POSIX and Object Distributed Storage Systems

Performance Comparison Studies With Real-Life Scenarios in an Experimental Data Taking Context Leveraging OpenStack Swift & Ceph

by Michael Poat, Dr. Jerome Lauret, Wayne Betts

Outline

- Motivation
- Architecture of OpenStack Swift & Ceph
- Hardware & Configuration of Nodes
- Performance Measurements
  - Ceph Object Storage vs. OpenStack Swift Object Storage
  - Ceph Object Storage vs. Ceph POSIX File System
  - Scale the Ceph cluster, re-ran performance measurements
- POSIX CephFS Storage
  - Scalability & Stress Tests

Motivation

- STAR has been interested in distributed storage aggregation for years
  - Introduced Xrootd on the RCF main compute farm to harvest local storage in 2005 (deployment over 100ds data servers).
  - Investigated the use of HadoopFS for online / real-time usage
    E. Sangaline, J. Lauret – "Experience, use and performance measurement of the Hadoop File System in a typical Nuclear Physics analysis workflow", *J. Phys.: Conf. Ser.* **523** 012006 doi:10.1088/1742-6596/523/1/012006 (2014).
- Simple problem: How can STAR reuse the online hardware (local disks) to create a file storage system available to users?
  - Redundancy (data safety at the experiment's beamLine is a MUST)
  - Power outages / crashes resilient i.e. no data loss
  - Possibly POSIX compliant system in mind (best for users)
- Usage example: offer users a storage system where they could recover Raw DATA files into an FS space, process from anywhere for Quality Assurance or Online-Calibration purposes, dump result back in the FS, …

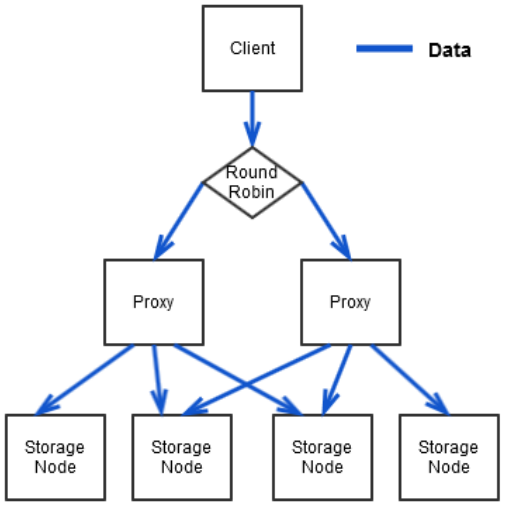
Processing pipeline done with a file system that can be accessed from anywhere online.

Architecture

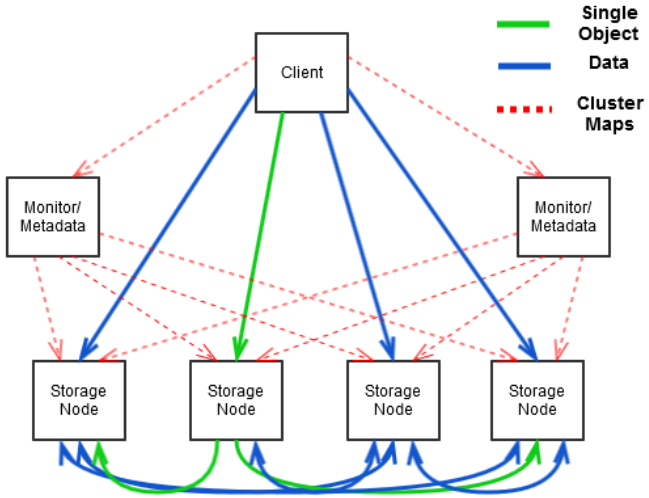
- ***OpenStack Swift*** transfers data through proxy servers which then distribute data to the Storage nodes.
- A Round Robin or Load Balancer must be used to distribute work load when using multiple proxies.
- The proxy nodes will be more CPU & I/O intensive than storage nodes due to data transferring.

- ***Ceph*** clients connect directly to the Storage nodes eliminating any bottleneck.
- Instead of proxies like Swift, Ceph uses monitors that distribute cluster maps to the clients and storage nodes.
- Cluster maps are guidelines for placement of data files.
- The monitor service can be run on same node running the OSD services.

OpenStack Swift

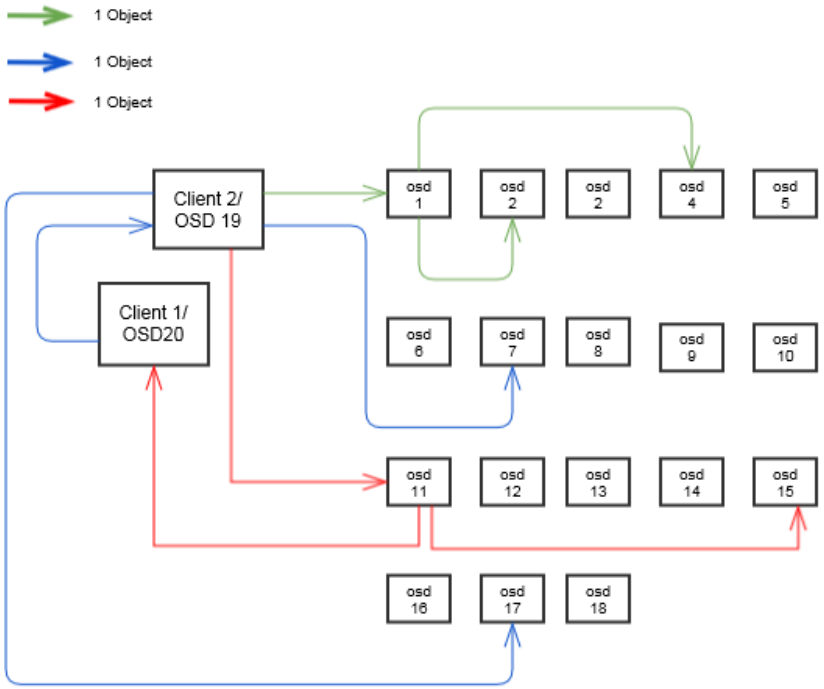


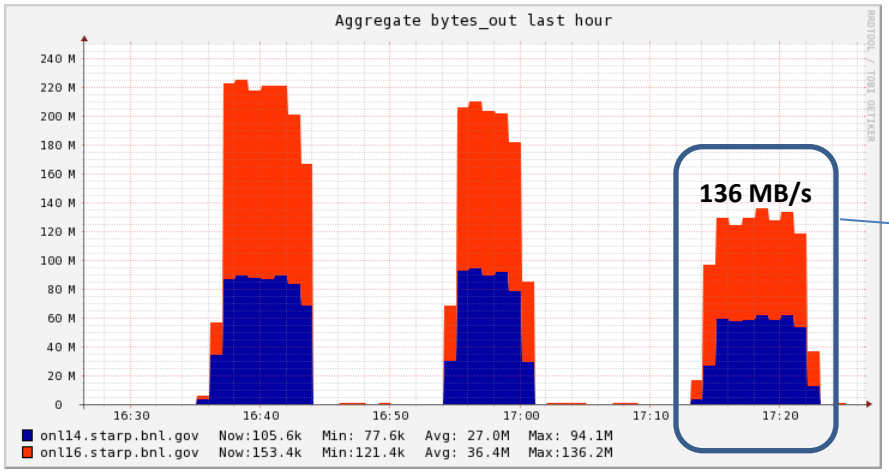
Ceph

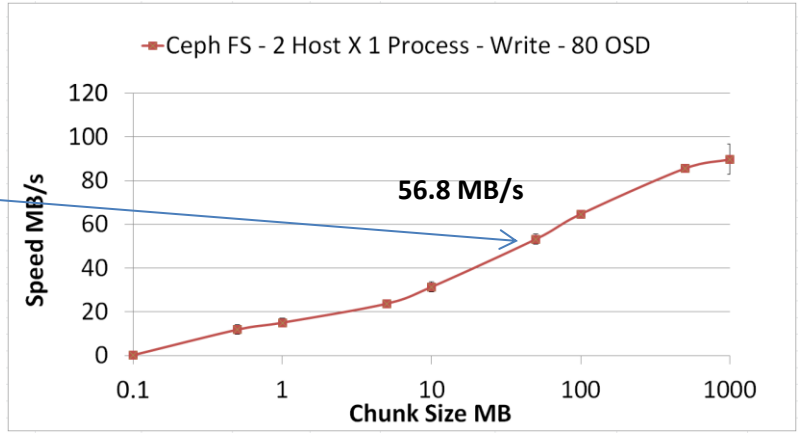


Ceph Communications

- Testbed: Total 20 Storage nodes. 2 nodes acting as clients and OSD's.
- Therefore 2/20 (10%) chance a Client will receive data from other OSD's that needs to be sent back out twice for replication (total replication 3).
- 2/20 * 2 = .2
- Total data output increases by .2
- Ex: 56.8MB/s * 2client * 1.2 = **136.32 MB/s** aggregate speed
- Ceph I/O is predictable and can be deduced from simple math.







Testbed

- 20 Dell PowerEdge 2950
- 6 – 2TB Seagate SAS drives
  - 4 Drives per node used for storage
  - XFS File System
- 2 Intel Xeon QC E5440 – 2.83GHZ
- Scientific Linux 6.6 x86_64
- Kernel: 3.10.67-1 (needed to mount CephFS)
  - Kernel: 2.6.32 latest version released by SL6
- 1 Gb Network Link

Configuration

OpenStack Swift

- Openstack Havana
- 40 Storage Daemons – 2TB each (10 nodes)
- 1 -> 2 -> 4 Proxy Servers
- Using Replication 3

Ceph

- Ceph Firefly 0.80.9 (LTS)
- 40 OSD (10 nodes) -> 80 OSD (20 nodes)
- 1 Monitor Server & 1 Metadata Server
- Using Replication 3

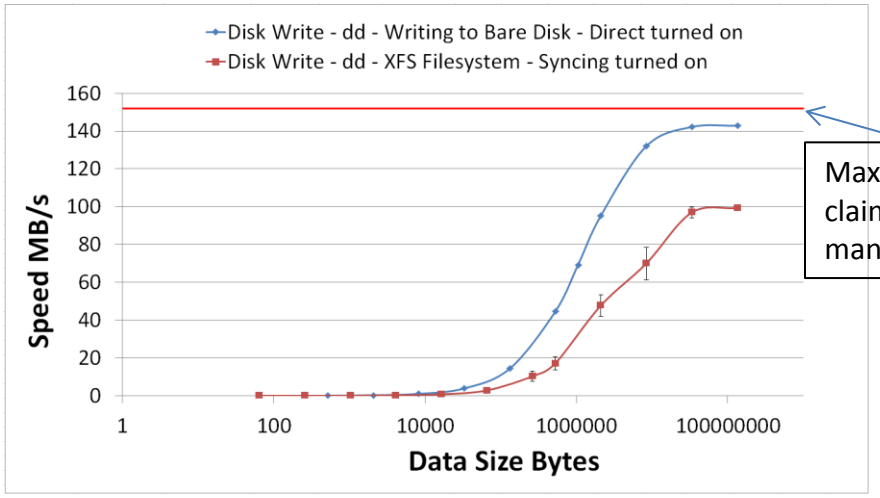
Tools and Tests

- Performance testing was done by scripts to simulate uses of the cluster.
- Utilized the built-in object store commands to interact with the cluster.
  - Swift -- `/usr/bin/swift`
  - Ceph -- `/usr/bin/rados`
- `dd` tests were performed to measure I/O speed of the CephFS and the I/O of the disks.
- Simulated real use of both clusters by running multiple processes on multiple client machines simultaneously.

`dd if=/dev/zero of=/dev/sdc bs=8388608 count=1000 oflag=direct`

`dd if=/dev/zero of=/mnt/sda bs=8388608 count=1000 oflag=sync`

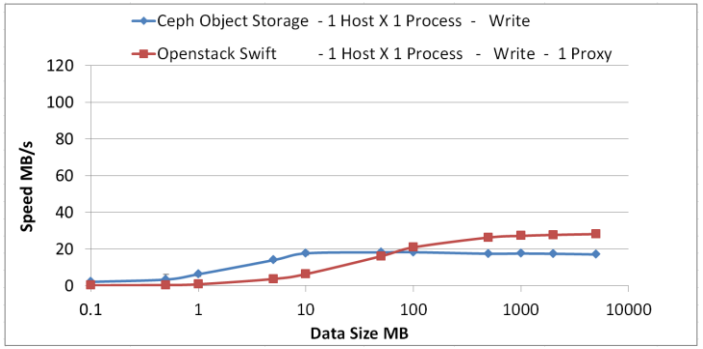
I/O Performance of 1 Disk



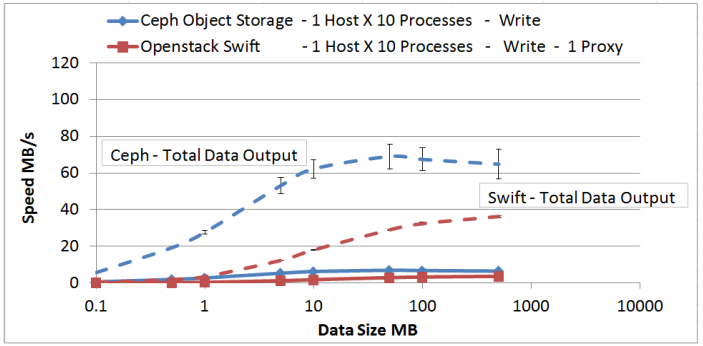
Object Storage I/O Comparison – Single Host

Swift vs. Ceph Object – Write Performance

- The Ceph I/O Performance scales over Swift due to client connecting directly to storage nodes.
- Swift I/O limited due to bottleneck of a single proxy node.
- Object storage breaks the object down into small files, storing multiple copies across the storage nodes.

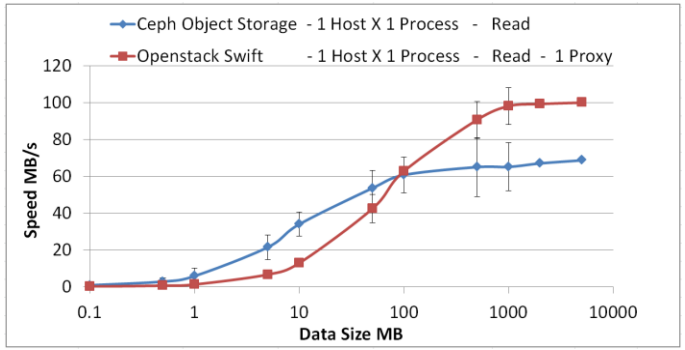


Write

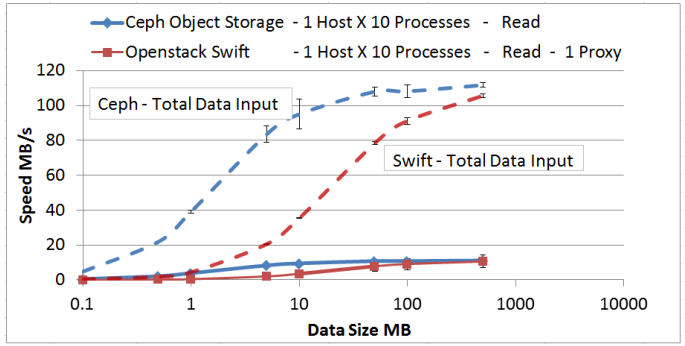


Swift vs. Ceph Object – Read Performance

- Ceph and OpenStack Swift object storage systems reassemble data on the fly when reading.
- In Ceph, when reading a single file the data is passed from a single storage node to the client.
- In Swift, when reading a single file the data is passed from the storage nodes, through the proxy then to the client.



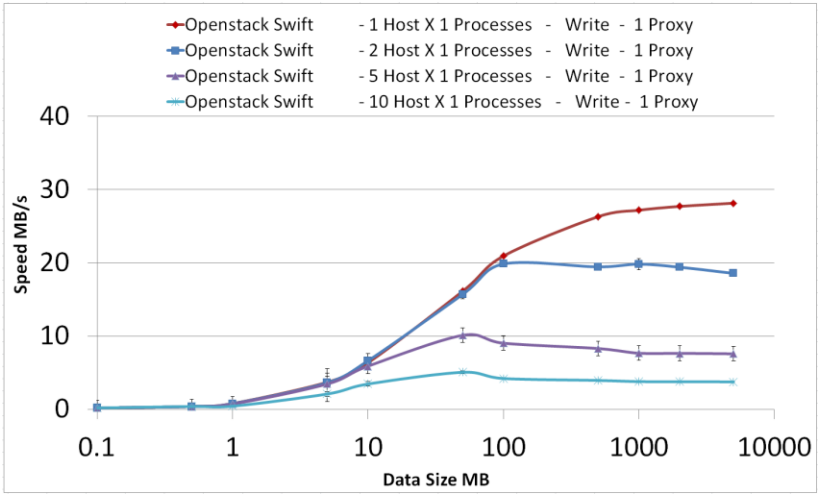
Read



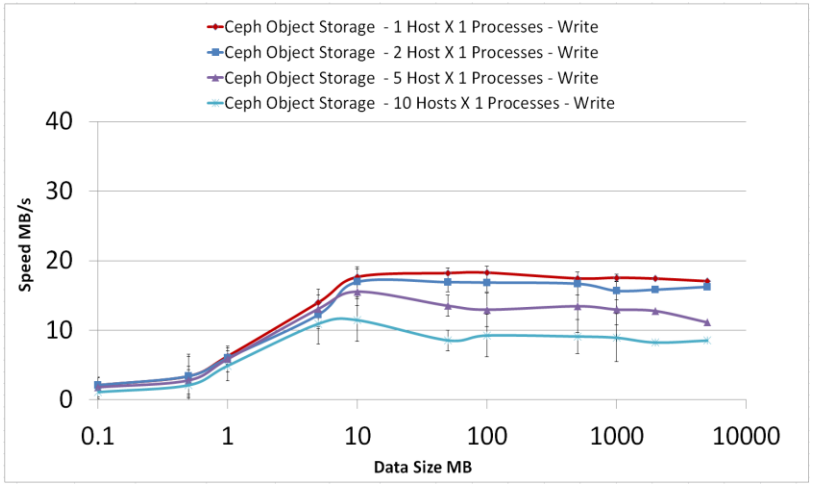
Object Storage I/O Comparison – Multiple Clients

Swift vs. Ceph – Multiple Client - Single File Write

- Comparison of Ceph Object Storage to OpenStack Swift Object Storage scaling from single to multiple clients.
- In a single user environment Swift would be ideal, however Ceph would be better for multi user environment as there is less performance degradation as clients increase.



Compare



With process concurrency, Ceph wins!

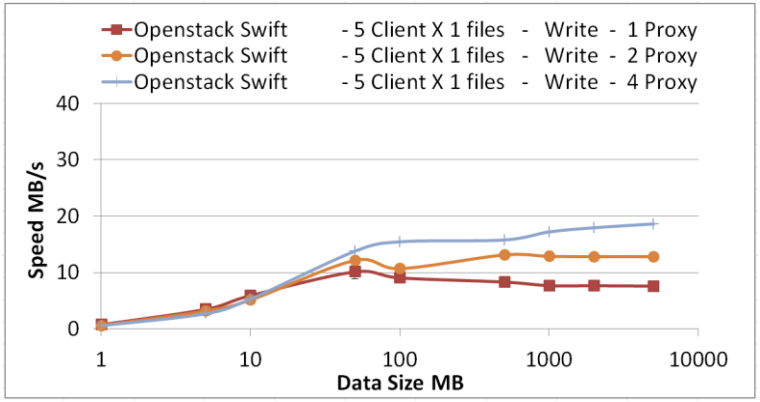
Performance Considerations

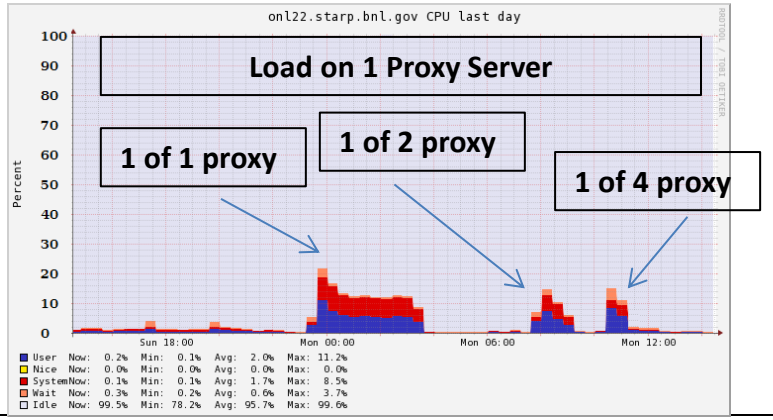
Ceph

- Adding private backbone network for immediate data replication reduces network workload.
- The Monitor service in Ceph is very light weight, it is possible to run the monitor service(s) and the OSD service(s) on the same node.

OpenStack Swift

- The addition of a backbone network for replication and inter-server communication is possible.
- Adding multiple proxy nodes will act as a load balancer reducing the I/O per node.
- Additional proxy servers are recommended to be dedicated proxy servers only.
- *Dedicating a fair amount of proxy servers for our cluster would make us lose 15% - 20% of total storage (1 : 5 – Proxy Node : Storage Node ratio) – not good*



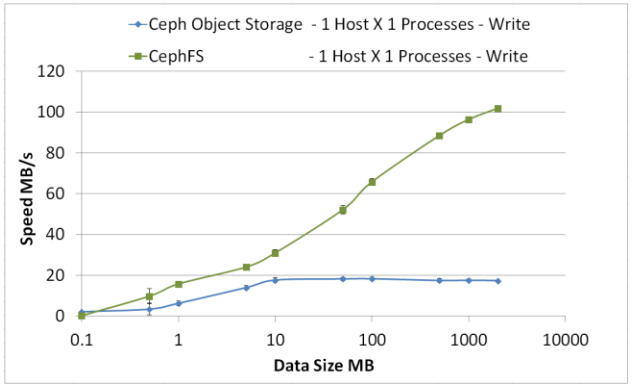


General Considerations

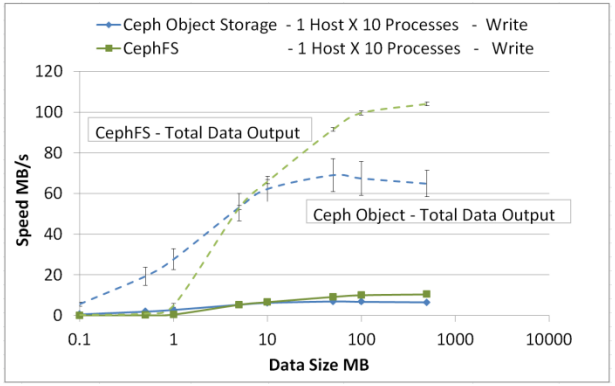
- STAR's online enclave is a multi use cluster.
- Openstack Swift proxy servers would have to be dedicated to the storage cluster, Ceph monitor servers can handle multi use.

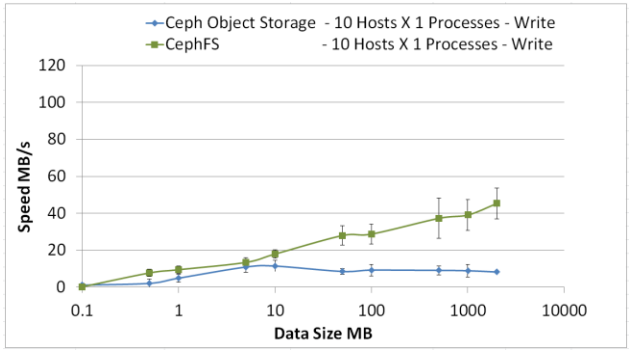
Ceph Object Storage vs. CephFS (POSIX)

- CephFS does scale over Ceph Object storage with a 1 host, 1 write process at a time scenario.
- CephFS will open multiple connections to Storage nodes when writing 1 file at a time, where as a client using Ceph object storage will only open 1 connection to 1 storage node at a time.
- Performance of CephFS writes when using memory caching will rise above the theoretical limit of disk I/O and network bandwidth. `dd` with syncing turned on will ensure the data is actually written.
- Using the `/bin/cp` command, the CLI returns while the I/O on the back end is still transferring data.

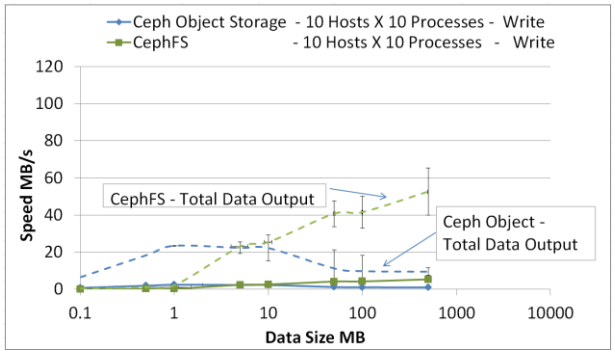


1 Client



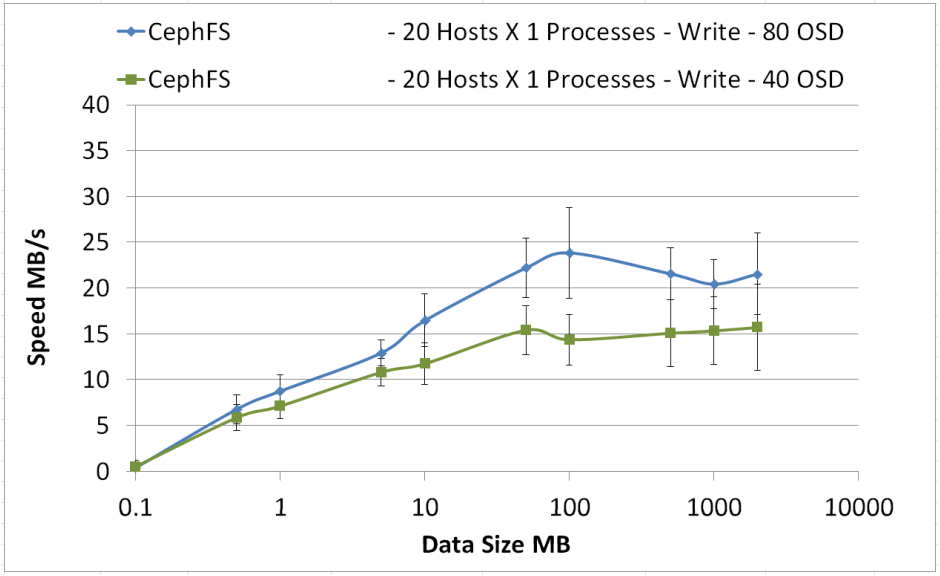


10 Client



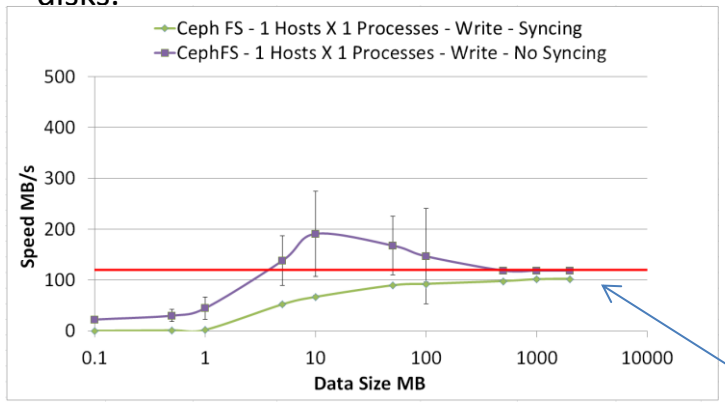
CephFS 40 OSD vs. CephFS 80 OSD

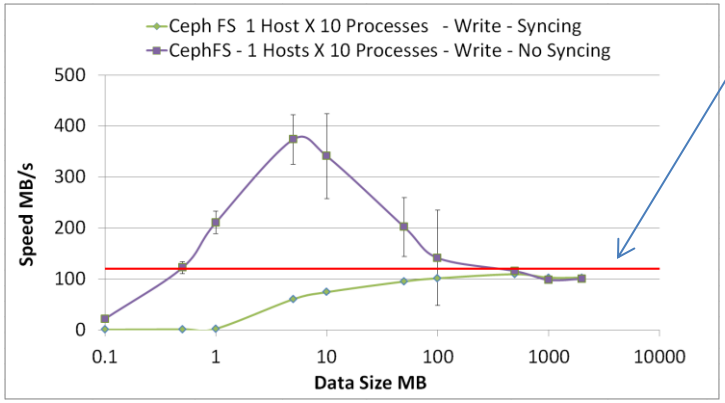
- Scaling the storage OSDs by x2.
- Write performance with 80 OSDs does scale over 40 OSDs with heavy I/O.
- It has been reported by Ceph, and the community that as you increase the number of OSDs the I/O performance of the cluster will increase.



CephFS 80 OSD Write Syncing vs. No Write Syncing

- Do users actually use syncing when writing data?
- Write performance is significantly faster with syncing off.
- With syncing turned off using `dd` the I/O reaches above the theoretical limit of the network link and disks.





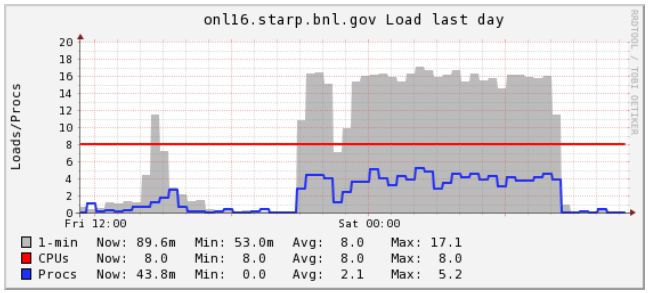
1Gb Network

Stress Tests

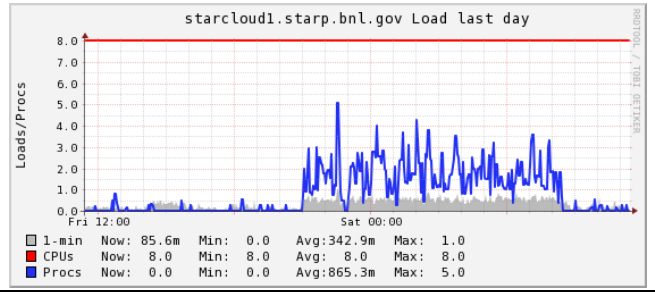
Maximizing the IOPs

- Lots of small I/O (lots of Ops) over a long period of time often causes problems for File Systems.
- Tested: 20 hosts running 8 `dd` processes at once (one per core) writing
  - 1 kilobyte for 1 million counts
  - Also did 1 byte x 1 M times – result identical
- 220 I/O Ops per node / sec (4400 IOPs for the entire cluster) inferred from time the test took.

Client & Storage Node

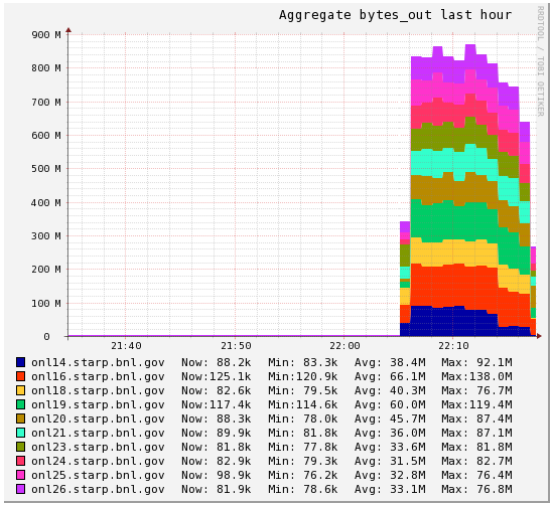


Monitor Node

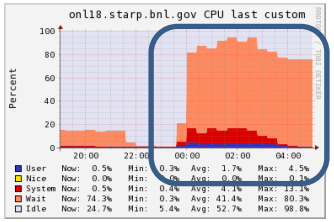


Maximizing the Throughput (+Lets Break It)

- 10 clients using CephFS (also half of storage nodes), 50 `dd` write processes per host.
- Scaled the block sizes from 100 KB to 100 MB.



Total IO ~ 850 MB/sec , balanced across the cluster + High load (IO Wait)

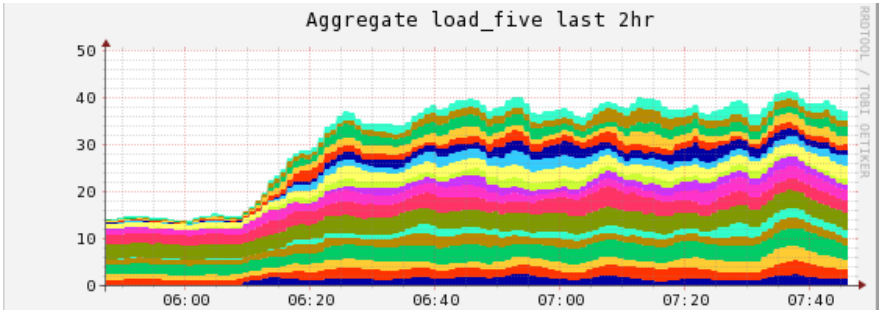


- Then asked the hosts for 50 X 500MB writes i.e. make it run out of memory (on purpose), and see.
- Nodes became unresponsive. Decided to shutdown nodes by hand and reboot.
- Ceph automatically salvaged the cluster after reboot (~1 hour for `Ceph -s ---> health HEALTH_OK`).
- **No data loss!**

Multi Use Cluster

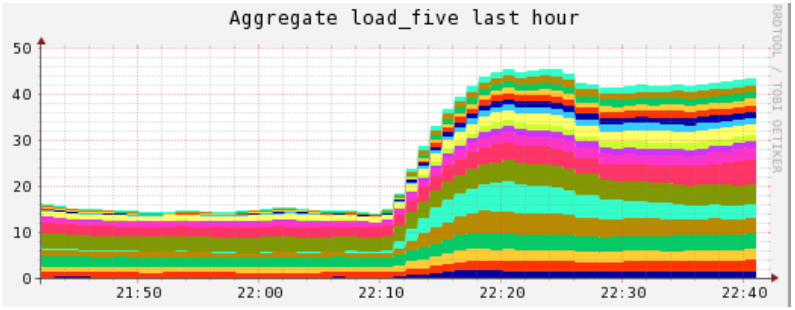
- Ceph OSD's sitting idle do inflict a 1% – 2% load on the CPU.
- Ceph OSD's will also perform 'scrubbing' which ensures data integrity (Light scrubbing daily, deep scrubbing weekly).

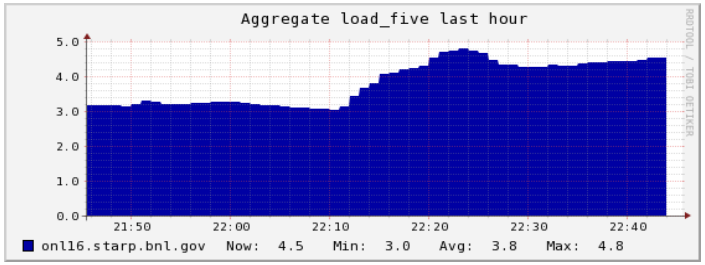
Deep Scrubbing across Cluster

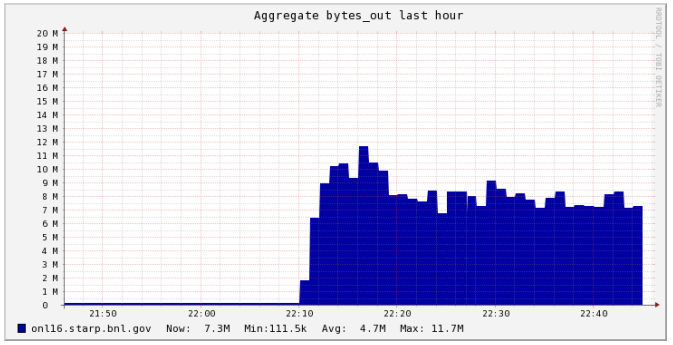


- We may be able to run jobs and the Ceph FS on our nodes simultaneously without overloading the nodes.
- Ceph does consume a lot of resource in heavy I/O scenarios.
- It is likely that we can use both but the Ceph load is non negligible.

- Utilizing all CPU for Ceph 1MB `dd` writes to the cluster only inflicts a small load on the overall cluster.







Conclusion

Performance summary

- Both object storage systems have similar performance with single file writes although Ceph Object Storage seems to have a limit with larger files (Swift is a better fit for large files)
- Ceph Object Storage outperforms OpenStack Swift with multiple file writing – Ceph scales better with IO concurrency
- With default parameters, CephFS scales better comparing to Ceph Object Storage – the difference is significant (NB: tuning of Object storage or striping may lead to different results). With large files, CephFS is x5 the performance of Ceph Object Storage
- While a better choice, Ceph does have a CPU load impact on your system (scrubbing or light load ⇔ ~ 30% CPU impact in our tests) – As for Xrootd in STAR, mixing compute power and storage possible within limits.
- Recycled storage on our testbed cluster shows 1.4 GB / sec (replication 3) ⇔ 400 MB/sec

Outcome & Perspectives

- CephFS offers versatility for our uses. Easy to mount with minimal install (later kernel and a few Ceph packages) + offers a POSIX interface that seem to outperform the Object Store out of the box.
- With Ceph POSIX storage system, we will be able to offer roughly 80TB of storage using replication 3 to all Online nodes. High reliability & availability, crash-safe tested, …
- We begun replacing the network with a 10 Gb backbone + initial testing of journals mounted on SSD's, may look promising. Will investigate further.