Event Services, Object Storage, and the ATLAS Experiment Experience at BNL

Hironori Ito
Brookhaven National Laboratory

Alexandr Zaytsev, Wen Guan, Doug Benjamin,
John Taylor Childers,Torre Wenus, etc...

Motivation for Object Store in ATLAS

- Its main goal is to effectively use opportunistic CPU cycles available around the world

  Opportunistic = non dedicated cycles

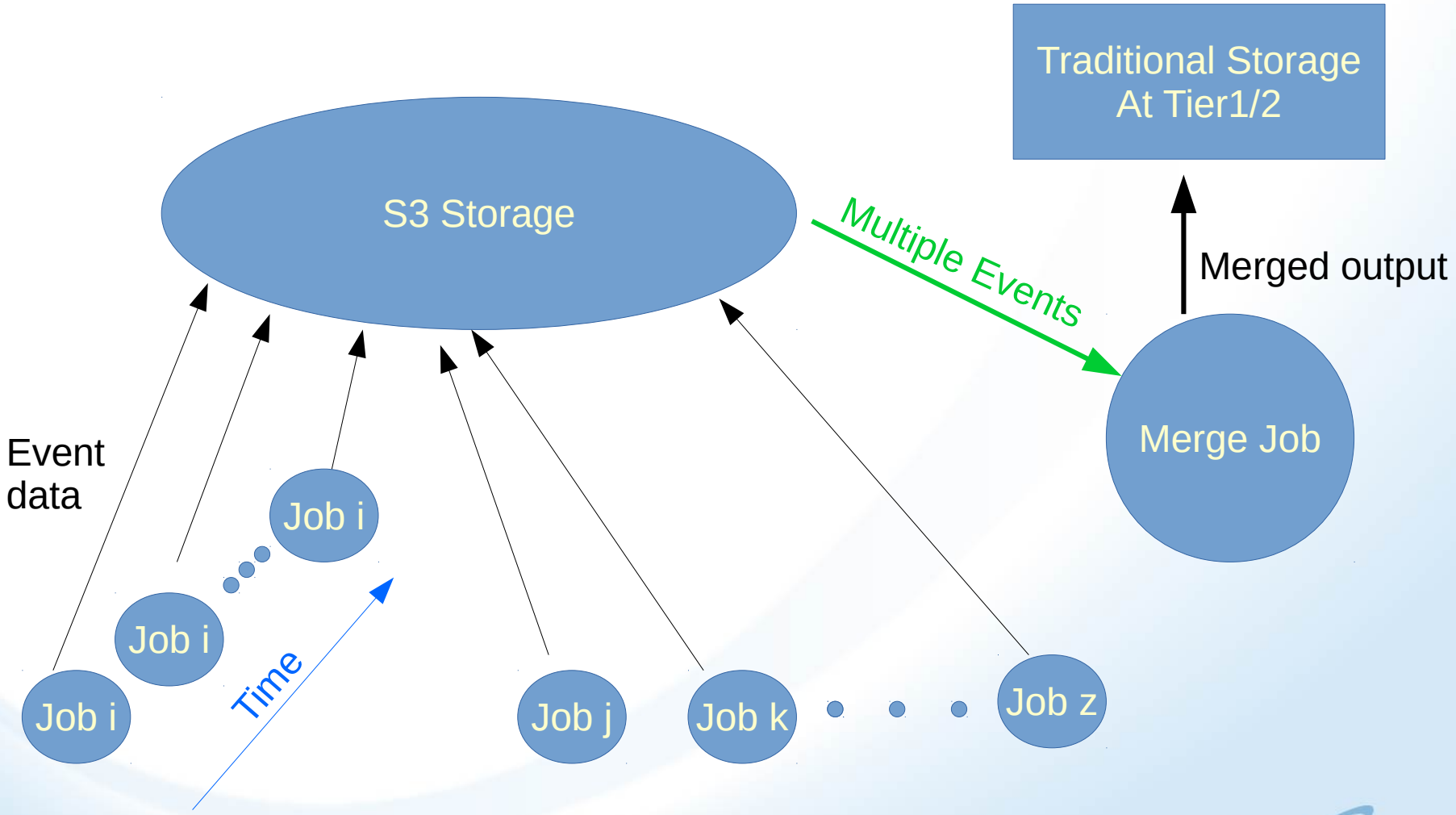
  - Commercial Clouds; Amazon, Google, etc…
    - Could even run free if your jobs get kicked out before the given time
  - HPC sites; Argonne Leadership Computing Facility, National Energy Research Scientific Computing Center,Oak Ridge Leadership Computing Facility, etc...
    - Possible large back-fill slots
  - Non-dedicated grid sites; Tier3, Non-ATLAS OSG sites, etc...
- Limitation of Non-dedicated sites
  - Lack of committed time for execution
    - Jobs could get kicked out from the execution before the completion

      → Wasted CPU cycles
- Solutions to reduce the waste: Check points
  - To write out results more often before the job gets kicked out
  - If kickd out, starts from the last check point and do only missing parts.

Object Store as Event Store Service

- In typical HEP data file, one of the smallest granularity of the data is an event. The file typically contains many events in the range of 10s to 100Ks or more.
- It is a matter of convenience as well as efficiency to store many events in one file. But, it is not necessary.
  - Traditional file system has large overhead or sever limitation of writing many small files.
- Object store without file system should have an advantage in storing many small files
- Event service
  - S3 storage using Ceph as well as Amazon is being used to store the data per event.
  - A singe job processing N number of events for a few hours can produce and store N number of outputs to S3 storage as it runs.
  - Many events are grouped together at later time (merging process) to produce a large file and store it on the traditional storage at the dedicated center (Tier1s/2s)
  - The time between the creation of the objects in S3 and merging process to produce larger files is expected to be short. As a result, S3 storage is really considered as a temporary data holder.

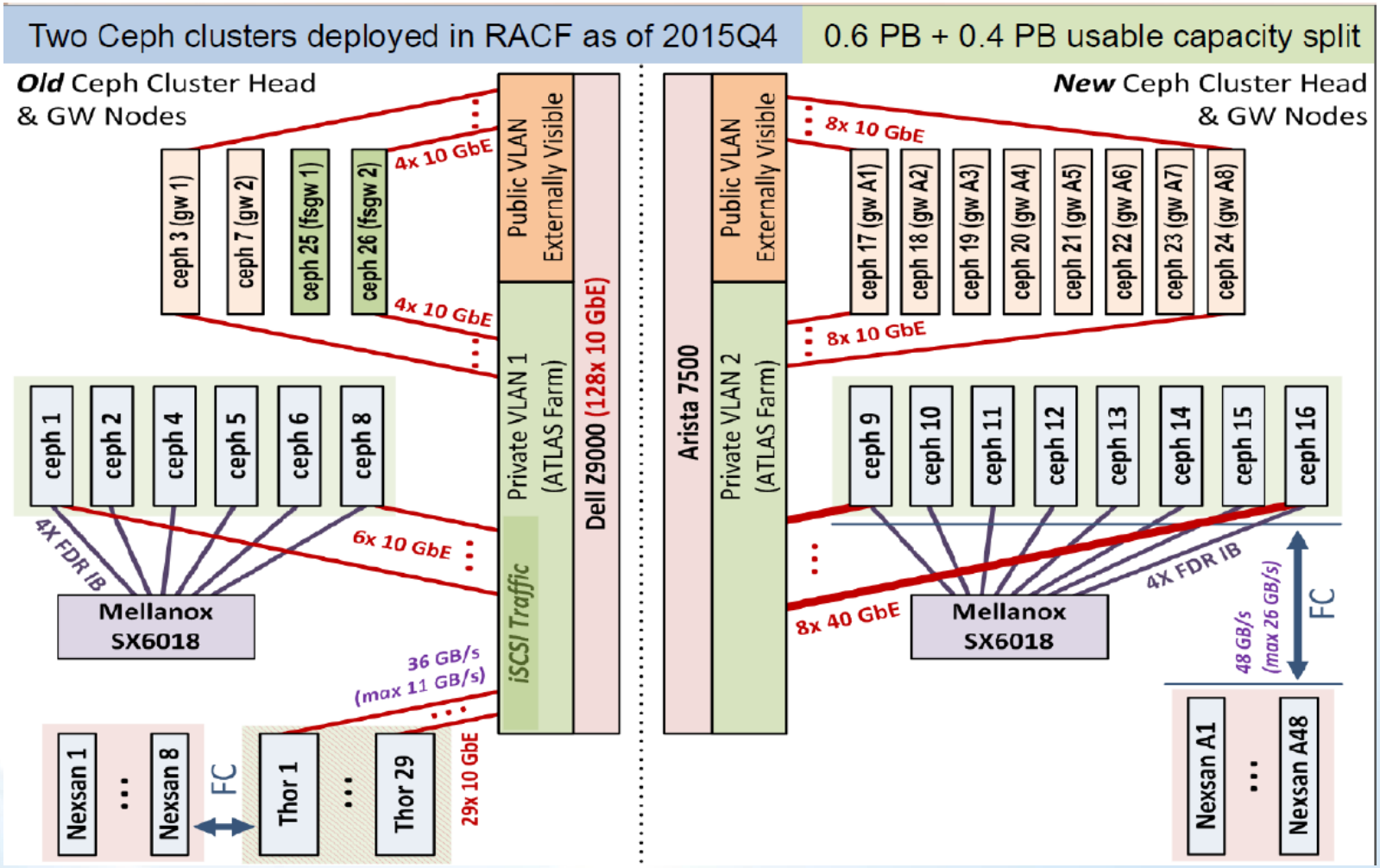
Event Service



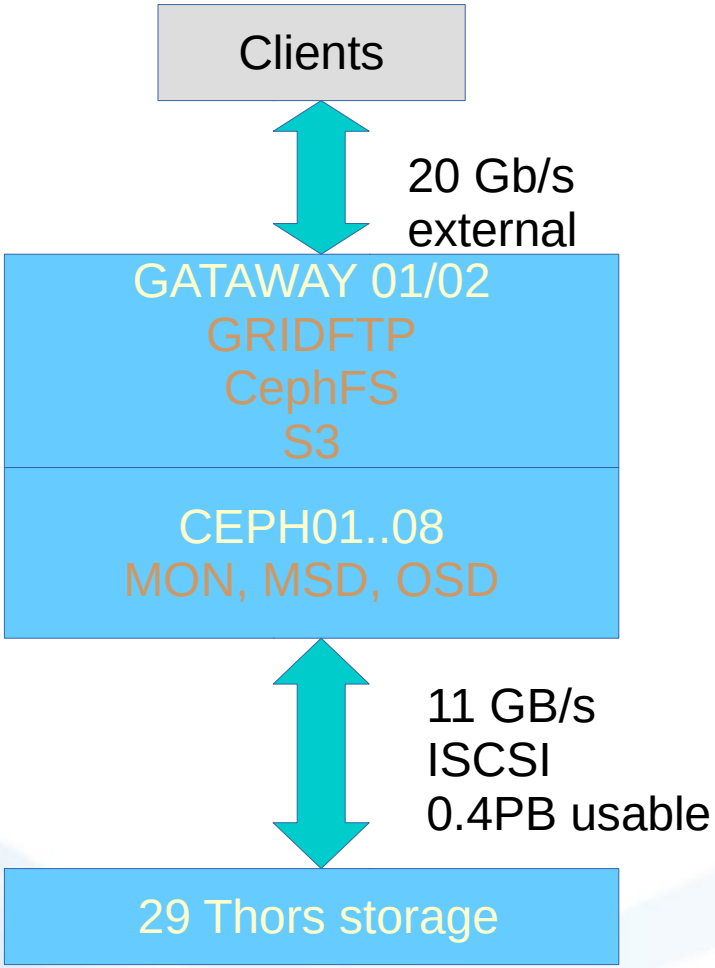
Ceph storage at BNL

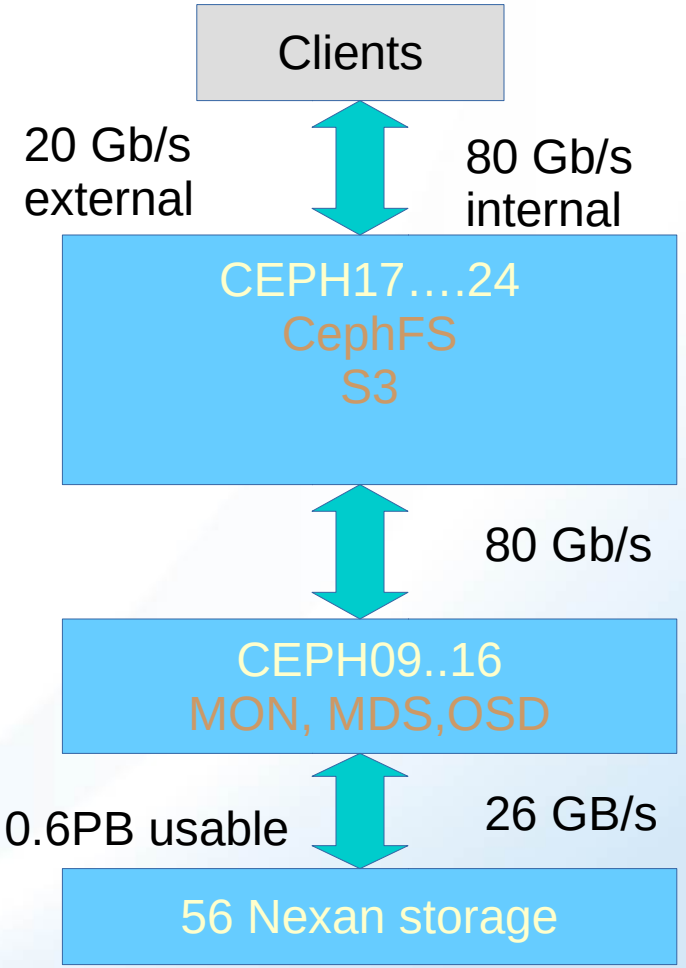
- BNL has been trying out Ceph for a last few years.
- Main goal is to make ourselves familiar with Ceph Storage as well as to provide the platform for other users to develop their use cases
  - BNL has been trying out all type of storage provided by Ceph; object storage, RBD, Cephfs
- The storage used for Ceph at BNL are mostly consisted of retired storage from Tier1 dCache service (5~6 years old)
  - Some new servers were purchased to attach these storage
  - Total of roughly 3PB (raw) are available as Ceph storage
  - Due to the use of old hardware with sever limitation, it was built primary to obtain enough capacity with reasonable data throughput
    - The setup is not designed to accommodate high random IO operation.

Hardware setup for Ceph at BNL

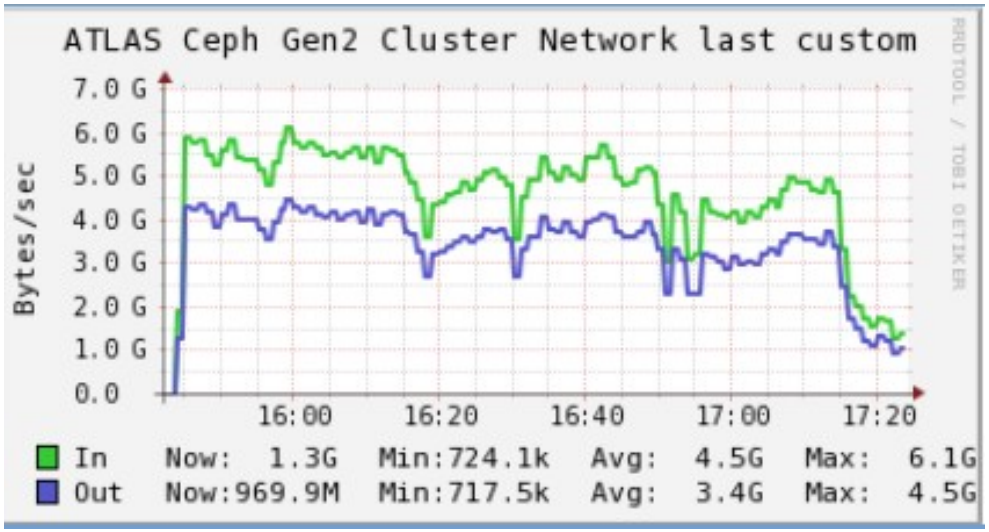


Hardware Setup and Limitation





Throughput performance of Ceph cluster at BNL



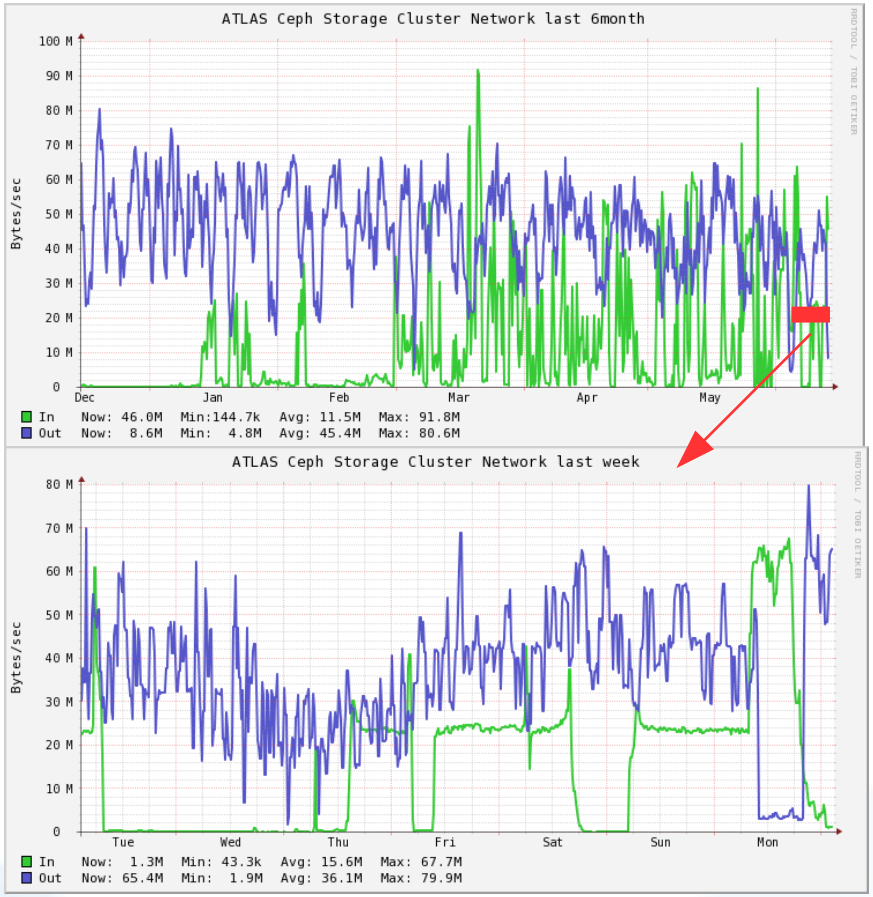
Actual total throughput is 1/3

Performance test of Cephfs using newer Cluster
74 concurrent clients with 1Gbps network
Writing large files to mounted directory in Cephfs

Event Service Tests

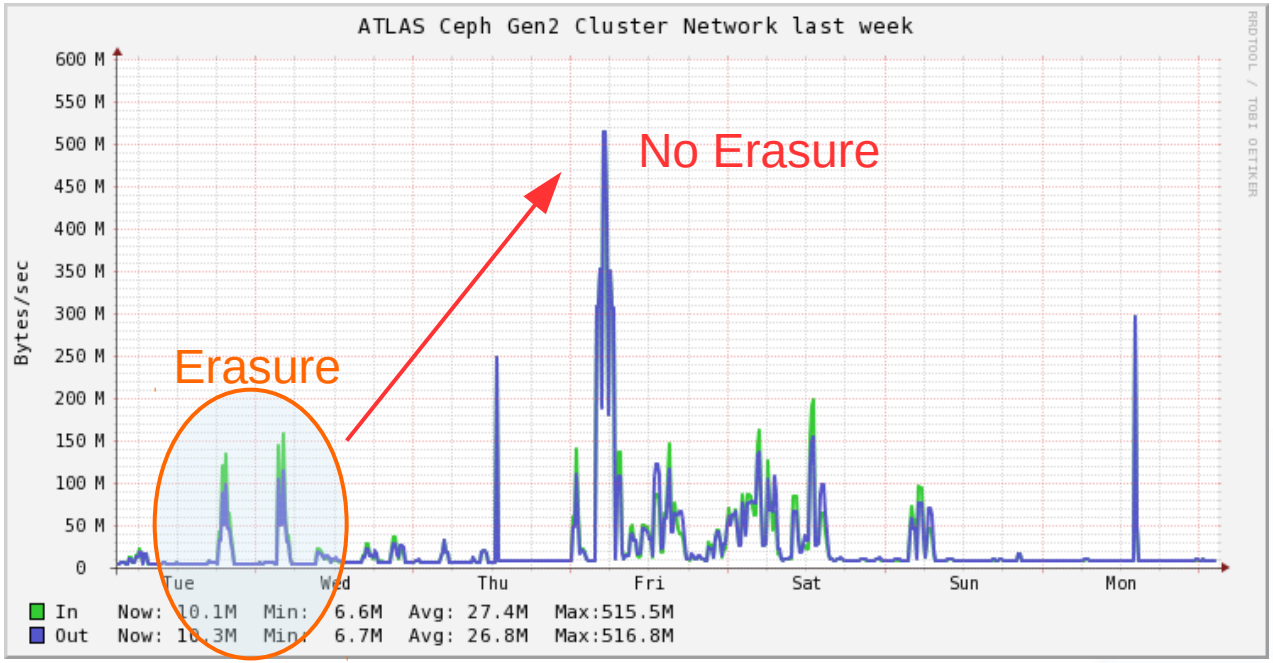
- NERSC HPCs
  - 700 nodes with 24 cores each ~17K job slots
    - Since each jobs are very similar and start about the same time, all jobs produce outputs in close proximity in time.
      - More bursty than regular jobs (?)
  - Each job process about 20 events in about hour
    - 700*24*20~340K objects per hour
  - Total amount of outputs are about 300GB
    - 300GB/340K ~ 1MB per object (very small compare with typical files)
  - RTT from BNL is ~70ms
  - Writing to the newer Ceph Cluster
- Use of opportunistic grid sites
  - Writing to the Ceph old cluster.
    - To split the load from the above HPC jobs
  - ~10K job slots
    - Each job starts and produces outputs at wider range of time

Current Status of older cluster



- The rate of activities has picked up in last few months
- Clients are seeing some timeouts, resulting in retry
  - Improvement is necessary.
- 5~10K job slots at BNL

Current Status of newer cluster



- Erasure code seemed to cause the large impact to the observed performance under our current setting
- Dropping erasure code and better tuning resulted in increase of client throughput by at least the factor of three
- Clients are still seeing timeout and retry, requiring further improvement

Future development

- Adding a new hardware and reconfigure to improve IO operation of mostly small objects
  - It is being used unlike the regular ATLAS storage where files with much larger size dominate total space and write IOs.
  - Cache Tiering?
    - Since it is very specific workload, it might significantly improve the performance
- Split the WAN network to BNL
  - BNL has 2 x 100 Gbps WAN; One active and one backup
    - If necessary, the backup can be used for network traffic between BNL and HPC sites, resulting in almost complete splits of regular ATLAS data traffic and event service traffic

Other developments using Ceph

- Use of Ceph as dCache storage pools
  - Take advantage of resiliency in Ceph with Swiss army knife of APIs from dCache
    - Familiarity of access modes and operation
- Use of Cephfs as resilient XRootD
  - XRootD without no single point of failures
    - No single data server with unique files because entire namespace are visible from all nodes