

XCache - XRootd Cache for SoCal

HEPIX Spring 2018

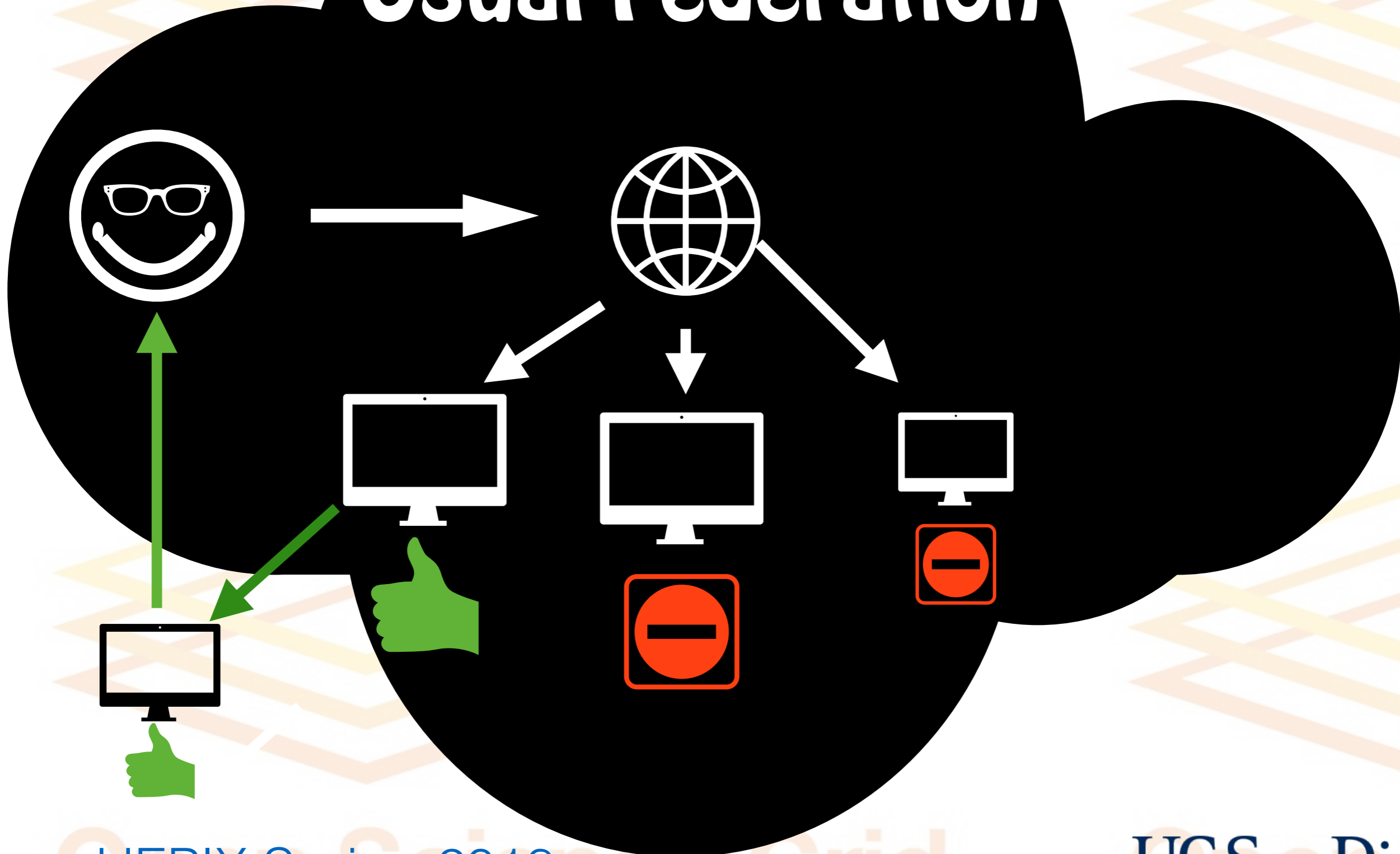
May 16 2018

**Edgar Fajardo (Presenting),
Terrence Martin, Frank Würthwein, Alja Tadel, Matevž
Tadel, Justas Balcas**

Motivation

- Have all the MiniAOD of Run 2 easily available for CMS users in SoCal
 - First step on potentially merging UCSD and Caltech's namespace.
 - Profit from the PRP 100 Gbit connection and 3ms latency between sites.

Usual Federation



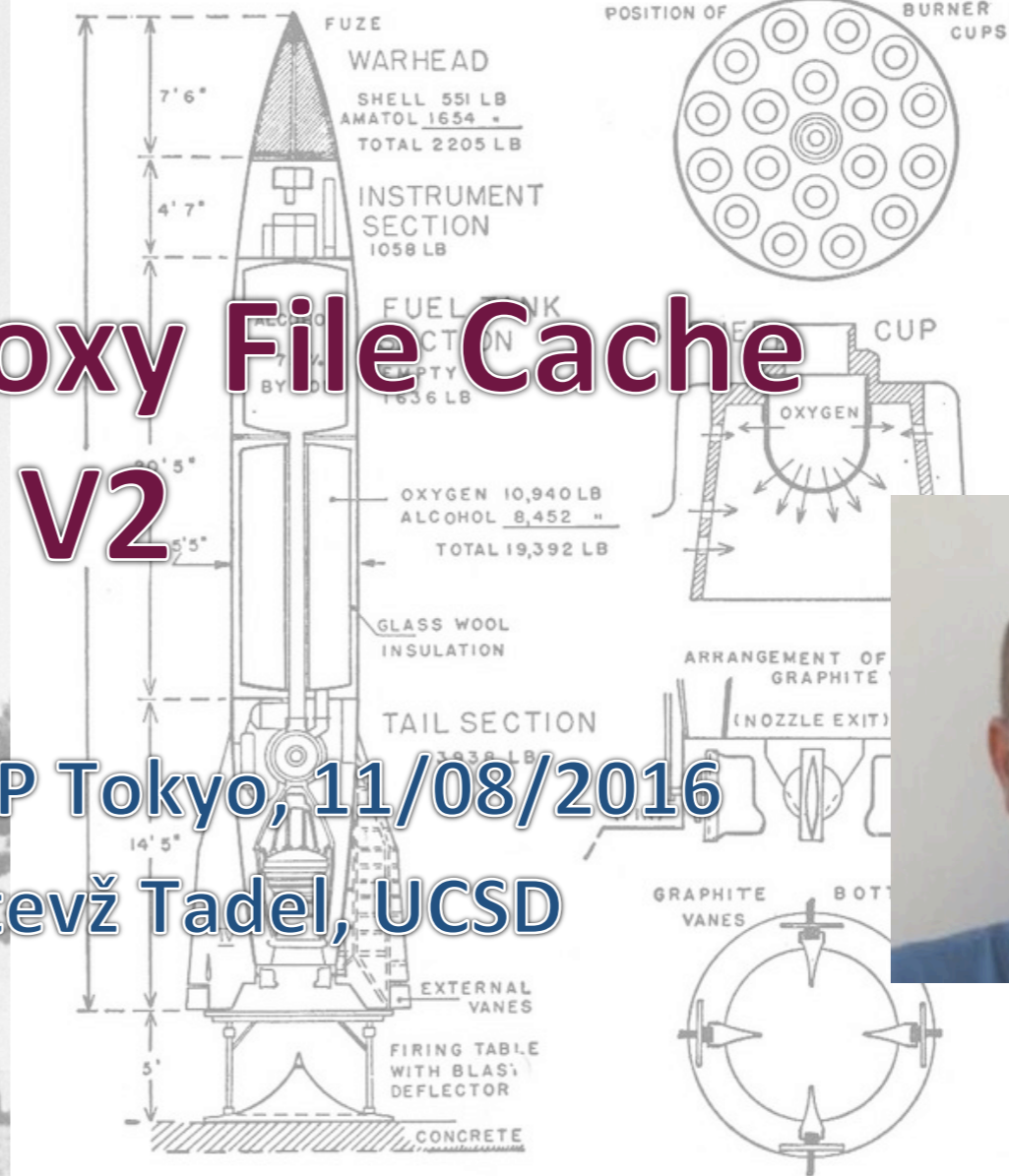
The idea of an Xrootd cache cluster was presented on Xrootd Tokyo meeting

XRootd Proxy File Cache

V2

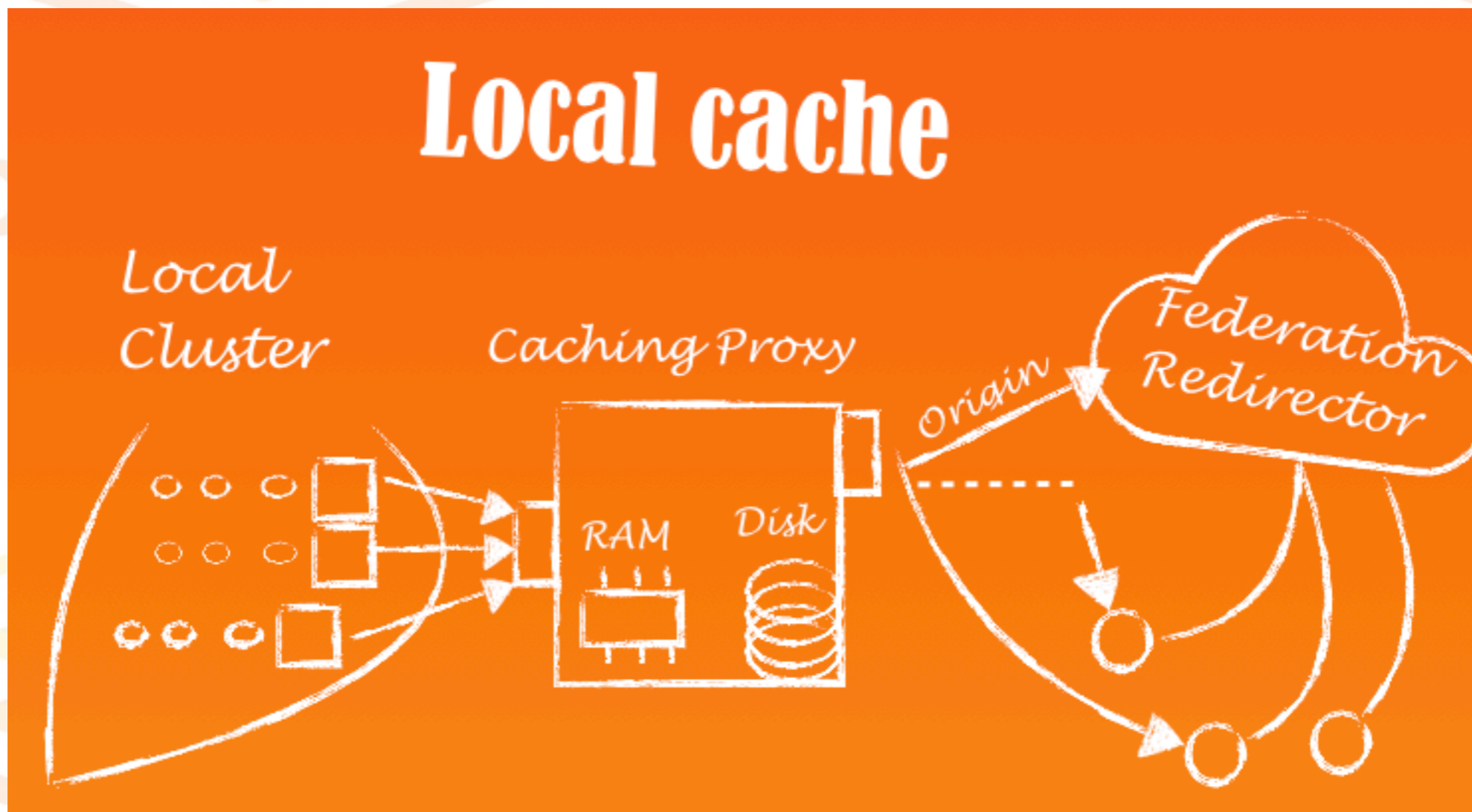
XRootd @ ICEPP Tokyo, 11/08/2016

Alja & Matevž Tadel, UCSD



Full talk available: [here](#)

Xrootd Local Cache



On ACAT 2017 the Xrootd cache cluster was scale tested. Full paper [here](#).

An Xrootd proxy cluster

Edgar Fajardo, Alja Tadel, Matevž Tadel, Frank Würthwein, Ben Steer, Terrence Martin

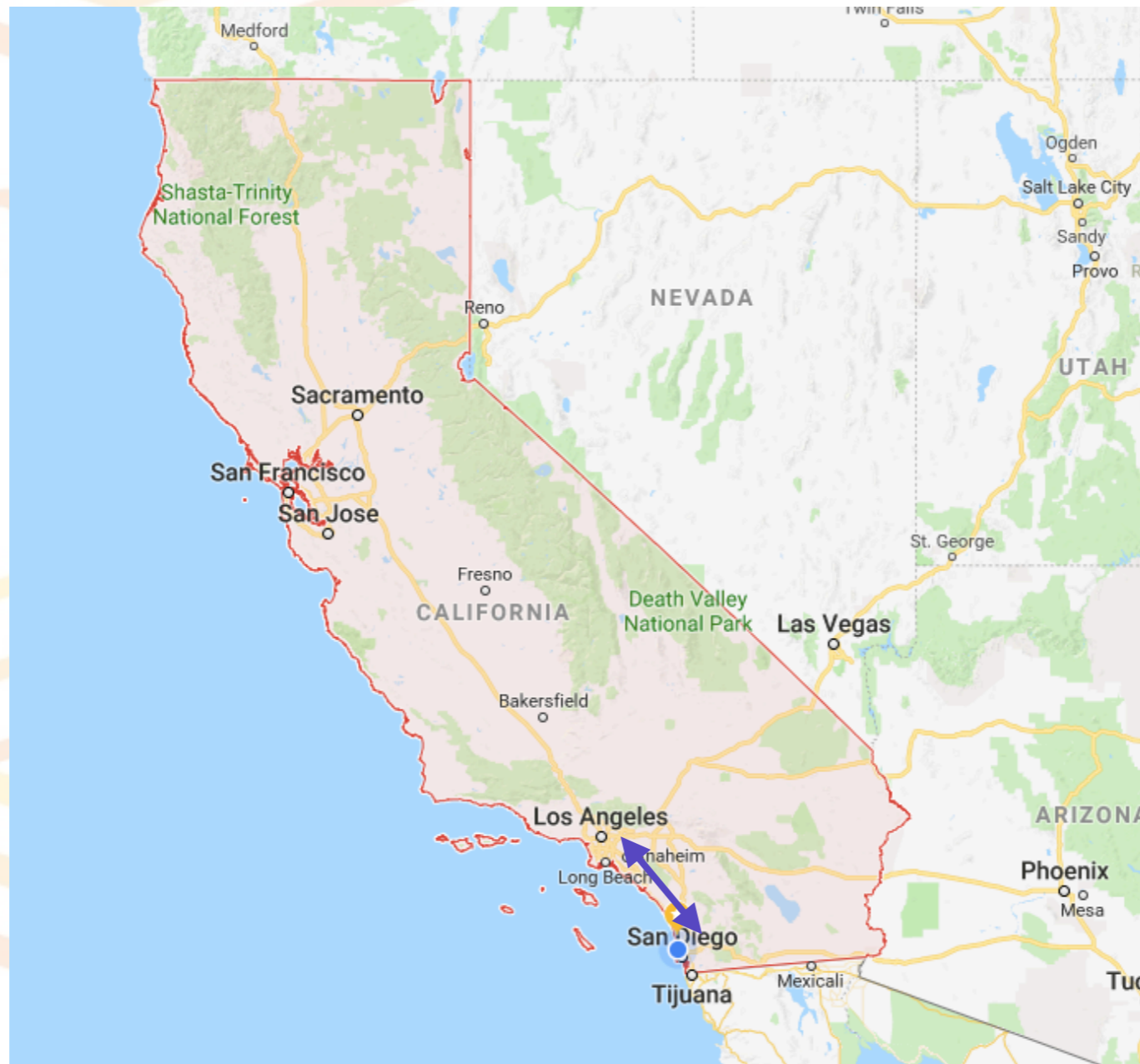


ACAT 2017

21-25 August 2017

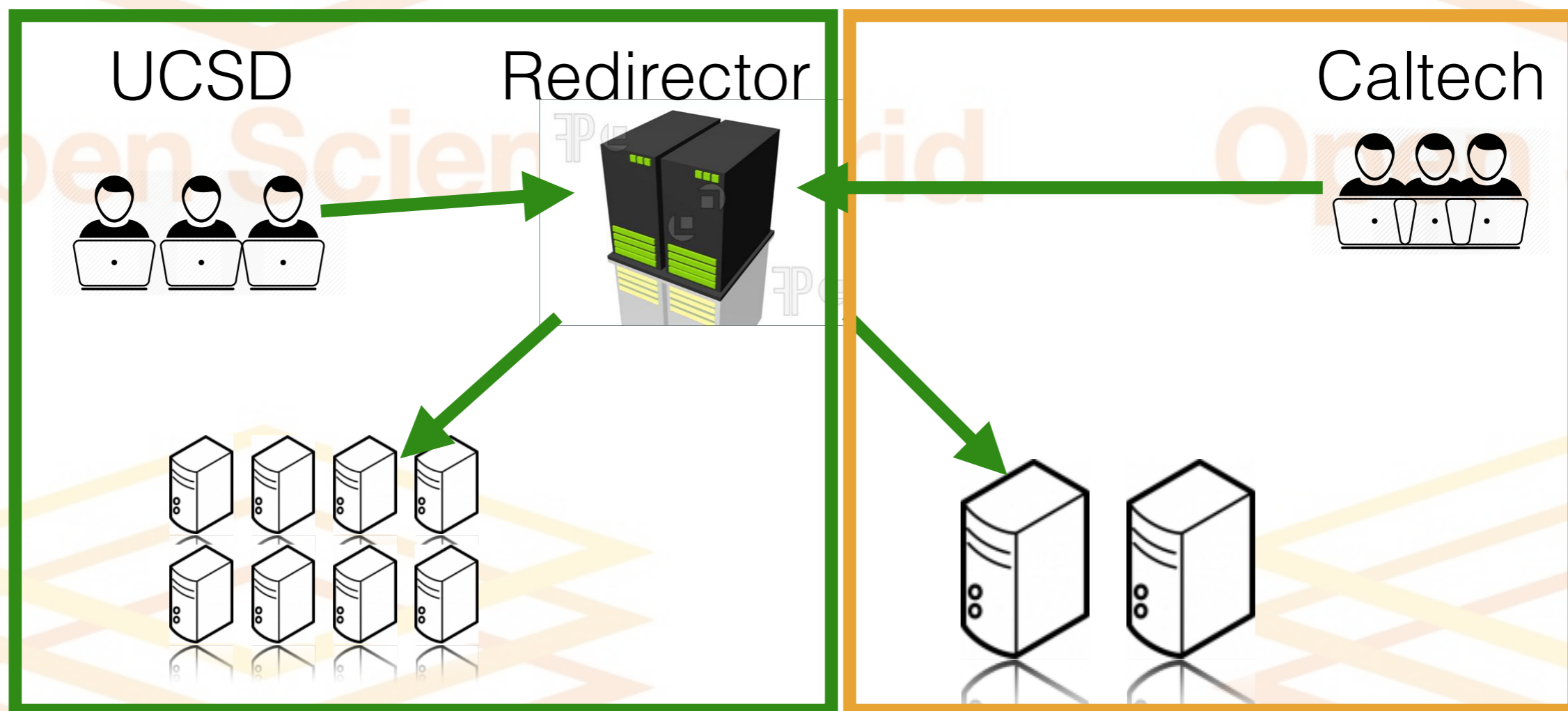
University of Washington, Seattle

Two caches become one



- 120 miles
- 100 Gbit/sec
- 3ms

SoCal Xrootd Cache (2018)



Jobs at UCSD and Caltech transparently use the cache

SoCal Xrootd Cache specs

	UCSD	Caltech
Nodes	11 (10 more coming)	2
Disk Capacity node	12x2TB = 24TB.	30 x 6TB disks (HGST Ultrastar 7K6000)
Network Card	10 Gbps	40 Gbps
Total Capacity	264 TB	360 TB

Space Needs

Datasets	Size (TB)
<code>/*/Run2016*-03Feb2017*/MINIAOD</code>	182.8
<code>/*/RunIISummer16MiniAODv2-PUMoriond17_80X_*/MINIAODSIM</code>	502.5
<code>/*/*RunIIFall17MiniAODv2*/MINIAODSIM</code>	211
<code>/*/*-31Mar2018*/MINIAOD</code>	137.9
Total	1041

Scaling tests

- We had a Monash University student (Ben Steer) that performed some scale testing on the UCSD cache.
- Thorough results of the scale tests can be found [here](#)
- The most important aspects are in the following slides
- Tests were revisited for HEPIX on spring 2018.
- UCSD student (Caitlin Hung) performed acceptance testing on the Caltech side.

Scaling Tests Description

- Baseline test used jobs reading at twice the CMS job usual spec 2MB/sec.
- We ran at different scales 1000/2000/3000/4000 jobs.
- Tests run jobs that simulate actual use of the system by clients using tool xrdfragcp (like xrdcp but controlling read rate)

How to test it?

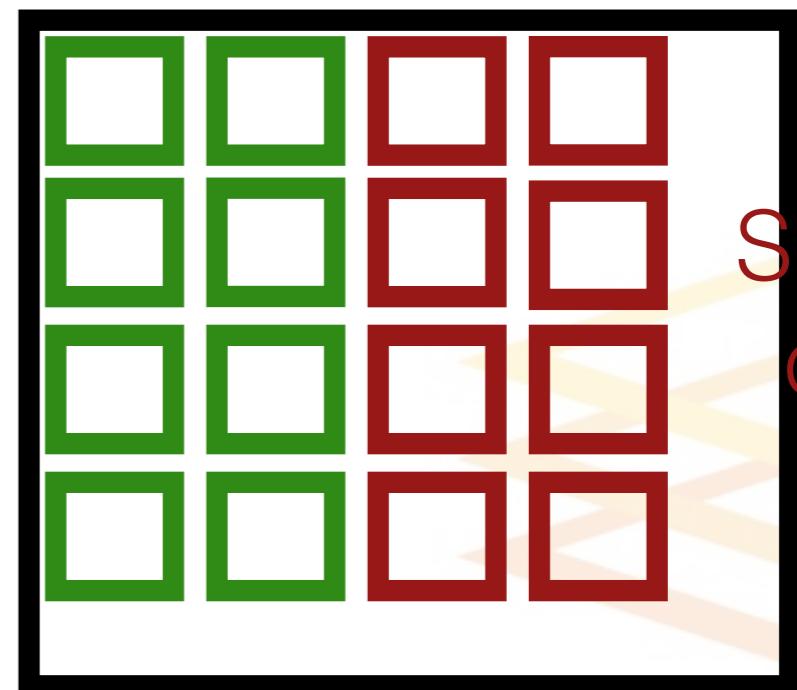
Use a sleeper pool

Not like this one:



Real Cores

Like This one:



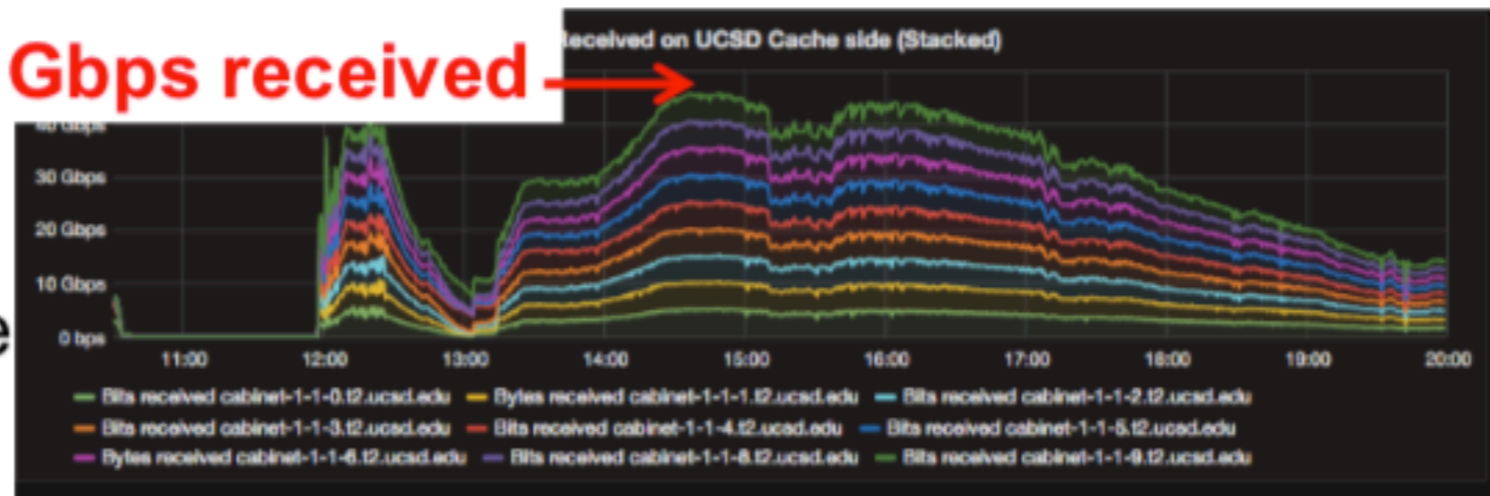
Sleeper
cores

A worker node

Aggregate view of all 9 NICs

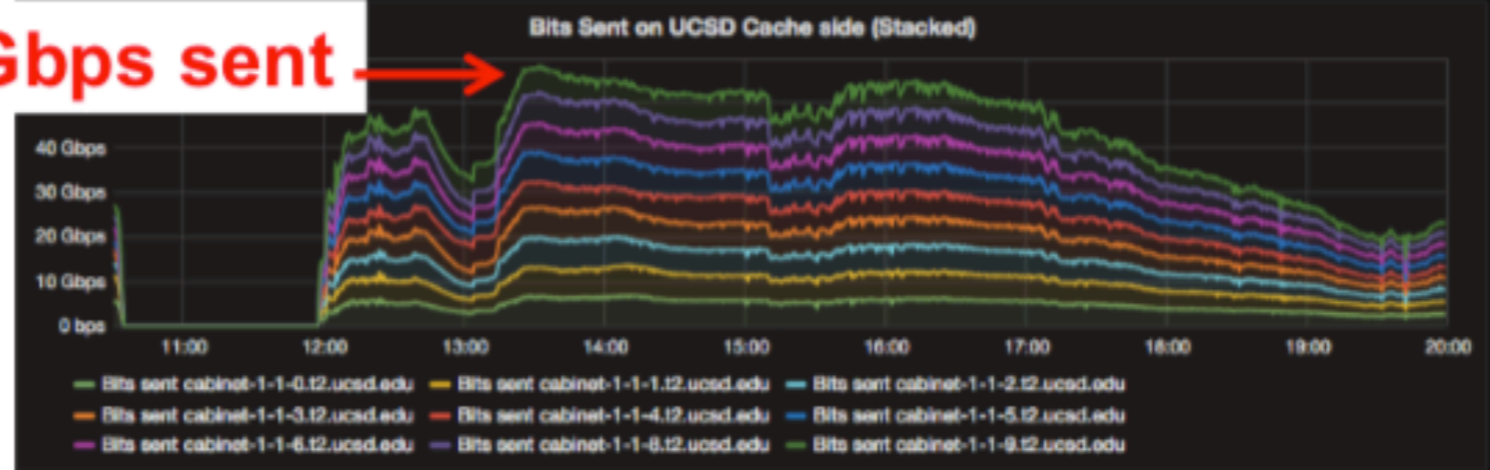
Peak of 45 Gbps received →

Bits received means IO from WAN to cache



Peak of 57.8 Gbps sent →

Bits send means IO from cache to jobs



Bits read from disk cache = Bits send – Bits received

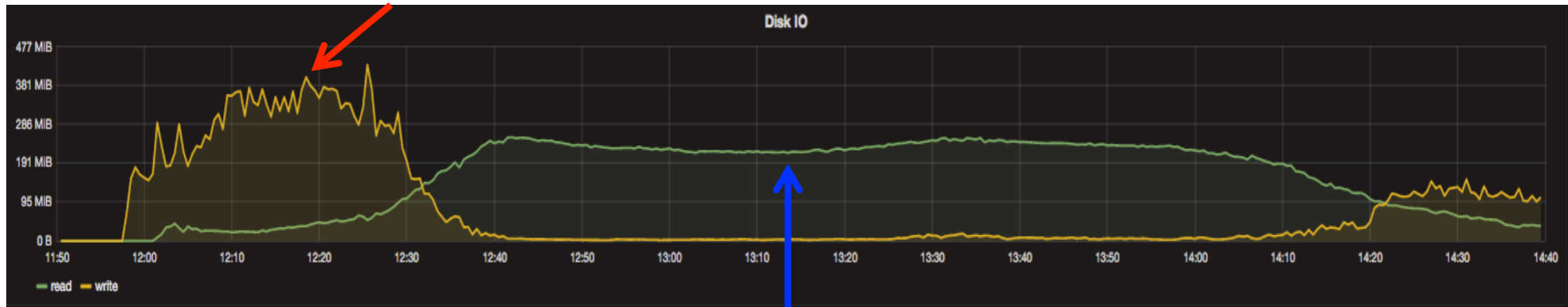
3/20/17

CENIC17

38

Single Server View

write to disk peak at ~3Gbps



Reads from disk pretty steady at ~2Gbps

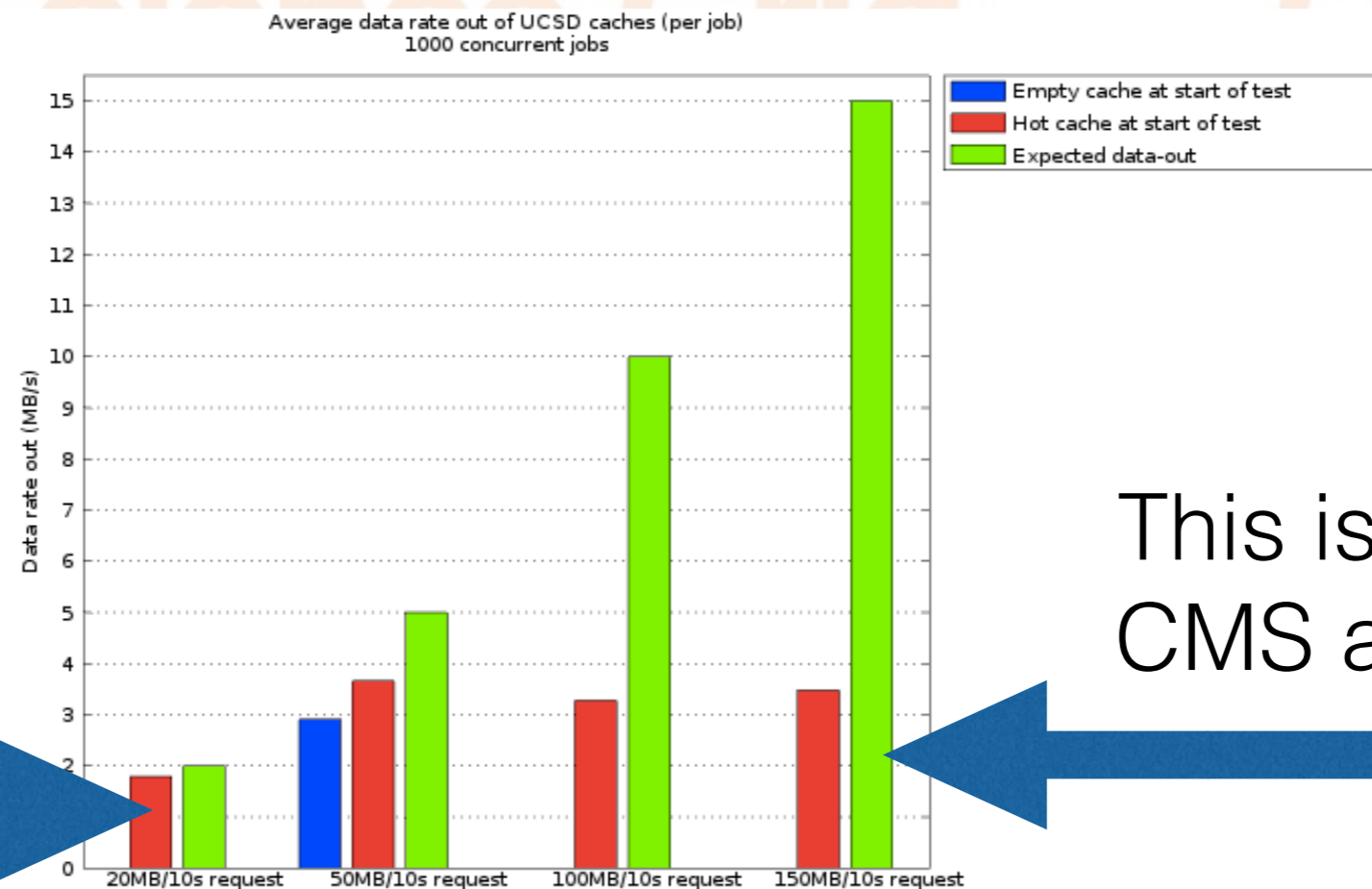
Note: both reads and writes are limited by complicated interplay of cache behavior, hardware performance, and requests from jobs.

3/20/17

CENIC17

40

Scaling tests (UCSD side only)



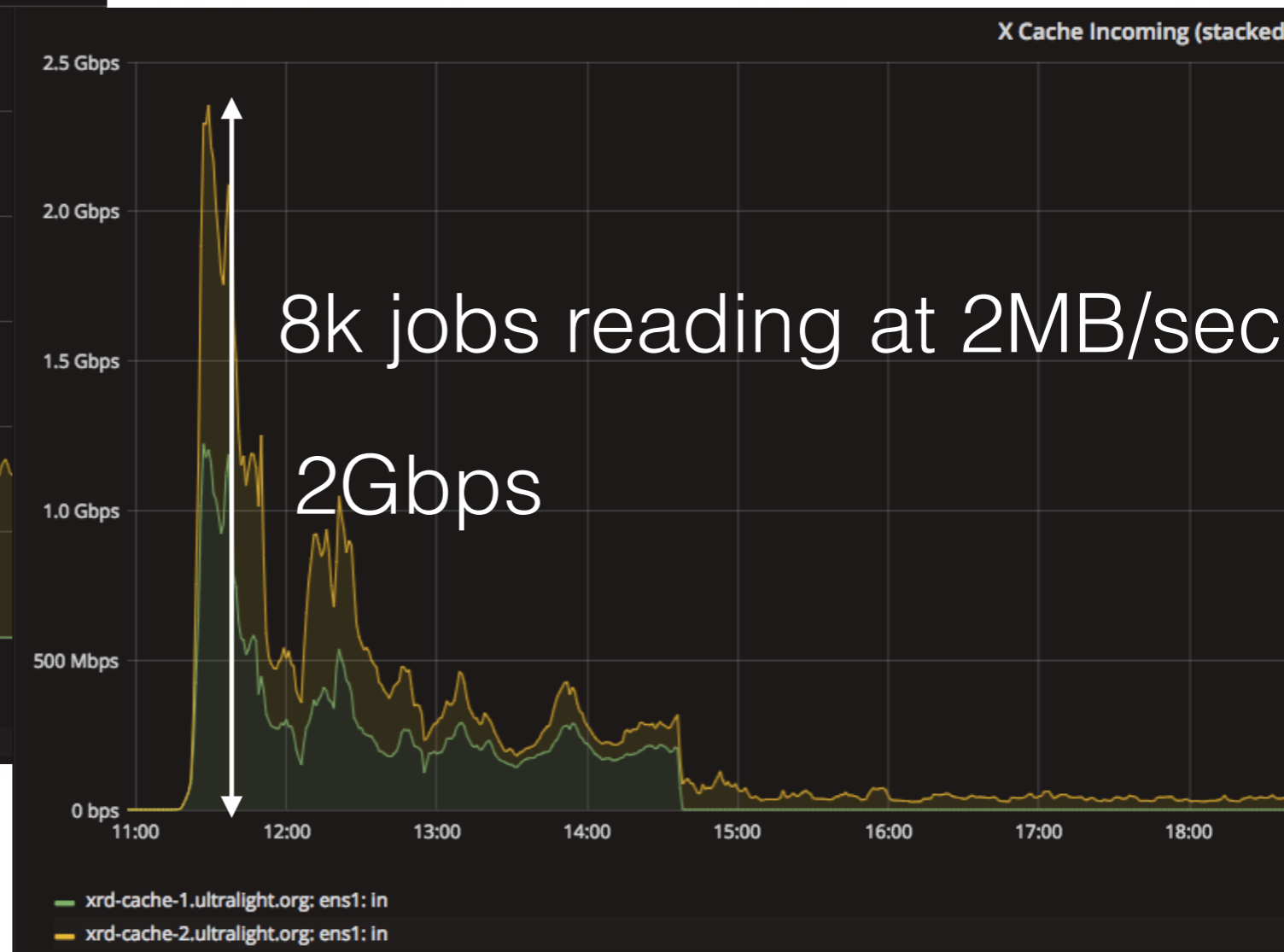
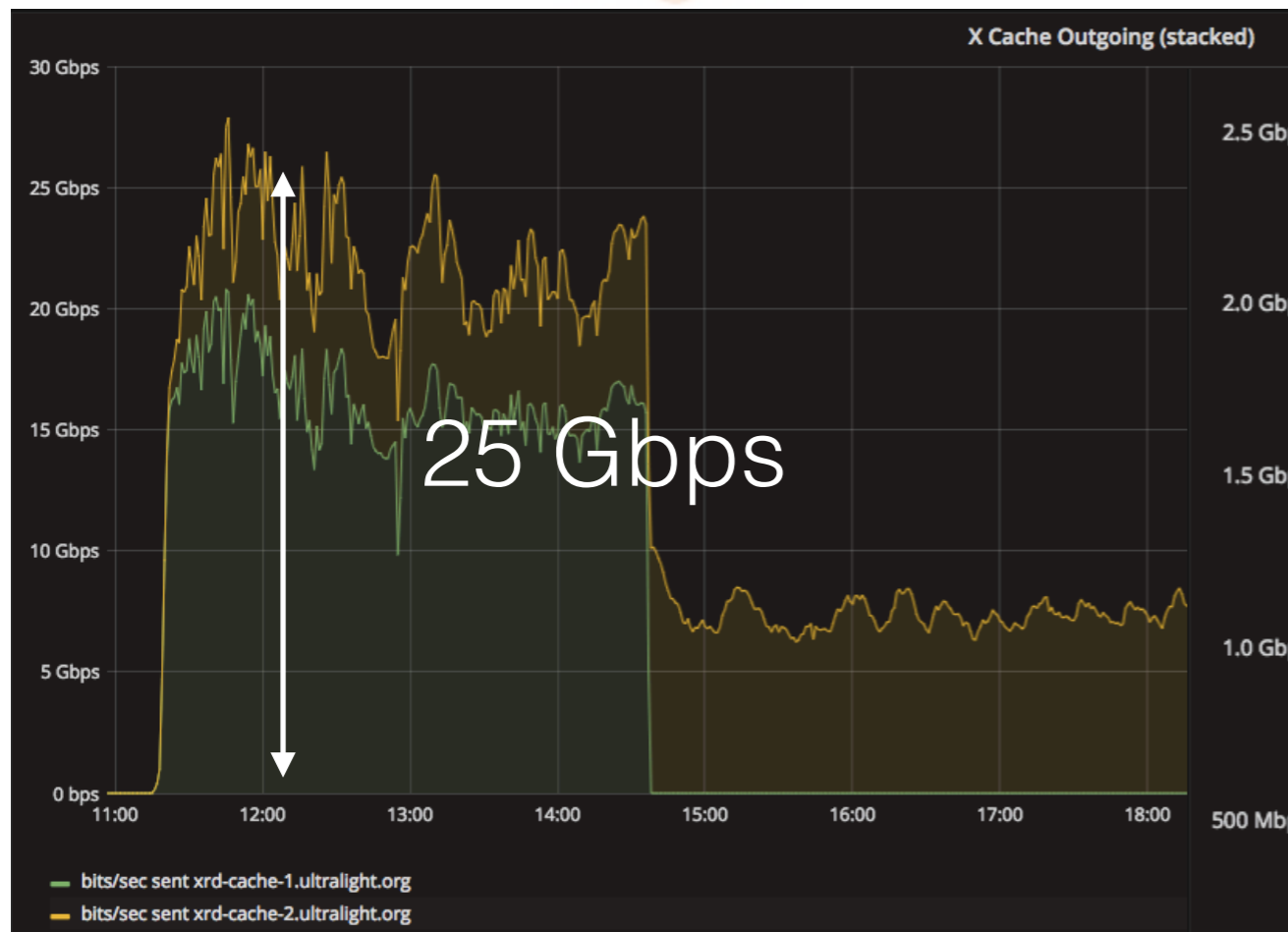
This is two times
CMS needs



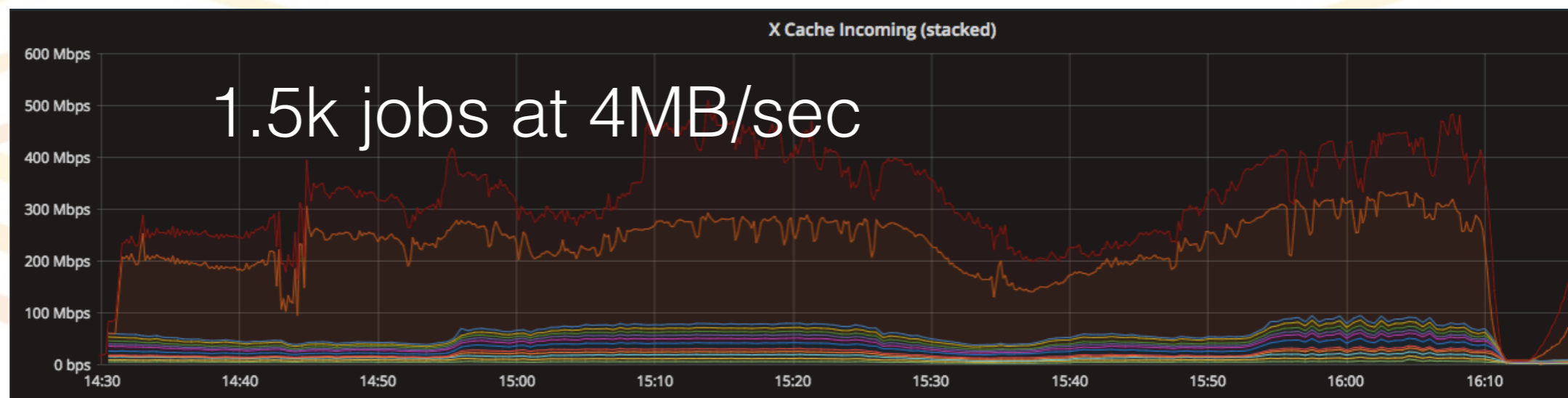
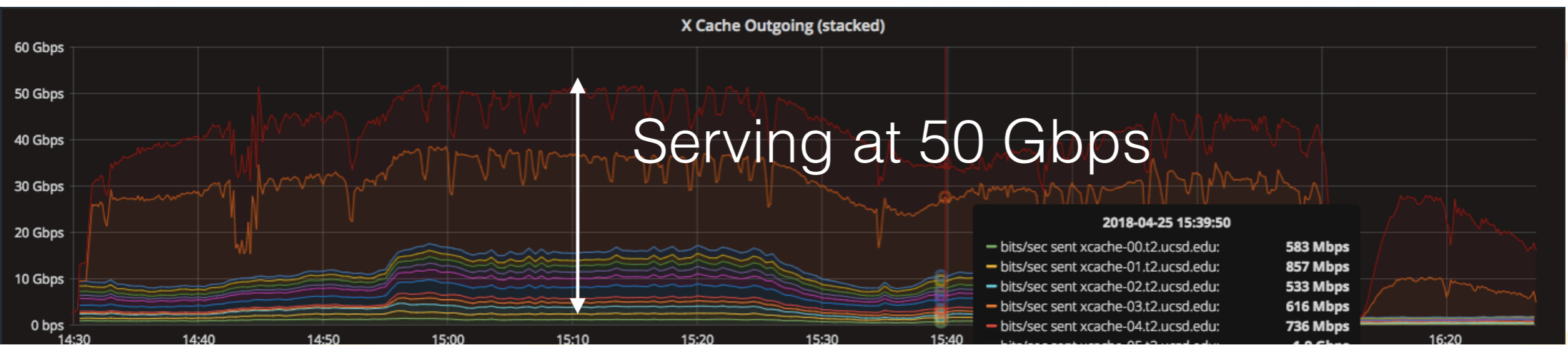
This is 150 times
CMS average job



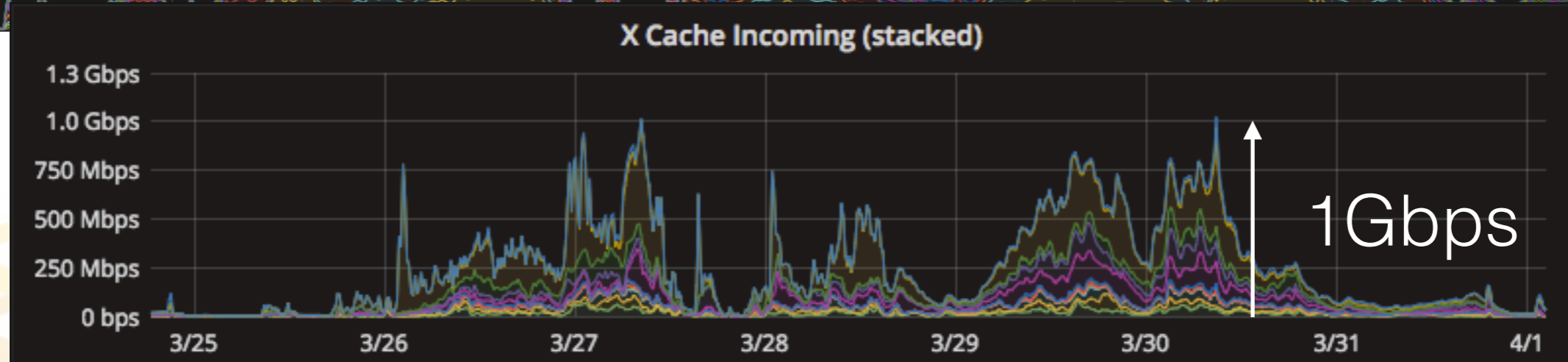
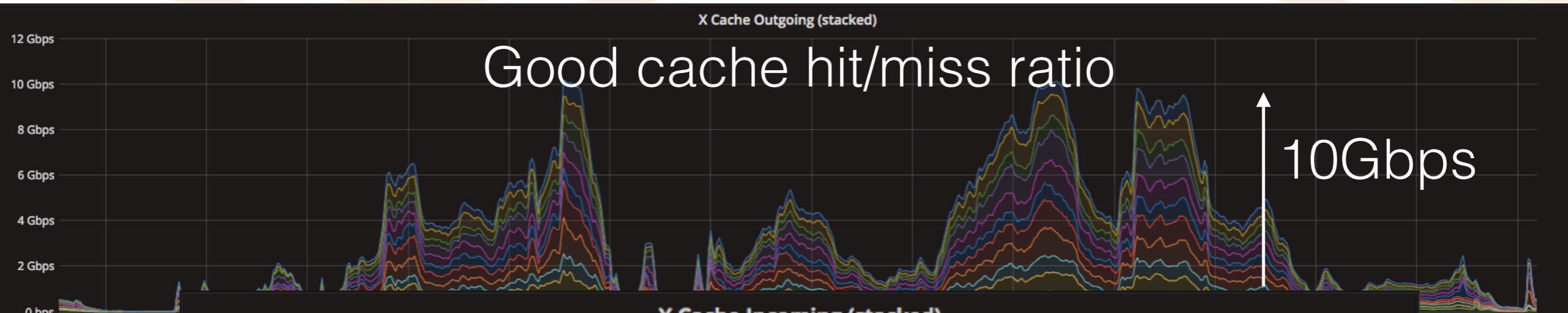
Scaling tests (Caltech Only)



Scaling tests (Both sites)



Production performance



	min	max	avg	current
xcache-00.t2.ucsd.edu: enp4s0: in	39 kbps	77 Mbps	13 Mbps	269 kbps
xcache-01.t2.ucsd.edu: enp4s0: in	39 kbps	84 Mbps	11 Mbps	195 kbps
xcache-02.t2.ucsd.edu: enp4s0: in	40 kbps	86 Mbps	17 Mbps	458 kbps
xcache-03.t2.ucsd.edu: enp4s0: in	40 kbps	23 Mbps	2 Mbps	252 kbps
xcache-04.t2.ucsd.edu: enp4s0: in	40 kbps	35 Mbps	2 Mbps	2 Mbps
xcache-05.t2.ucsd.edu: enp4s0: in	39 kbps	27 Mbps	4 Mbps	161 kbps

How to bring the cache in production

Storage: Rule in storage.xml

```
<lfn-to-pfn protocol="direct" destination-match=".*"  
  path-match="/+store/(data/Run2016[A-Z]/[^/]+/MINIAOD/03Feb2017.*)"  
  result="root://xrootd.t2.ucsd.edu:2050//store/$1"/>  
<lfn-to-pfn protocol="direct" destination-match=".*"  
  path-match="/+store/(mc/RunIISummer16MiniAODv2/[^/]+/MINIAODSIM/  
PUMoriond17_80X_.*)"  
  result="root://xrootd.t2.ucsd.edu:2050//store/$1"/>
```

Computing: New group in frontend

```
<group name="overflow-xcache-socal" enabled="True"><start_expr='(regexp("/*/Run2016.*-03Feb2017.*"/MINIAOD", DESIRED_CMSDataset) || regexp("/*/RunIISummer16MiniAODv2-PUMoriond17_80X_.*"/MINIAODSIM", DESIRED_CMSDataset))>
```

Future Work

- Having other Tier3's in SoCal use the cache: i.e
T3_US_UCR
- Going out of California: Colorado
- What about NorCal? T3_US_UCD
- Basically doing large scale latency tests.
- Learning how to operate a collection of Caches at all US
Tier 2's, which polices? each site own policy?
- Grow cache size on the fly with kubernetes (We have
some students that will work on this over the summer).

Questions?

Contact us at:

1-900-Xrootd-Cache-Masters

Just Kidding

Contact us:

emfajard@ucsd.edu

Thank You