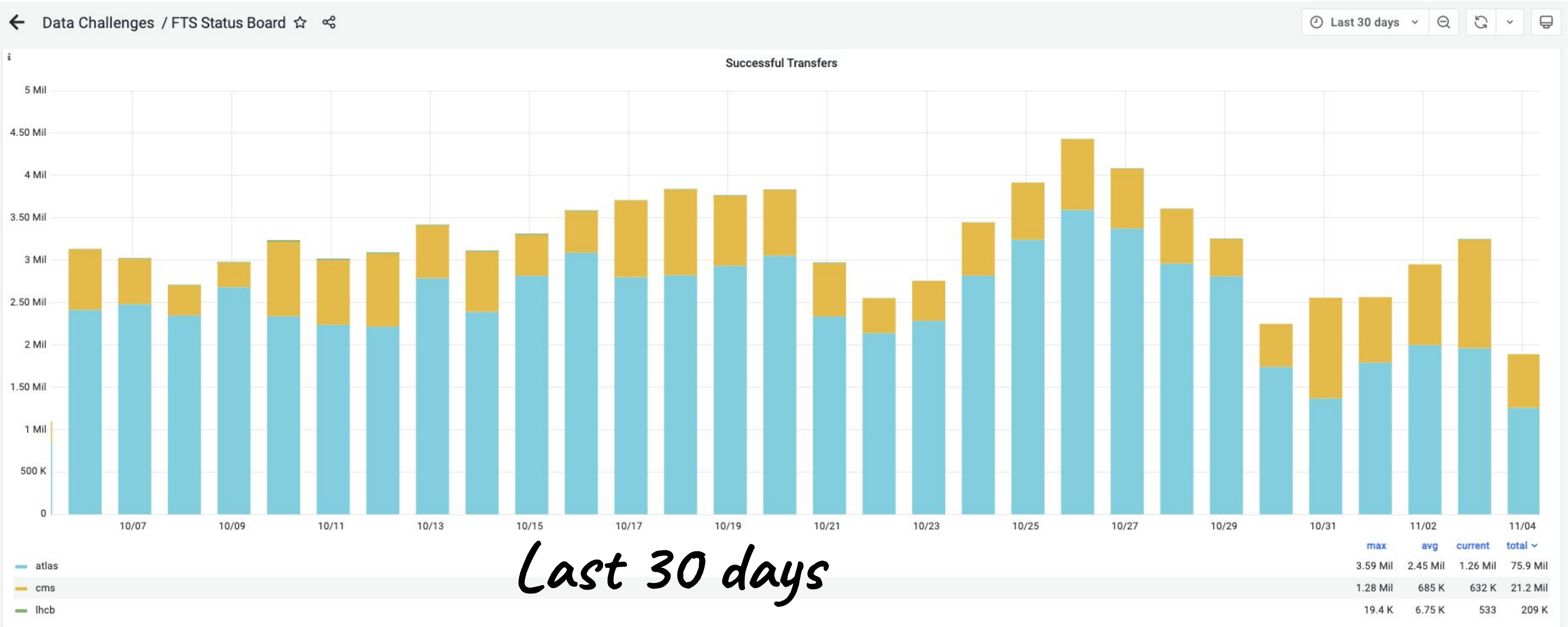
Network Management Enhancements for the High Luminosity Era

Frank Würthwein, Jonathan Guiang, Aashay Arora, **Diego Davila**, John Graham, Dima Mishin, Thomas Hutton, Igor Sfiligoi, Harvey Newman, Justas Balcas, Preeti Bhat, Tom Lehman, Xi Yang, Chin Guok, Oliver Gutsche, Phil Demar, Marcos Schwarz

WLCG workshop, Nov 2022

Motivation: Millions of transfers every day

3Mil



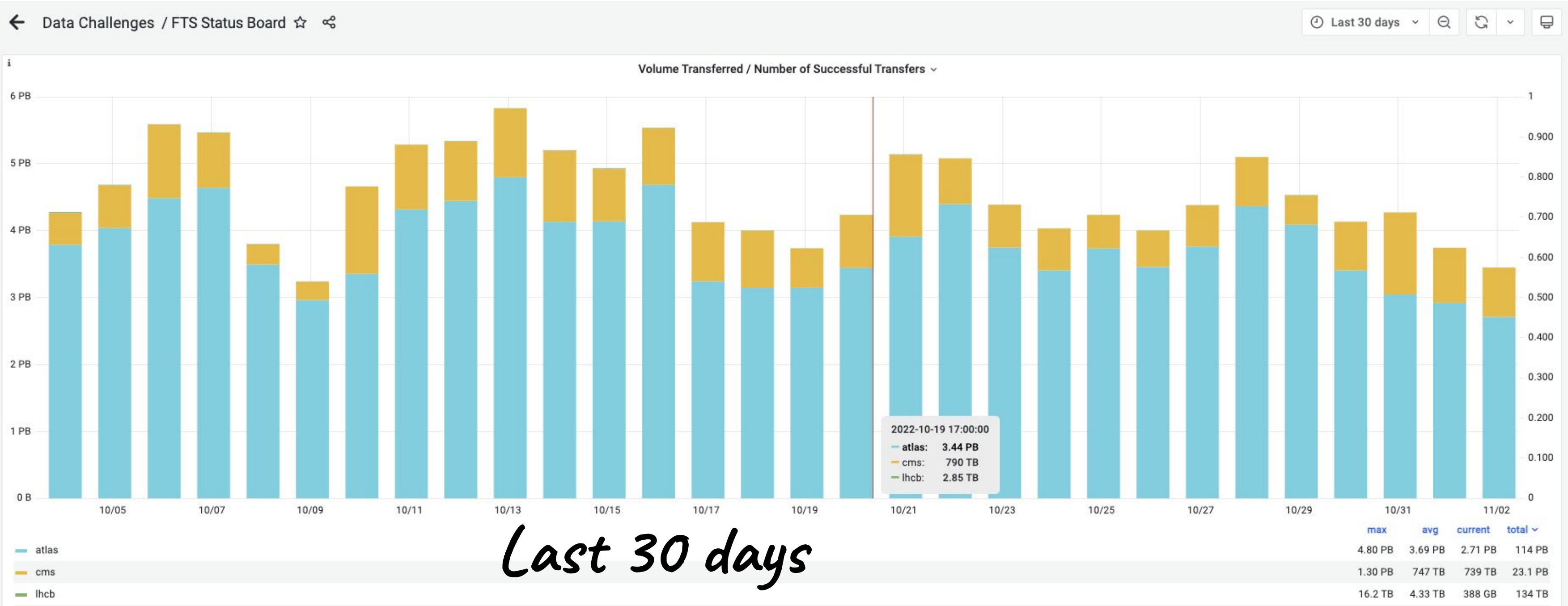
	max	avg	current	total
atlas	3.59 Mil	2.45 Mil	1.26 Mil	75.9 Mil
cms	1.28 Mil	685 K	632 K	21.2 Mil
lhcb	19.4 K	6.75 K	533	209 K

Last 30 days

https://monit-grafana.cern.ch/goto/WeXOcGDVk?orgId=20

Motivation: PBs of data every day

4PB →



Data Challenges / FTS Status Board

Last 30 days

Volume Transferred / Number of Successful Transfers

2022-10-19 17:00:00
atlas: 3.44 PB
cms: 790 TB
lhcb: 2.85 TB

	max	avg	current	total
atlas	4.80 PB	3.69 PB	2.71 PB	114 PB
cms	1.30 PB	747 TB	739 TB	23.1 PB
lhcb	16.2 TB	4.33 TB	388 GB	134 TB

Last 30 days

https://monit-grafana.cern.ch/goto/aeOjFMvVz?orgId=20

Motivation: CMS estimated numbers during HL-LHC

Notice: The previous plots show only Third Party Copy Successful transfers. XRootD reads **not** shown there

During **High Luminosity** LHC CMS itself expects:

- more than **half an exabyte of new data** for each year of operations
- one annual processing workflow of a few **hundred PBs**
- one **ExaByte scale re-processing** workflow every 3 years

ESnet/Data Challenge estimates a min bandwidth requirement of **1.4Tbps across the Atlantic** for CMS and ATLAS alone up to **2.7Tbps** for the “flexible scenario”

Total aggregate data flows are expected to be **dominated by the largest flows**

What’s the issue with large flows

Think of Data Taking

Q. How large has to be the buffer at CERN?

A> It depends on how fast we can move data to the T1s

Q. How fast we can move data to the T1s?

A> It depends on how much network traffic is in and in-between the sites

Lack of predictability makes planning harder

What if we could fine grain manage our largest data flows?

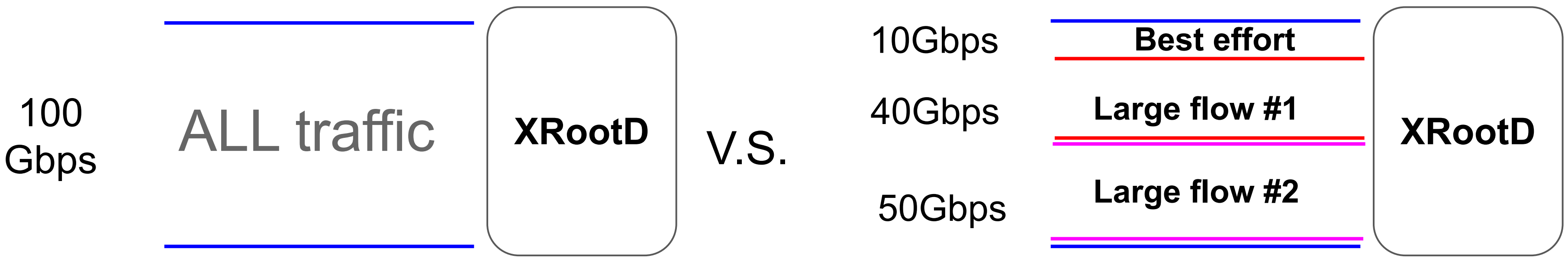
- What if we could…
  - **Isolate and guarantee** a minimum bandwidth for any given data flow
  - Assemble that minimum as aggregate across the **best network paths** available
- Then we could:
  - **Predict the duration** of these data flows
  - **Find and fix** issues when transfers performs poorly
  - Prioritization

Goal: be able to fine grain manage our largest data flows

What if we could fine grain manage our largest data flows?

- What if we could…
  - Isolate and guarantee a minimum bandwidth for any given data flow

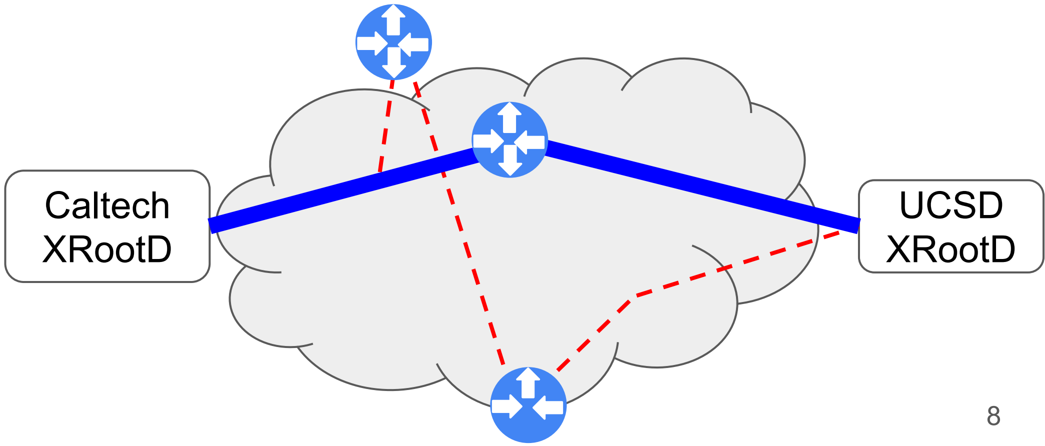
Configure SEs with multi-endpoints to isolate flows and **Quality of Service (QoS)** to assign/allocate bandwidth



What if we could fine grain manage our largest data flows?

- What if we could…
  - Assemble that minimum as aggregate across the **best network paths** available

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



THE WLCG IF WE COULD FINE GRAIN MANAGE

OUR LARGEST DATA FLOWS

What we propose

Integration of Rucio and SENSE

Why?

Rucio:

- Knows everything about our datasets
- Triggers and keeps track of our transfer requests
- Knows our priorities

SENSE knows how to build multi-domain network services e.g. Quality of Service (QoS) and Virtual Private Network (VPN)

Let’s make Rucio express its wishes to SENSE

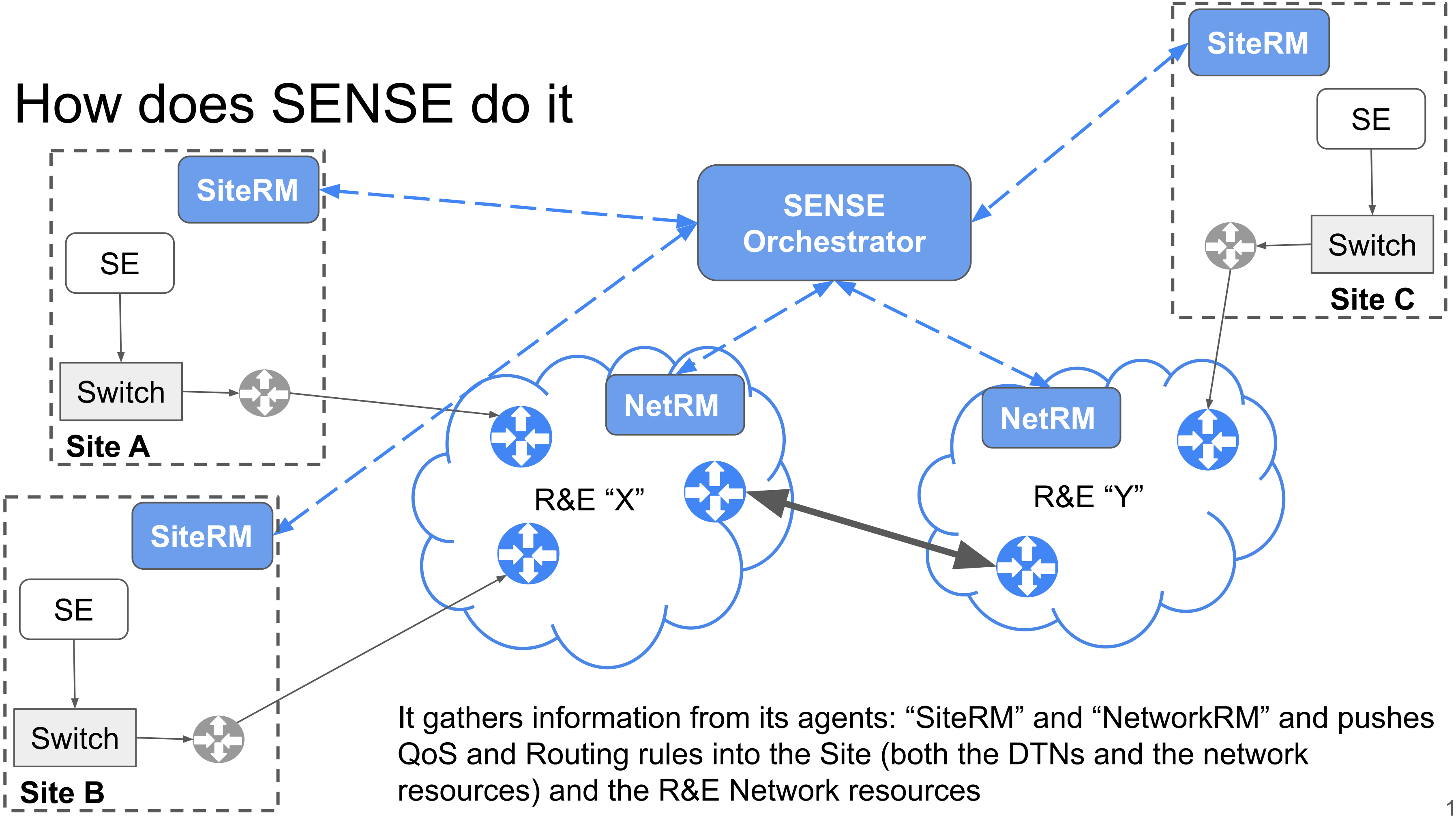
Why?

Rucio knows what we want to do

SENSE knows how to do it

- QoS => bandwidth guarantees
- VPN => fixed network paths

How does SENSE do it

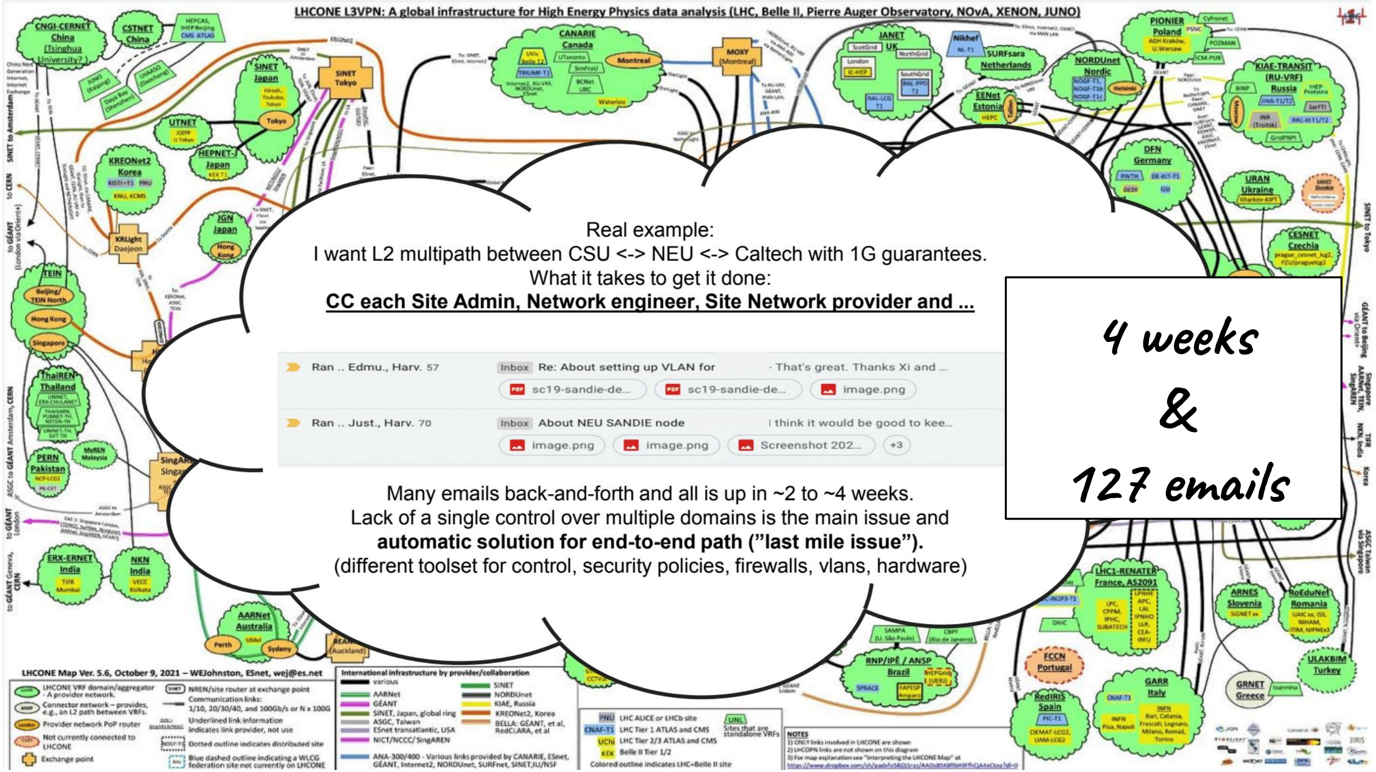


It gathers information from its agents: "SiteRM" and "NetworkRM" and pushes QoS and Routing rules into the Site (both the DTNs and the network resources) and the R&E Network resources

Building network services **without** SENSE

Stolen slide from Justas' presentation on the LHCOPN/LHCONE

Full presentation available here



Real example:
I want L2 multipath between CSU <-> NEU <-> Caltech with 1G guarantees.
What it takes to get it done:
CC each Site Admin, Network engineer, Site Network provider and ...

Ran .. Edmu., Harv. 57 Inbox Re: About setting up VLAN for - That's great. Thanks Xi and ...
sc19-sandie-de... sc19-sandie-de... image.png

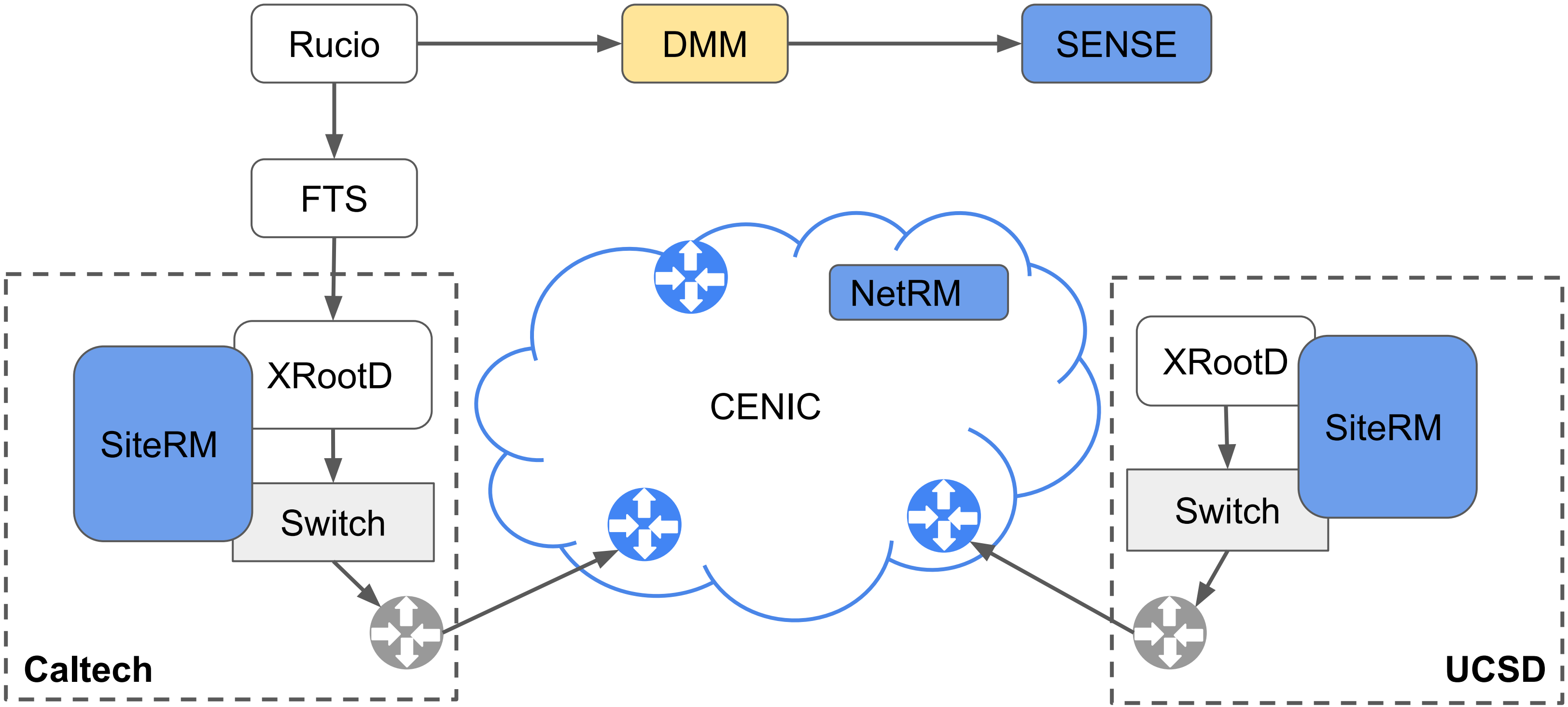
Ran .. Just., Harv. 70 Inbox About NEU SANDIE node I think it would be good to kee...
image.png image.png Screenshot 202... +3

Many emails back-and-forth and all is up in ~2 to ~4 weeks.
Lack of a single control over multiple domains is the main issue and
automatic solution for end-to-end path ("last mile issue").
(different toolset for control, security policies, firewalls, vlans, hardware)

4 weeks
&
127 emails

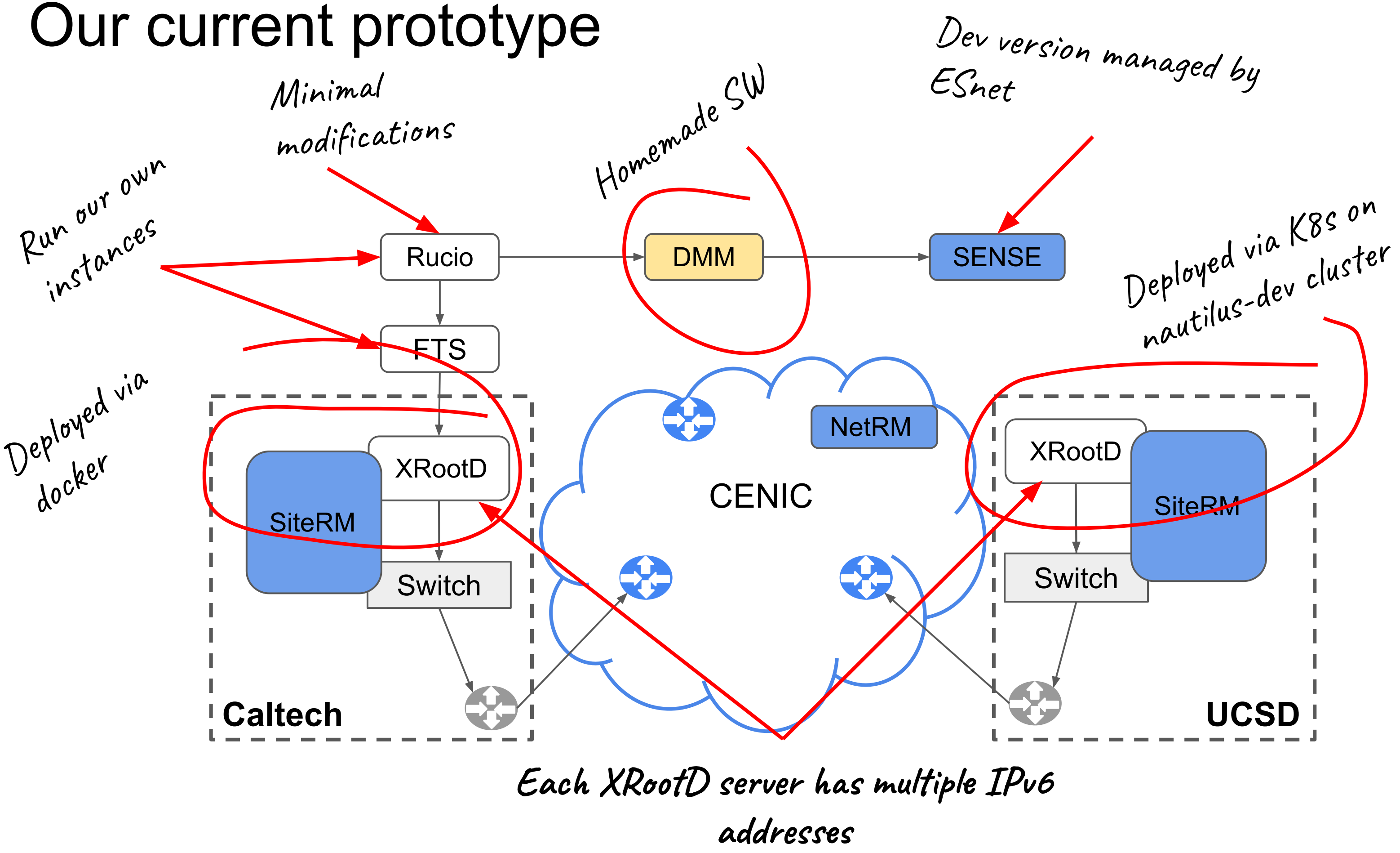
https://indico.cern.ch/event/1146558/contributions/5030701/attachments/2534532/4361680/Justas-B-CMS-Rucio-LHCONE-latest.pdf

How does Rucio + SENSE looks like



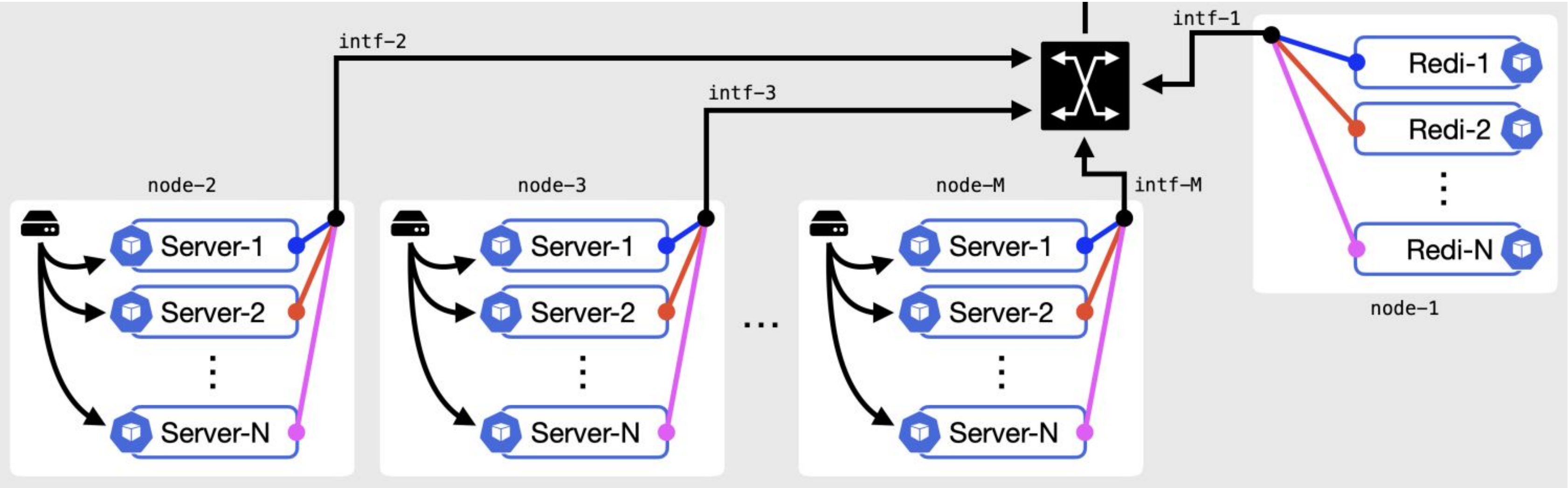
DMM: Data Movement Manager (interface between Rucio and SENSE … and much more)

Our current prototype



XRootD multi-endpoint

- Priority services (QoS and VPN) are established on a subnet basis
- An XRootD cluster requires N different subnets to participate in N priority services.
- An XRootD cluster with M servers will require M x N IP addresses i.e. every server will have an IP in each subnet



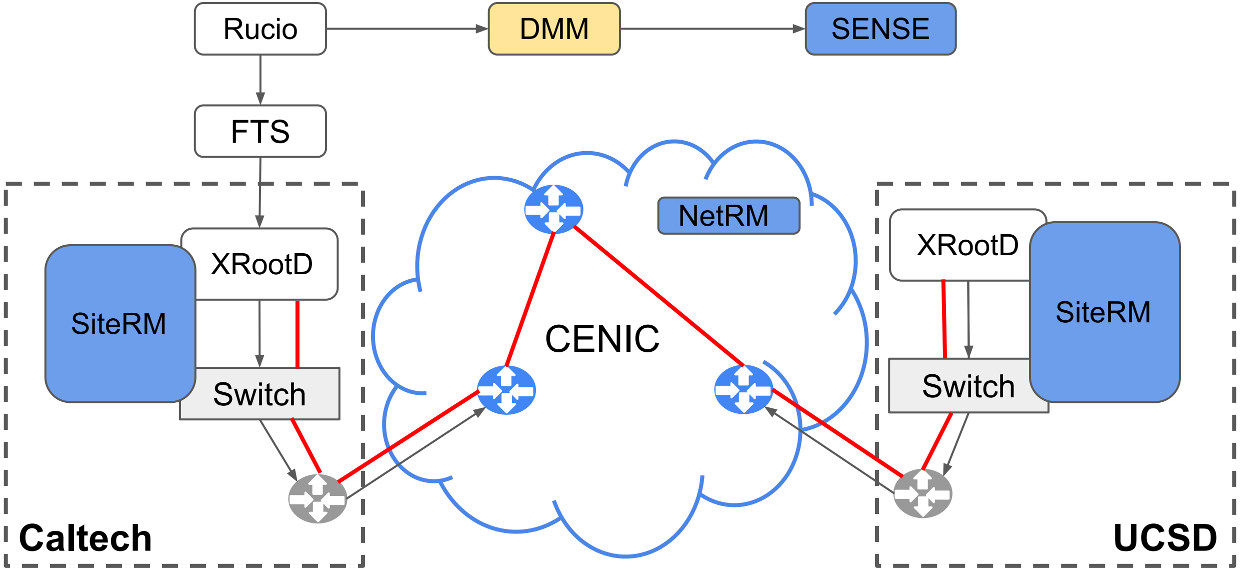
XRootD cluster with M servers and N subnets, Every color represents a different subnet

XRootD multi-endpoint (cont’d)

- As M and N grow you run out of IP addresses quickly
- We use IPv6 because they are “cheap”
- In principle this should work with IPv4 as well

For the sake of making things simpler let’s think of the case of a single XRootD server with N different IPv6 addresses on each Site.

How it works? For a **non-priority** Rucio request

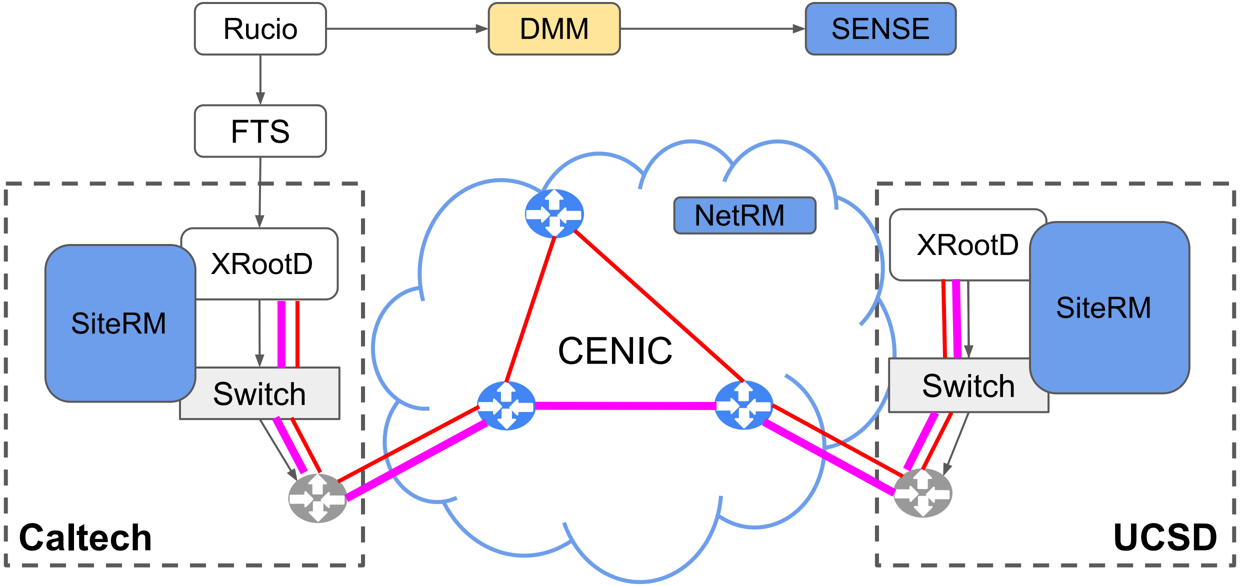


For every Rucio request, Rucio contacts DMM to ask for the endpoints (IP addresses) to use before contacting FTS

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

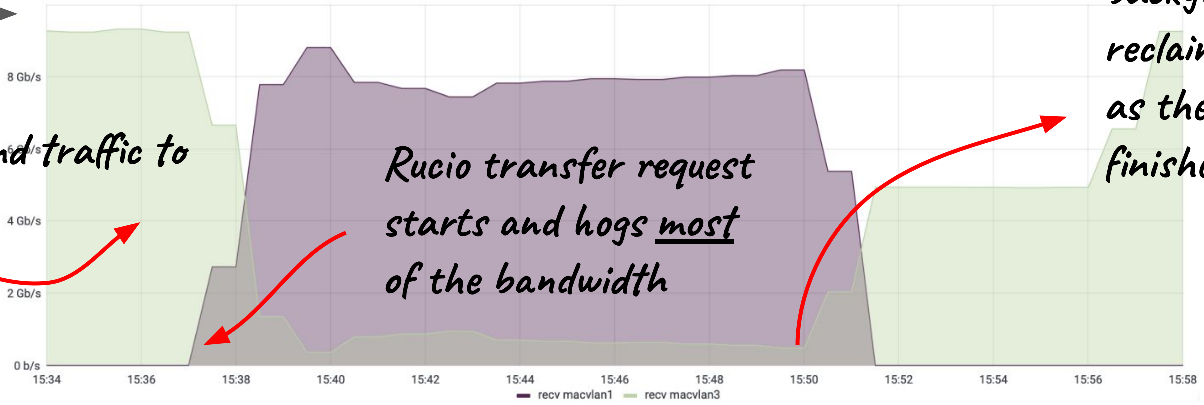
SENSE instructs NetworkRM to implement specific routing and apply QoS in CENIC nodes in between the 2 IPv6 endpoints

When the transfer is finished Rucio signals DMM which request the deallocation of the priority services

Our Proof of Concept

As a PofC we wanted to prove that we could create a priority service between 2 sites:

- On demand i.e. triggered solely by the creation of a rule in Rucio
- On a congested network path (to show QoS)
- Just for the duration of the transfer request in question



10Gbps

Artificial background traffic to produce congestion

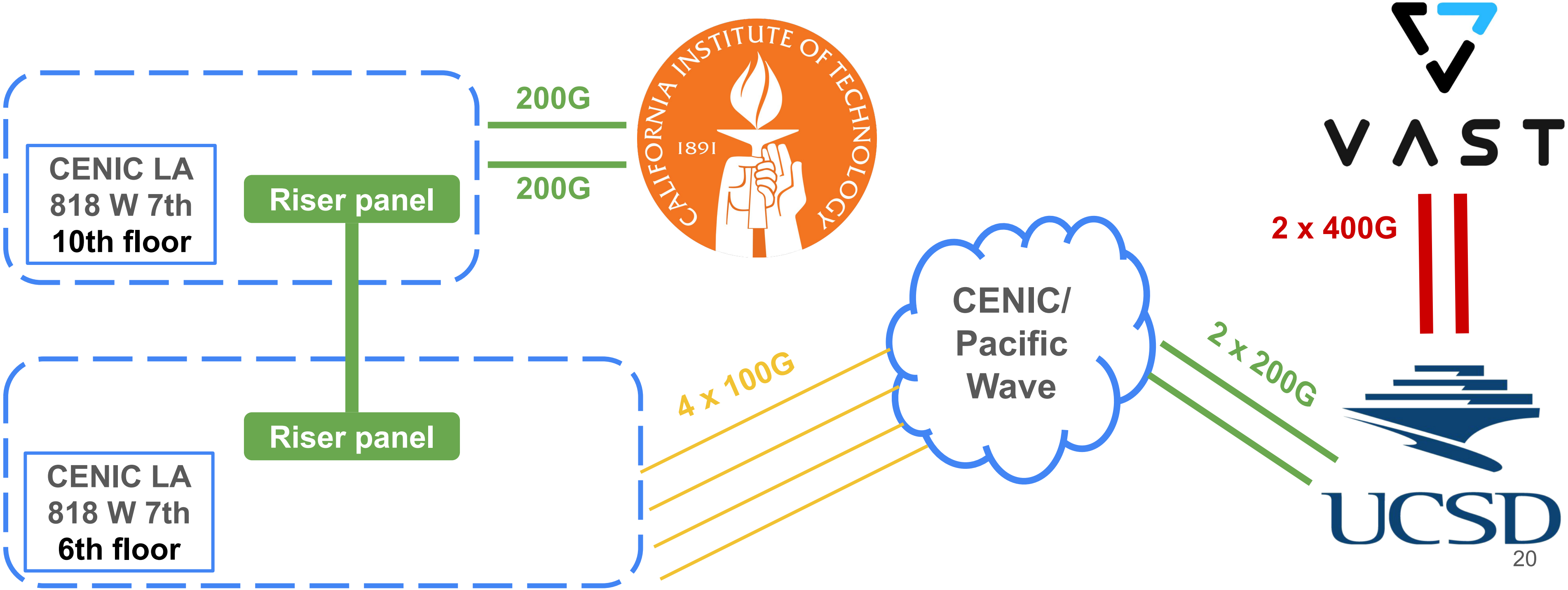
Rucio transfer request starts and hogs <u>most</u> of the bandwidth

background traffic reclaiming bandwidth as the transfer finishes

Network traffic on 2 different virtual interfaces in the receiving XRootD server

Coming soon: new test at 400Gbps

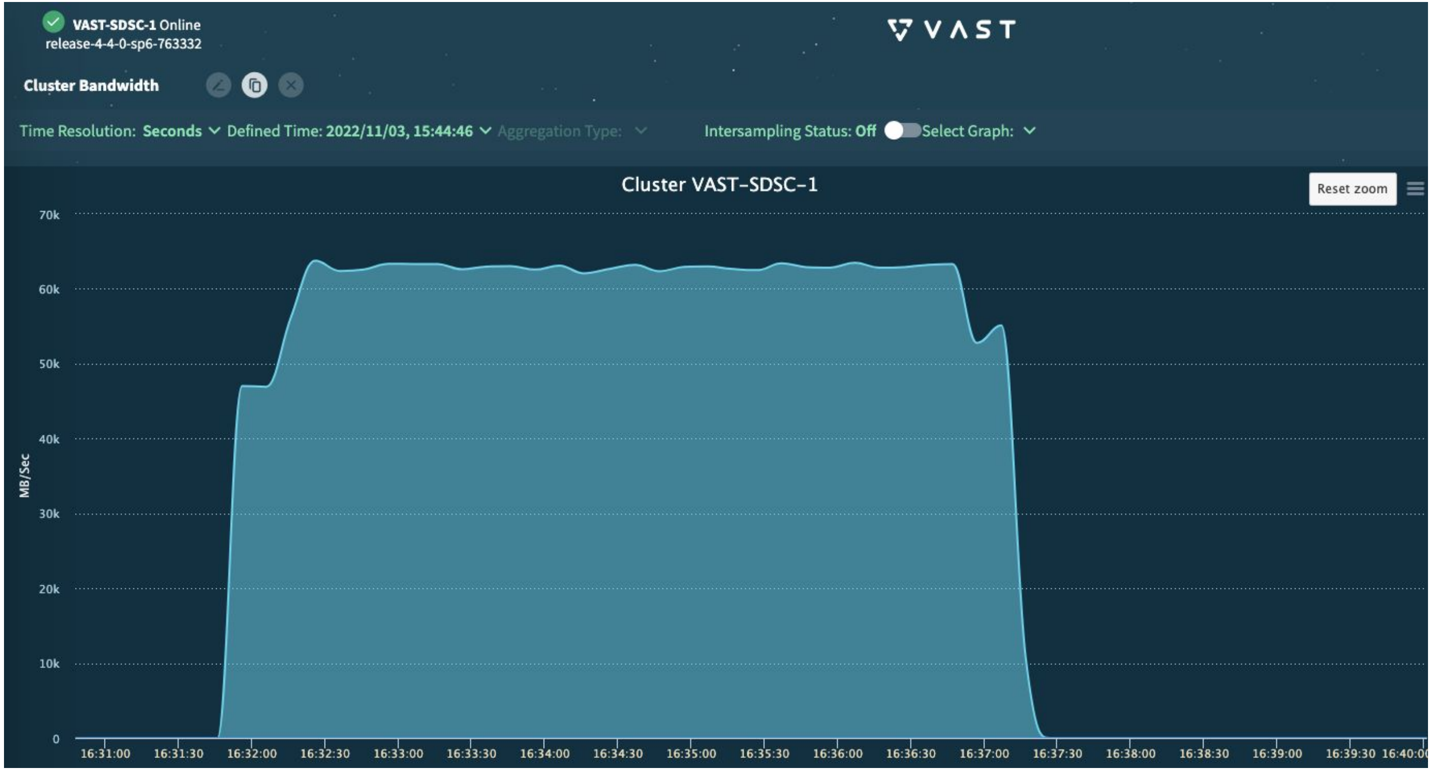
The PofC was done at 10Gbps. In principle this should work at any scale … but it would be nice to show: *“How the future of transfer requests will look”*



400Gbps test status

UCSD Storage to DTNs read rate: >480Gbps

480 Gbps



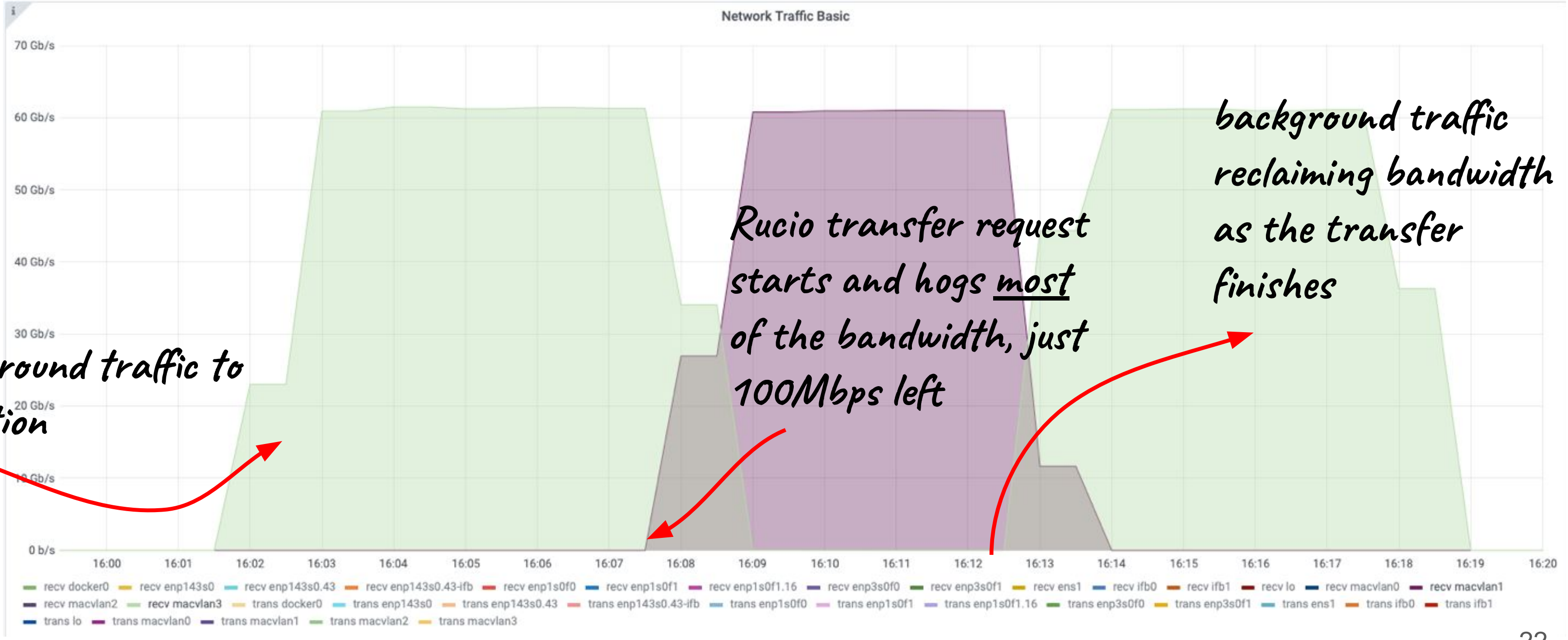
VAST-SDSC-1 Online
release-4-4-0-sp6-763332

Cluster Bandwidth

Time Resolution: Seconds
Defined Time: 2022/11/03, 15:44:46
Aggregation Type:
Intersampling Status: Off
Select Graph:

400Gbps test status (cont'd)

Currently a max of **60Gbps**, still far from the target…



60 Gbps

Artificial background traffic to produce congestion

Rucio transfer request starts and hogs <u>most</u> of the bandwidth, just 100Mbps left

background traffic reclaiming bandwidth as the transfer finishes

Future plans

- Implement monitoring
  - Compare allocated vs achieved bandwidth using DTN network traffic + FTS records
- Add more sites to our testbed
  - Coming soon: Fermilab (T1), Nebraska (T2), Vanderbilt (T2), Sprace (T2)
  - Looking for European sites.
- DMM policy implementation and simulation (More on my talk on Friday)
- Participate as a prototype in the WLCG Data Challenge 2024
- Add support for more NOS (Network Operating Systems) in SiteRM
- DTN-as-a-Service – Auto Start/Stop Transfer Service on Request
- How can we include Sites without network control?

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.

Other Networking Activities

The **R**esearch **N**etworking **T**echnical **W**orking **G**roup (**RNTWG**), was formed in 2020 after the LHCONE/LHCOPN meeting

GOAL: To be able to identify the owner and purpose of any R&E network flow anywhere in the network.

Motivation: The poor experience for WLCG trying to understand network flows, especially across the Atlantic

WHY??: Many reasons:

- It is vital to **understand the sources network congestion** and work with users to better orchestrate.
- **R&E networks want to understand their users** and associated flows and optimize how they are served.
- Science collaborations are often unaware of the negative **impact that tuning or changing their workflows** can have on the wide area network

Other Networking Activities (cont'd)

RNTWG was created to cover 3 areas:

- **Network visibility:**
    - focus on **Packet and Flow Marking**
    - has spawned a new initiative call SciTags
- **Network flow optimization** (not ramped up yet)
    - focus on **traffic pacing** and **protocol optimization**
    - to allow more efficient use of our networks
- **Network orchestration**: GNA-g, NOTED, **SENSE**

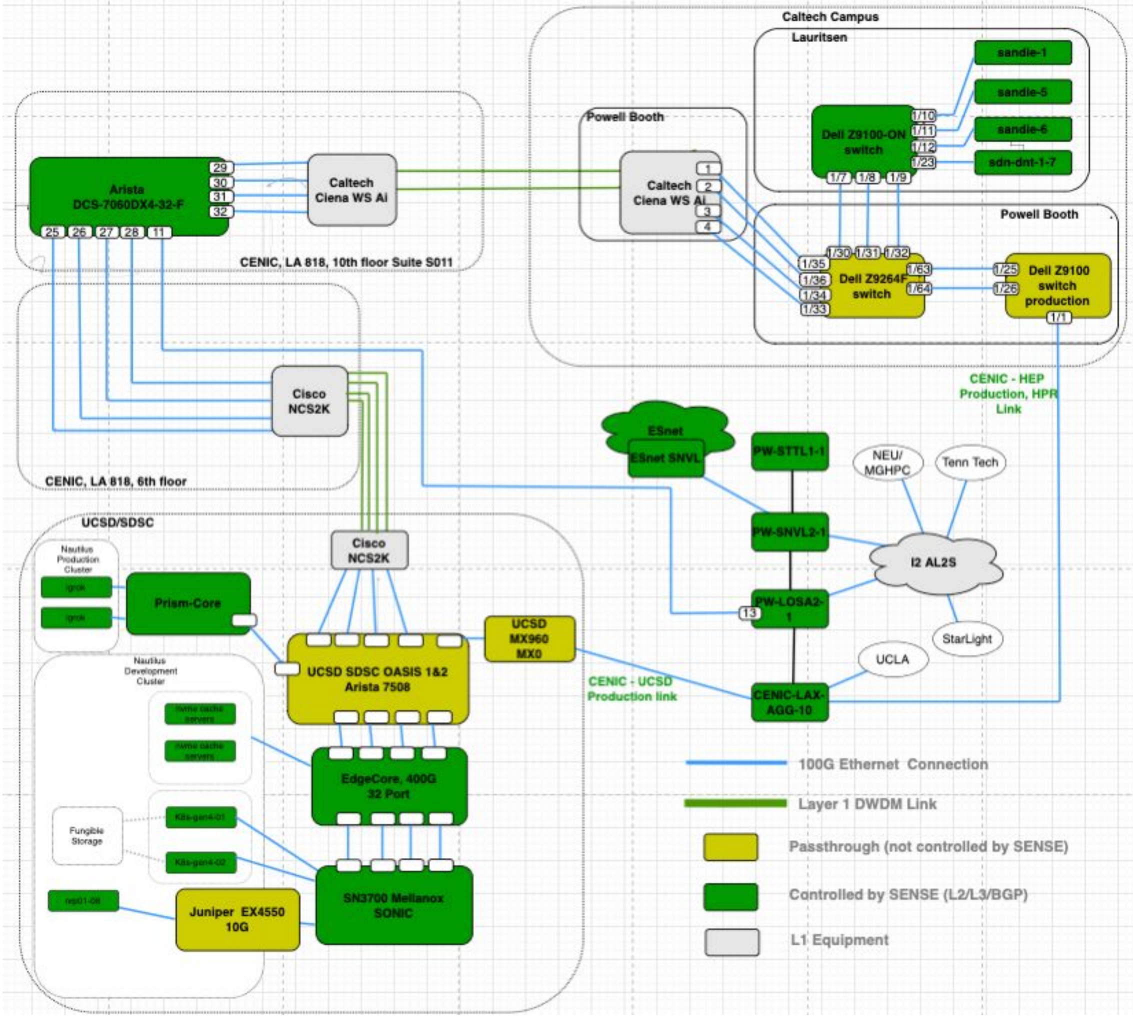
They also note the work of the WLCG Network Throughput working group which deploys, manages and monitors a global perfSONAR infrastructure

More about this on Shawn McKee's presentation on the Rucio workshop:
Network Packet Marking and Flow Labeling: the Technical Details
(Friday at 9:30am)

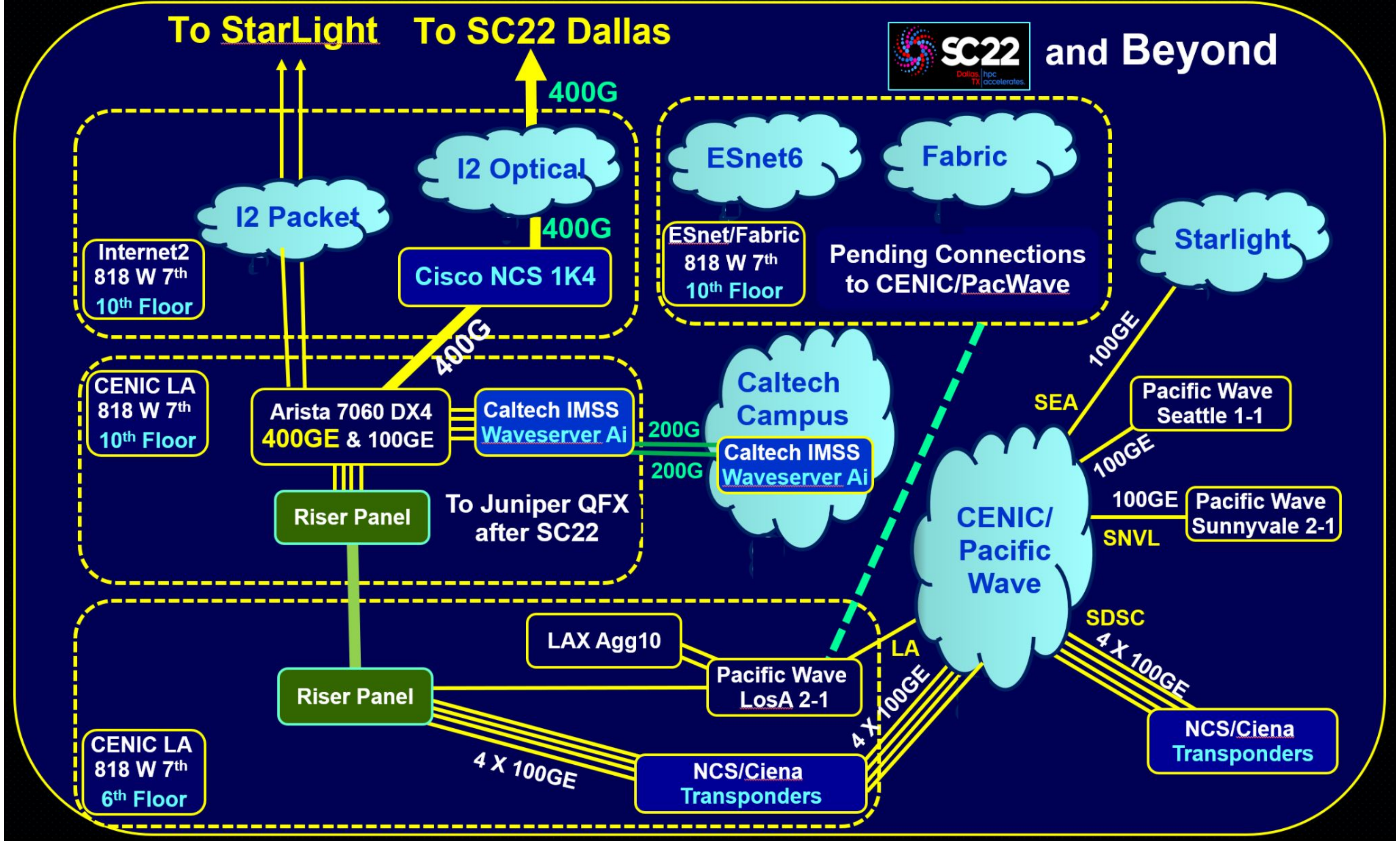
Thank you for listening, questions?

Want to join SENSE
Testbed? Or ask questions?
Drop an email to SENSE
Group: **sense-info@es.net**

Background slides


Caltech Campus
Lauritsen
sandie-1
sandie-5
sandie-6
sdn-dnt-1-7
Dell Z9100-ON switch
1/10
1/11
1/12
1/23
1/7
1/8
1/9
Powell Booth
Caltech Ciena WS Ai
1
2
3
4
Powell Booth
1/30
1/31
1/32
1/35
1/36
1/34
1/33
Dell Z9264F switch
1/63
1/64
1/25
1/26
Dell Z9100 switch production
1/1
Arista DCS-7060DX4-32-F
29
30
31
32
25
26
27
28
11
Caltech Ciena WS Ai
CENIC, LA 818, 10th floor Suite S011
Cisco NCS2K
CENIC, LA 818, 6th floor
CENIC - HEP Production, HPR Link
ESnet
ESnet SNVL
PW-STTL1-1
PW-SNVL2-1
PW-LOSA2-1
13
CENIC-LAX-AGG-10
NEU/ MGHPC
Tenn Tech
I2 AL2S
StarLight
UCLA
UCSD/SDSC
Nautilus Production Cluster
igrok
igrok
Prism-Core
Cisco NCS2K
UCSD SDSC OASIS 1&2 Arista 7508
UCSD MX960 MX0
CENIC - UCSD Production link
Nautilus Development Cluster
nvme cache servers
nvme cache servers
EdgeCore, 400G 32 Port
Fungible Storage
K8s-gen4-01
K8s-gen4-02
nrp01-08
Juniper EX4550 10G
SN3700 Mellanox SONIC
100G Ethernet Connection
Layer 1 DWDM Link
Passthrough (not controlled by SENSE)
Controlled by SENSE (L2/L3/BGP)
L1 Equipment

A New Generation Persistent 400G/100G Super-DMZ: CENIC, Pacific Wave, ESnet, Internet2, Caltech, UCSD, StarLight ++

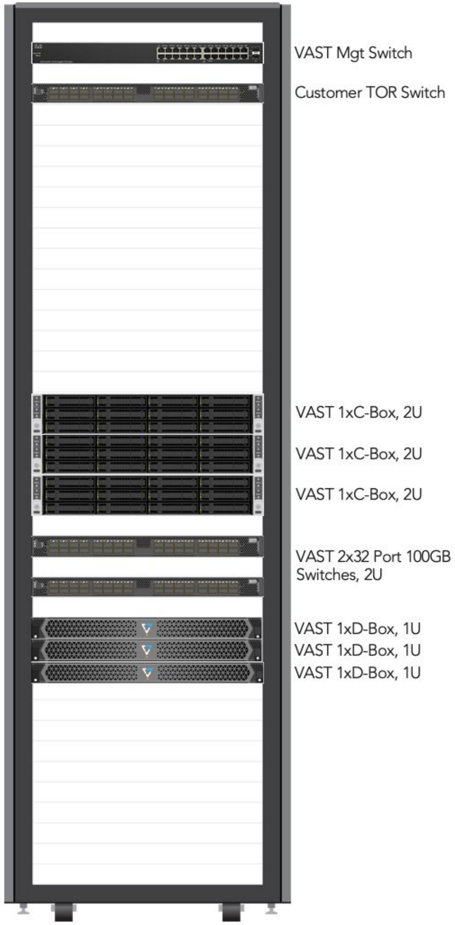


VAST 3X3 Rack Elevation

- 3 C-Boxes
  - 4 Protocol Servers each
- 3 D-Boxes
  - 4 Bluefield-1 DPU accelerators each
  - 22 X 15TB E.1L “ruler” flash SSD each
  - 8 X Kioxia FL-6 Storage Class Memory modules each

System Performance (Throughput)

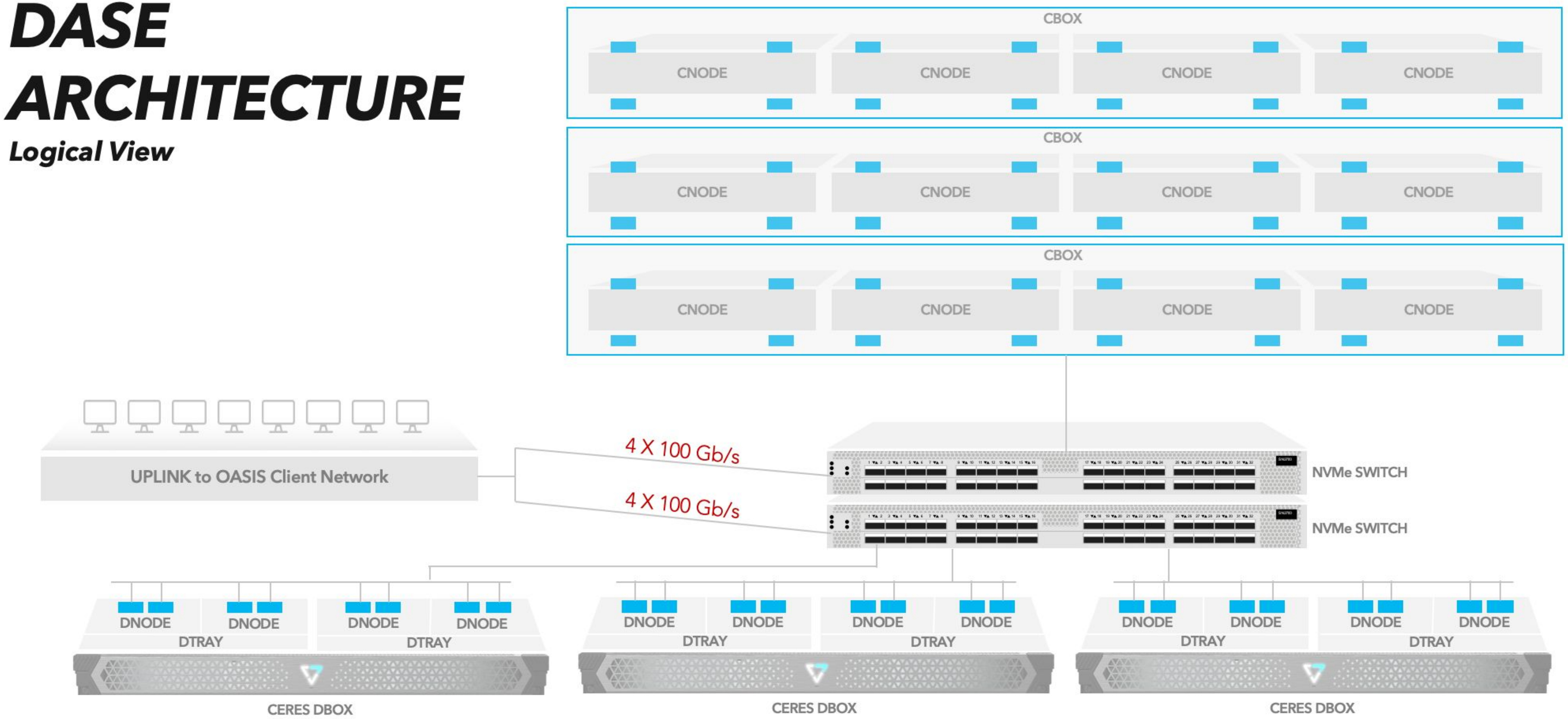
- Random write 12 GB/s
- Random read 120 GB/s
- (Limited by uplinks to ~75GB/s)
- Linear scale-out to grow capacity and performance



Bezel Diagram

DASE ARCHITECTURE

Logical View



VAST