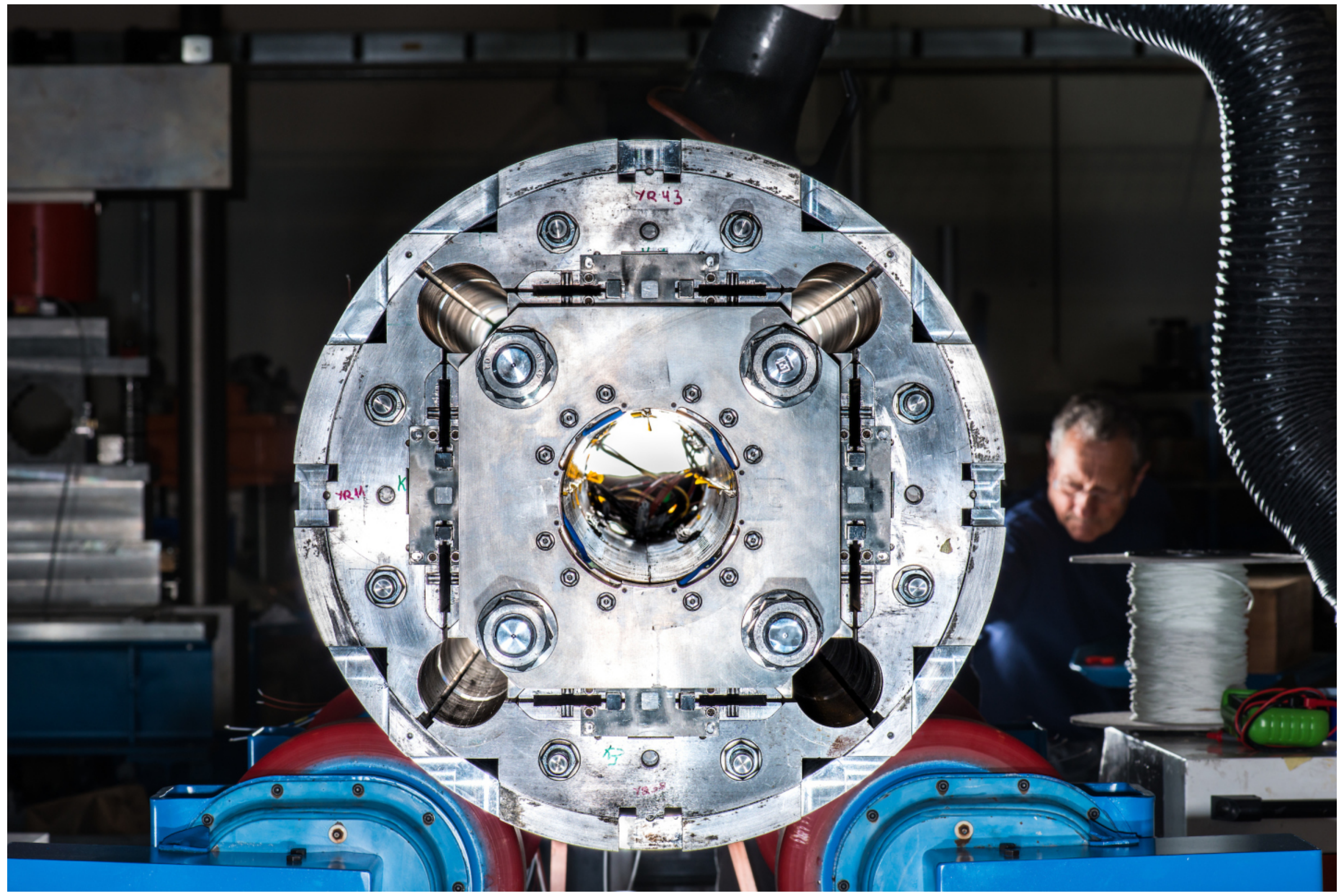
# 400Gbps Benchmark of XRootD-HTTP third-party-copy Transfers

Aashay Arora, Jonathan Guiang, Diego Davila, Frank Würthwein, Justas Balcas, Harvey Newman

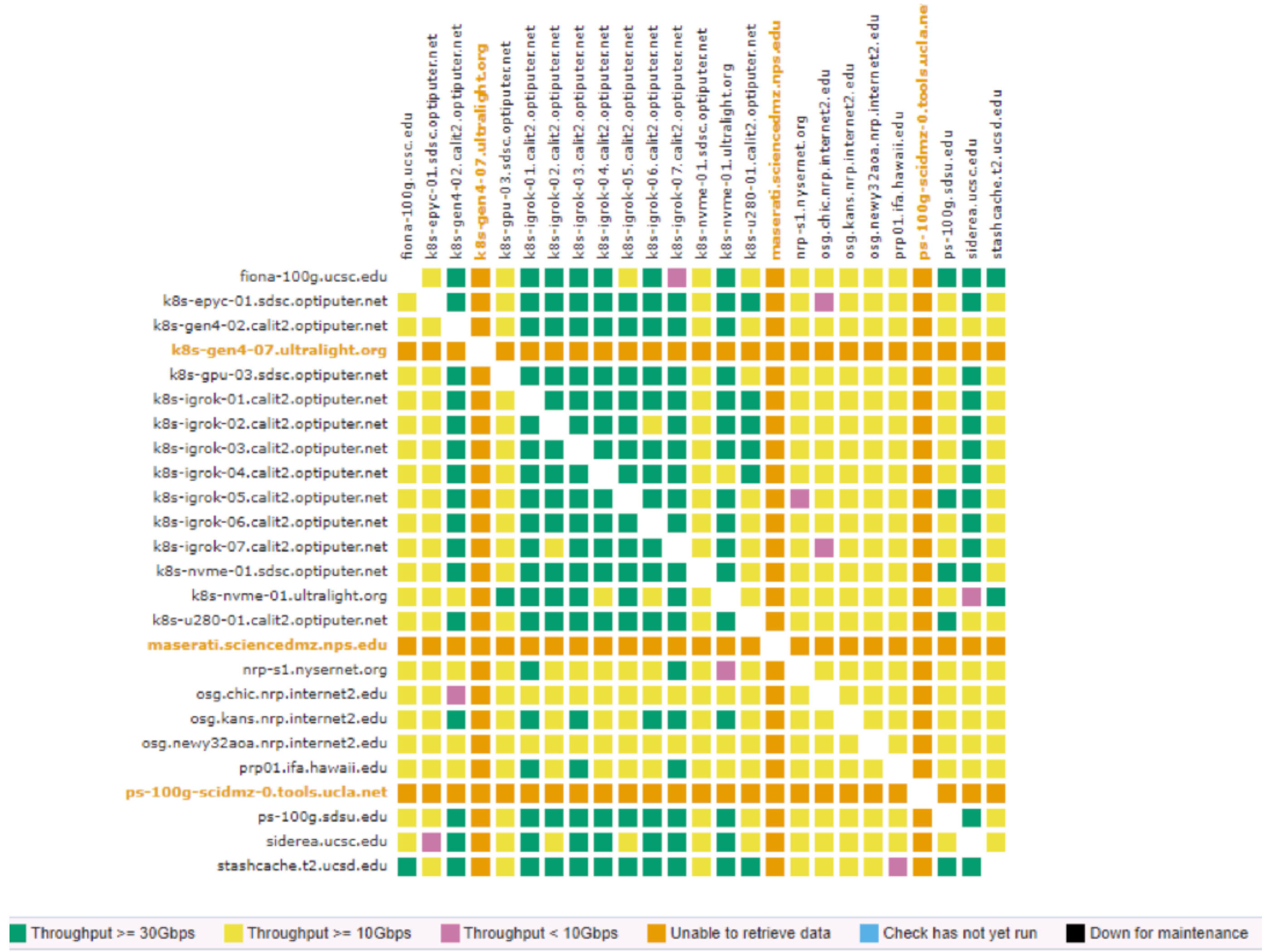
# Introduction / Motivation

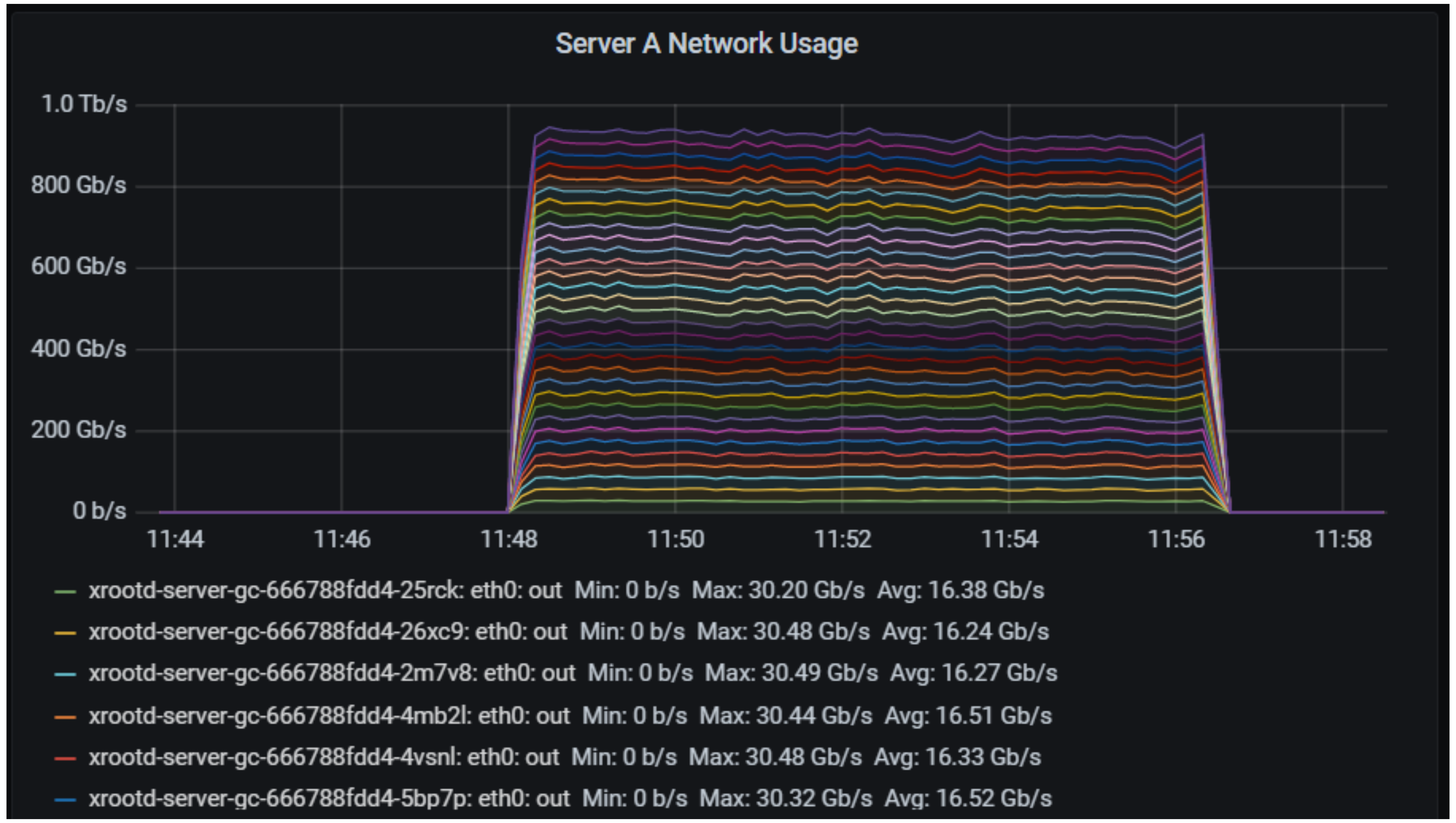


- The High-Luminosity LHC era will bring huge data challenges. We predict, ATLAS and CMS combined will accumulate on the order of an exabyte of raw data every year [1].
    - ESNet quotes (for T2 sites), "*[Throughput] projections for the HL-LHC, with a planned start in 2027, are a 100 Gbps average over the year, with 400 Gbps bursts lasting hours*" [1]
- Therefore, in addition to hardware upgrades, we need to verify the robustness of our software stack to make sure it can support this high throughput.
- HTTP third-party-copy (HTTP-TPC) is now the default for data transfers between LHC sites.
- In the US, all Tier 2 centers support XRootD for data access.
- **What configuration of HTTP-TPC + XRootD give us the throughput we need? What are the minimum hardware requirements to run this configuration?**

# Previous Studies

- We studied the overhead of using XRootD-HTTPS over the globus protocol for data transfers
  - Transferred data over several 100Gbps links using the two protocols with varying RTTs between endpoints to test and compare the throughputs
  - Concluded that XRootD-HTTPS performs slightly better on average over high throughput links. [2]

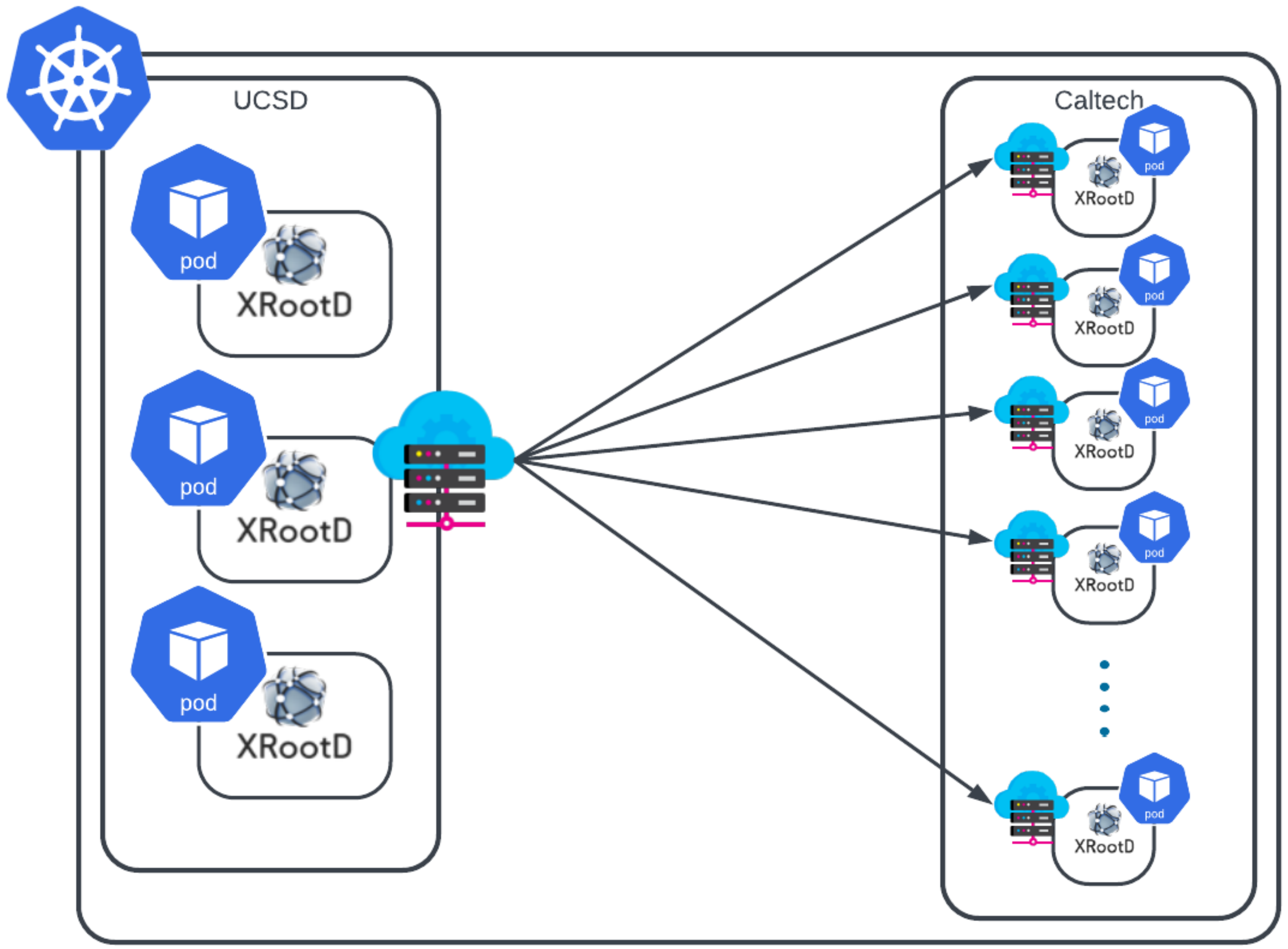


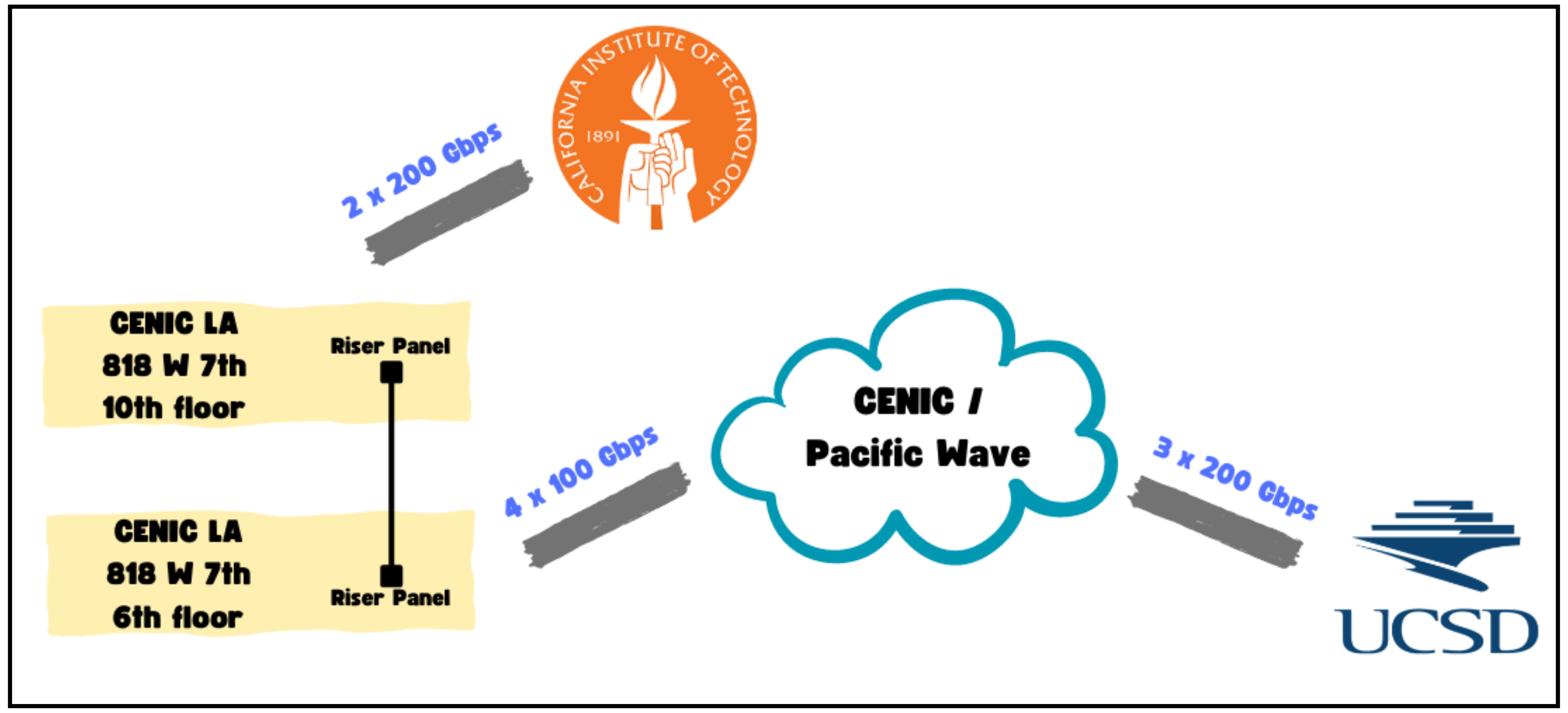




- We benchmarked the performance on empty-links on low latency networks (Microsoft Azure)
  - We were able to get ~ 1 Tbps within the same region
  - Want to test over higher RTT

# Current Hardware Setup

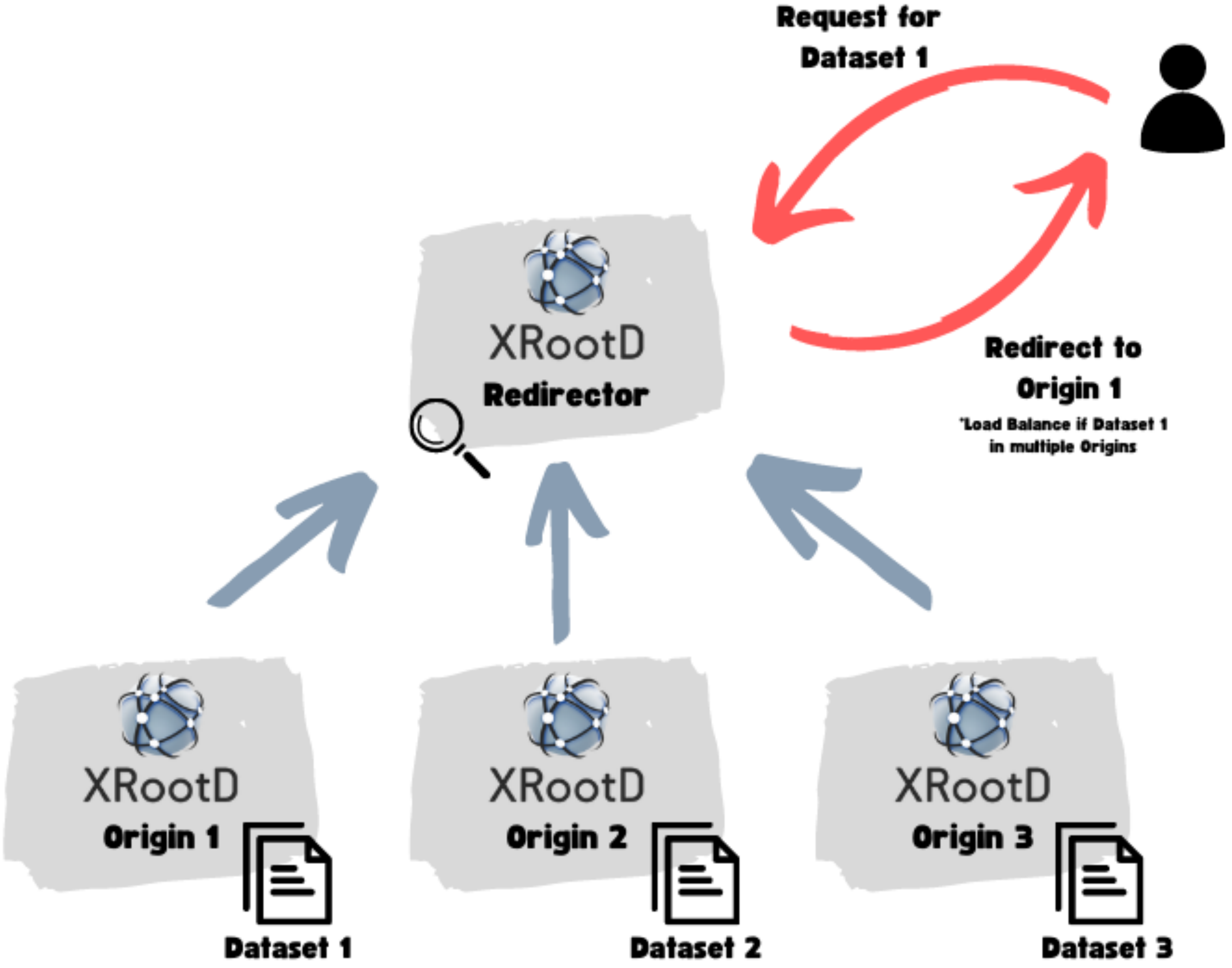
- We have 13 data-transfer-nodes (DTNs) at Caltech (Full Specs in backup)
- 1 **BIG** DTN at UCSD
  - 2 x AMD EPYC 7763 64-Core Processor (with SMT)
  - 2.0 Ti Memory
  - 3 x 200 G Links (ConnectX-6)
- All hosts managed using Kubernetes
- Running 3 pods on UCSD host (each running its own XRootD service and separate interface), and 1 pod on each Caltech Host
- Dedicated network paths provisioned using SENSE-Autogole



# XRootD Configuration

- XRootD configured in a cluster using a non-shared filesystem (each origin has its own set of files)
  - Multiple data origins subscribed to a single redirector.
- Both origins and redirectors configured with the HTTP(S) directive.
  - Authentication using X509
  - Macaroons for delegation and authorization.



# Results

- We can reach 400* Gbps and sustain it for hours!
  - Well, 345 Gbps over a network path capable of doing 350 Gbps.
  - Using 40 streams of 1 GB files for each of the 13 servers with Caltech as sink, i.e. 520 streams coming out of UCSD

```
sense@sn3700:~$ show interfaces counters -i Ethernet4,Ethernet16,Ethernet20,Ethernet124
      IFACE    STATE          RX_OK      RX_BPS    RX_UTIL    RX_ERR    RX_DRP    RX_OVR           TX_OK         TX_BPS    TX_UTIL    TX_ERR
TX_DRP   TX_OVR
-----------  -------  -------------  ----------  ---------  --------  --------  --------  --------------  -------------  ---------  --------
----------  --------
  Ethernet4        U  8,982,229,719  36.80 MB/s      0.29%         0         0         0  60,127,506,940   9958.94 MB/s     79.67%         0
40,750,689         0
 Ethernet16        U  9,002,491,671  38.91 MB/s      0.31%         0         0         0  58,470,633,925  11048.23 MB/s     88.39%         0
24,316,754         0
 Ethernet20        U  8,847,434,599  31.74 MB/s      0.25%         0         0         0  60,157,515,021   9912.32 MB/s     79.30%         0
29,518,679         0
Ethernet124        U  8,845,126,430  36.77 MB/s      0.29%         0         0         0  58,698,987,555  11940.03 MB/s     95.52%         0
26,414,746         0
```

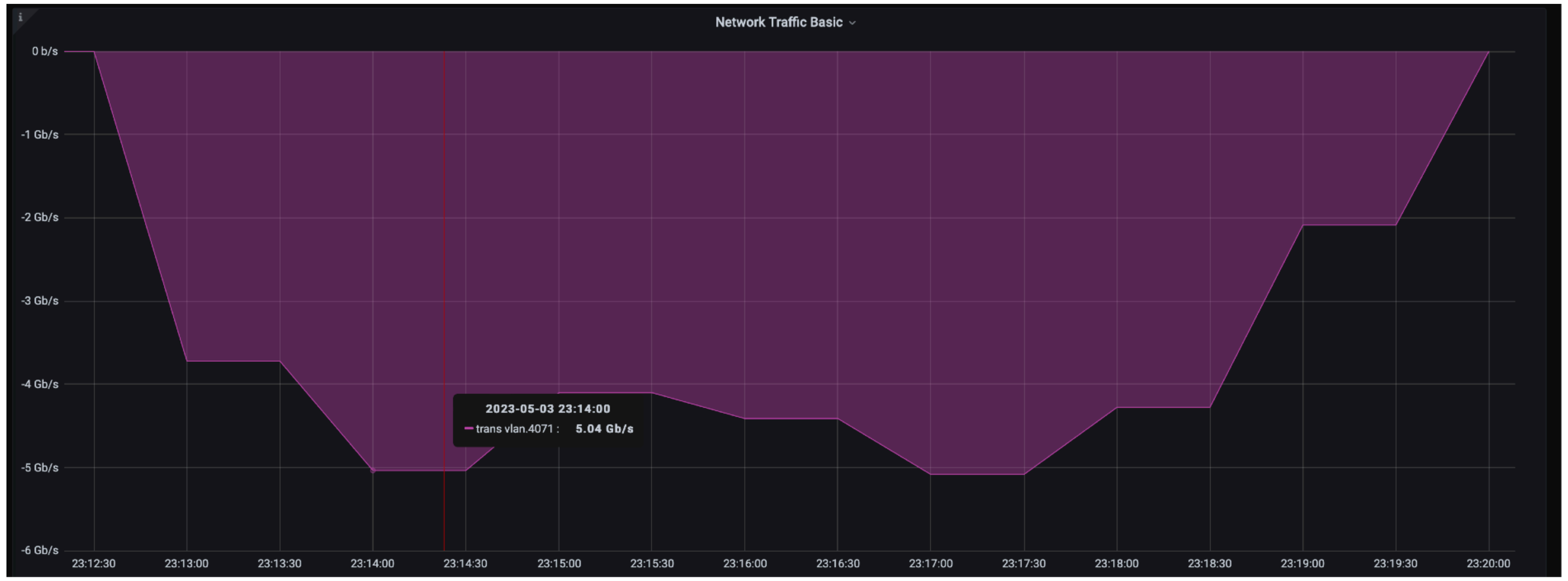
= 345 Gbps

# Interesting Findings

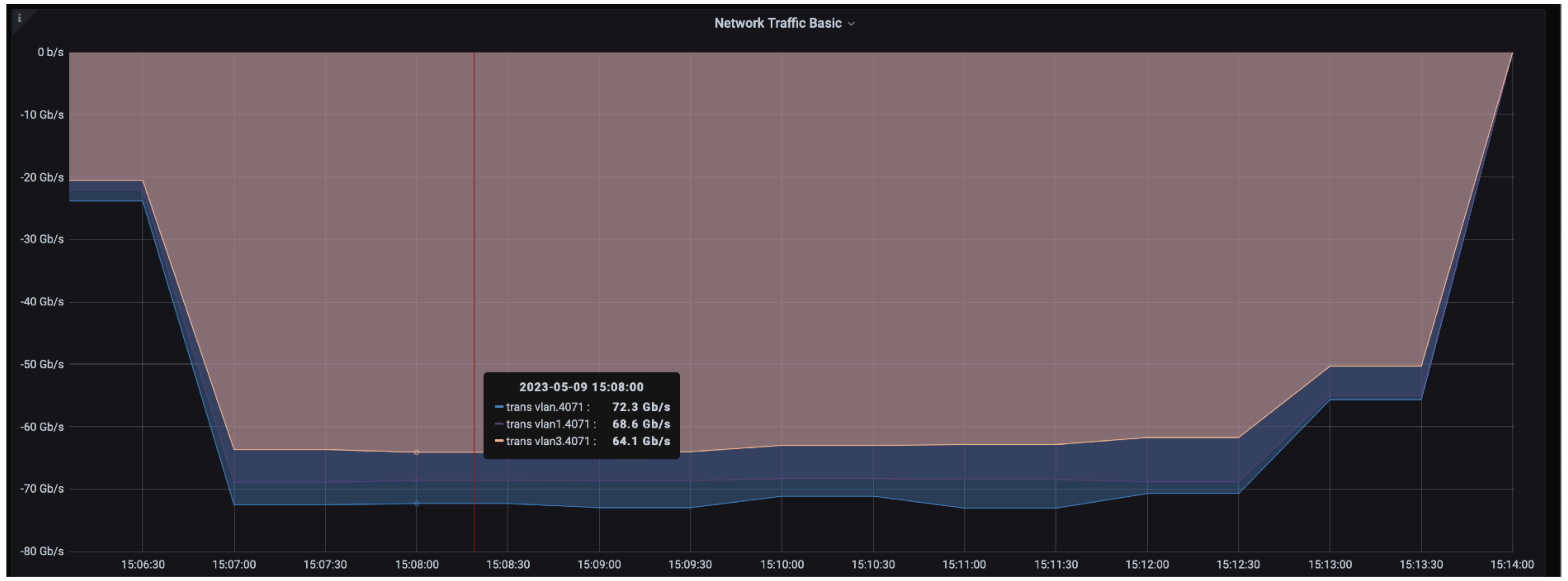
## 1. How does number of streams affect the throughput?

- Not surprising, Drastically! Over the same 200 Gbps capable links

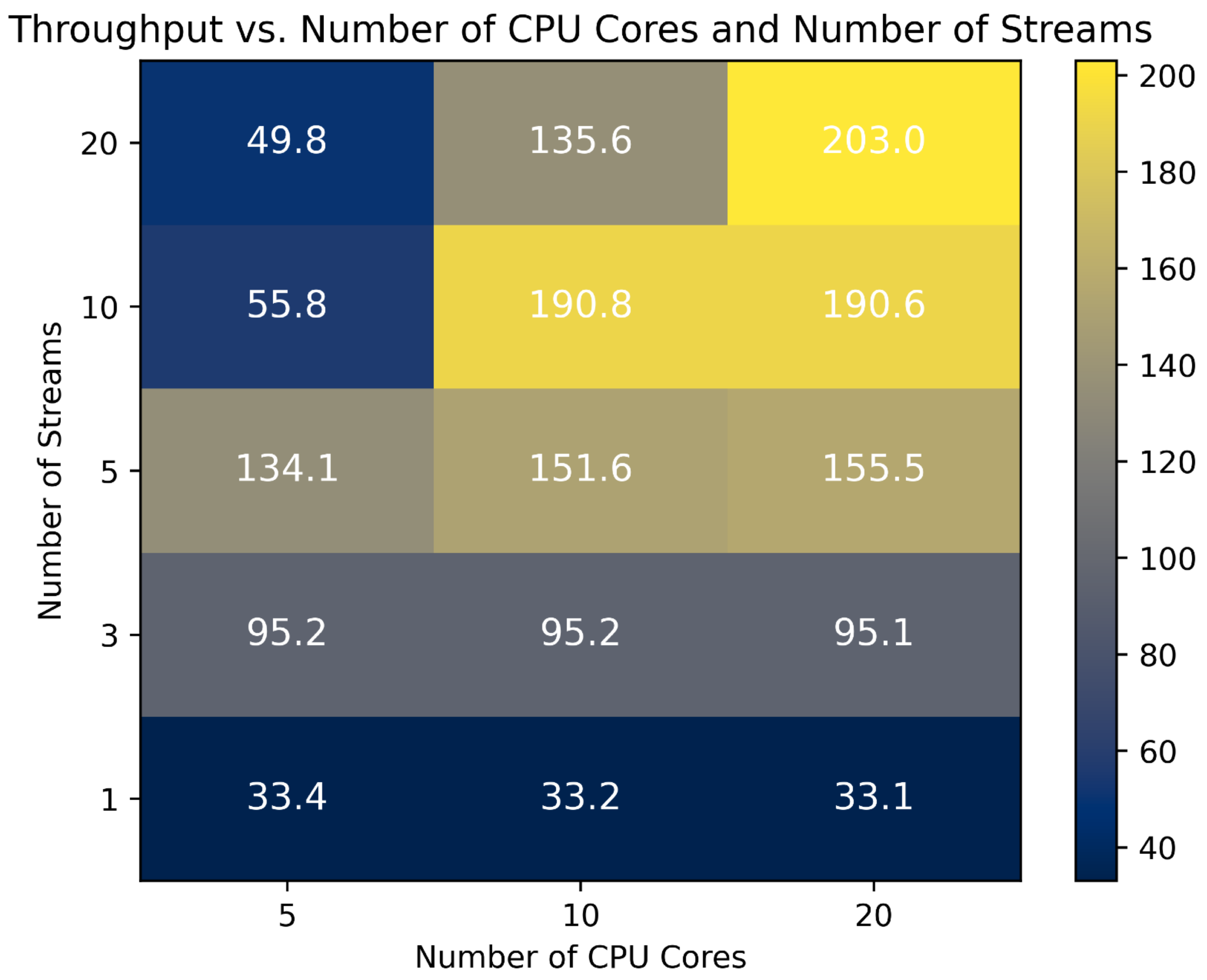
1 100GB file in-flight gives us
**5 Gbps**




200 1GB files in-flight gives us
**200 Gbps**



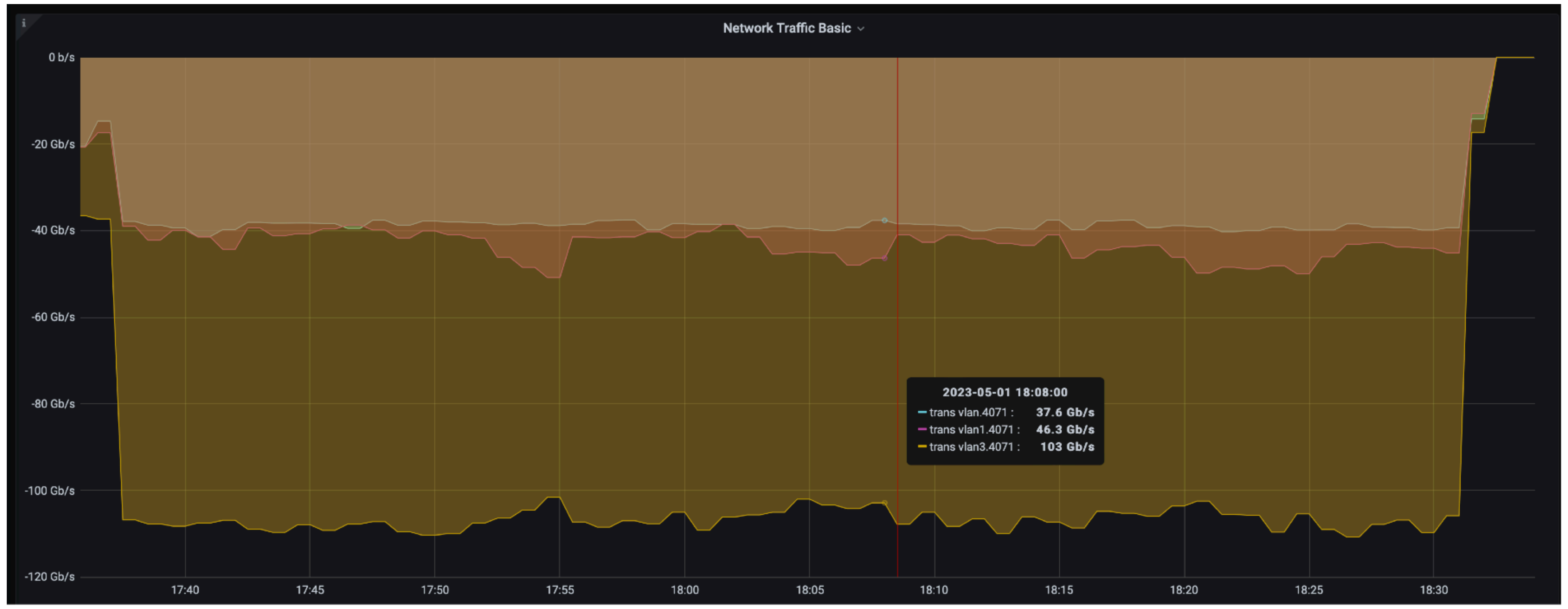
# In fact, how is throughput parametrized by number of cpu cores and streams?



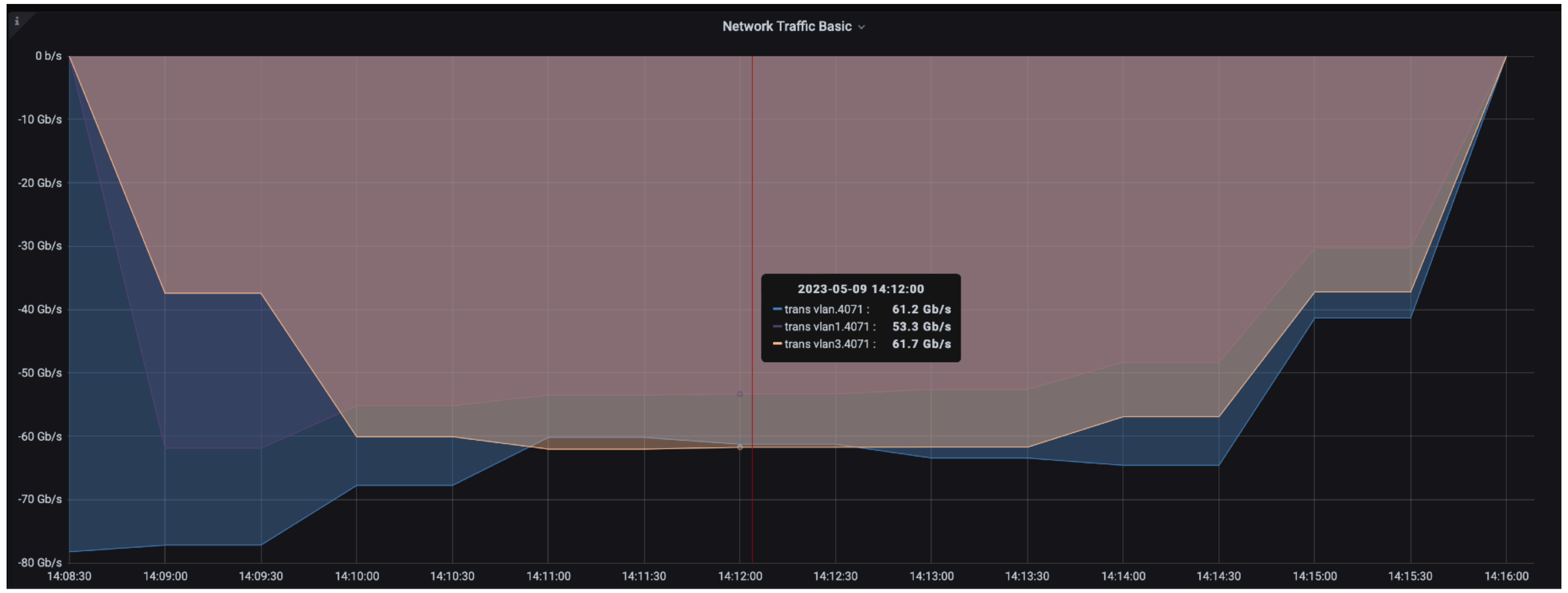
## 2. What is the overhead of adding a redirector?

- Almost None! We get the same performance transferring data between clusters as we do transferring directly between origins.

cluster-to-cluster (using redirectors): 500 streams



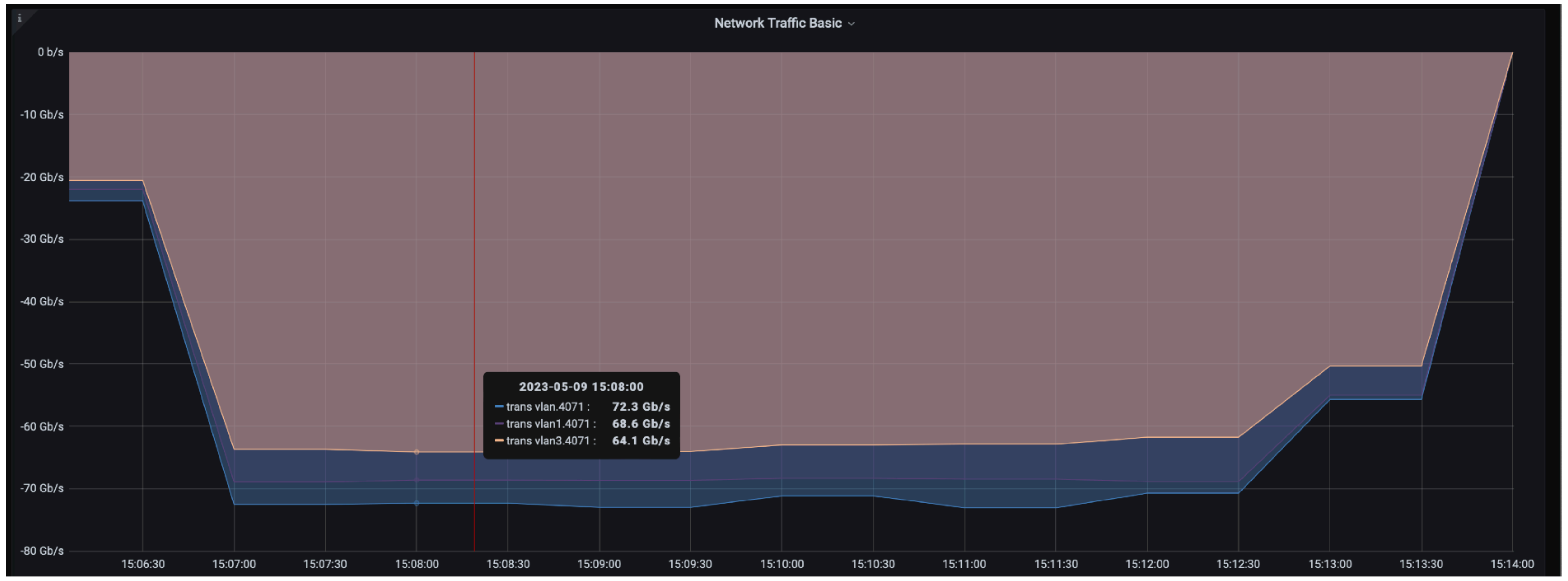

origin-to-origin (manual load balancing): 40 streams per caltech host



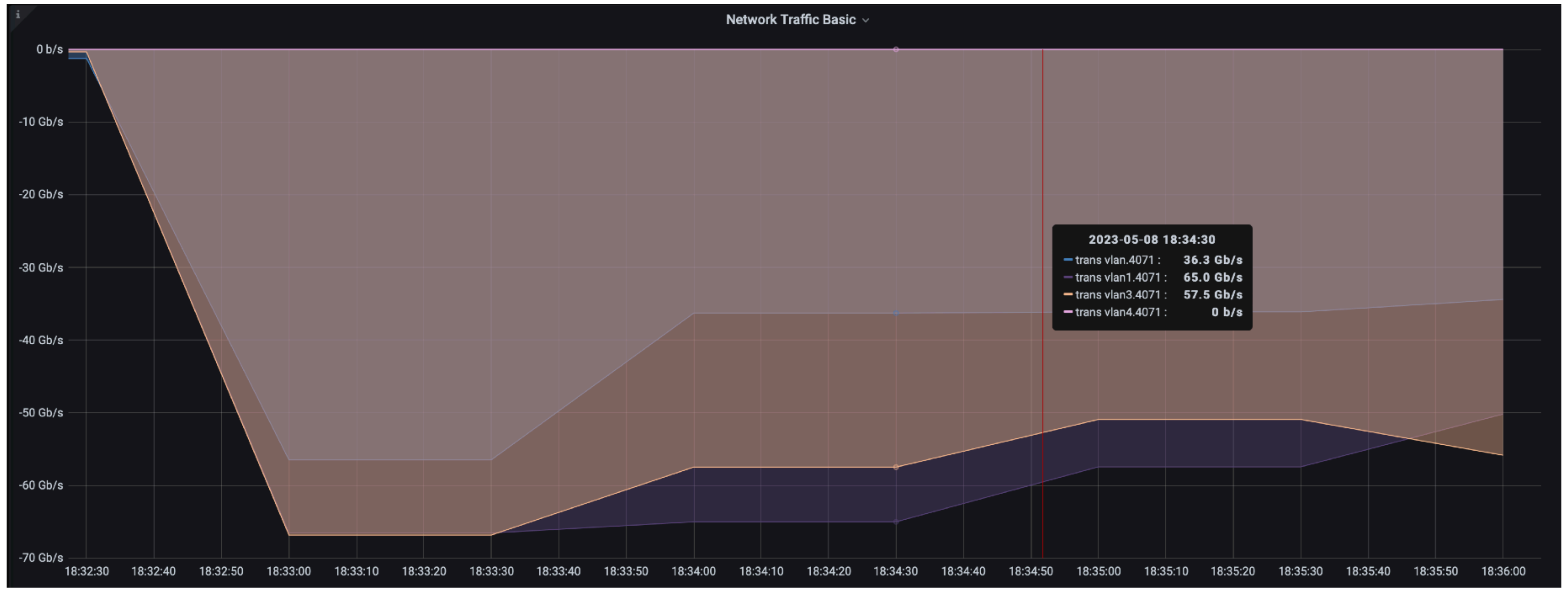
# 3. How does the transfer tool affect throughput?

- We get slightly different performance when we transfer gfal-copy and curl (write-then-delete-then-repeat)

Using gfal-copy we get **200 Gbps**




Using curl, we get **170 Gbps**



# Conclusion + Summary

- XRootD-HTTP is capable of supporting the high throughputs required for the HL-LHC era.
  - Systematically running transfers can enable us to parameterize by number of CPU cores, number of streams, etc.
- Need at least $\mathcal{O}(10)$ streams per XRootD instance for ideal throughput.
- Use of redirectors does not affect performance.
- Choice of transfer tool does affect throughput.

# Acknowledgement

The authors would like to thank the different funding agencies for this work, in particular the National Science Foundation through the following grants: OAC-1836650, OAC-2030508, OAC-1836650, MPS-1148698 and OAC-1541349

# References


- Zurawski, J.; Brown, B.; Carder, D.; Colby, E.; Dart, E.; Miller, K., et al. (2021). 2020 High Energy Physics Network Requirements Review Final Report. In 2020 High Energy Physics Network Requirements Review Final Report. Lawrence Berkeley National Laboratory. Report #: LBNL-2001398. Retrieved from https://escholarship.org/uc/item/78j3c9v4

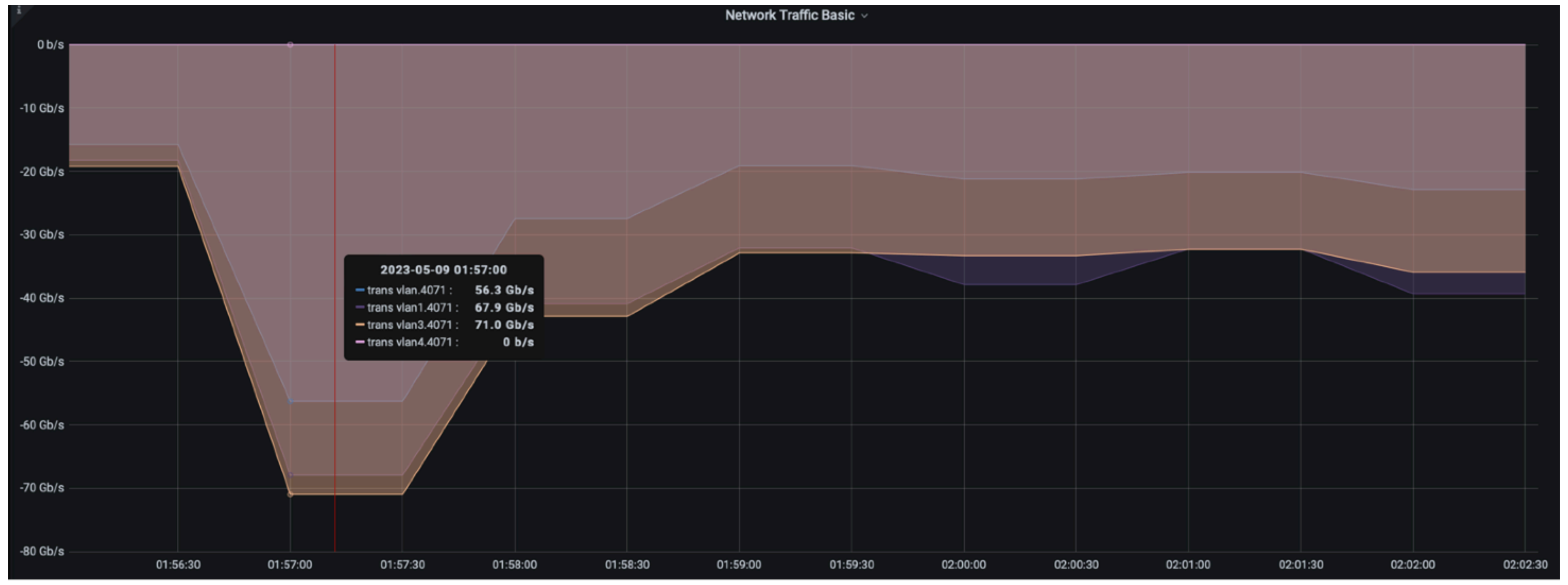
- Fajardo, E., Arora, A., Davila, D., Gao, R., Würthwein, F., & Bockelman, B. (2021). Systematic benchmarking of HTTPS third party copy on 100Gbps links using XRootD. EPJ Web of Conferences, 251, 02001. https://doi.org/10.1051/epjconf/202125102001

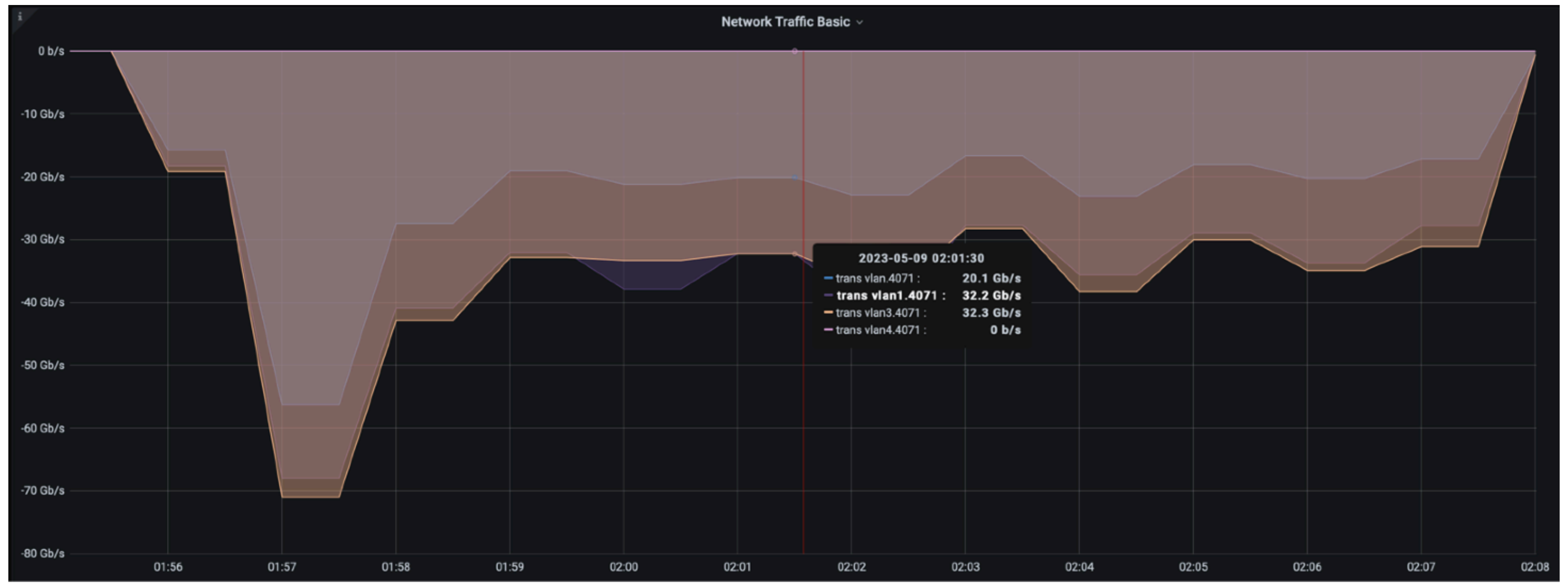
# Thank You!

# Backup

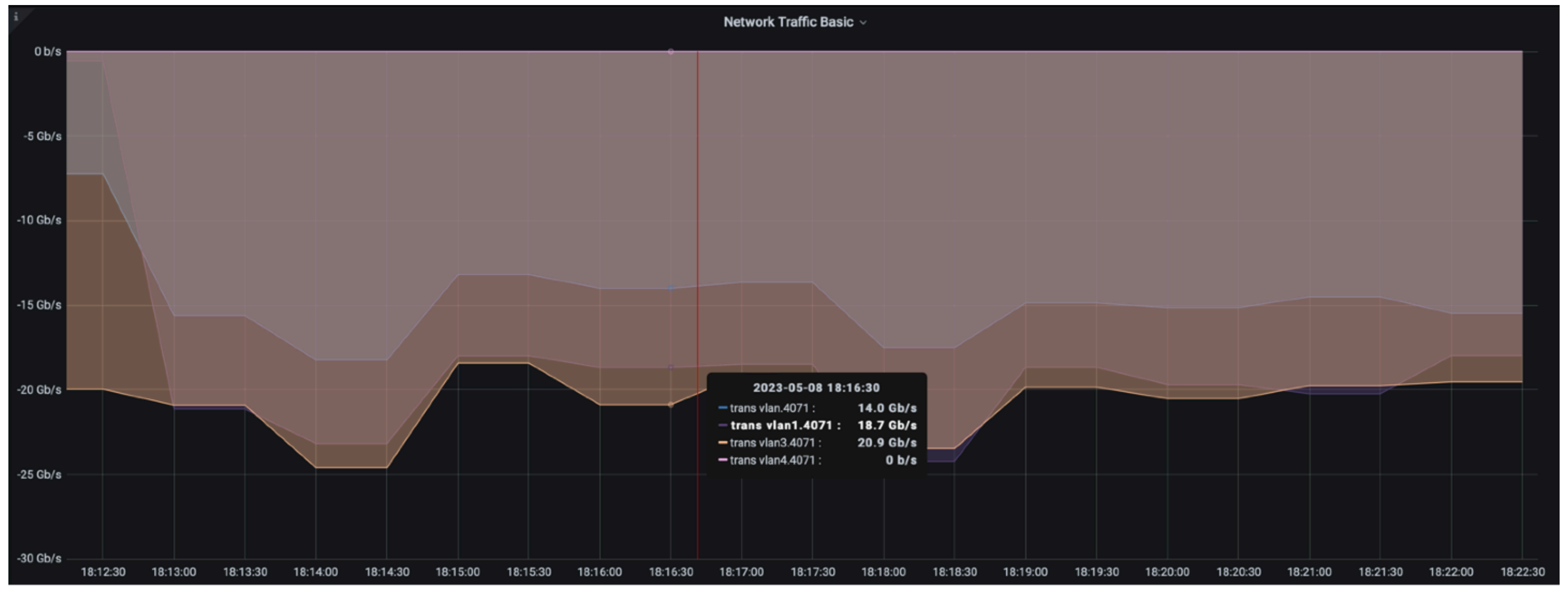
- We see interesting behavior when overwriting files,

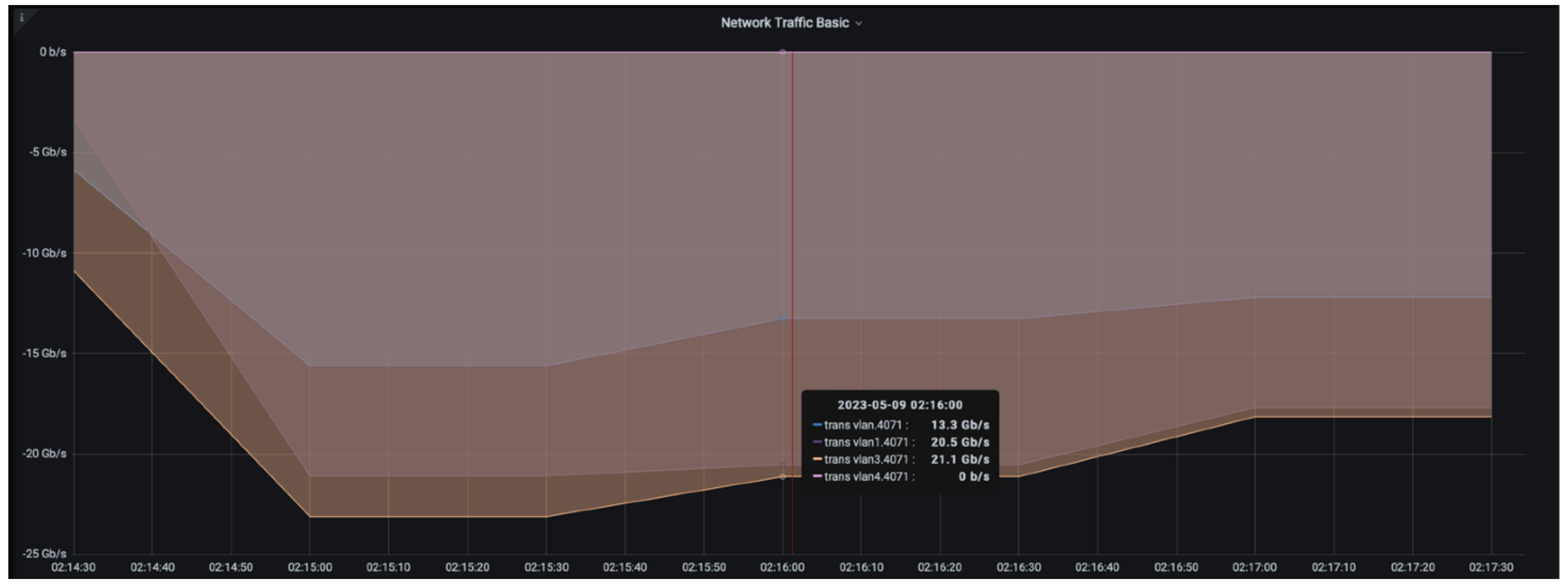
With gfal-copy, we get 170 Gbps initially, then it drops down to 80 Gbps




Similar behavior with curl,

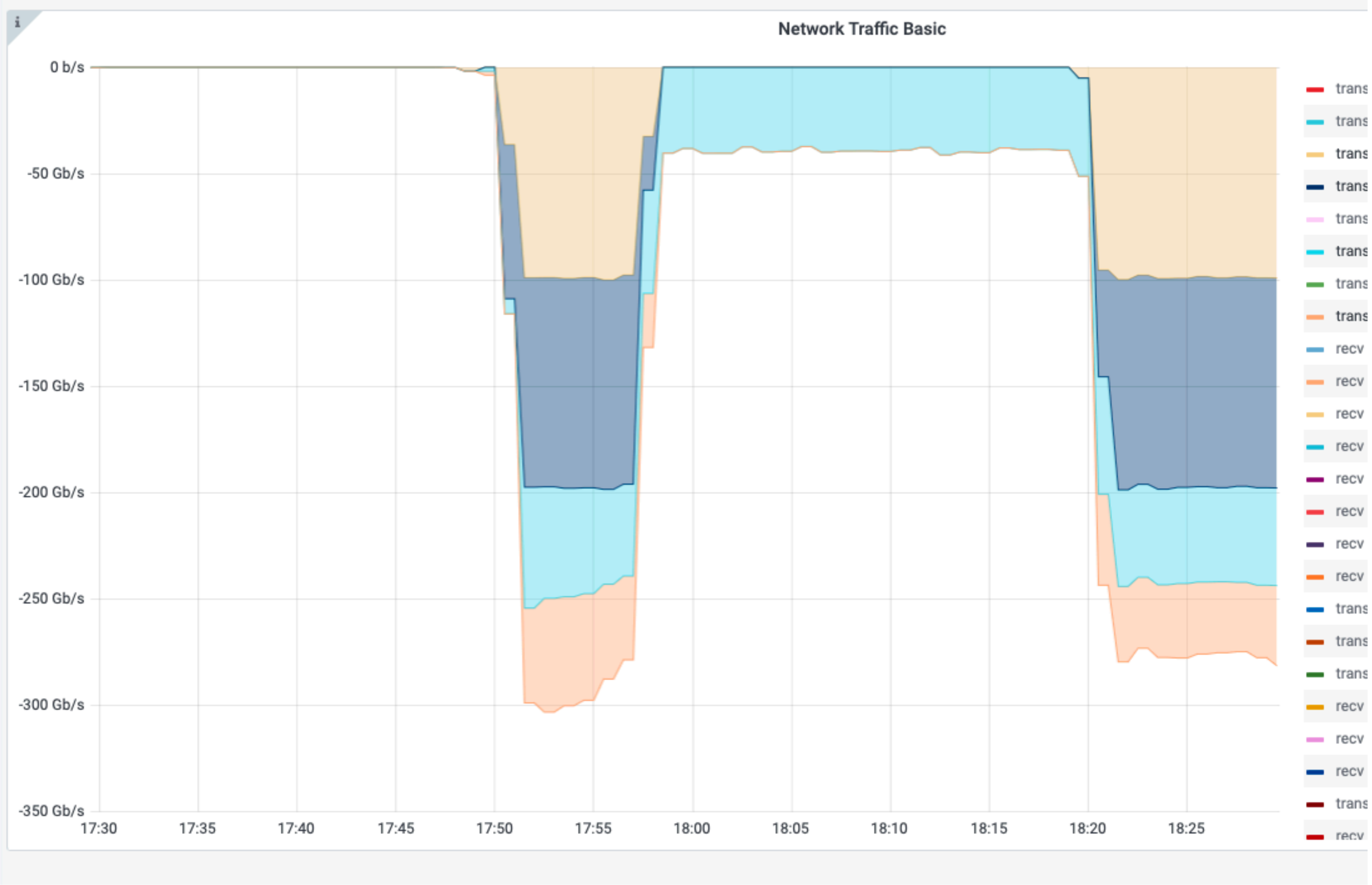


# Caltech Nodes' Specs

| Server | CPU | Cores | Mem | NIC |
|---|---|---|---|---|
| **sandie-1** | 2x E5-2667 v3 @ 3.20GHz | 32 (HT ON) | 16 x Kingston DDR4 16GB 2400 | MT27700 ConnectX-4 (100G) |
| **sandie-3** | 2x Silver 4110 CPU @ 2.10GHz | 32 (HT ON) | 12 x Micron DDR4 8GB 2666 | MT27500 ConnectX-3 (40G) |
| **sandie-5** | 1x AMD 7551P @ 2Ghz | 64 (SMT ON) | 8 x Kingston DDR4 32GB 2666 | MT28800 ConnectX-5 (100G) |
| **sandie-6** | 1x AMD 7551P @ 2Ghz | 64 (SMT ON) | 8 x Kingston DDR4 32GB 2666 | MT28800 ConnectX-5 (100G) |
| **sdn-dtn-1-7** | 2x E5-2687W v3 @ 3.10GHz | 20 (HT OFF) | 8 x Micron DDR4 16GB 2133 | MT27700 ConnectX-4 (100G) |
| **sdn-dtn-2-09** | 2x E5-2690 v2 @ 3.00GHz | 40 (HT ON) | 16 x Samsung DDR3 8GB 1600 | MT27500 ConnectX-3 (40G) |
| **sdn-dtn-2-11** | 2x E5-2670 v3 @ 2.30GHz | 48 (HT ON) | 16 x Micron DDR4 8GB 2133 | MT28800 ConnectX-5 (100G) |
| **neu-sc-01** | 2x E5-2667 v4 @ 3.20GHz | 32 (HT ON) | 8 x Hynix DDR4 16GB 2133 | MT28908 ConnectX-6 (100G) |
| **sdn-sc-03** | 2x E5-2667 v4 @ 3.20GHz | 32 (HT ON) | 8 x Hynix DDR4 16GB 2133 | MT28908 ConnectX-6 (100G) |
| **sdn-sc-04** | 2x E5-2667 v4 @ 3.20GHz | 32 (HT ON) | 8 x Hynix DDR4 16GB 2133 | MT28908 ConnectX-6 (100G) |
| **sdn-sc-05** | 2x E5-2667 v4 @ 3.20GHz | 32 (HT ON) | 8 x Hynix DDR4 16GB 2133 | MT28908 ConnectX-6 (100G) |
| **sdn-sc-06** | 2x E5-2667 v4 @ 3.20GHz | 32 (HT ON) | 8 x Hynix DDR4 16GB 2133 | MT28908 ConnectX-6 (100G) |
| **sandie-9** | 2x E5-2667 v3 @ 3.20GHz | 32 (HT ON) | 8 x Hynix DDR4 16GB 2133 | MT28800 ConnectX-5 (100G) |

# Plot Showing 300 Gbps sustained



# Plot showing throughput vs. latency