Managed Network Services for Exascale Data Movement Across Large Global Scientific Collaborations

Frank Würthwein, Jonathan Guiang, Aashay Arora, **Diego Davila**, John Graham, Dima Mishin, Thomas Hutton, Igor Sfiligoi
Harvey Newman, Justas Balcas
Tom Lehman, Xi Yang, Chin Guok

August 31st, 2022

Using SENSE   To move data in Rucio

~~Managed Network Services~~ for ~~Exascale Data Movement~~ Across Large ~~Global Scientific Collaborations~~

CMS sites

But let’s keep in mind that this work is extensible to any collaboration that uses Rucio to move data across sites

Motivation = HL-LHC

- CMS expects more than half **an exabyte of new data for each year of LHC** operations during the High-Luminosity LHC era from about 2028-2040
  - One annual processing workflow of few hundred PBs
  - Every 3 years, exabyte scale re-processing workflow
- Total aggregate data flows expected to be **dominated by the largest flows**.

- Given that we don’t have infinite $$, we cannot just throw hardware to the problem which means we need to be smart and **use our resources efficiently**

Objective #1 = Rucio/SENSE integration

Make Rucio capable of using SENSE to **schedule transfers on the network**.

In other words:

Give Rucio the ability of saying: “Transferring this Dataset is more important than this other one so it should get more bandwidth and/or better routing.”

E.g. Transfering a DT-dataset V.S. Replicating a random dataset

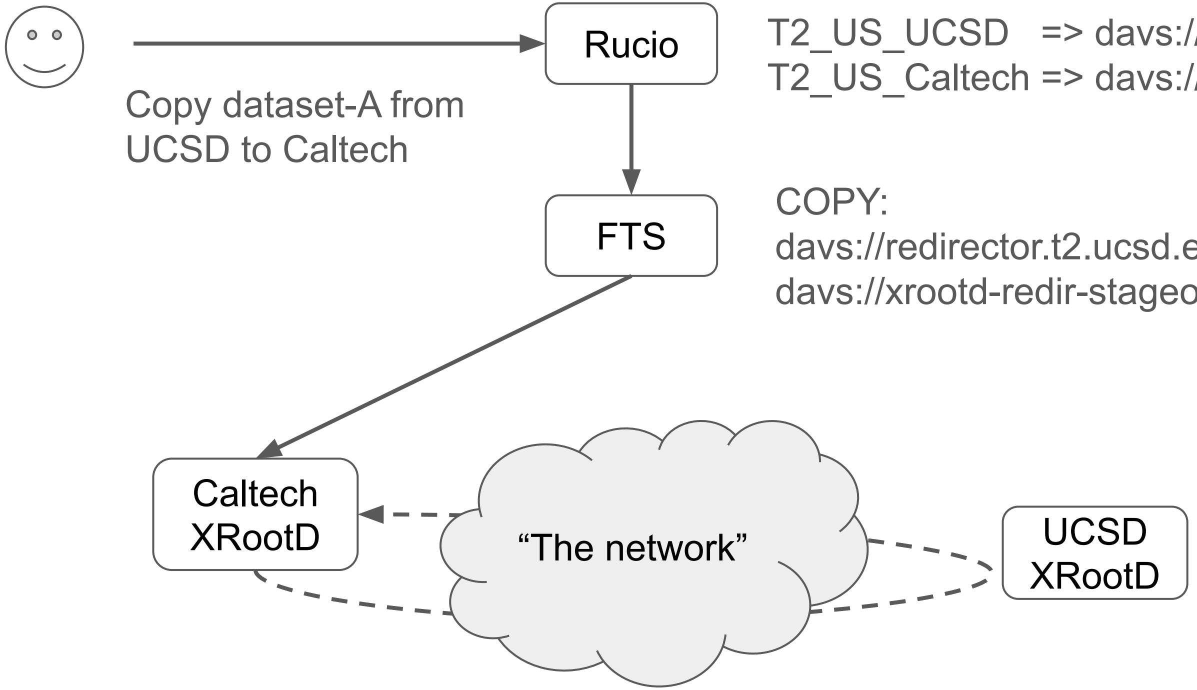
Objective #2

Improve accountability:

Fine-grain managed transfers can be also fine-grain monitored since they travel alone within a well identified network channel

Comparing **Achieved V.S. Allocated bandwidth** will make network & endpoint issues evident.

Quick review: How transfers work nowadays?



Copy dataset-A from UCSD to Caltech

Rucio

FTS

Caltech XRootD

“The network”

UCSD XRootD

T2_US_UCSD   => davs://redirector.t2.ucsd.edu
T2_US_Caltech => davs://xrootd-redir-stageout.ultralight.org

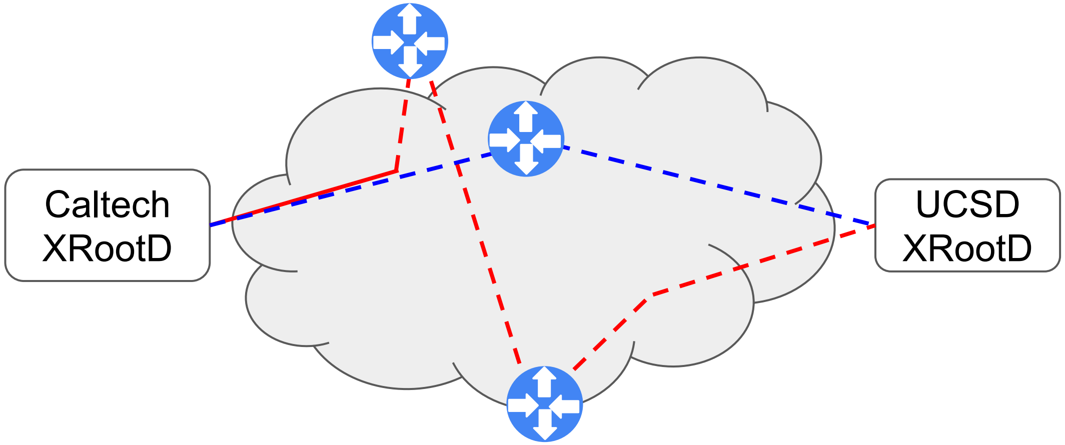
COPY:
davs://redirector.t2.ucsd.edu/dataset-A =>
davs://xrootd-redir-stageout.ultralight.org/dataset-A

GET davs://redirector.t2.ucsd.edu/dataset-A

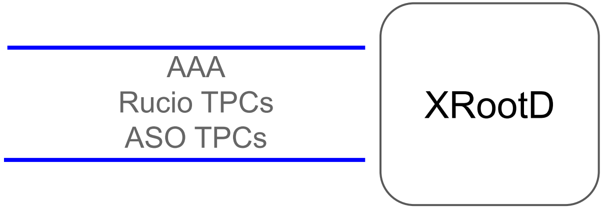
What can be improved?

We have to STOP using the network as a black box

There can be multiple paths between 2 SEs, but we don't get to pick which one to use.
The routing algorithm know nothing about our priorities

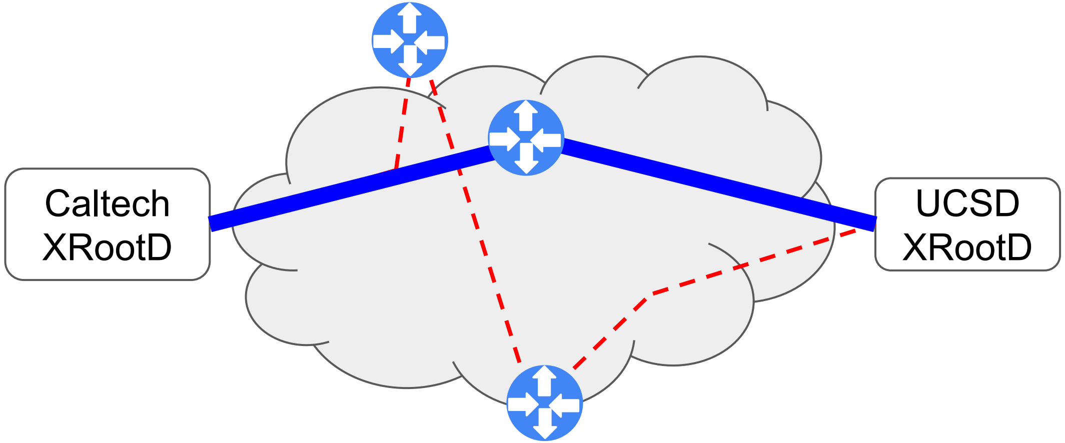


At the SE we put all transfers in the same bucket, no prioritization

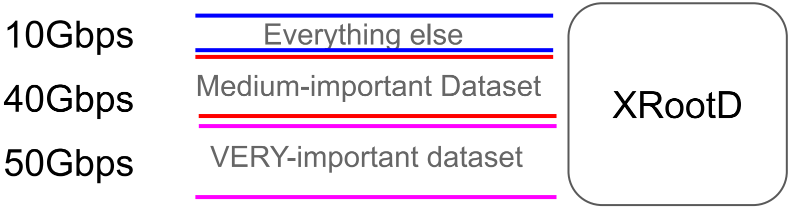


How do we improve things? Using SENSE we can:

Configure **VPNs** between SEs so we can enforce a given path to be used for specific set of transfers



Implement **QoS** so that we can prioritize certain transfers at the DTN level



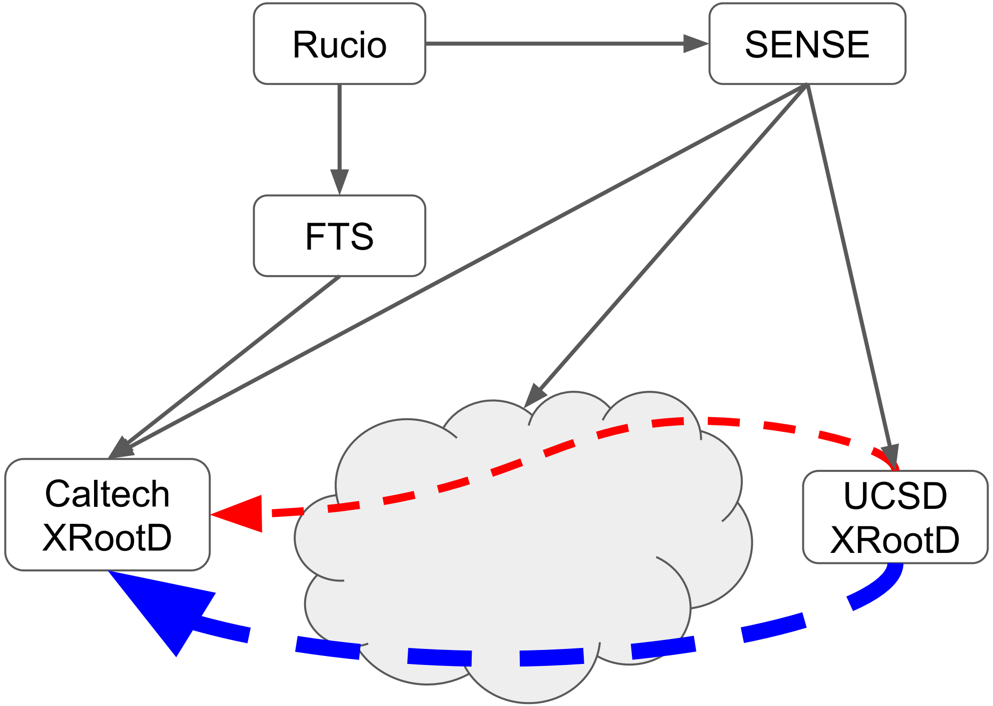
What we envision

An integration of Rucio and SENSE that can be used to create priority channels on-demand between SEs. These links would be:

- of different capacities
- using specific paths
- created across different (*)NRENs

Dataset transfers with different priority will travel in different channels

A given percentage of the SE capacity is **always reserved for best effort** transfers and can make use of the entire capacity when it is not in use



(*)NREN - National Research and Education Network e.g. CENIC, ESnet, Internet2, etc

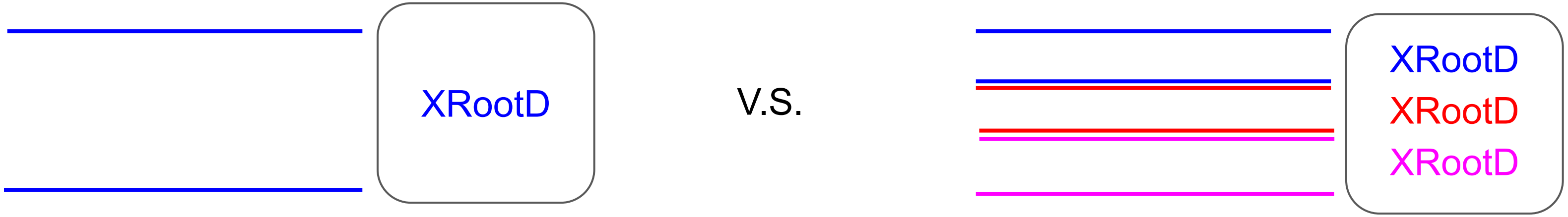
What we need

In order for the above to work we need:

1. **An interface between Rucio and SENSE**
   a. We need to translate Rucio priorities into SENSE requests
2. **Make our SEs capable of supporting multiple communication channels**
   a. Form the network perspective our SEs currently have a single entry point
   b. We need to give SENSE many entry points to use
3. **Run SENSE agents at:**
   a. the SEs (siteRM)
   b. the different NREN (NetworkRM)

Making an SE support multiple communication channels

Using multiple IPv6 addresses, virtual interfaces and XRootD special configs (**no extra HW needed)** we were able to configure, in a single server, multiple instances of XrootD each of them listening on a different IPv6.



Under this approach a single XRootD server can expose N different IPv6 addresses that can be used by SENSE to create different VPNs and to implement QoS among them.

This is easily extendible to an XRootD cluster

The interface between Rucio and SENSE

For our prototype we have developed a software component that acts as the middle man between Rucio and SENSE. This component is called “Data Movement Manager (DMM)”

DMM’s main functionality is to:

- Translate Rucio priorities into SENSE requests
- **Manage the available bandwidth and the different IPv6 endpoints of all the SEs** – this has been proven to be more difficult than it looks and requires further work and thinking

We expect DMM to be a temporary construct. As we figure out what DMM functionality should go into Rucio vs SENSE, DMM will eventually disappear.

SENSE agents

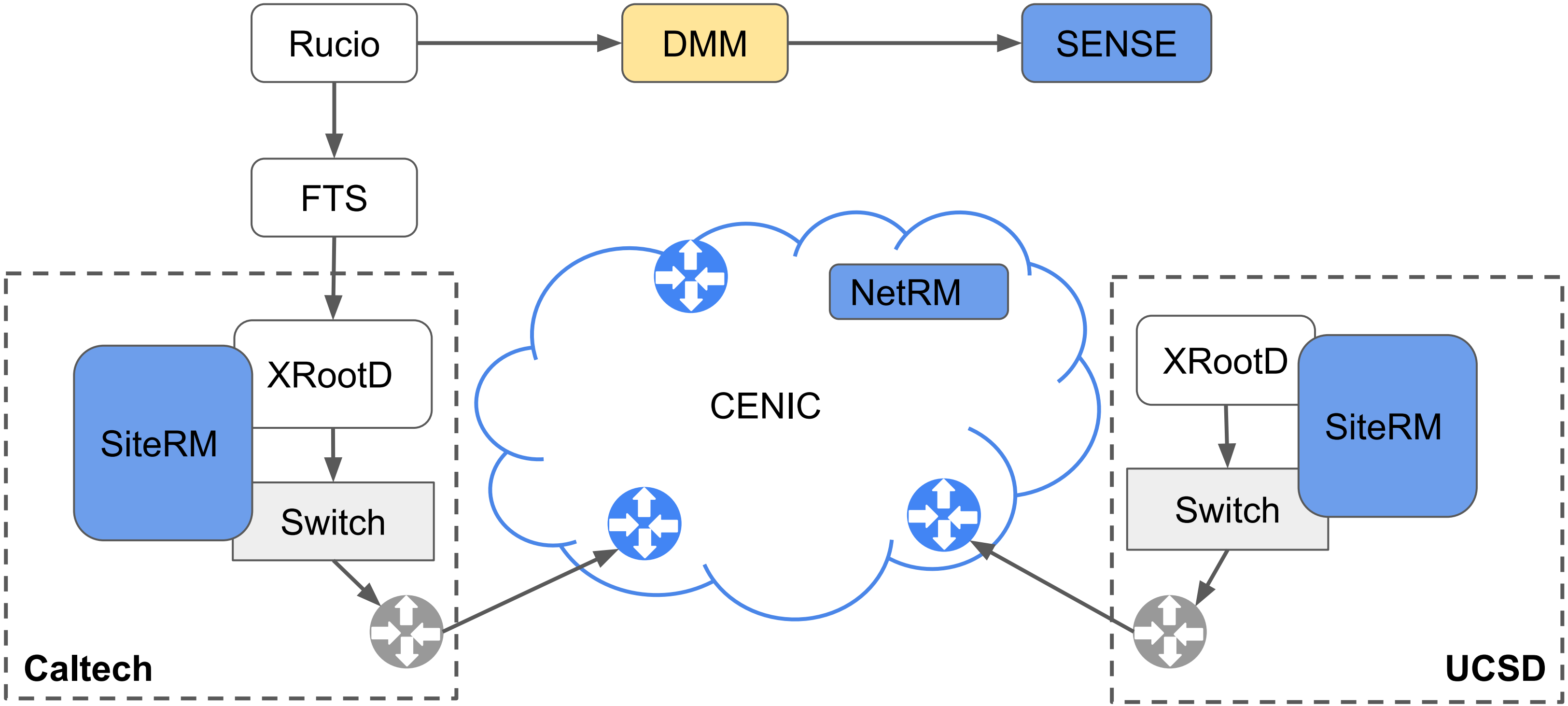
SiteRM

- Provides SENSE with information about the **site** e.g. maximum bandwidth
- Implements routing on the site
- Implements QoS at the DTN level

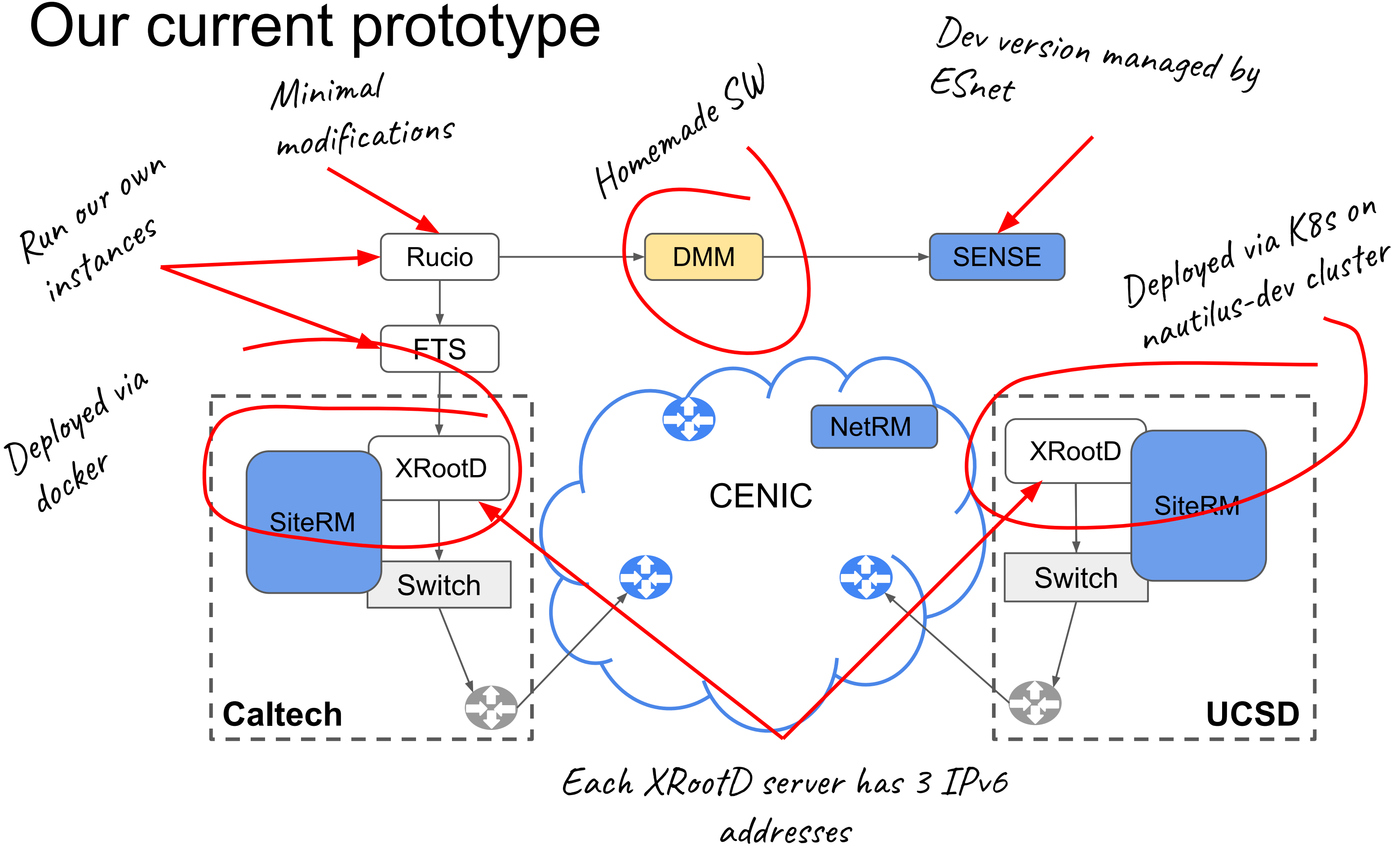
NetworkRM

- Provides SENSE with information about the **network** e.g. topology
- Implements routing on the network
- Implements QoS on the network

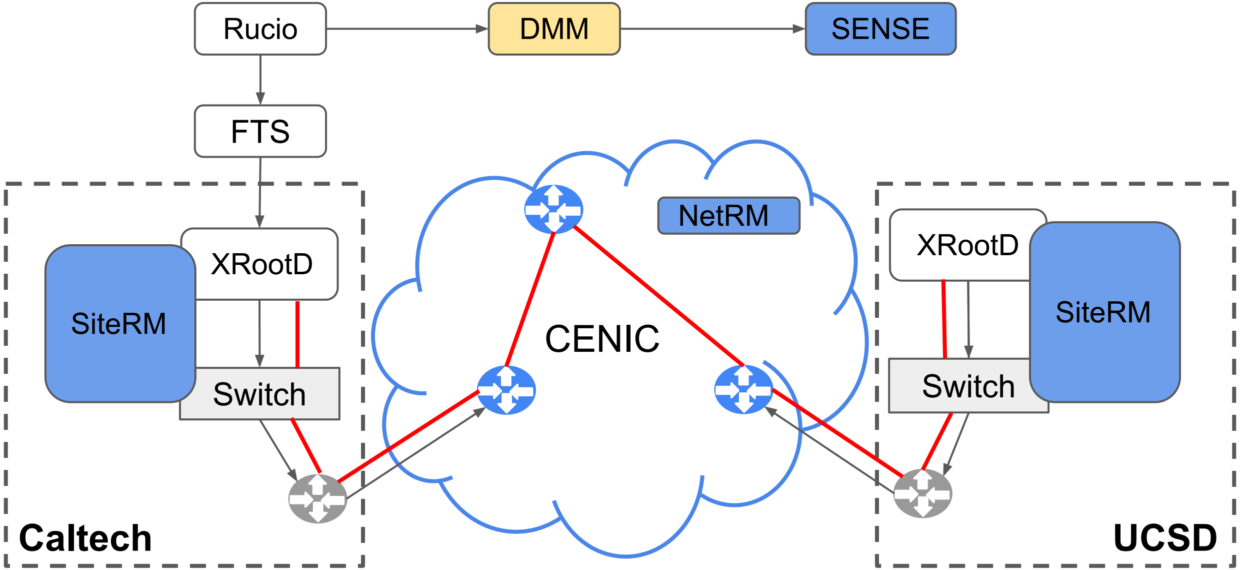
Our current prototype



Our current prototype



How it works? For a non-priority Rucio request

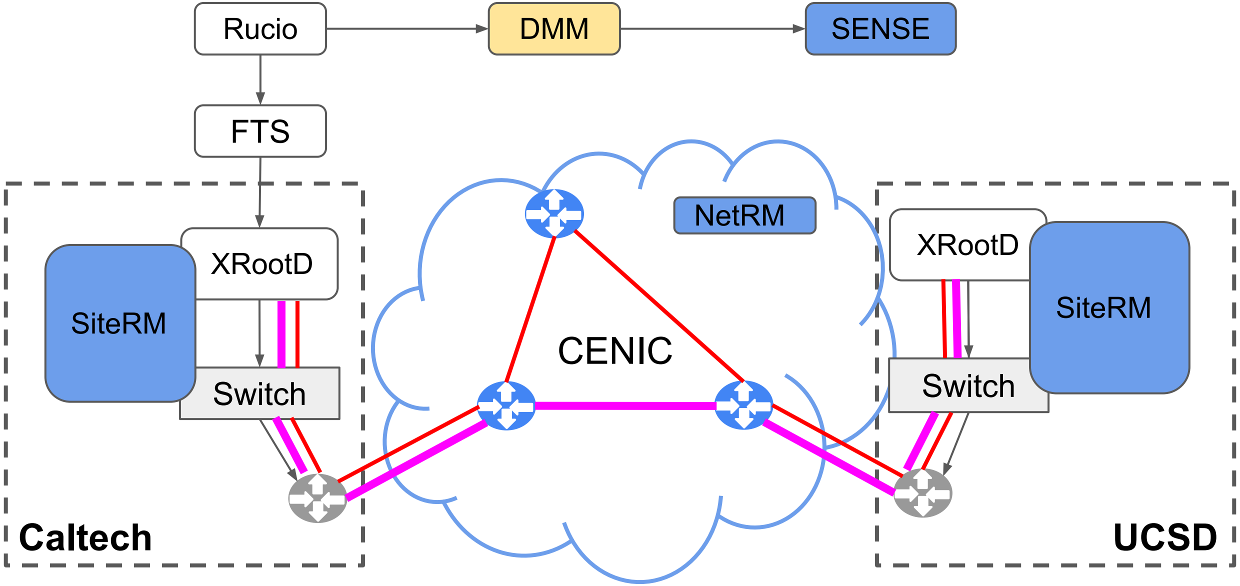


For every Rucio request, Rucio contacts DMM to ask for the IPv6 endpoints to use before contacting FTS

For a regular request (red) DMM will return the IPv6 addresses selected for “best effort”

SENSE is only contacted by DMM in order to get the set of IPv6 addresses of the 2 sites involved in the transfer. This information is cached

How it works? For a priority Rucio request



For a priority Rucio request (pink) DMM picks a pair of free IPv6s and requests a bandwidth allocation on them to SENSE

DMM return the selected pair of IPv6s to Rucio

SENSE instructs SiteRM to implement specific routing and QoS on the given IPv6s at the site level

SENSE instructs NetworkRM to implement specific routing and apply QoS in CENIC nodes in between the 2 IPv6 endpoints

Current status = Just finished our PofC

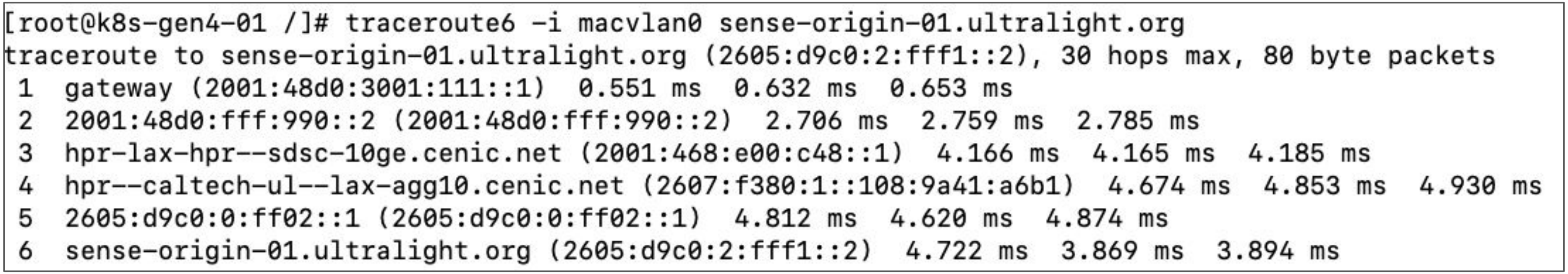
Objective: demonstrate that we can create, in a fully automated way, a priority link between 2 XRootD servers (UCSD and Caltech) triggered by the insertion of a rule in Rucio.

Process:

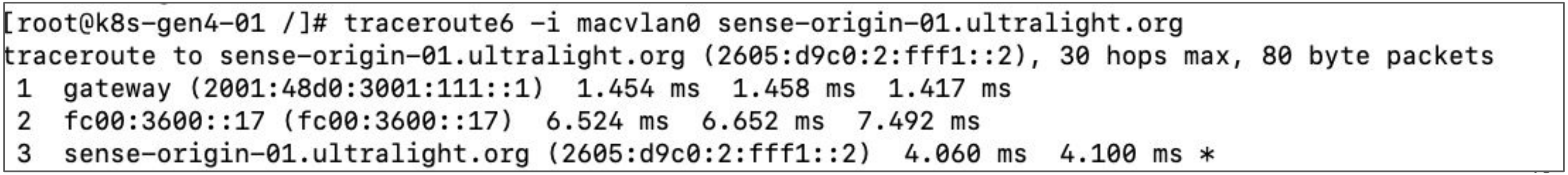
1. Initiate enough background traffic to fill the available bandwidth between the 2 sites using Iperf
2. Insert a priority rule in Rucio to replicate a Dataset from UCSD to Caltech

Proof of Concept

The VPN creation was verified by looking at traceroute between the 2 endpoints before and after the creation of the priority link.



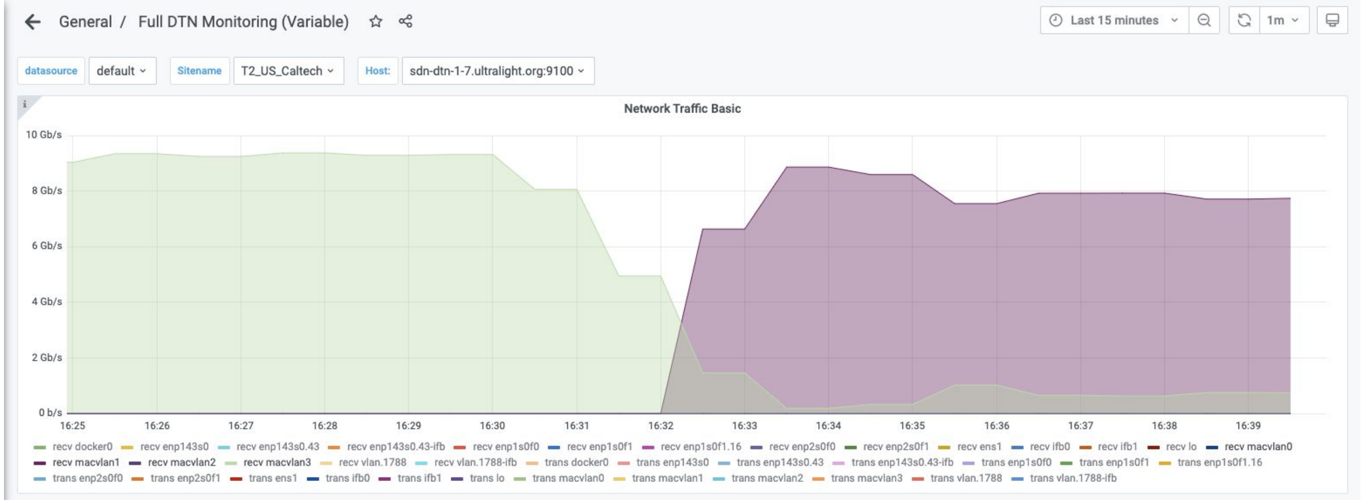
```
[root@k8s-gen4-01 /]# traceroute6 -i macvlan0 sense-origin-01.ultralight.org
traceroute to sense-origin-01.ultralight.org (2605:d9c0:2:fff1::2), 30 hops max, 80 byte packets
 1  gateway (2001:48d0:3001:111::1)  0.551 ms  0.632 ms  0.653 ms
 2  2001:48d0:fff:990::2 (2001:48d0:fff:990::2)  2.706 ms  2.759 ms  2.785 ms
 3  hpr-lax-hpr--sdsc-10ge.cenic.net (2001:468:e00:c48::1)  4.166 ms  4.165 ms  4.185 ms
 4  hpr--caltech-ul--lax-agg10.cenic.net (2607:f380:1::108:9a41:a6b1)  4.674 ms  4.853 ms  4.930 ms
 5  2605:d9c0:0:ff02::1 (2605:d9c0:0:ff02::1)  4.812 ms  4.620 ms  4.874 ms
 6  sense-origin-01.ultralight.org (2605:d9c0:2:fff1::2)  4.722 ms  3.869 ms  3.894 ms
```



```
[root@k8s-gen4-01 /]# traceroute6 -i macvlan0 sense-origin-01.ultralight.org
traceroute to sense-origin-01.ultralight.org (2605:d9c0:2:fff1::2), 30 hops max, 80 byte packets
 1  gateway (2001:48d0:3001:111::1)  1.454 ms  1.458 ms  1.417 ms
 2  fc00:3600::17 (fc00:3600::17)  6.524 ms  6.652 ms  7.492 ms
 3  sense-origin-01.ultralight.org (2605:d9c0:2:fff1::2)  4.060 ms  4.100 ms *
```

Proof of Concept

QoS implementation was verified looking at the network traffic pattern at the virtual interfaces of the destination site.



General / Full DTN Monitoring (Variable)

datasource: default

Sitename: T2_US_Caltech

Host: sdn-dtn-1-7.ultralight.org:9100

Last 15 minutes · 1m

We can see how the background traffic (green) gets shrunk in favor of the priority traffic (purple)

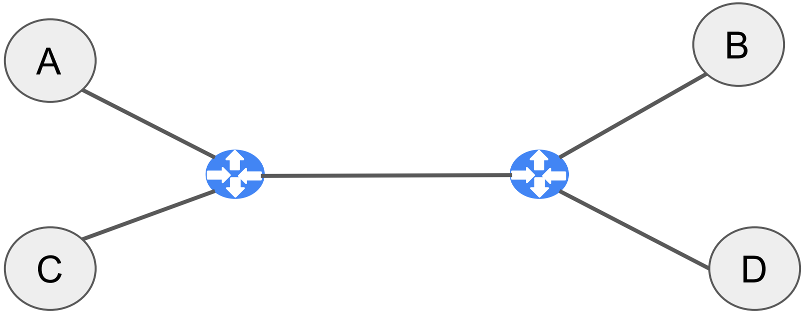
Simulation

Designing effective policies on how bandwidth should be shared is one of the main tasks of DMM and also **a key conceptual challenge for this project** in the long term.

Implementing effective fair sharing is not a trivial task due to the possibility of having several independent transfers using overlapping segments of the network.

Example. In the following topology, consider 2 independent transfers between :
A=>B
C=>D
Without knowing the topology it would be impossible to know that both transfers go across an overlapping segment

Things get quickly more complicated as we start adding more sites



Simulation

To facilitate exploration of this problem we have started developing a simulation of the entire system surrounding DMM including the network topology.

The main objectives of the simulation are:

1. **Validate our observations** of the behavior of the testbed
2. Playback annual sequences of actual transfers to **show SENSE benefits**
    a. We plan to use the monitoring records from Rucio and/or FTS for that
3. Collaborate with CS researchers to **develop policies for network bandwidth allocation**

What’s next?

- Design and implement the monitoring
    - SiteRM already records traffic on all the interfaces of the DTNs
    - We should be able to compare allocated vs achieved bandwidth without too much trouble
- Add more sites to our testbed
    - This increments the complexity of the policies to implement in DMM
    - Give us more use cases to test
    - FNAL and Nebraska are the next in the list
- Simulate the effects of different policies that DMM can implement
- Participate as a prototype in the WLCG Data Challenge 2023

Summary

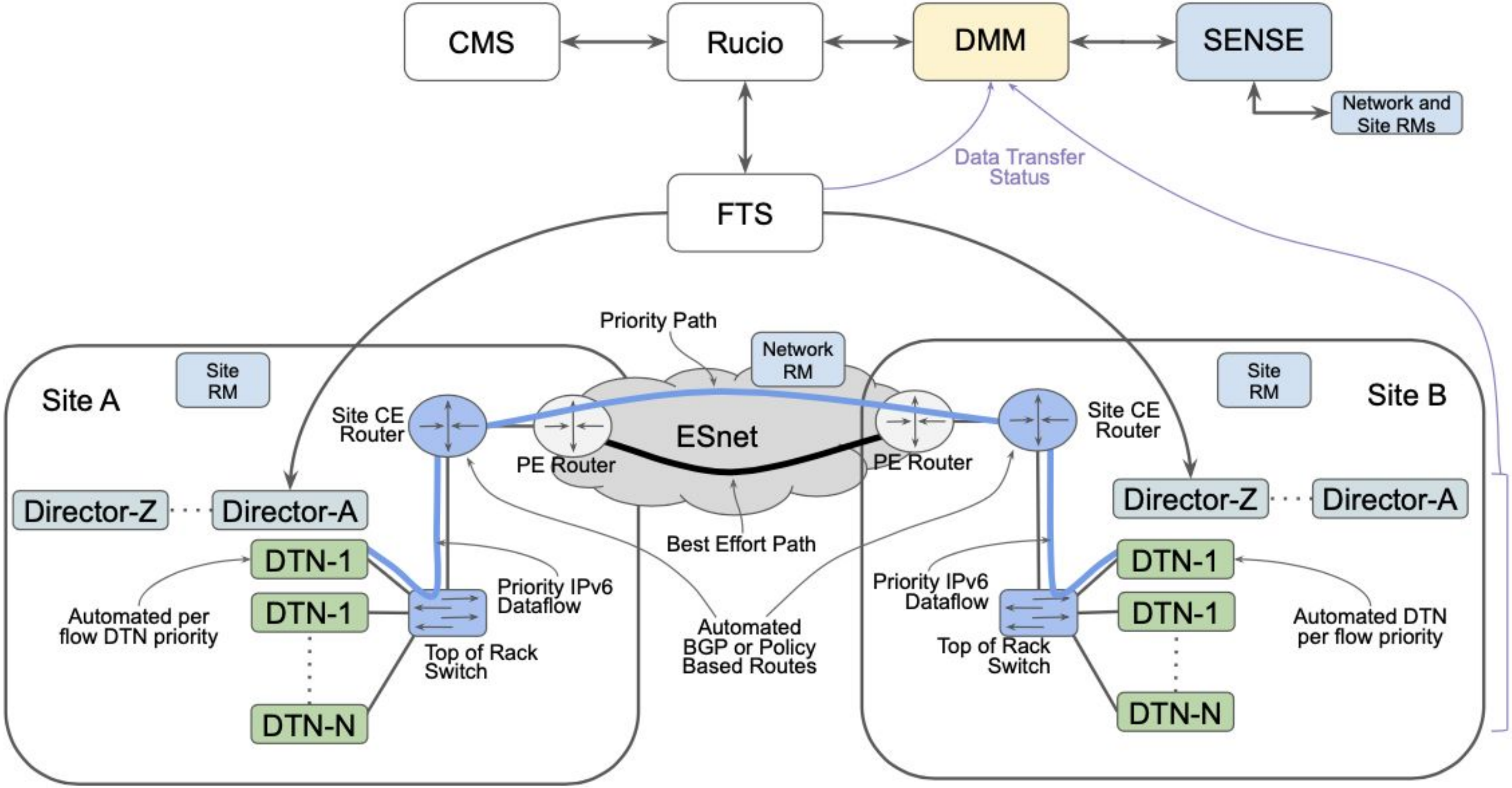
- In order to meet the increased requirements of the HL-LHC we need to use our resources efficiently
- SENSE can help us to make a better use of the network resources
- We have demonstrated that Rucio and SENSE can be integrated in a fully automated way
- Fine-grain managed transfers should, in principle, be easy to fine-grain monitor
- Designing fair sharing policies for network allocation does NOT seem trivial
- Simulation can help us speed up the development of fair share policies

Questions?

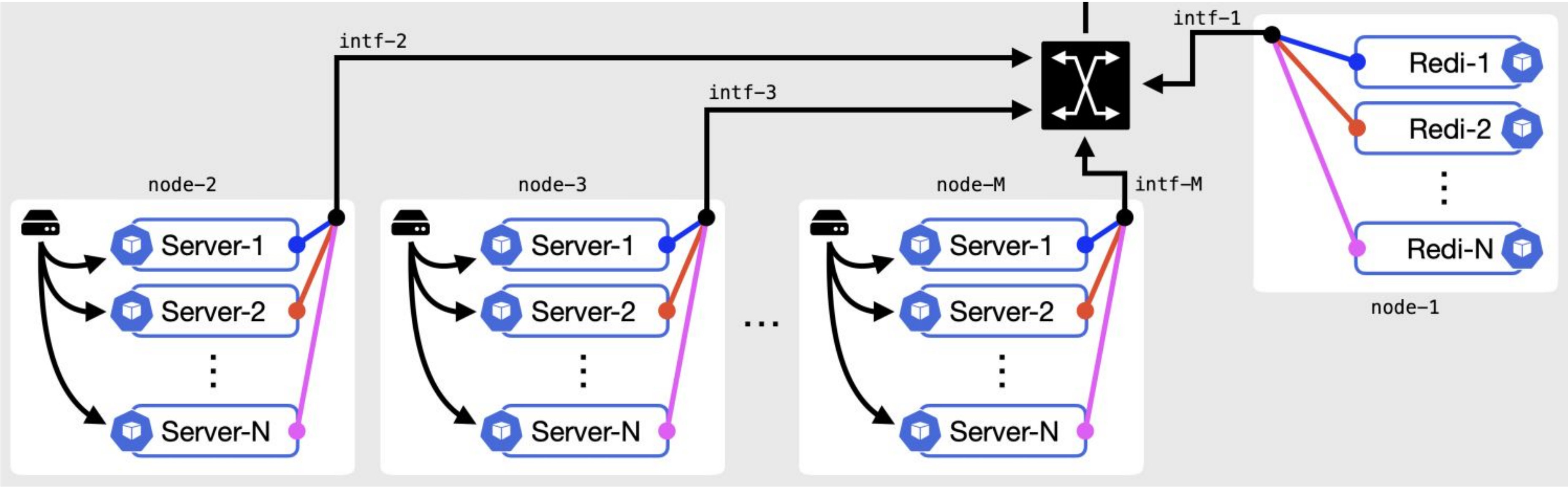
ACKNOWLEDGMENTS

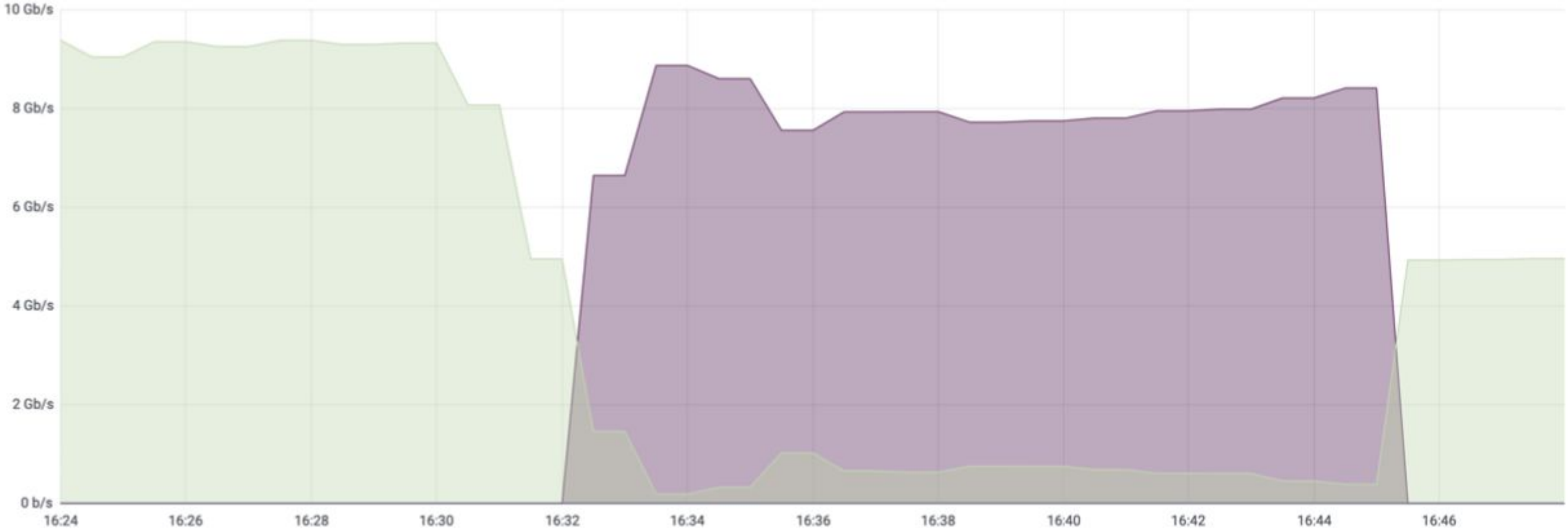
This ongoing work is partially supported by the US National Science Foundation (NSF) Grants OAC-2030508, OAC- 1841530, OAC-1836650, MPS-1148698, and PHY-1624356. In addition, the development of SENSE is supported by the US Department of Energy (DOE) Grants DE-SC0015527, DE- SC0015528, DE-SC0016585, and FP-00002494. Finally, this work would not be possible without the significant contributions of collaborators at ESNet, Caltech, and SDSC.

Backup slides


CMS
Rucio
DMM
SENSE
Network and Site RMs
Data Transfer Status
FTS
Priority Path
Network RM
Site A
Site RM
Site CE Router
PE Router
ESnet
PE Router
Site CE Router
Site RM
Site B
Director-Z
Director-A
Director-Z
Director-A
DTN-1
DTN-1
DTN-N
Automated per flow DTN priority
Priority IPv6 Dataflow
Top of Rack Switch
Best Effort Path
Automated BGP or Policy Based Routes
Priority IPv6 Dataflow
Top of Rack Switch
DTN-1
DTN-1
DTN-N
Automated DTN per flow priority

Multi-channel XRootD cluster




10 Gb/s
8 Gb/s
6 Gb/s
4 Gb/s
2 Gb/s
0 b/s
16:24
16:26
16:28
16:30
16:32
16:34
16:36
16:38
16:40
16:42
16:44
16:46