Ceph overview:
https://umich.app.box.com/s/f8ftr82smlbuf5x8r256hay7660soafk

Orchestration



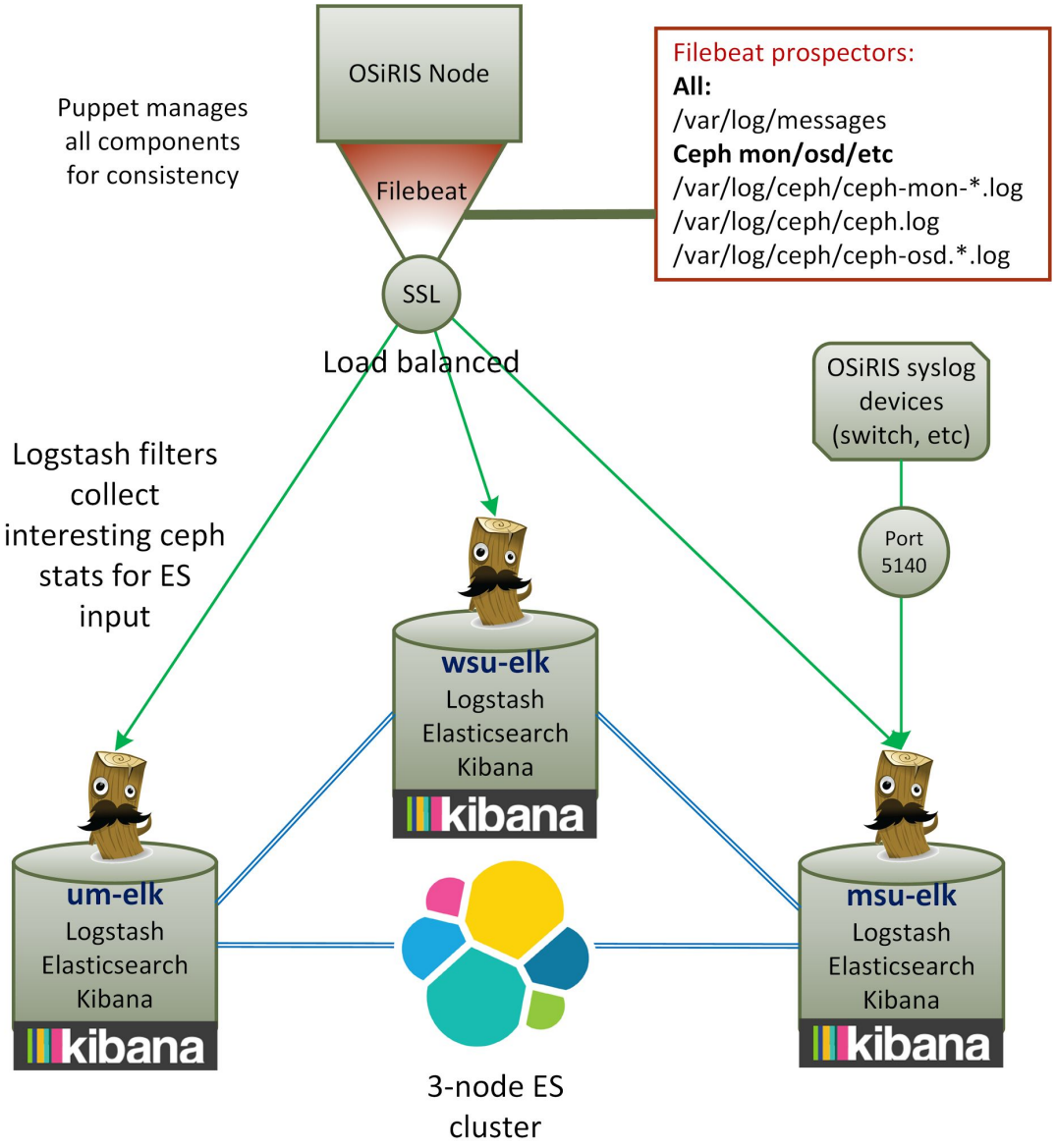
Deploying and extending our infrastructure relies heavily on orchestration with Puppet and Foreman

We can easily deploy bare-metal or VMs at any of the three sites and have services configured correctly from the first boot

Except: OSD activation requires a manual step

Openvswitch (scripted setup)

Monitoring with ELK

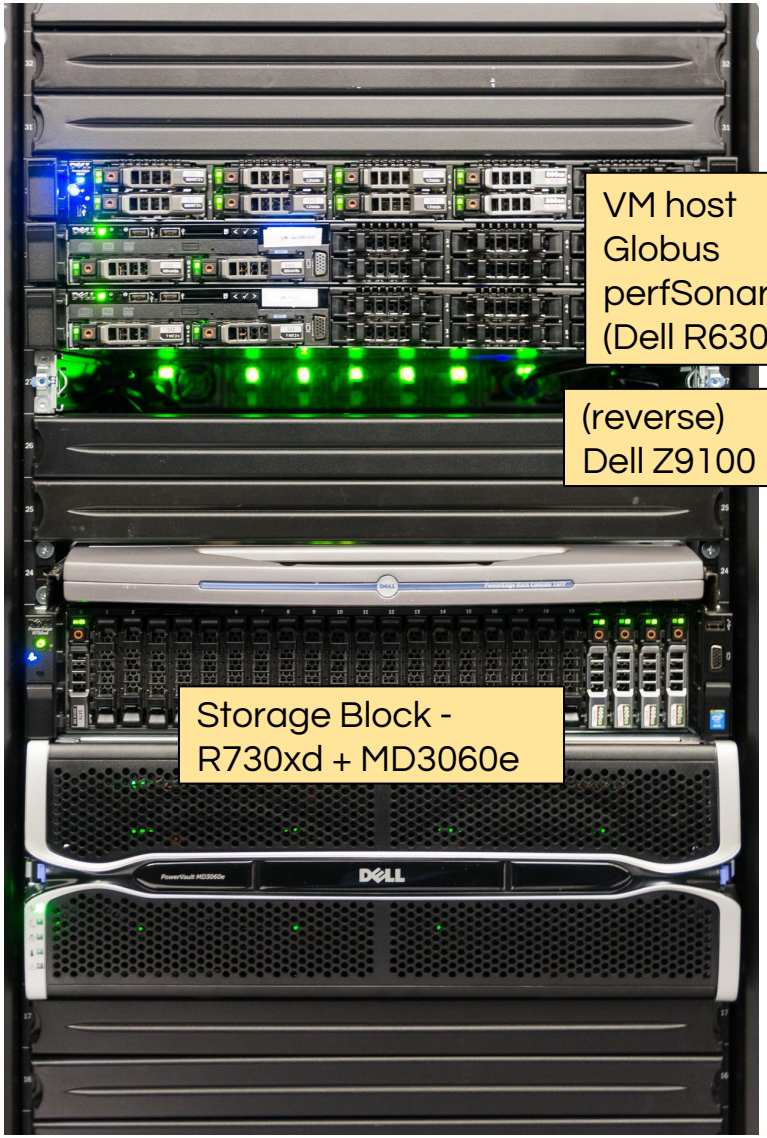


A resilient logging infrastructure is important to understand problems and long-term trends

The 3 node arrangement means we are not reliant on any one or even two sites being online to continue collecting logs

Ceph cluster logs give insights into cluster performance and health we can visualize with Kibana

Status



The OSiRIS project requested proposals to meet our hardware needs in October 2015 (9 bids)

November 2015 we decided on Dell servers, HGST 8TB drives, Mellanox ConnectX 4 NICs

Orders out in December 2015

Equipment arrived in January/February 2016

Sites are all fully operational

Currently engaging with ATLAS and Naval Oceanics group to begin placing data on OSiRIS

Have extensive tests, instrumentation, etc in place for production monitoring (covered on other slides)

Network Monitoring

Because networks underlie distributed cyberinfrastructure, monitoring their behavior is very important

The research and education networks have developed perfSONAR as a extensible infrastructure to measure and debug networks (http://www.perfsonar.net)

The CC*DNI DIBBs program recognized this and required the incorporation of perfSONAR as part of any proposal

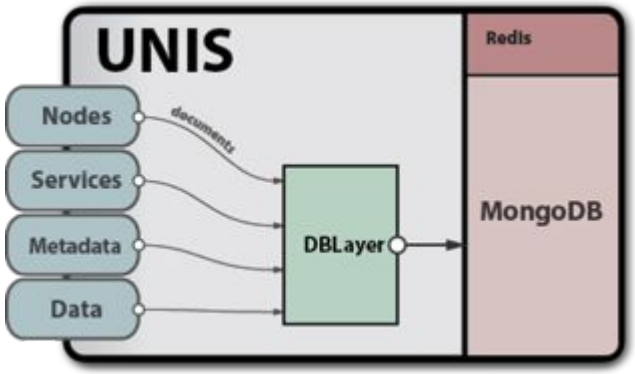
For OSiRIS, we were well positioned since one of our PIs Shawn McKee leads the worldwide perfSONAR deployment effort for the LHC community: https://twiki.cern.ch/twiki/bin/view/LCG/NetworkTransferMetrics

We intend to extend perfSONAR to enable the discovery of all network paths that exist between instances

SDN can then be used to optimize how those paths are used for OSiRIS

NMAL

The OSiRIS Network Management Abstraction Layer is a key part of managing our network as a dynamic resource



Captures site topology and routing information in UNIS from multiple sources: SNMP, LLDP, sflow, SDN controllers, and existing topology and looking glass services.

Package and deploy conflict-free measurement scheduler (HELM) along with measurement agents (Basic Lightweight Periscope Probe - BLiPP)

Correlate long-term performance measurements with passive metrics

Defining best-practices for SDN controller and reactive agent deployments within OSiRIS.

BLiPP/UNIS

The monitoring and topology discovery components being worked on by Indiana University/CREST are key parts of OSiRIS NMAL SDN

UNIS Topology and Measurement Store

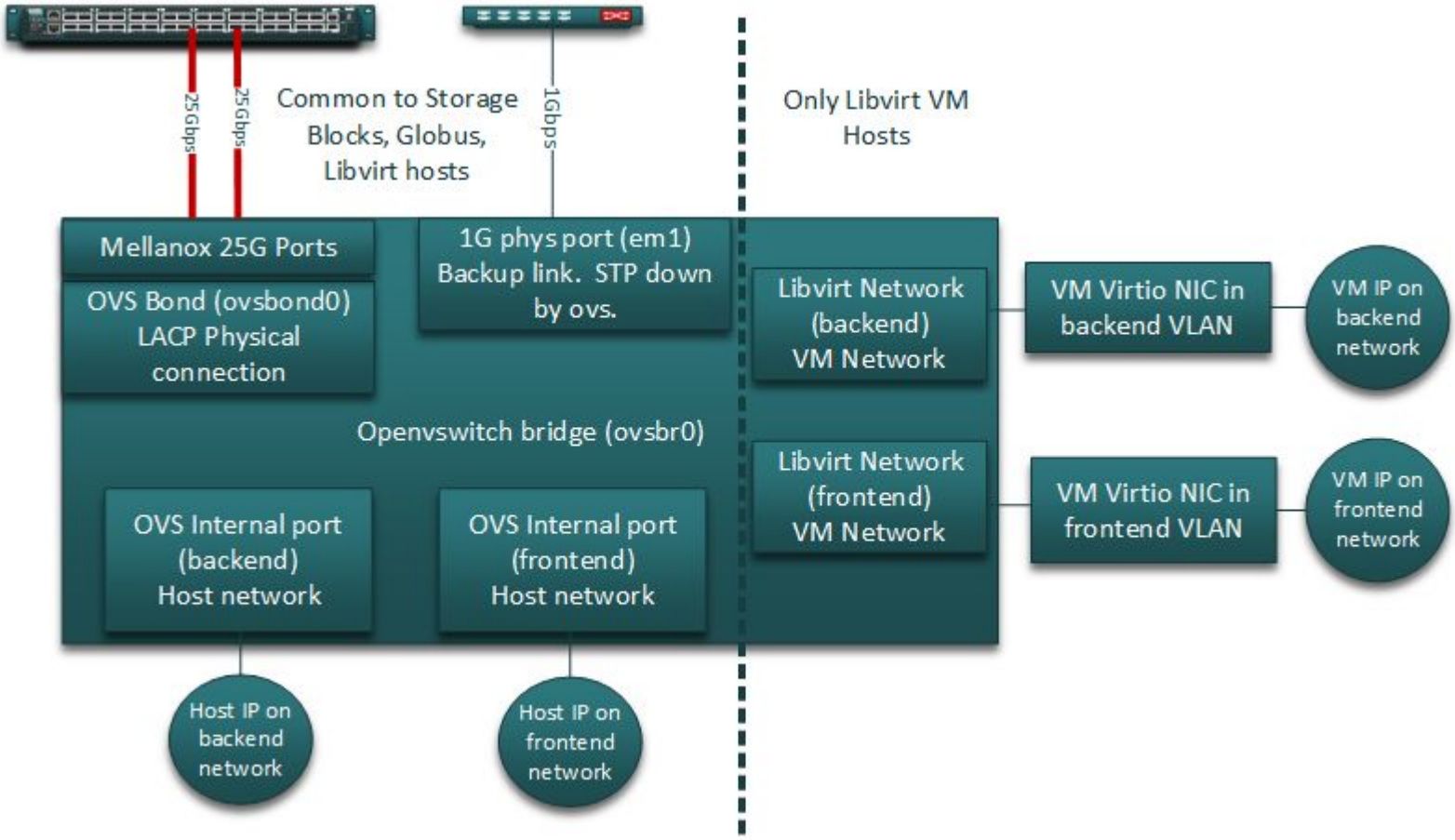
- Exposes a RESTful interface for information necessary to perform data logistics
  - Measurements from BLiPP
  - Network topology inferred through various agents
- Provides subscription endpoints for event-driven clients

Basic Lightweight Periscope Probe (BLiPP)

- Distributed probe agent system
- BLiPP agents execute measurement tasks received from UNIS and report back results for further analysis.
- BLiPP agents may reside in both the end hosts (monitoring end-to-end network status) and dedicated diagnose hosts inside networks

SDN - Open vSwitch

OSiRIS storage blocks, transfer gateways (S3, globus), and virtualization hosts incorporate Open vSwitch to allow fine-grained control dynamic network flows and integration with OpenFlow controllers



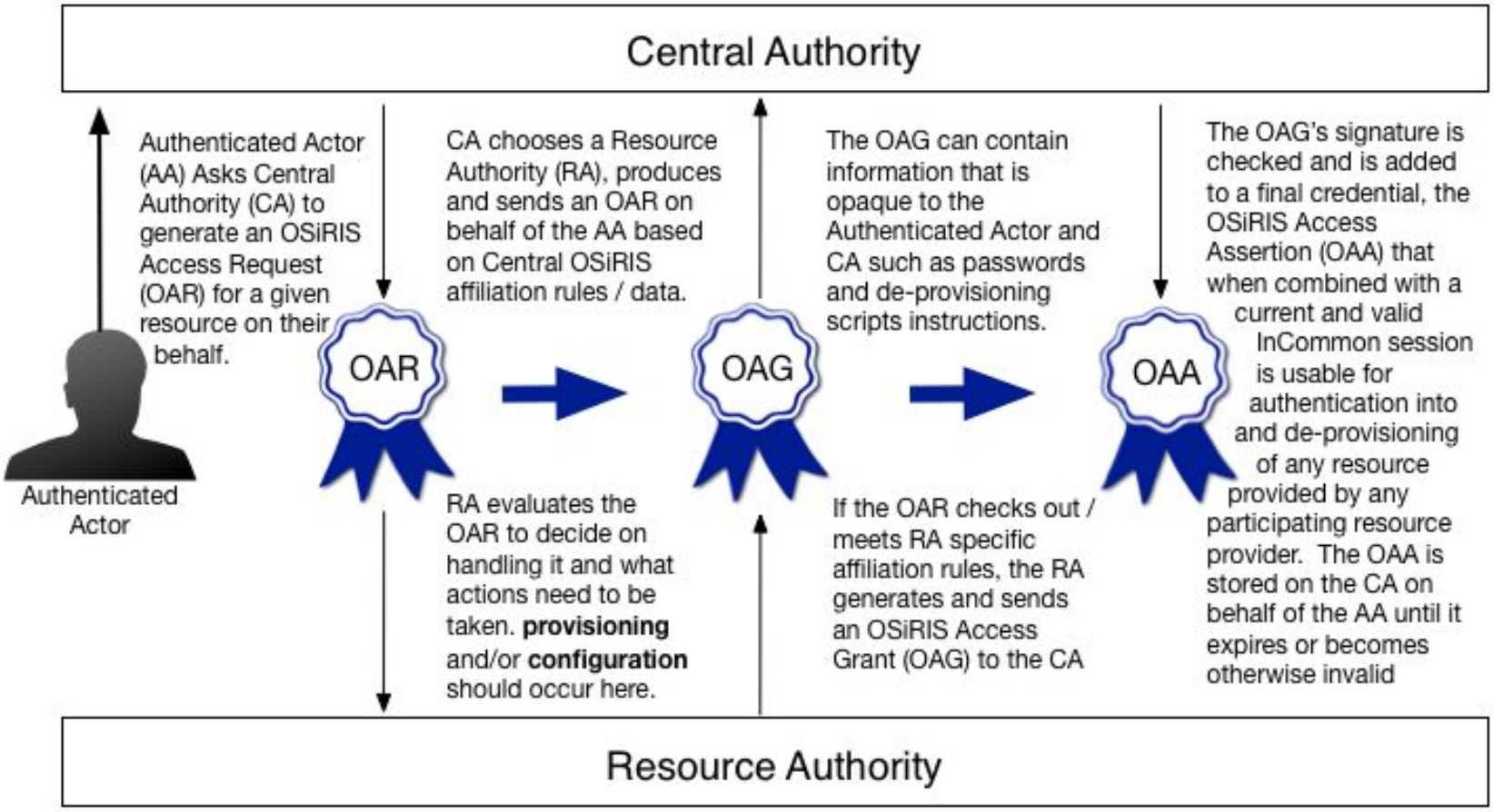
Authentication and Authorization

Session and affiliation data are first pulled into OSiRIS from SAML2 Assertions made by IdPs at configured or InCommon participant organizations

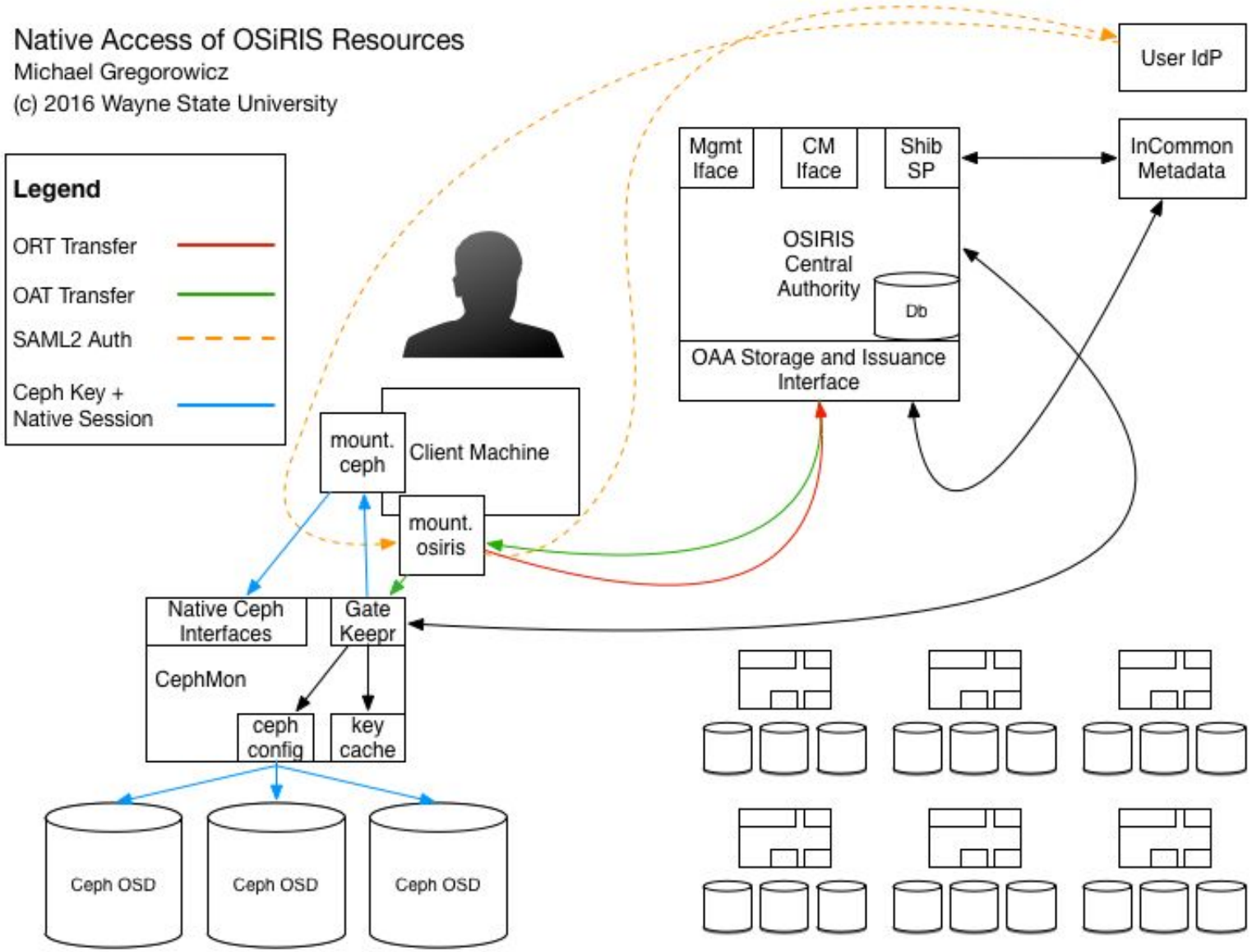
Valid SAML2 sessions are combined with OSiRIS Access Assertions to create Bearer Tokens that users may use with OSiRIS' wide array of interfaces / use cases

Authentication and Authorization

OSiRIS Access Assertions: Overview and Lifecycle



Authentication and Authorization



Physical Ocean Modeling and OSiRIS

Still in the early stages of engagement

The **Naval Research Lab** is collaborating with researchers at **UM** to share their high-resolution ocean models with the broader community

- This data is not classified but is stored on Navy computers that are not easily accessible to many researchers

Discussions are underway to determine a suitable interface and transfer method to put this data into OSiRIS for wider use

We are exploring S3/RGW with objects mapped to a URL to provide high-level organization of the objects (e.g., the URL defines the type/location of the object data)

ATLAS and OSiRIS

ATLAS will use the **OSiRIS** Ceph S3 gateway to read/write single events

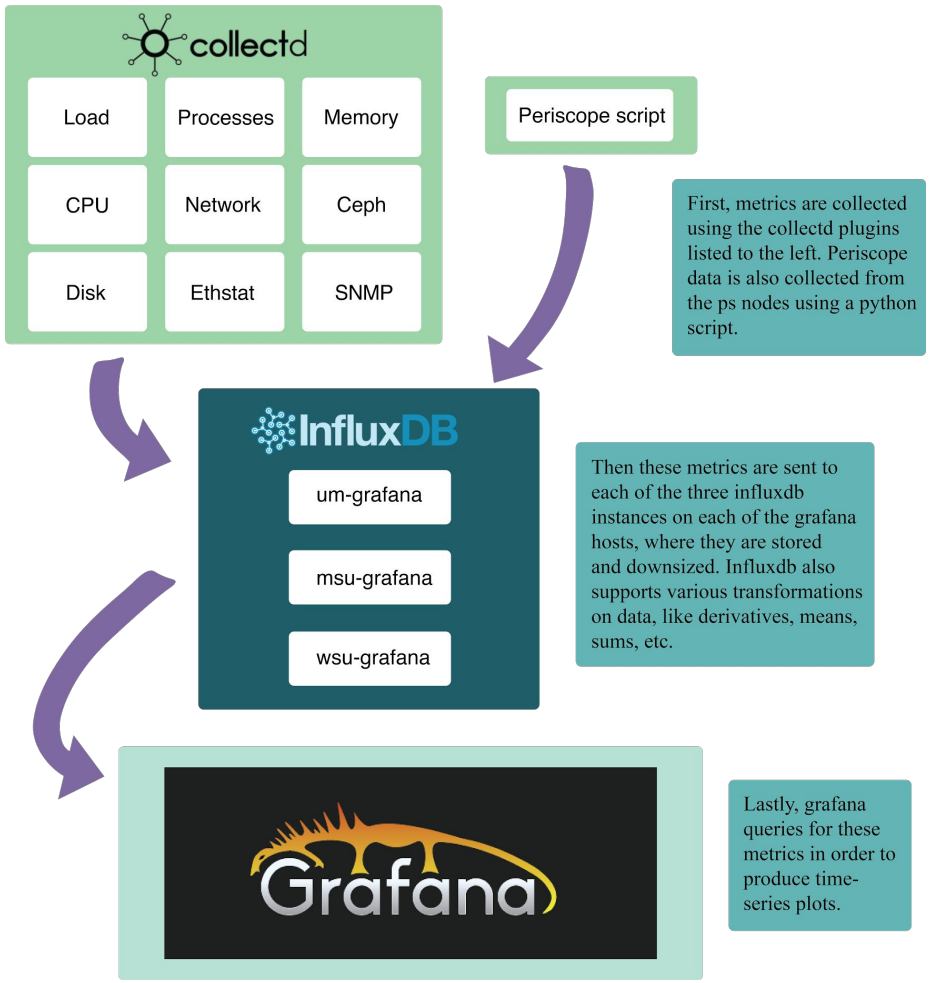
Allows leveraging transient computing resources - lost job means only 1 lost event

Authentication is still in the early stages - doesn't yet tie in with **ATLAS** authentication.

Plots of running test event code

Instrumentation

System Performance Monitoring

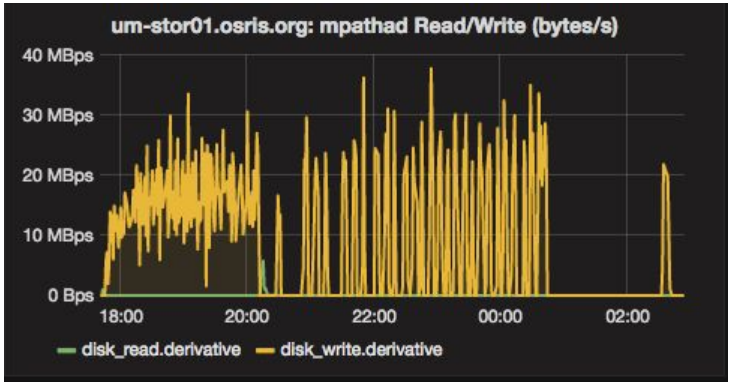


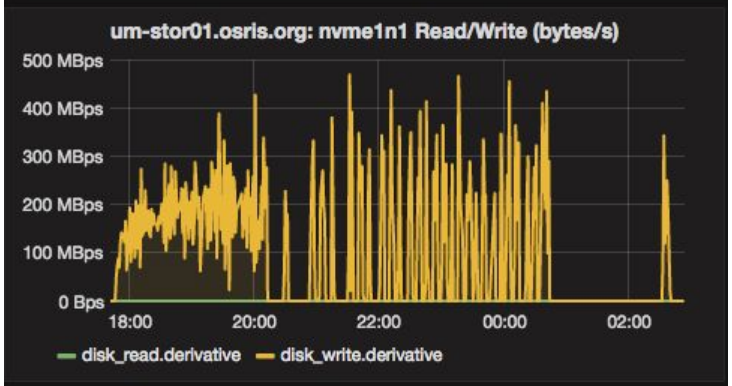
We want insight into each layer of our architecture - systems, storage, network, and Ceph itself

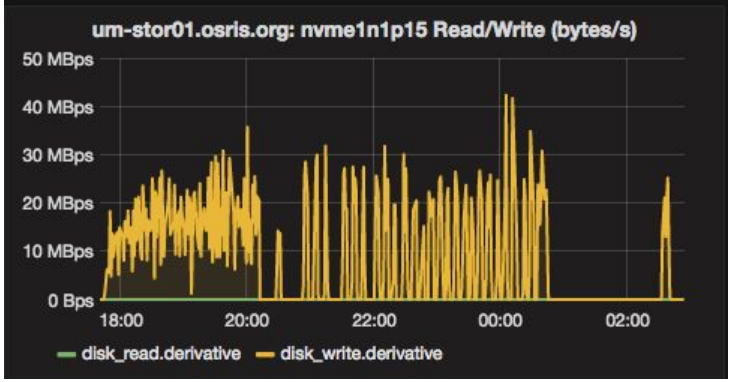
We've been able to leverage collectd and its large collection of plugins

All of our systems feed data to instances of Influxdb and we can plot any series of data with Grafana (examples to follow)

Instrumentation

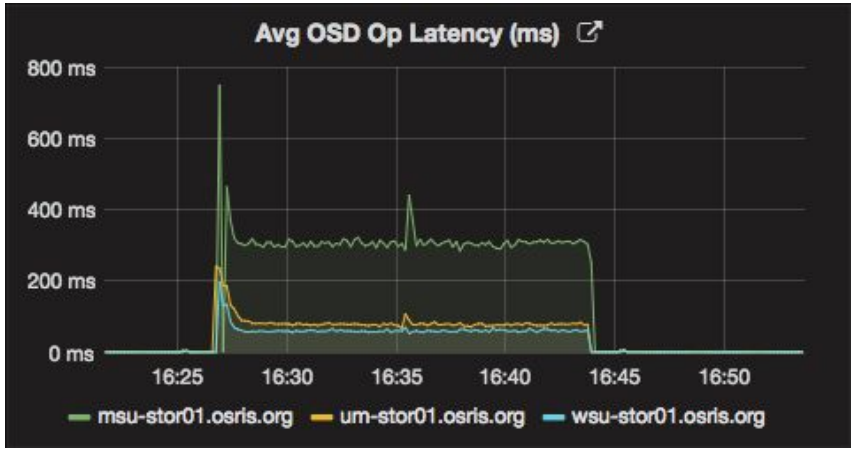






One example - data from collectd-disk plugin focused on one OSD device and its related journal partition as well as the whole device.

Below, collectd-ceph plugin collects stats directly from OSD daemon sockets. We take an average of the latency and group by host.



Latency Experiments

Part of our project is also testing the viable limits of Ceph.

We did a variety of scenarios - at left is a plot of gradually increasing latency to a single storage block during recovery to another node

At 320 ms there are major problems

Things don't really recover fully until we back off to 80ms...likely 80ms is the max safe latency.

Used tc and netem to simulate delay (example slide in supplements)

Latency Experiments

Some conclusions on Ceph and latency (also see http://www.osris.org/performance/latency)

When latency to any OSD host hits 160ms RTT, throughput and ops are effectively 0. Before this point we see a steady and predictable decrease from the maximum but still usable

Max RTT latency to any given OSD is probably about 80ms. Beyond this recovery ops may not be reliable.

Ceph Monitors are not particularly sensitive to latency. At up to 800ms RTT to one monitor there were not any issues and cluster ops, including mon ops such as creating auth keys, were fine. Somewhat slower for some interactive ops.

We didn't try to adjust any timeouts, etc upward to compensate, but this may be possible.

The Goal: Enabling Science

The OSiRIS project goal is enable scientists to collaborate on data easily and without building their own infrastructure

The science domains mentioned all want to be able to directly work with their data without having to move it to their compute clusters, transform it and move results back

Each science domain has different requirements about what is important for their storage use-cases: capacity, I/O capability, throughput and resiliency. OSiRIS has lots of ways to tune for these attributes.

Questions?

Questions or comments?

We gratefully acknowledge the support of the National Science Foundation . Grant information at https://nsf.gov/awardsearch/showAward?AWD_ID=1541335&HistoricalAwards=false

More Information

For more information about OSiRIS:
http://www.osris.org

For more information on our network and system tunings for Ceph:
http://www.osris.org/performance/tuning.html

More information on latency testing:
http://www.osris.org/performance/latency

Monitoring and logging:
http://www.osris.org/components/monitoring.html

NMAL:
http://www.osris.org/nmal

Tuning for Ceph

Ceph spawns a lot of processes, and we have a lot of daemons on dense nodes.

fs.file-max = 78718144

kernel.pid_max = 4194303

We also tune swap tendency down and prefer to retain inode caches under memory pressure

vm.swappiness = 0

vm.vfs_cache_pressure = 1

Besides these we apply a variety of network tunings to all our hosts. Many of these are well known from other sources. For more information please see http://www.osris.org/performance/tuning.html

Netem Example

```
# we need an ifb device to set a delay on ingress packets

modprobe ifb  # load 'intermediate functional block device'
ip link set dev ifb0 up
tc qdisc add dev eth0 ingress
tc filter add dev eth0 parent ffff: \
        protocol ip u32 match u32 0 0 flowid 1:1 action mirred egress
redirect dev ifb0

# now set the netem delay filter on ingress/egress
# last 3 params are delay, variation, and correlation of delay variation
tc qdisc add dev ifb0 root netem delay 10ms 10 25%
tc qdisc add dev eth0 root netem delay 10ms 10 25%
```