ESnet-US ATLAS Xcache pilot Summary

Alex Sim
Computational Research Division
Lawrence Berkeley National Laboratory
&
Eli Dart, Chin Guok, Yatish Kumar, Inder Monga, Eric Pouyoul, Adam Slagell
(Energy Sciences Network)
Rob Gardner, Ilija Vukotic, Lincoln Bryant, Chris Weaver (Univ. of Chicago)
Wei Yang, Andy Hanushevsky (SLAC)

Introduction

- We all know about
  - Data volume increase in experiments and simulations
  - Data volume moving through network also increases
  - Network bandwidth requirement gets higher
- Observation
  - Significant portion of the popular dataset is transferred multiple times to different users as well as to the same user

Data sharing

- **Sharing data**
  - Reduce the redundant data transfers
  - Save network traffic volume, consequently.
- **Lower data access latency**
  - Overall application performance is expected to be improved

Pilot experiment

- In-network temporary data cache for data sharing
- Goals
  - Study how network cache storage helps network traffic performance and the overall application performance
  - Accumulate experience on how the DOE scientific experiments and simulations share data among their users.

Experimental setup

- Collaborate with US ATLAS
- ESnet
  - Provide a temporary storage cache node in our network
  - Monitor the network activities (SNMP, tstat)
  - Adam Slagell, Chin Guok, Eli Dart, Eric Pouyoul, Inder Monga, Yatish Kumar, Alex Sim
- US ATLAS
  - Deploy/operate the Xrootd/Xcache software stack
  - Recruit users
  - Application-level monitoring
  - Rob Gardner, Ilija Vukotic, Lincoln Bryant, Chris Weaver at U. of Chicago,
  - Wei Yang, Andy Hanushevsky at SLAC
- Hardware
  - 20TB storage capable of 30Gbps I/O and 40Gbps networking capability
- Schedule:
  - Storage host ready - May 24, 2019
  - Project end - July 31, 2019

Analysis jobs

- **Wei Yang at SLAC: 300 concurrent simulated analysis jobs for 5 times between 6/24/2019 and 7/5/2019**
  - **Not necessarily the same as real analysis jobs as the diversity of those analysis jobs is large.**
  - **Emulate the behavior of analysis jobs**
    - **Random number of reads, each read in random size, and each read starting from a random offset.**
    - **Simply discard the data just read, and move on to the next read, while real analysis jobs may spend some time on the data.**
- **Jobs**
  - **Dataset: 4381 files, 13.8 TB, file sizes [449.098KB, 11.678GB]**
  - **Each of the 300 concurrent jobs sequentially open 100 random files, with random offset and length.**
    - **Each file can be opened and read by a random number of different jobs.**
  - **3 hour run time limit**
    - **It may affect the total number of requests to the cache.**

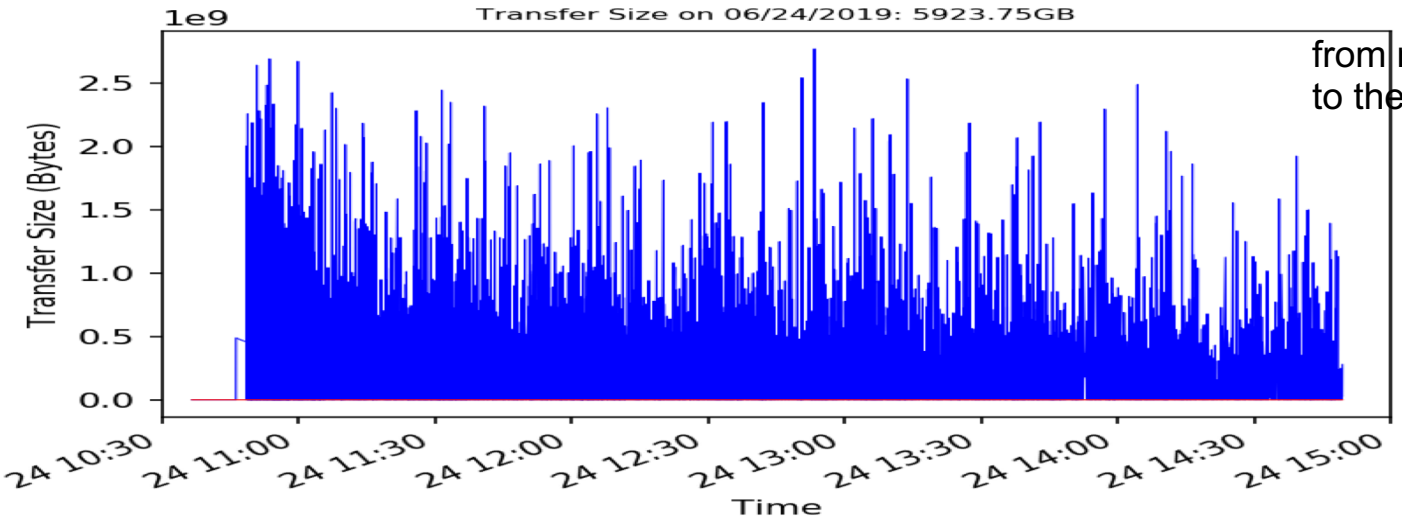
Xrootd/Xcache data management behavior

- **Data block**
  - Xcache views a file as a sequence of 1MB data blocks
  - Only brings in the data blocks needed by the clients
    - If a client requests a file from offset 100, length 1000, then the Xrootd cache only requests a block of offset 0, length 1MB from the remote file source to the cache.
    - If a client requests from offset 2M+10, length 1M+5 (e.g. this request cross the boundary of block 2 and 3), then the cache requests two blocks from the remote data source: offset 2M, length 2M.
- **Data movement**
  - Only a part of each file is moved out of the cache for each job.
  - Xrootd keeps track of cached partial data in a file using a sparse file: files with holes
    - For example, open(), seek(100), write(10), close(), then you have a file of 110 bytes. The first 100 bytes is a hole. It uses another file, datafile.cinfo to keep trace of the holes in the data file.

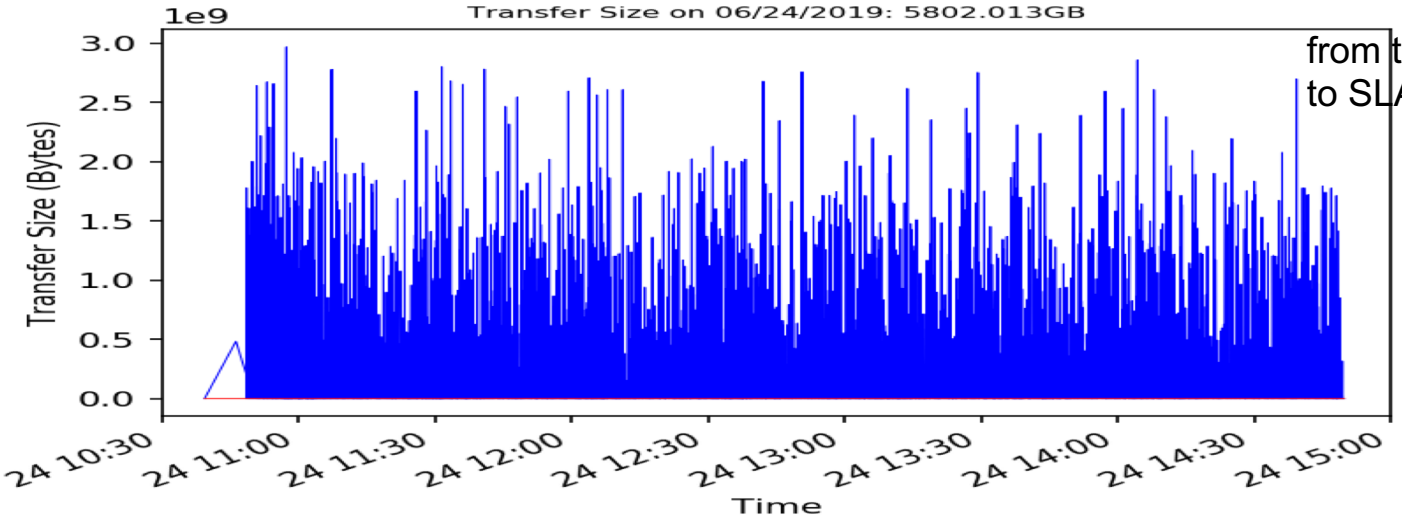
Analysis job #1

- Retrieve data into the cache
  - Connected 208 unique remote IPs for 25440 times
  - Transferred 5923.75 GB into the cache node
  - Cached 42.93% of the total dataset
    - 5923.75/13800
- From the cache to the analysis job
  - SLAC asked 34922 times to access data
  - Transferred out 5802.01 GB in total
  - ~0.1934 GB (=5802.01/30000) in each request.
- Network traffic volume saving from various sites to the cache node
  - 0 GB
  - Started from cold cache

Analysis job #1 (2)
Transfer size over time

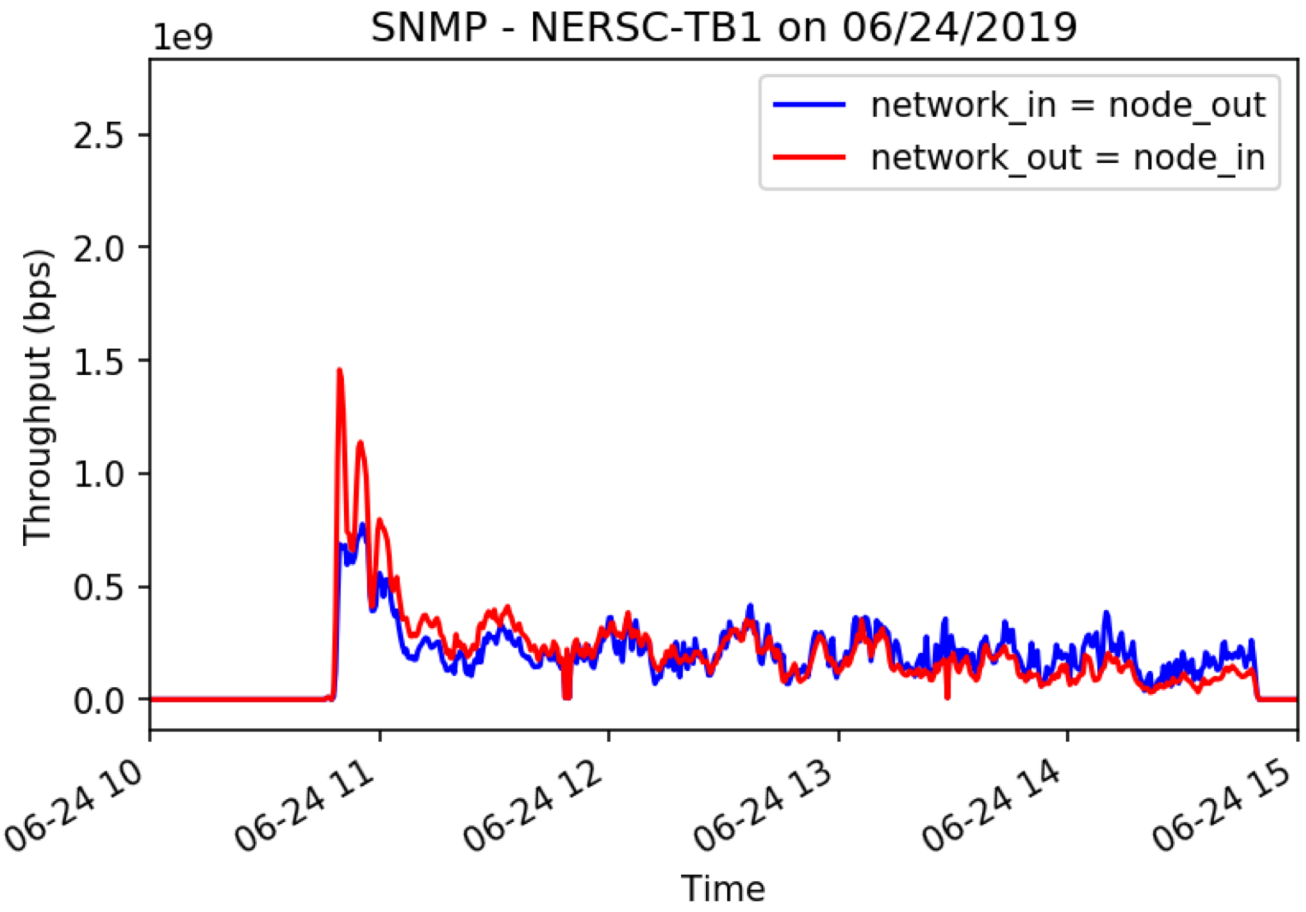


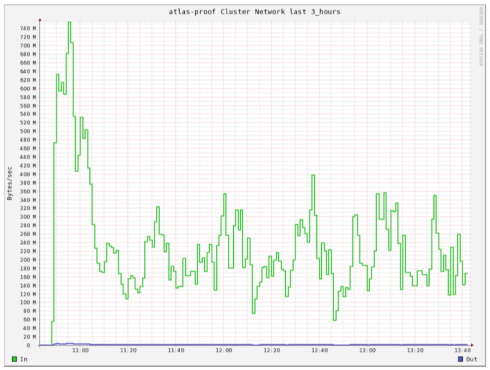
from remote sites
to the local cache



from the local cache
to SLAC analysis requests

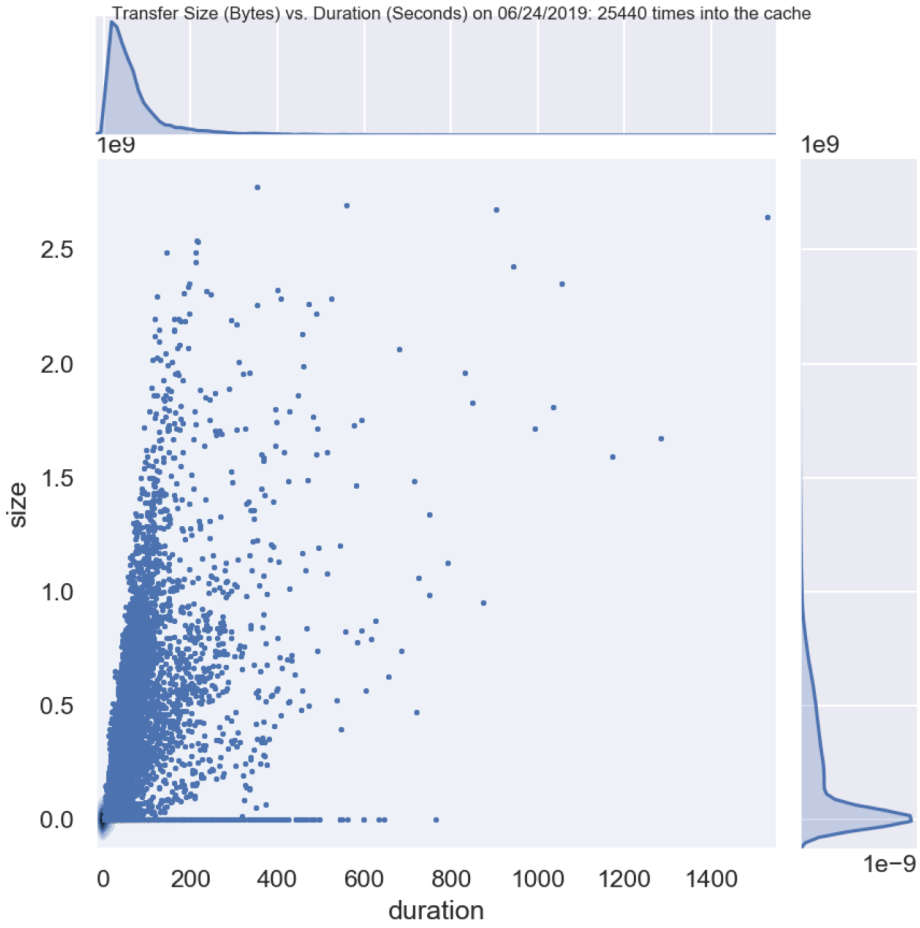
Analysis job #1 (3)
SNMP



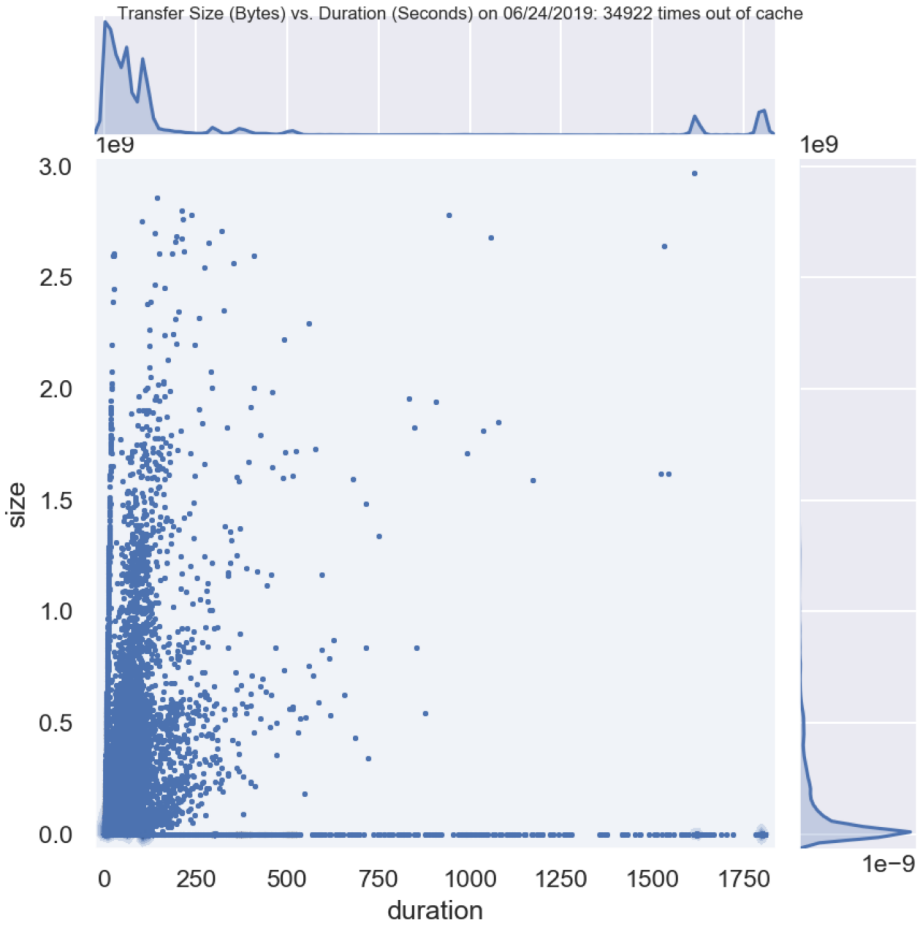


Analysis job #1 (4)

Transfer duration vs Data size with Distributions

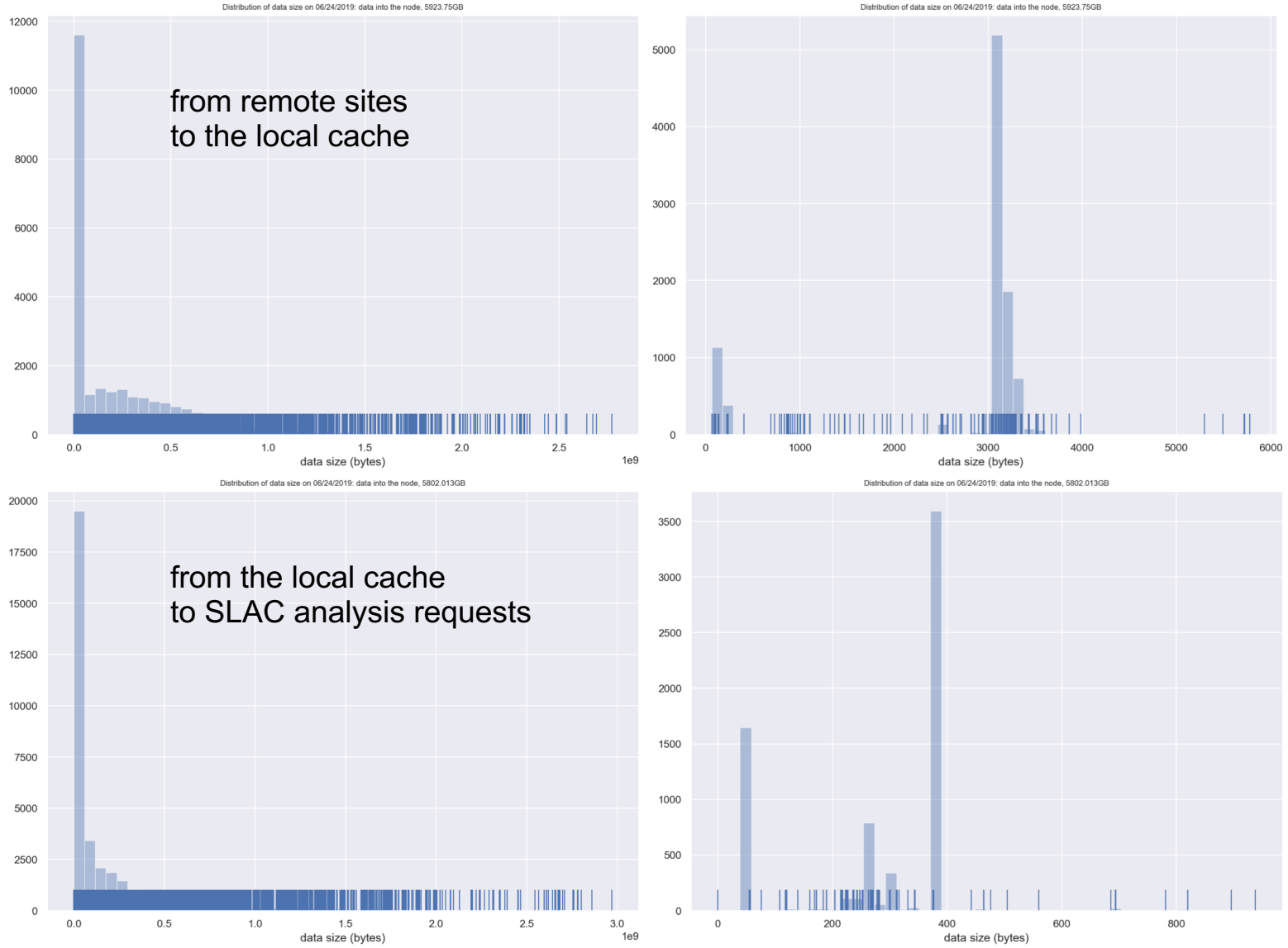


from remote sites to the local cache

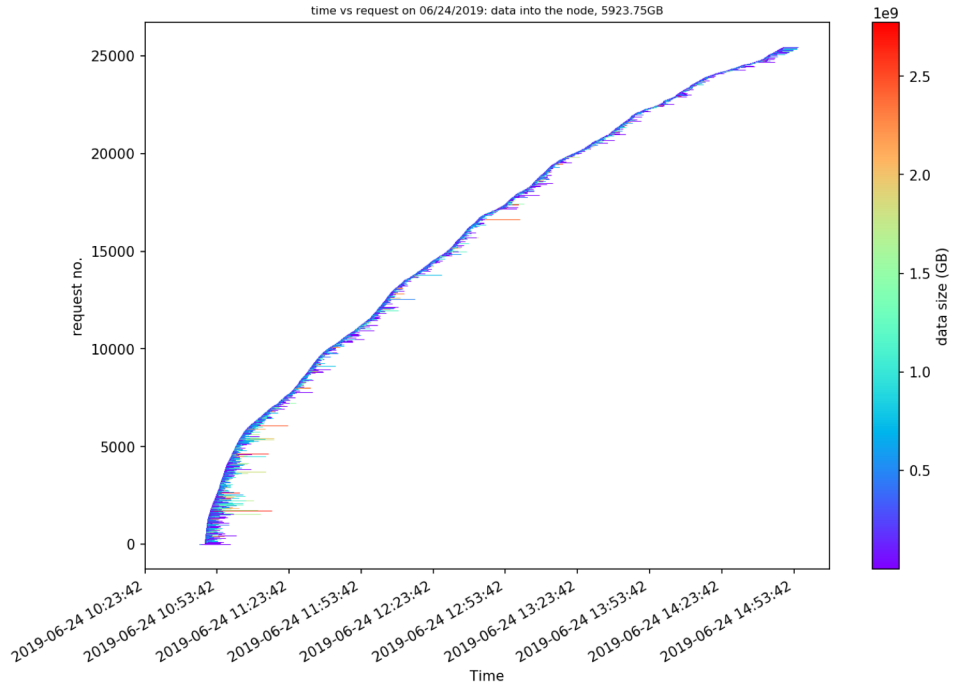


from the local cache to SLAC analysis requests

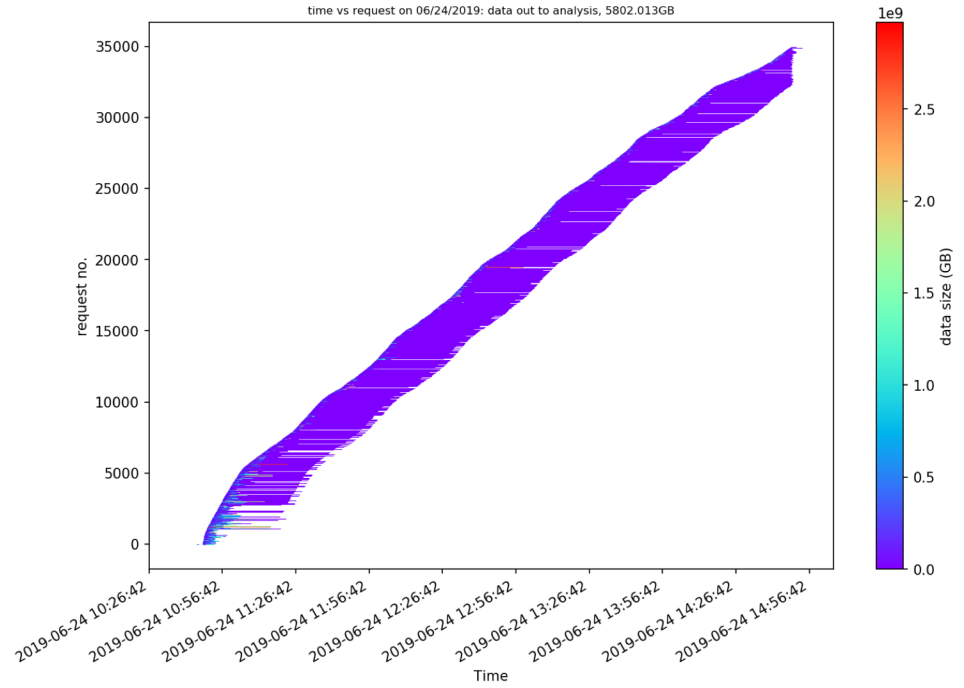
Analysis job #1 (5)
Transferred data size distribution



Analysis job #1 (6)
Transfer request completion time



from remote sites to the local cache

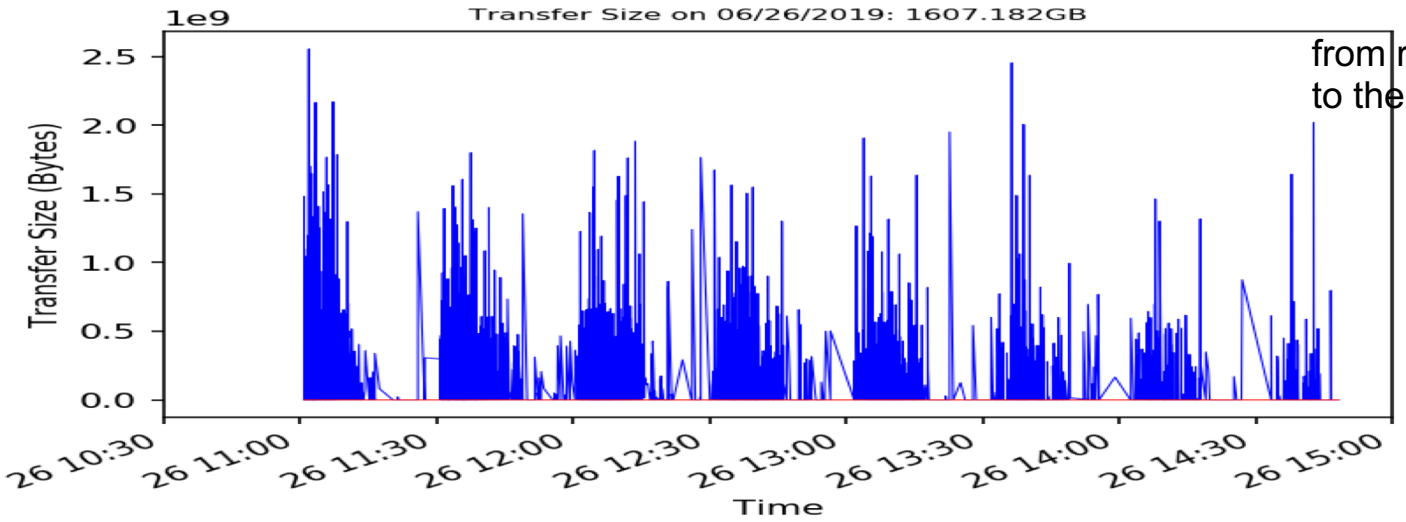


from the local cache to SLAC analysis requests

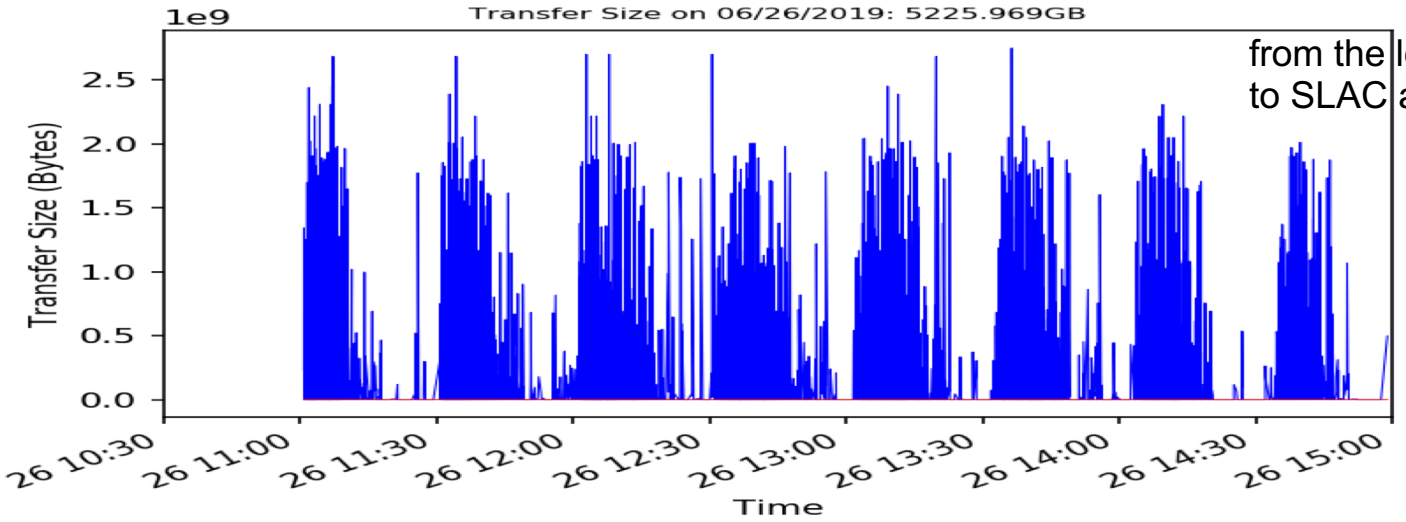
Analysis job #2

- **Retrieve data into the cache**
  - Connected 281 unique remote IPs for 13662 times
  - Transferred 1607.18 GB into the cache node
  - Cached 54.57% of the total dataset
    - (5923.75+1607.18)/13800
- **From the cache to the analysis job**
  - SLAC asked 30466 times to access data
  - Transferred out 5225.97 GB in total
  - ~0.1742 GB (=5225.97/30000) in each request.
- **Network traffic volume saving from various sites to the cache node**
  - ~3618.79 GB (=5225.97-1607.18)

Analysis job #2 (2)
Transfer size over time

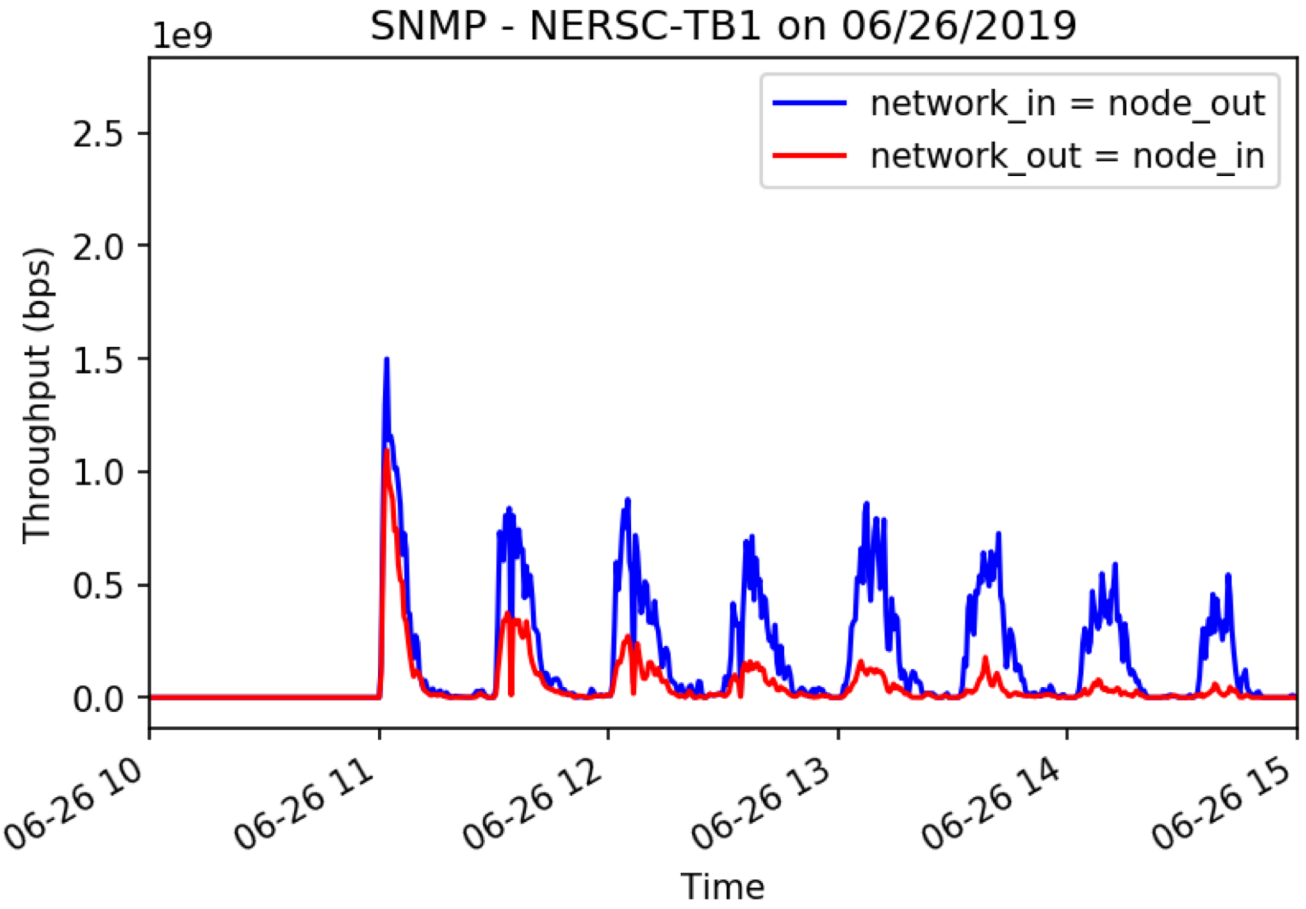


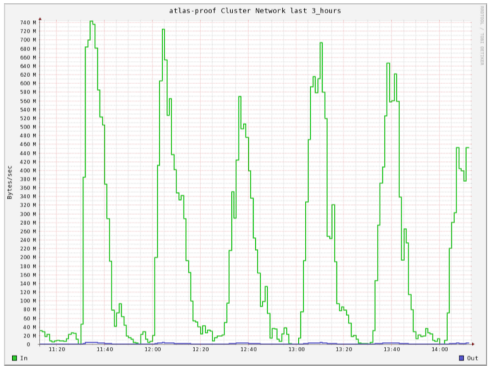
from remote sites
to the local cache



from the local cache
to SLAC analysis requests

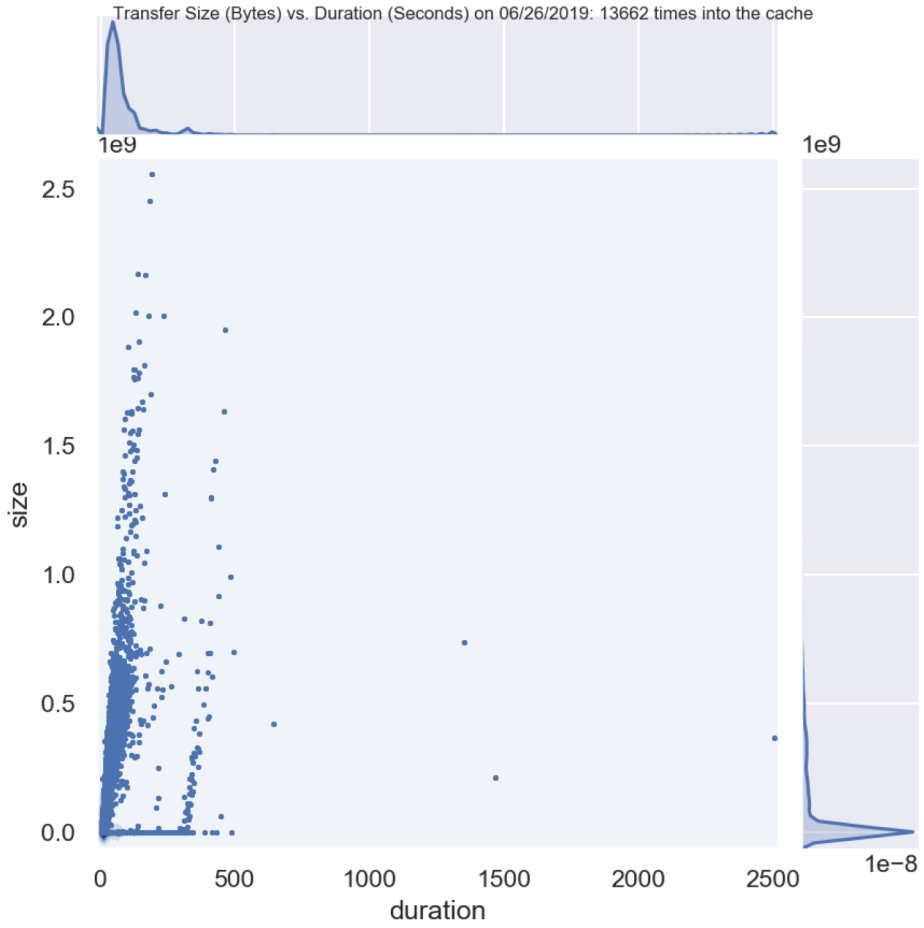
Analysis job #2 (3)
SNMP



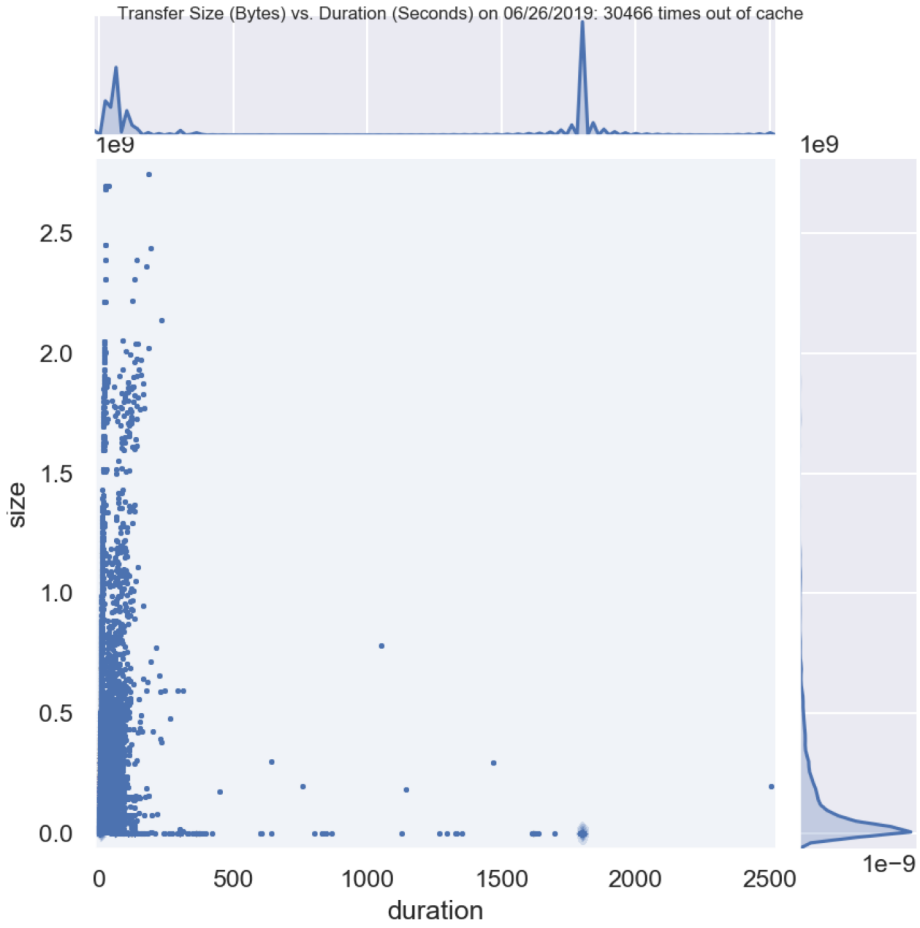


Analysis job #2 (4)

Transfer duration vs Data size with Distributions

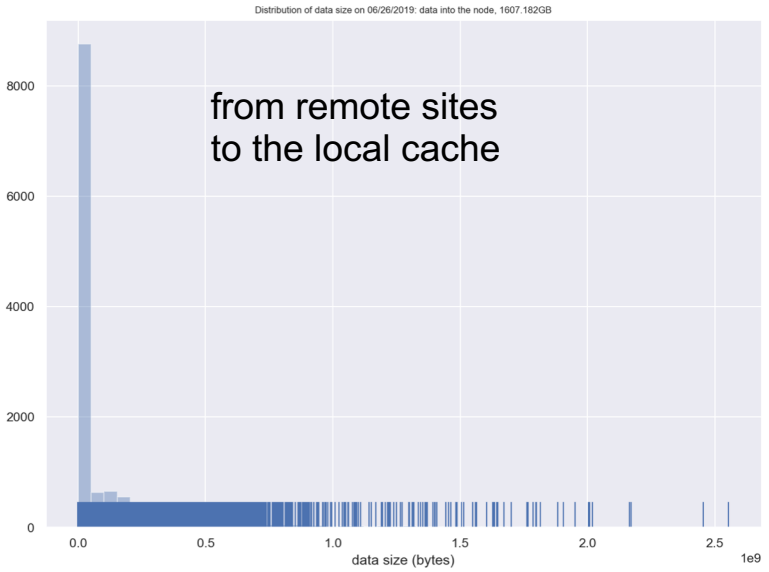


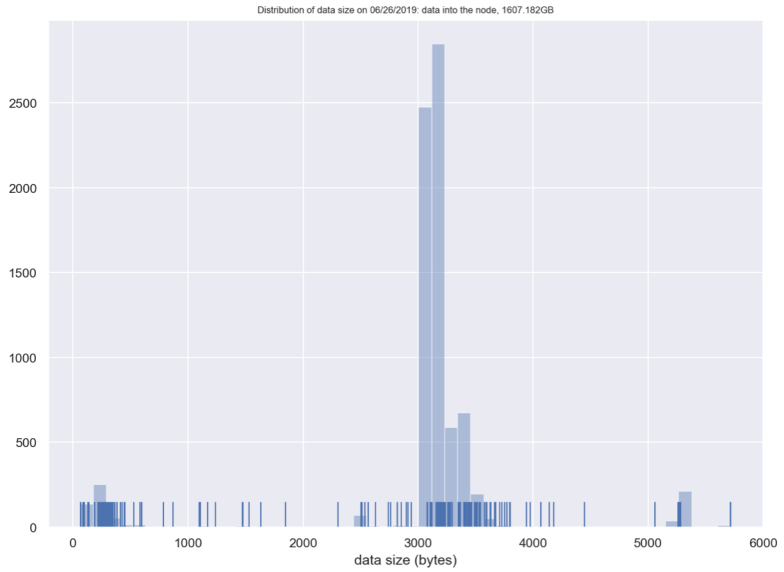
from remote sites to the local cache

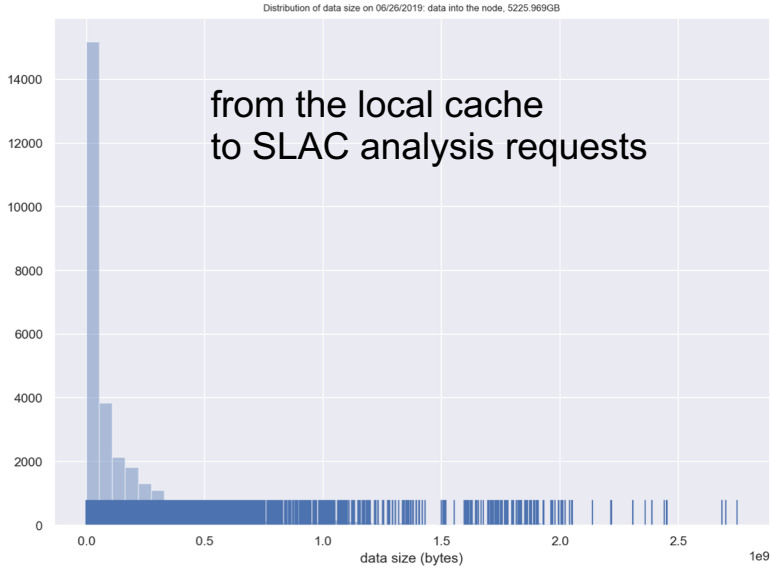


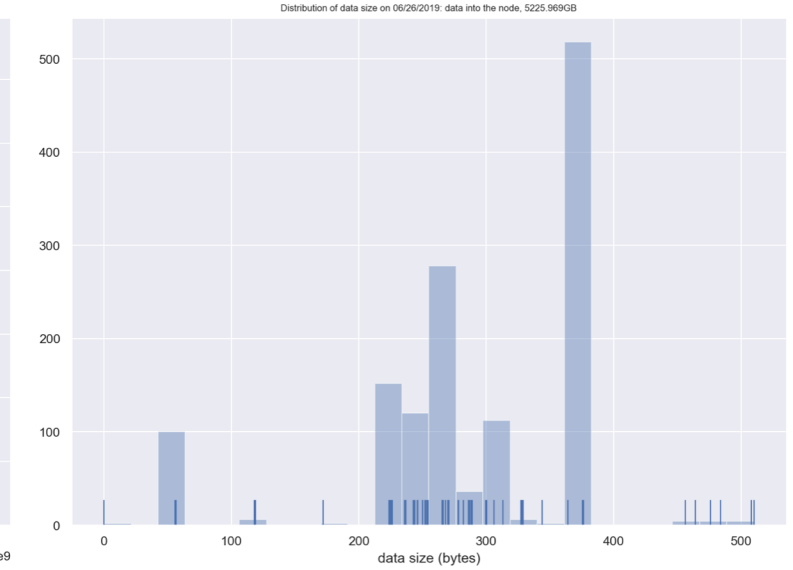
from the local cache to SLAC analysis requests

Analysis job #2 (5)
Transferred data size distribution

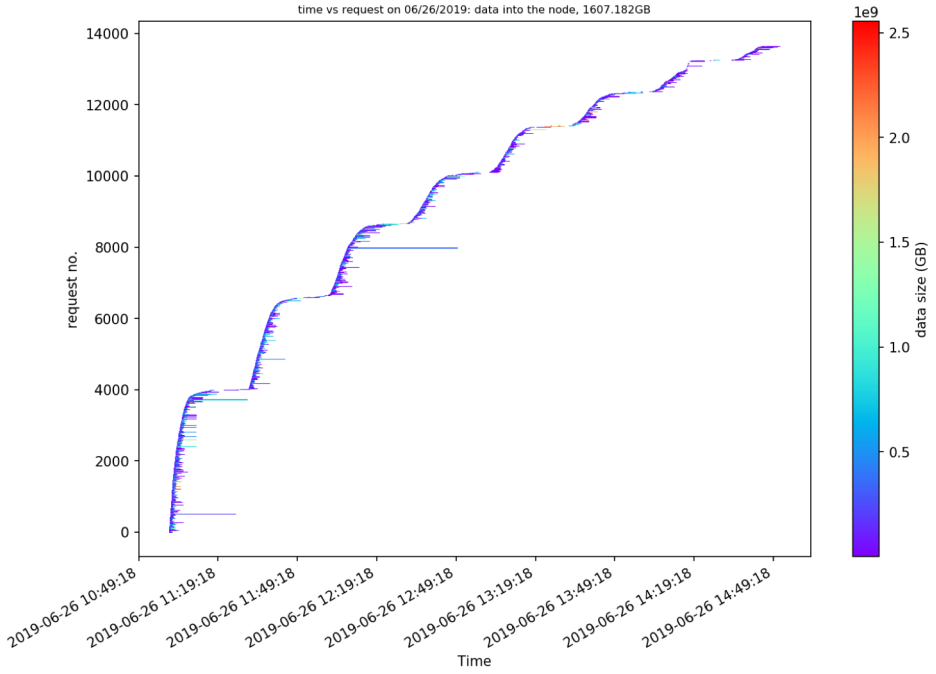




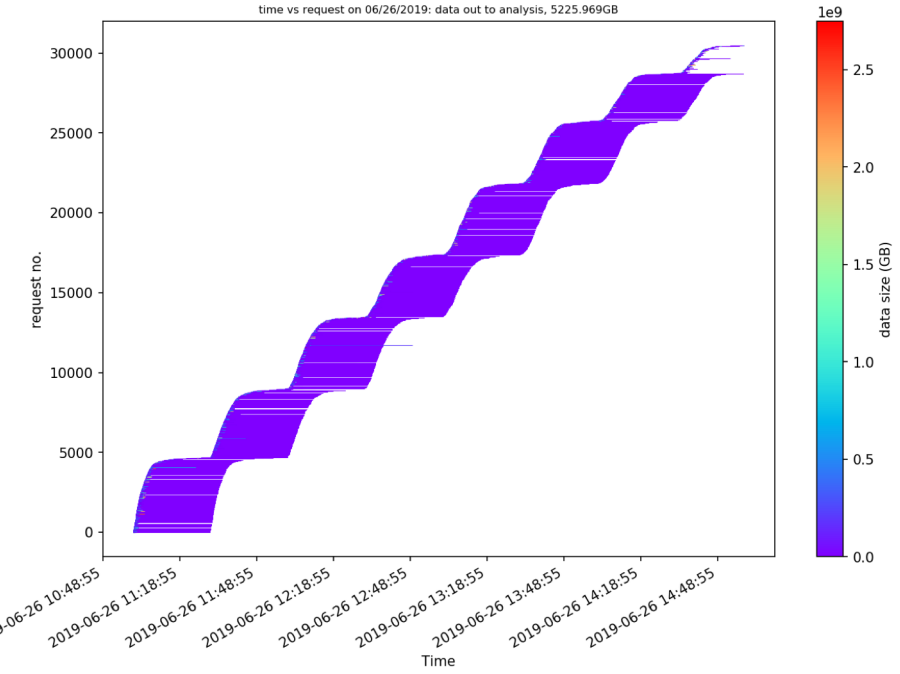




Analysis job #2 (6)
Transfer request completion time



from remote sites to the local cache

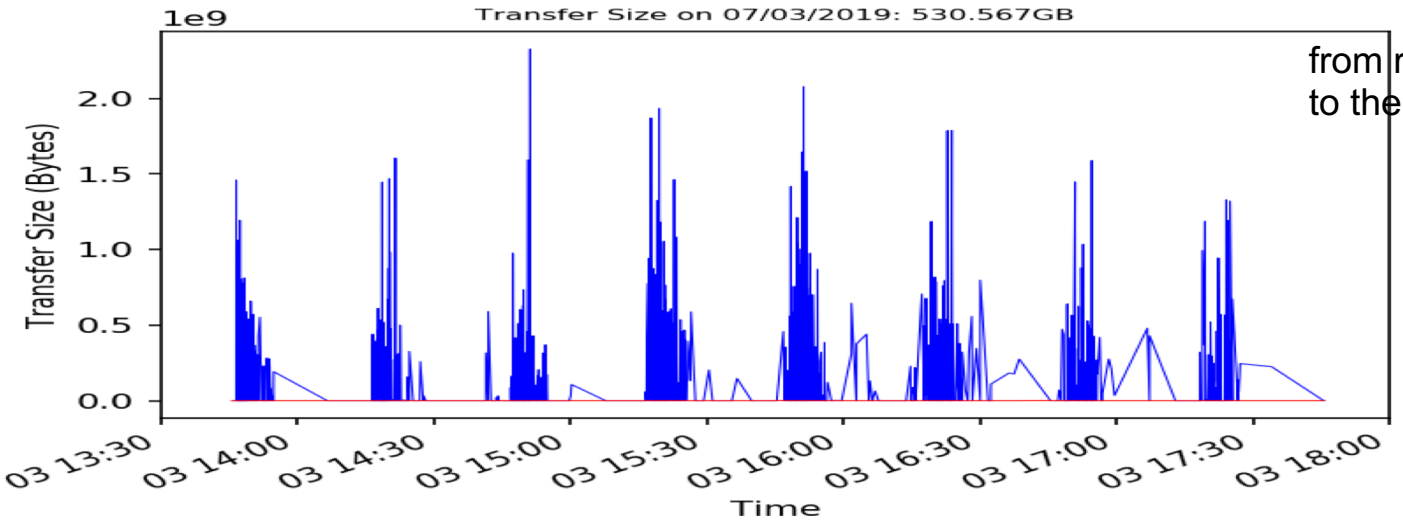


from the local cache to SLAC analysis requests

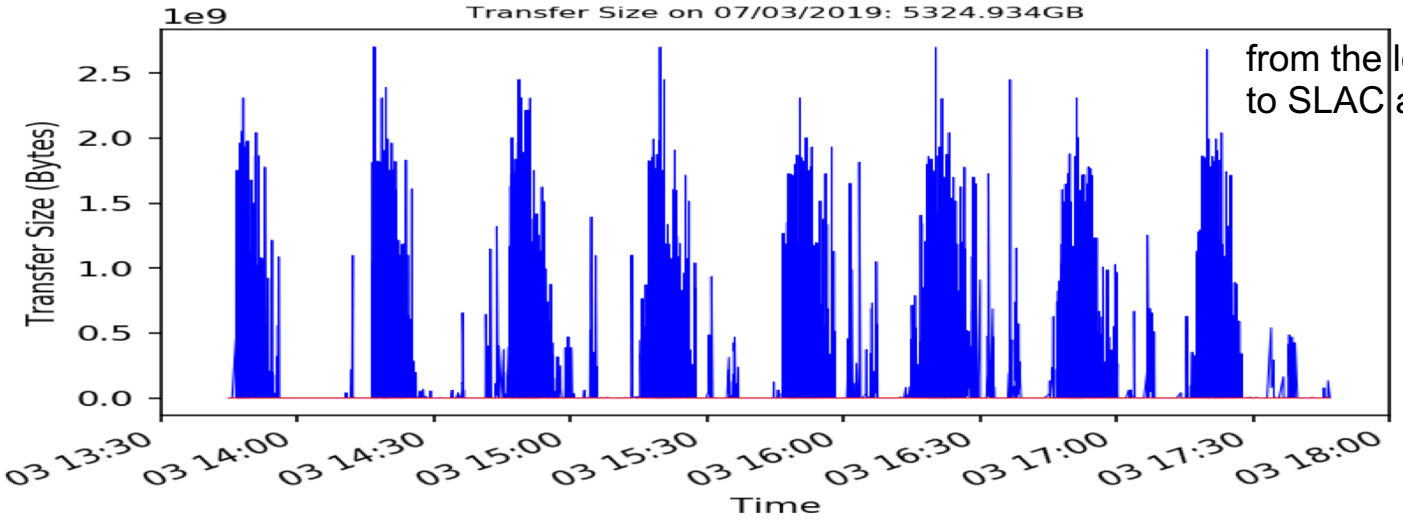
Analysis job #3

- **Retrieve data into the cache**
  - Connected 169 unique remote IPs for 4900 times
  - Transferred 530.57 GB into the cache node
  - Cached 58.42% of the total dataset
    - (5923.75+1607.18+530.57)/13800
- **From the cache to the analysis job**
  - SLAC asked 29930 times to access data
  - Transferred out 5225.97 GB in total
  - ~0.1775 GB (=5324.93/30000) in each request.
- **Network traffic volume saving from various sites to the cache node**
  - ~4794.36 GB (=5324.93-530.57)

Analysis job #3 (2)
Transfer size over time

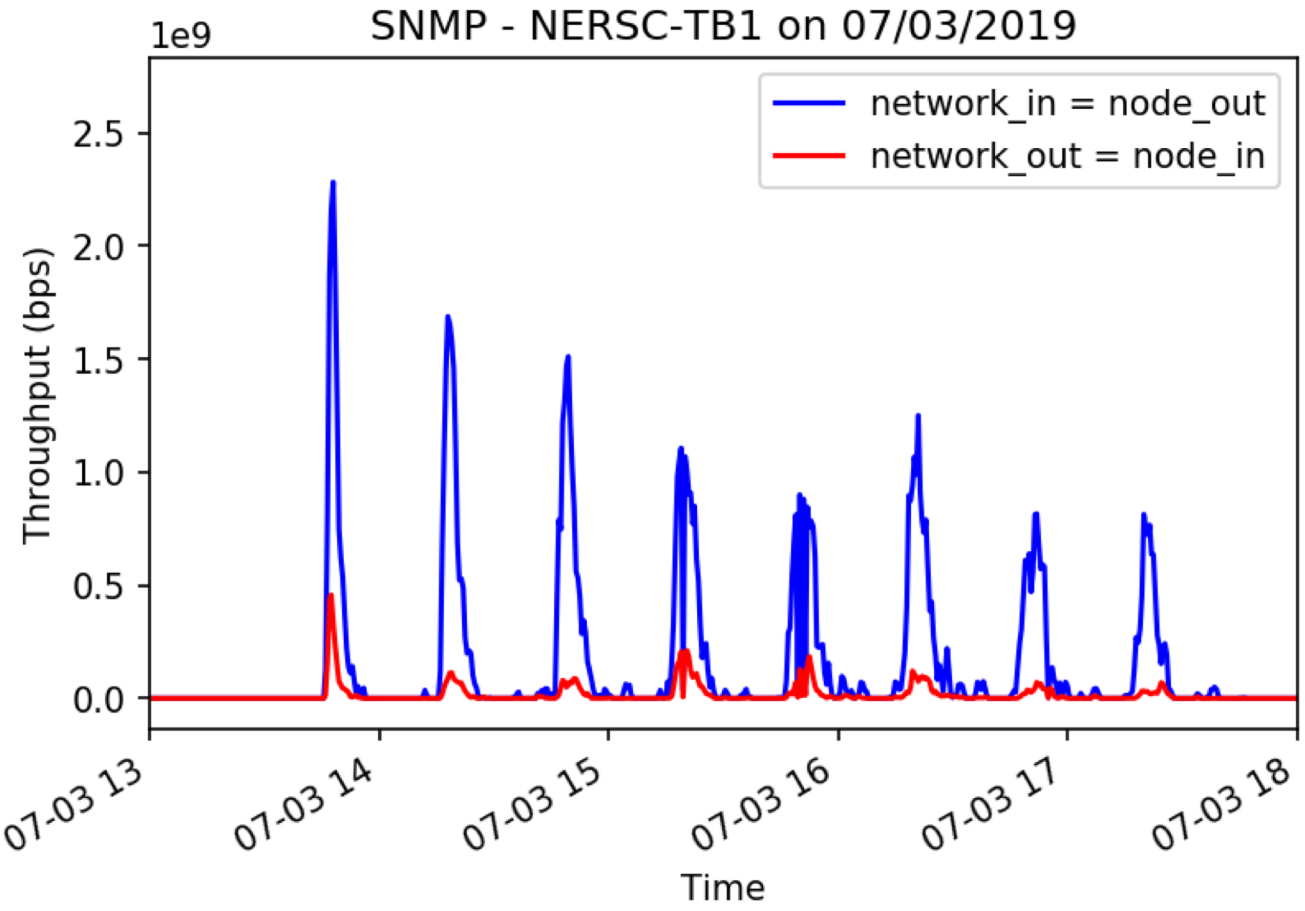


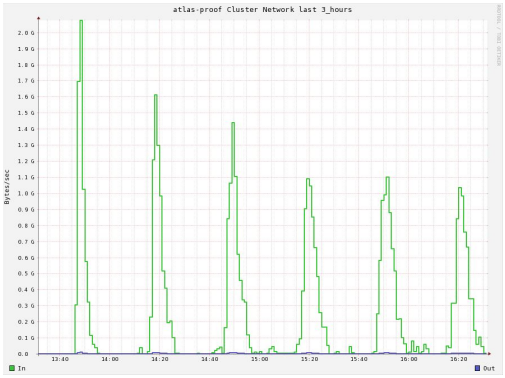
from remote sites
to the local cache



from the local cache
to SLAC analysis requests

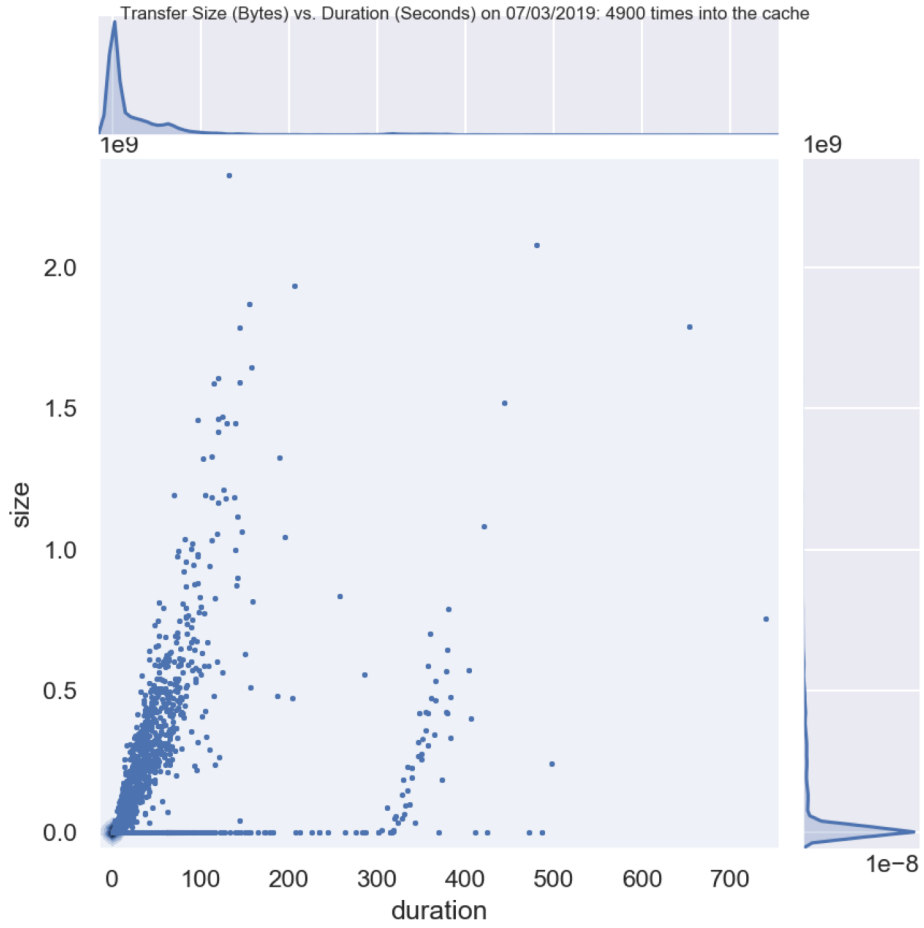
Analysis job #3 (3)
SNMP



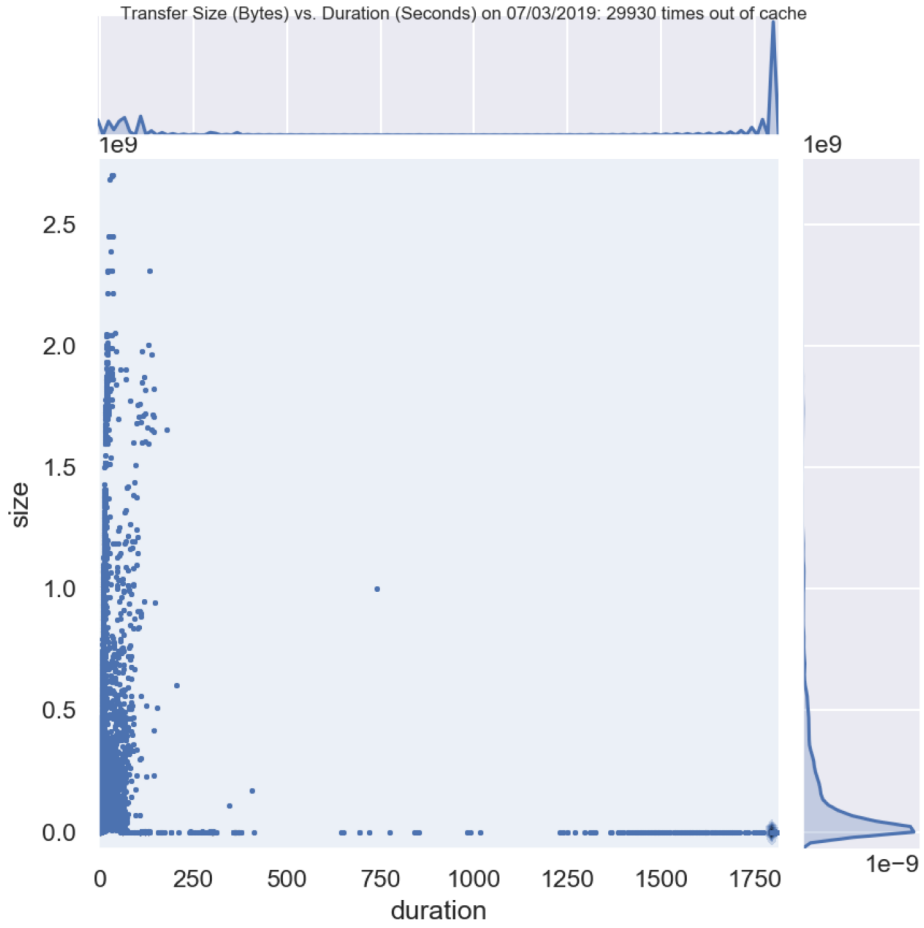


Analysis job #3 (4)

Transfer duration vs Data size with Distributions

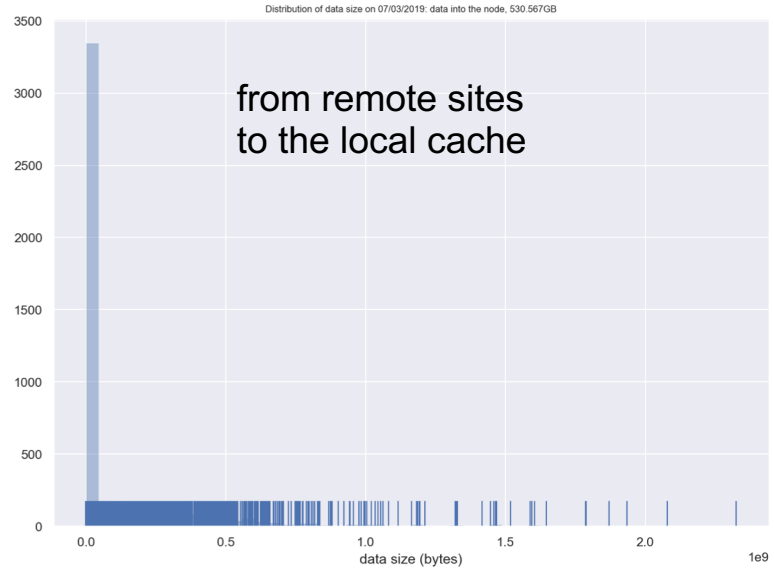


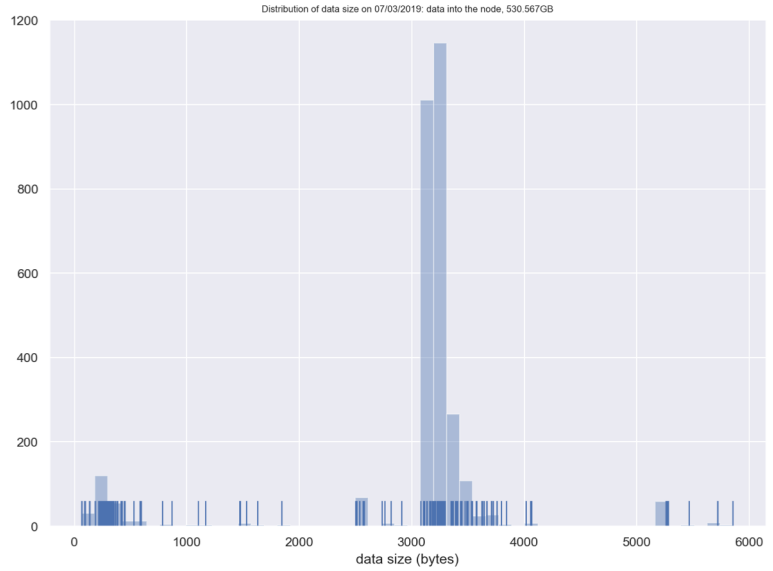
from remote sites to the local cache

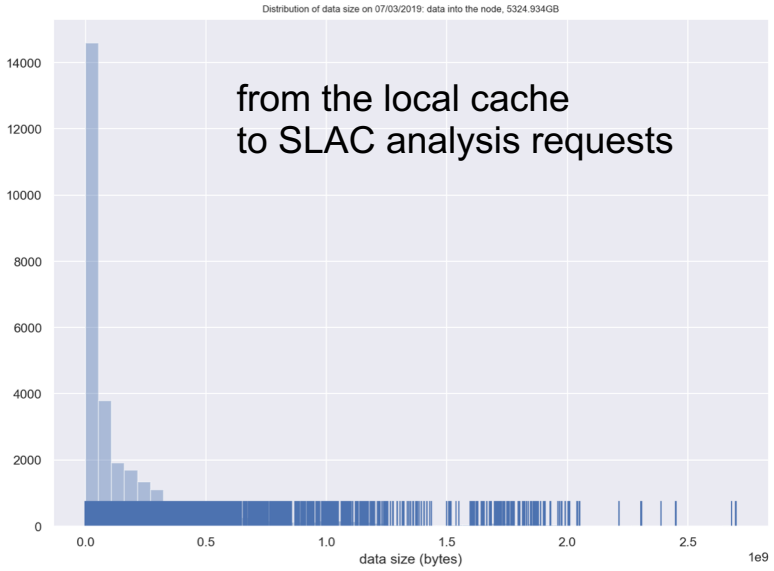


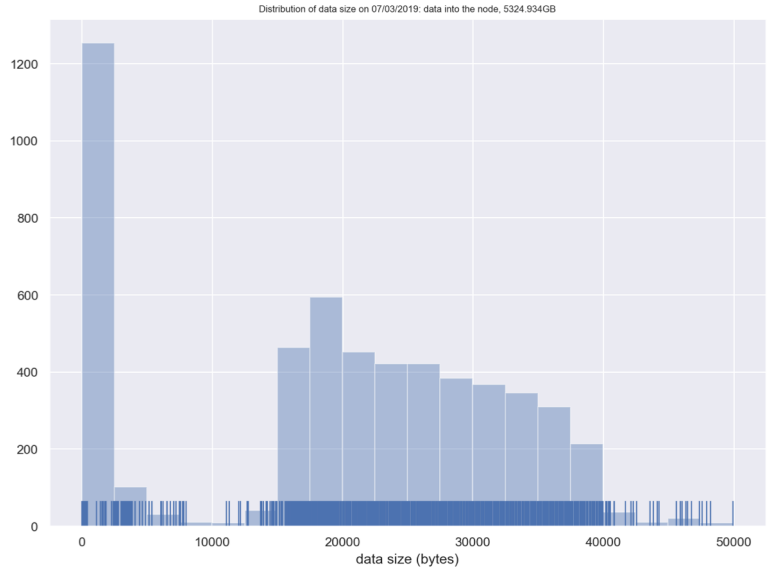
from the local cache to SLAC analysis requests

Analysis job #3 (5)
Transferred data size distribution

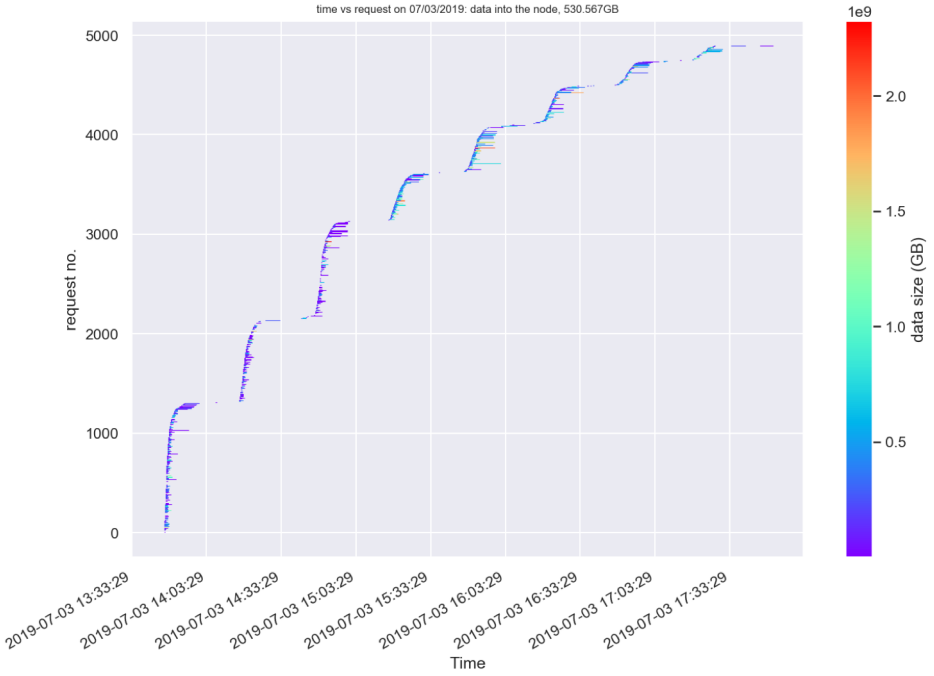




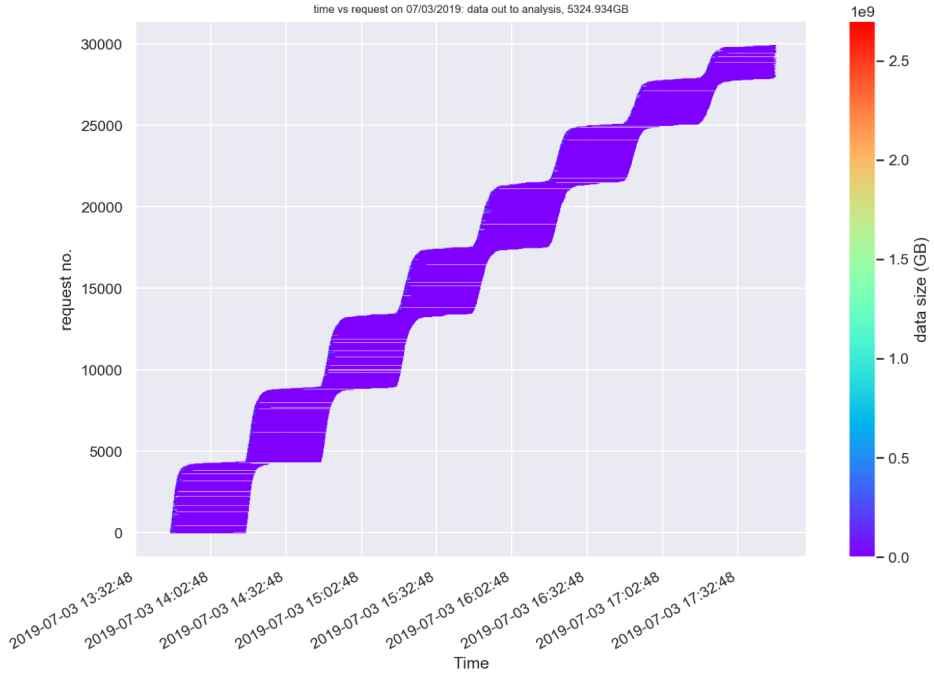




Analysis job #3 (6)
Transfer request completion time



from remote sites to the local cache

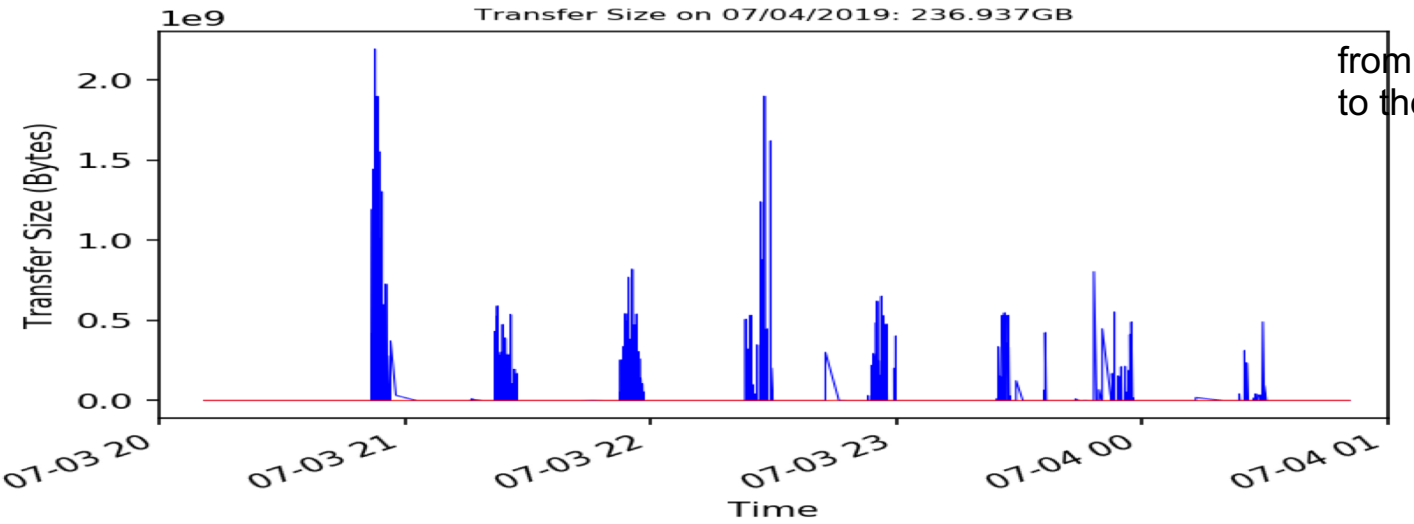


from the local cache to SLAC analysis requests

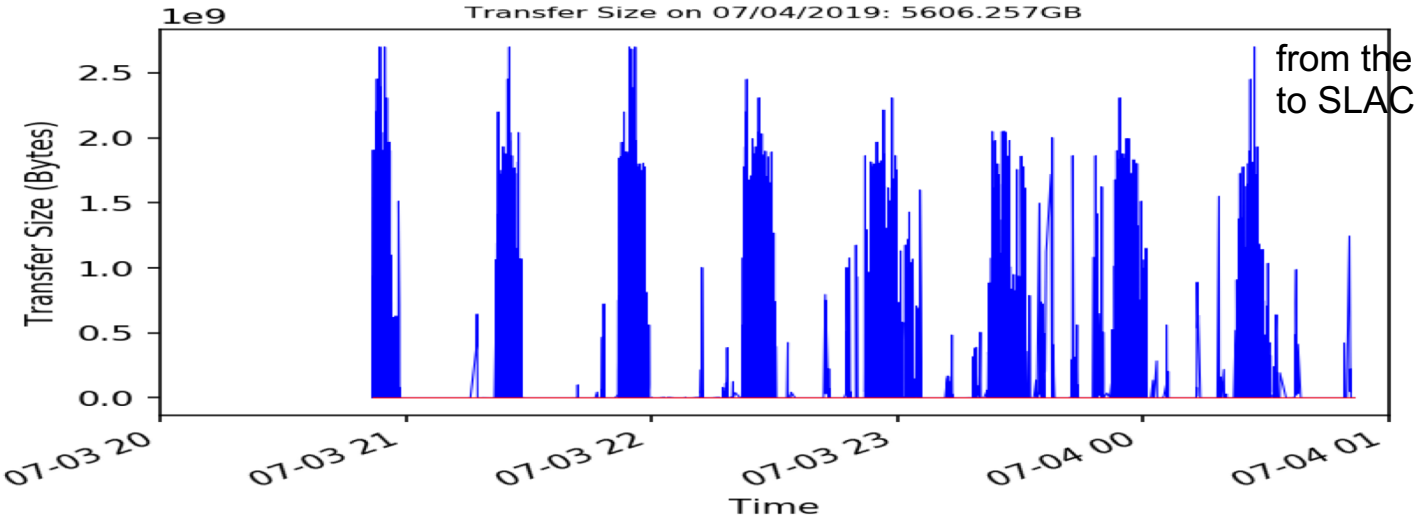
Analysis job #4

- **Retrieve data into the cache**
  - Connected 141 unique remote IPs for 3058 times
  - Transferred 236.94 GB into the cache node
  - Cached 60.13 % of the total dataset
    - (5923.75+1607.18+530.57+236.94)/13800
- **From the cache to the analysis job**
  - SLAC asked 29694 times to access data
  - Transferred out 5606.26 GB in total
  - ~0.1869 GB (=5606.26/30000) in each request.
- **Network traffic volume saving from various sites to the cache node**
  - ~5369.32 GB (=5606.26 - 236.94)

Analysis job #4 (2)
Transfer size over time

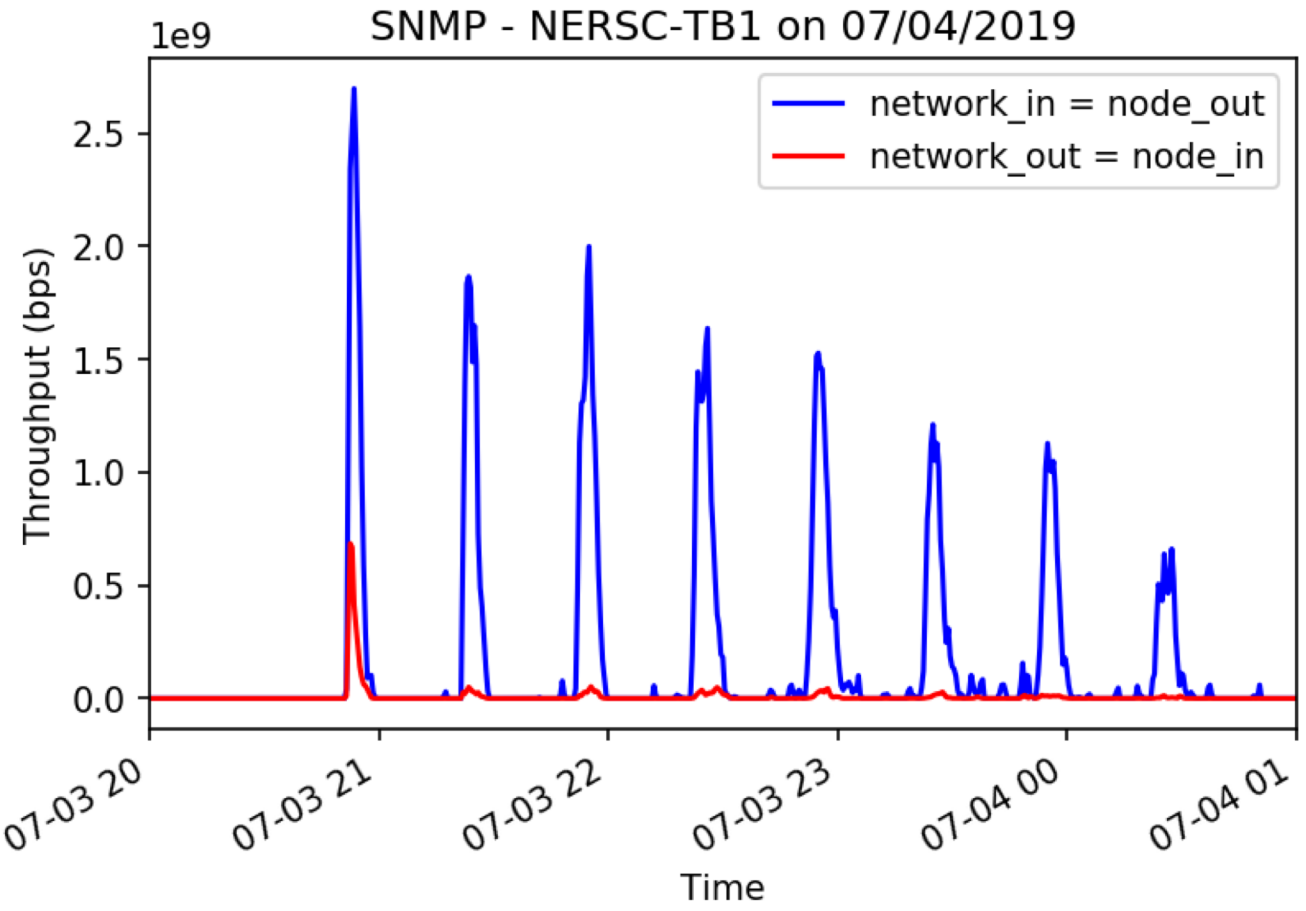


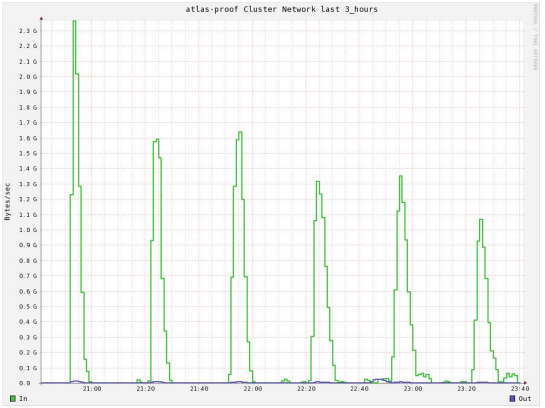
from remote sites
to the local cache



from the local cache
to SLAC analysis requests

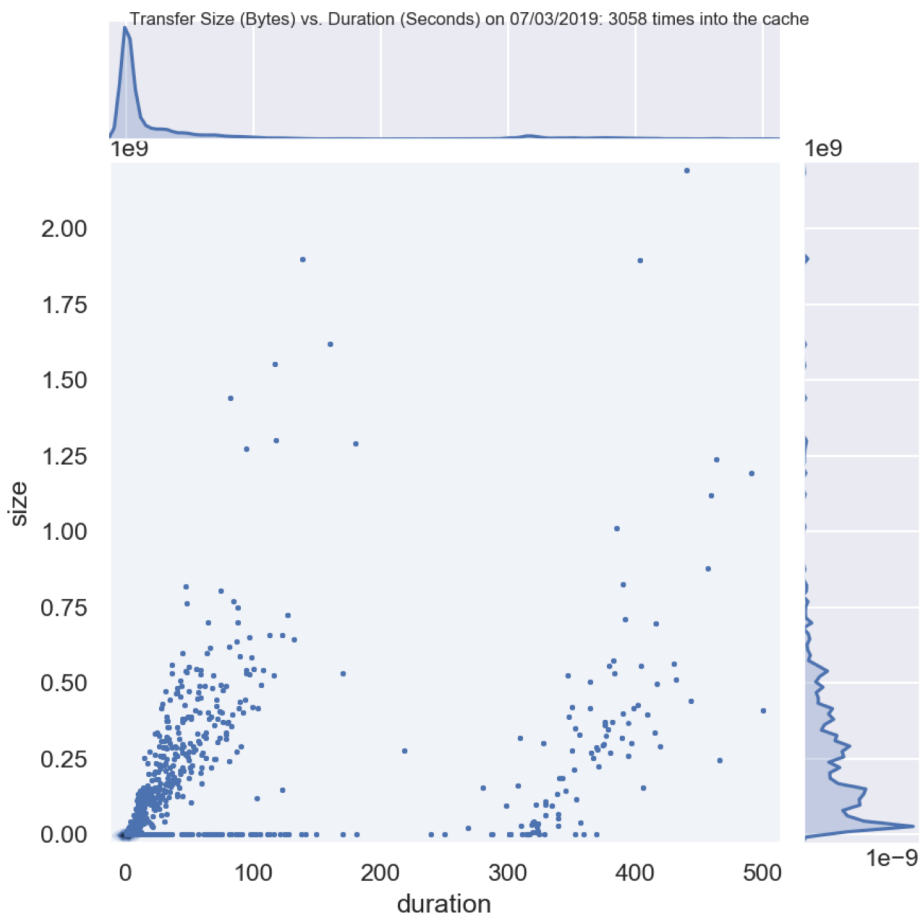
Analysis job #4 (3)
SNMP



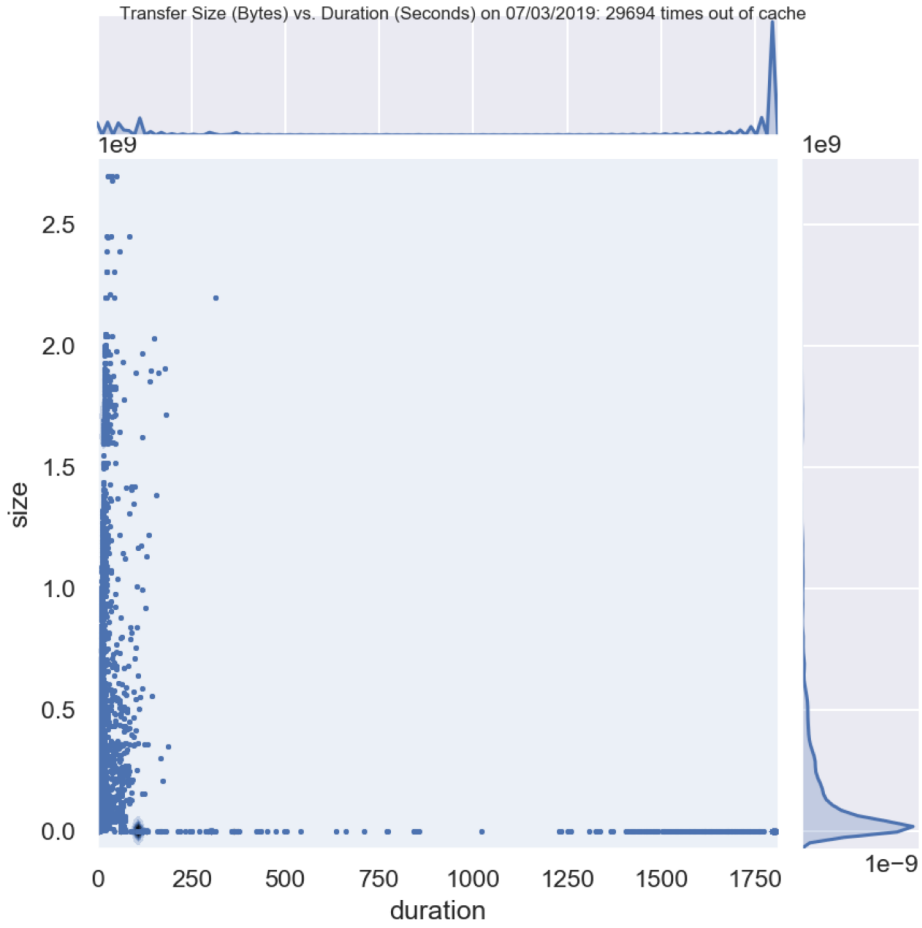


Analysis job #4 (4)

Transfer duration vs Data size with Distributions

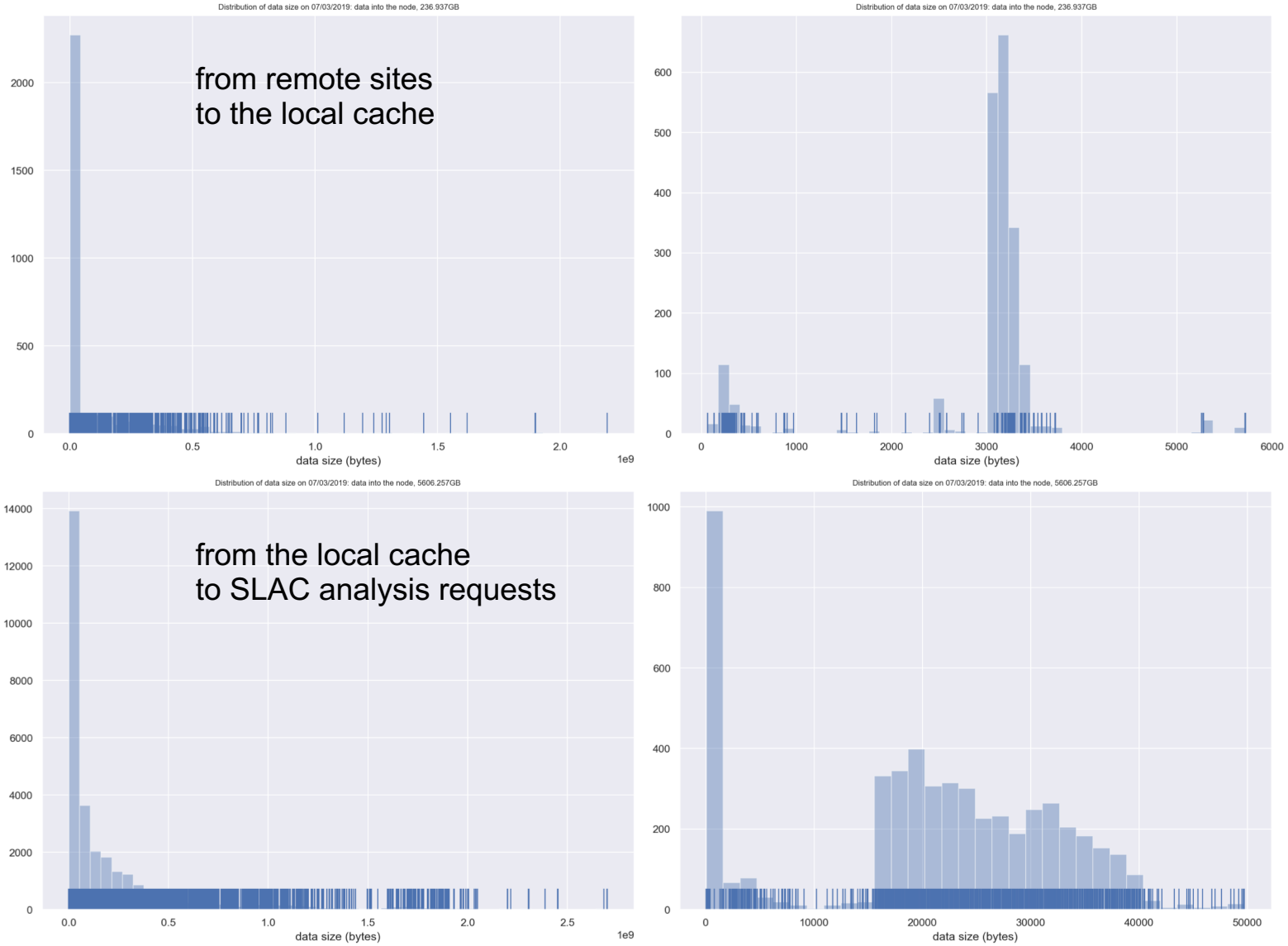


from remote sites to the local cache

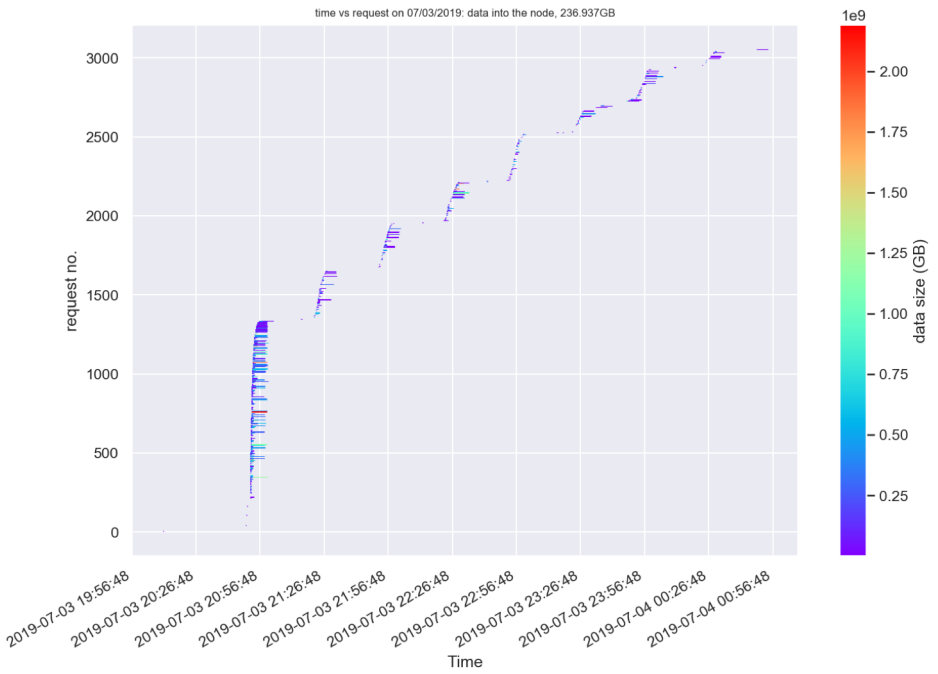


from the local cache to SLAC analysis requests

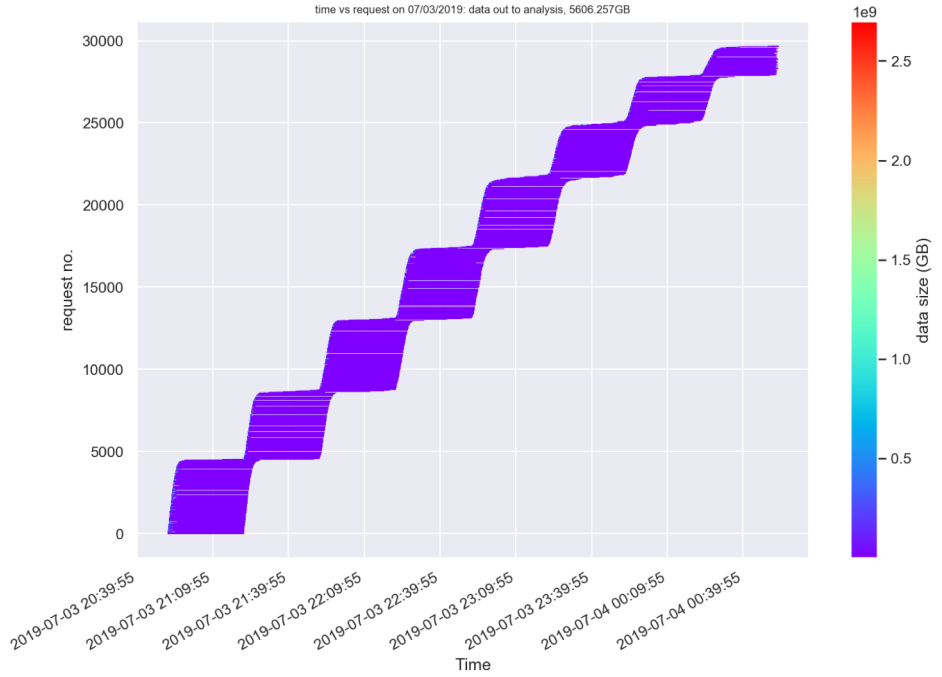
Analysis job #4 (5)
Transferred data size distribution



Analysis job #4 (6)
Transfer request completion time



from remote sites to the local cache

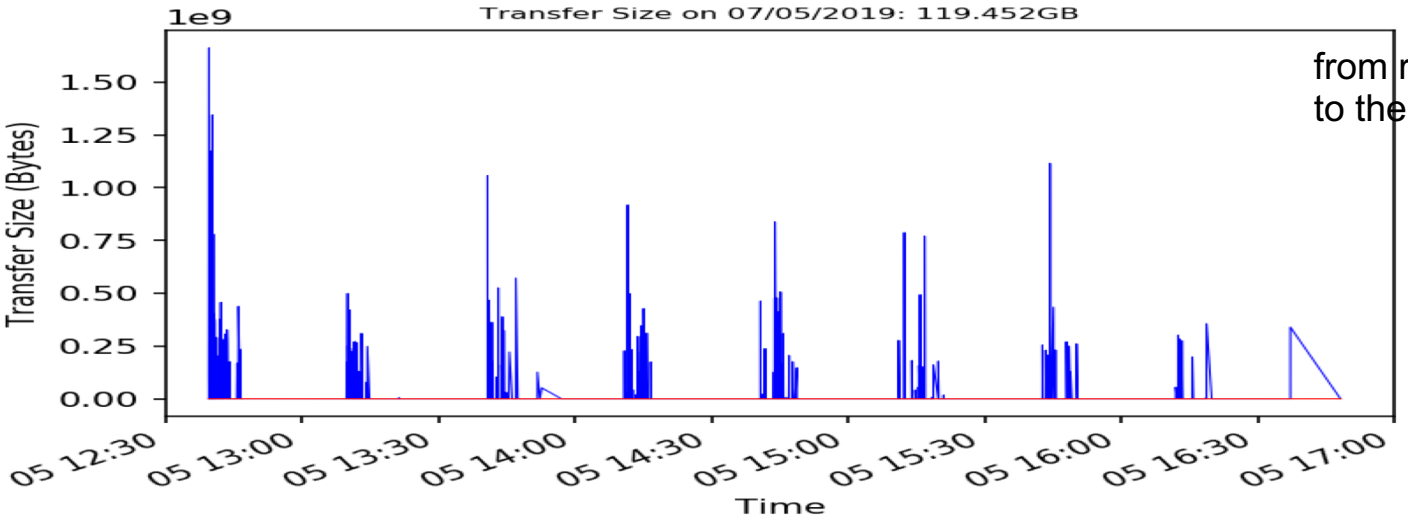


from the local cache to SLAC analysis requests

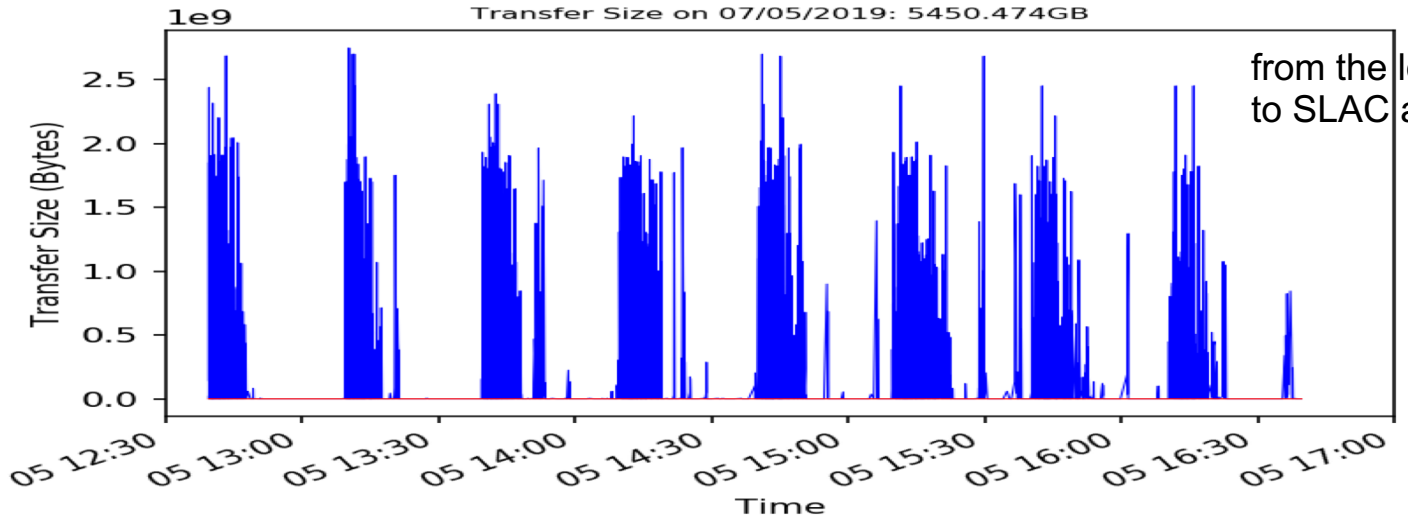
Analysis job #5

- **Retrieve data into the cache**
  - Connected 133 unique remote IPs for 2706 times
  - Transferred 119.45 GB into the cache node
  - Cached 60.99 % of the total dataset
    - (5923.75+1607.18+530.57+236.94+119.45)/13800
- **From the cache to the analysis job**
  - SLAC asked 29604 times to access data
  - Transferred out 5450.47 GB in total
  - ~0.1817 GB (=5450.47/30000) in each request.
- **Network traffic volume saving from various sites to the cache node**
  - ~5331.02 GB (=5450.47 - 119.45)

Analysis job #5 (2)
Transfer size over time

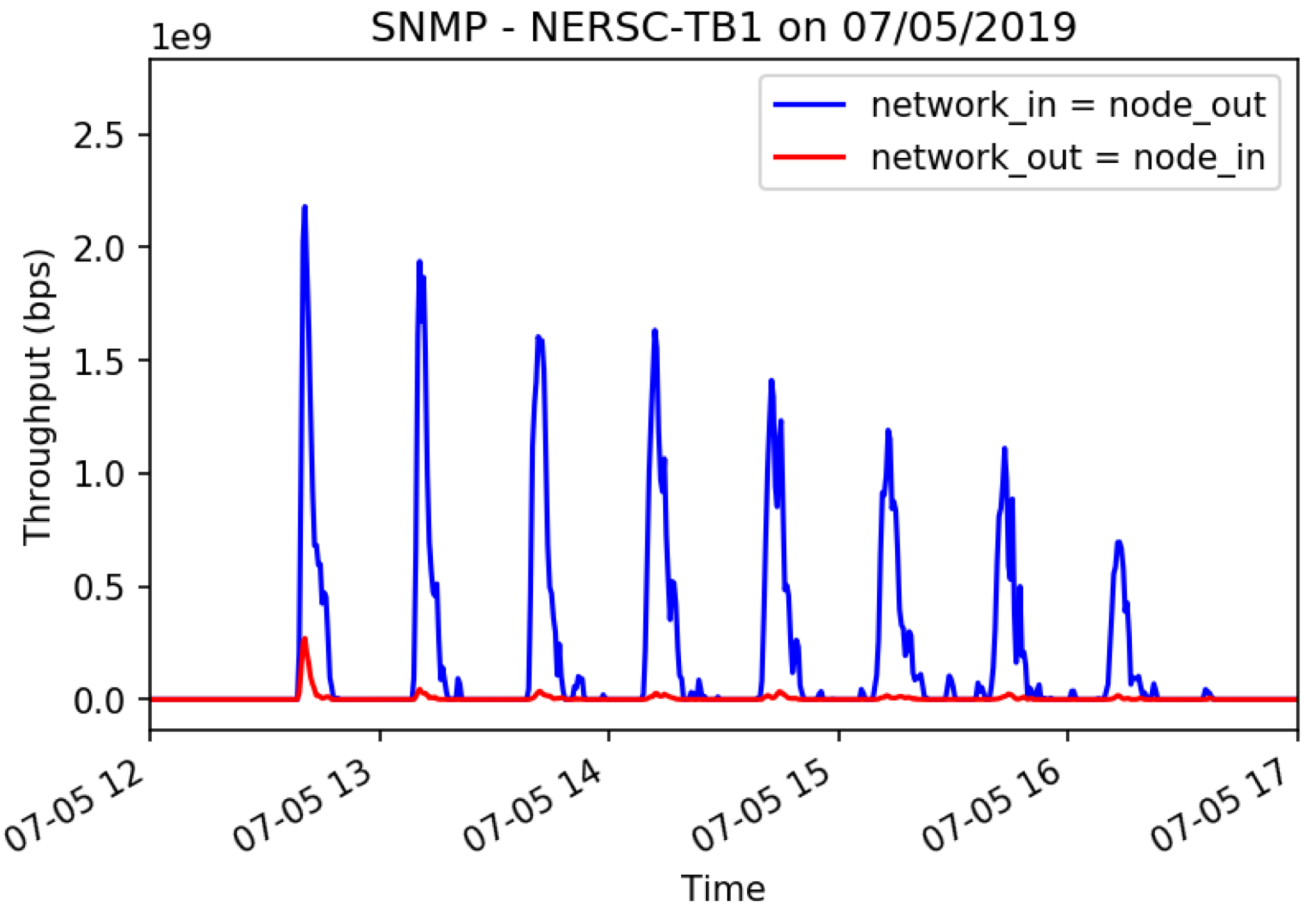


from remote sites
to the local cache



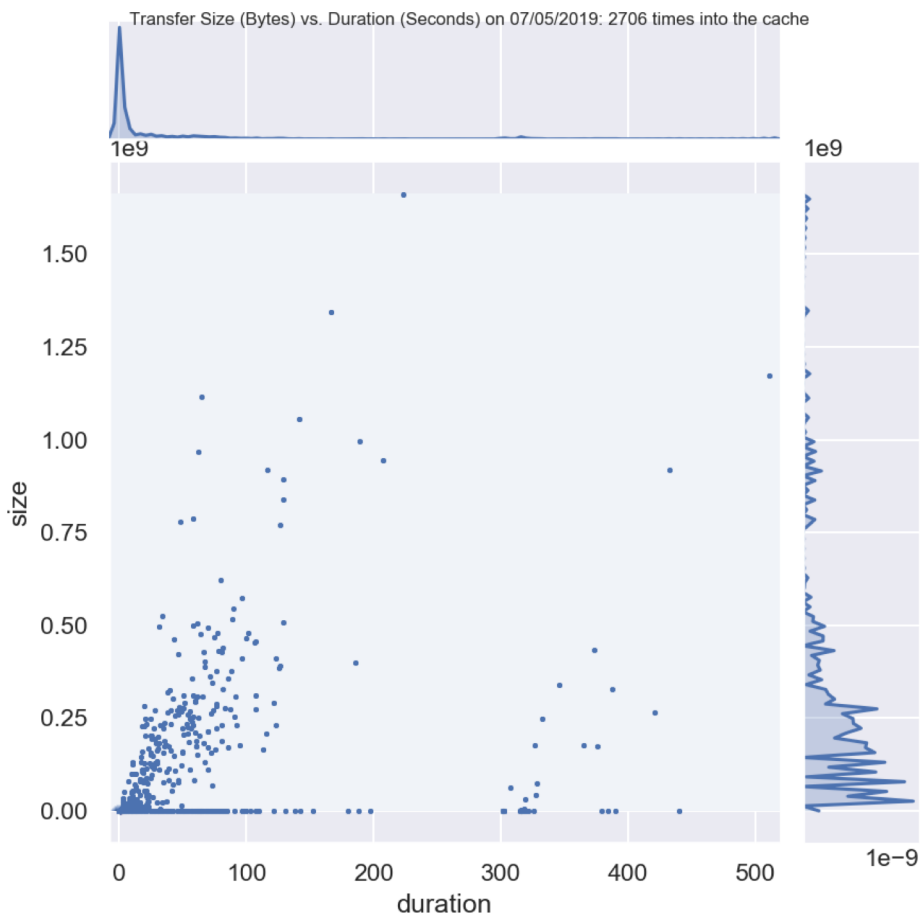
from the local cache
to SLAC analysis requests

Analysis job #5 (3)
SNMP

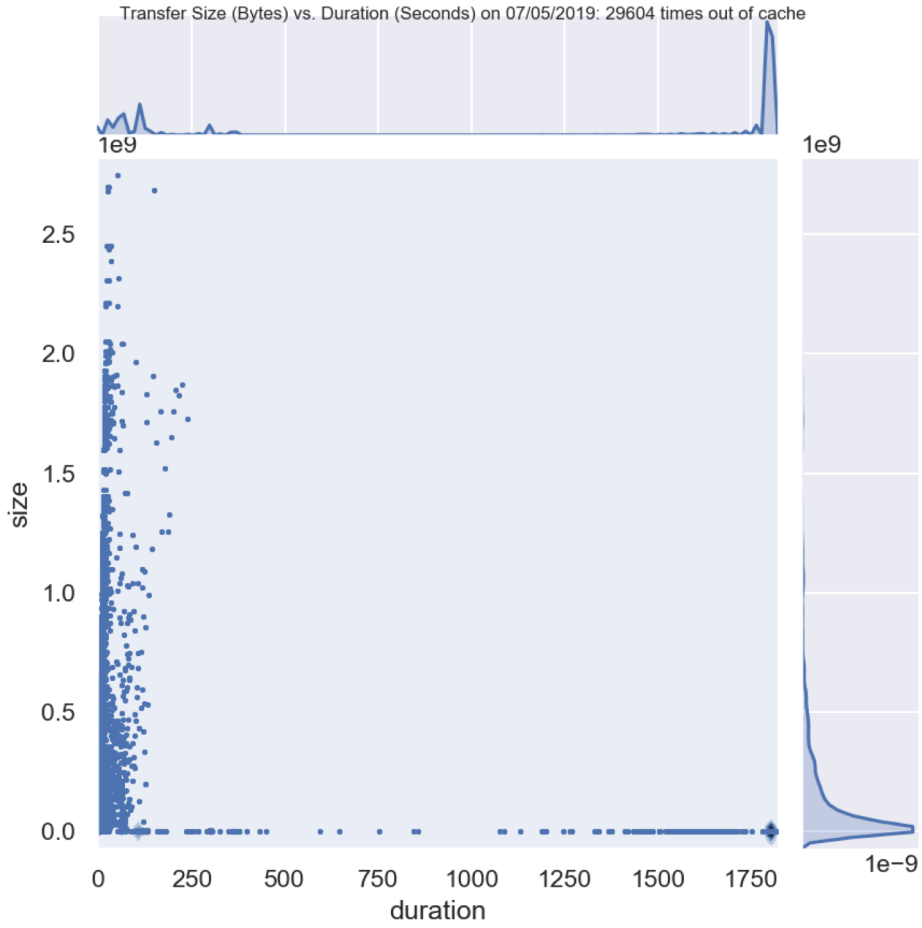


Analysis job #5 (4)

Transfer duration vs Data size with Distributions



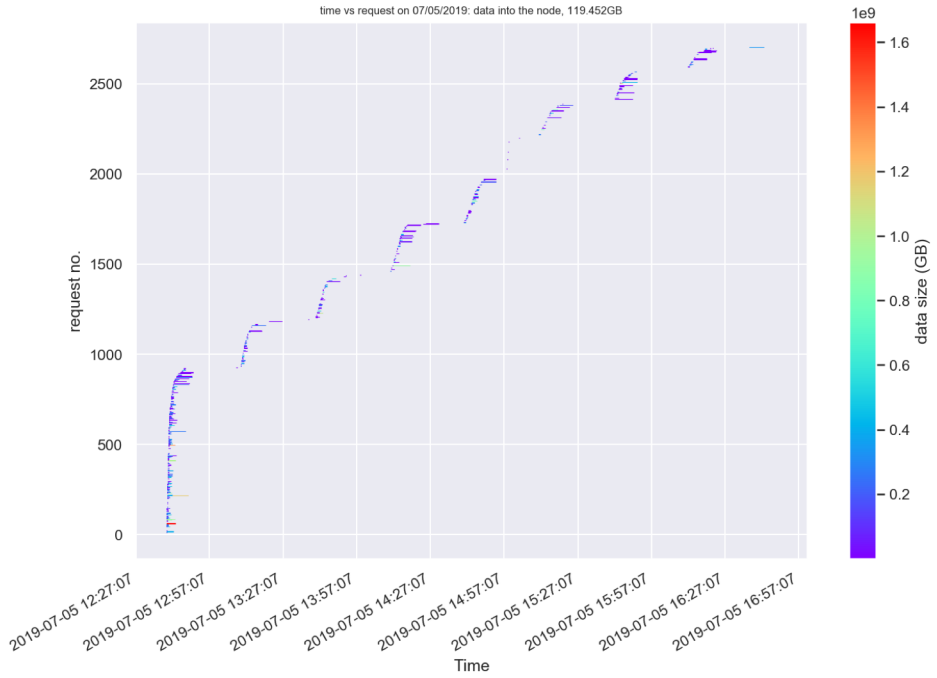
from remote sites to the local cache



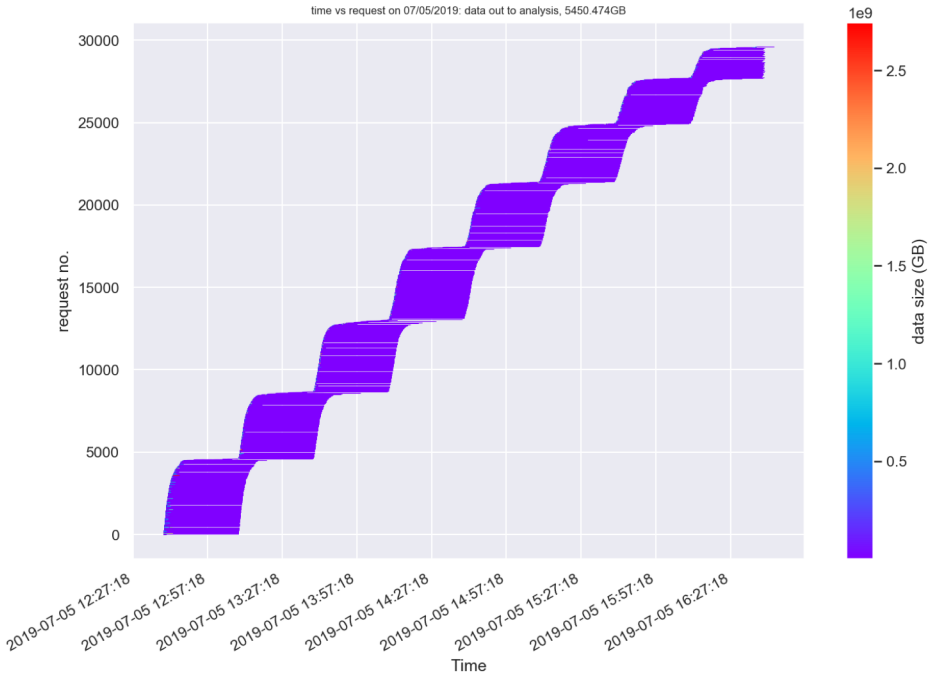
from the local cache to SLAC analysis requests

Analysis job #5 (5)
Transferred data size distribution

Analysis job #5 (6)
Transfer request completion time



from remote sites to the local cache



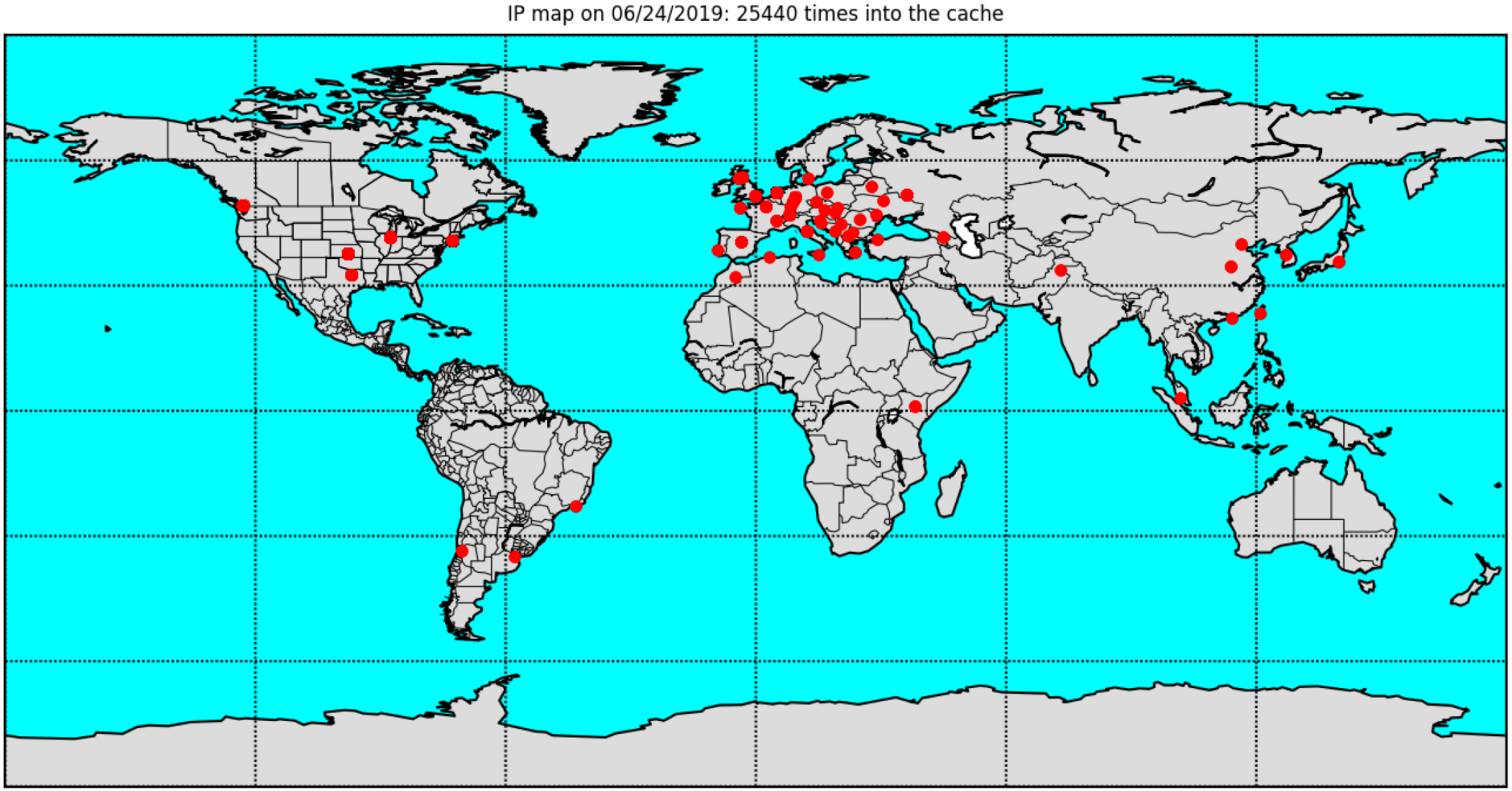
from the local cache to SLAC analysis requests

Summary

- Demonstrated the capability of a network based data cache
- Shared data caching mechanism
  - Reduced the redundant data transfers
  - Saved network traffic volume consequently
  - Experimental summary of the 5 jobs on 13.8 TB dataset
    - Saved 19.11349 TB of network traffic volume
    - Cached 8.41789TB out of 13.8TB (60.99%)
    - Each file is shared about 7 times on average
- Remaining questions
  - Why there are less than 30K requests for later jobs
    - Maybe 3 hour wall time limit
  - How many files are actually shared for how many times
  - Why it took so long to retrieve some data
  - These questions may be answered with Xrootd application logs

Backup

Analysis job #1 (7)
IP address map



- IP location accuracy is low. It shows (37.751, -97.822) for 134.79.129.116 (SLAC) with geoip2 (github.com/maxmind/GeoIP2-python) and DB from dev.maxmind.com/geoip. Same results as geoip-db.com/json.
- It should be (37.4201, -122.202) from db-ip.com or (37.4538, -122.1822) from ipinfo.io.