Ceph - Distributed Storage System

Rahul Raghatate[1] **and Snehal Chemburkar**[1]

[1] *School of Informatics and Computing, Bloomington, IN 47408, U.S.A.*
[*] *Corresponding authors: rragahta@iu.edu, snehchem@iu.edu*

paper-1, February 28, 2017

Ceph is a unique storage solution that delivers all critical storage system capabilities:open-source, software-defined, enterprise-class and unified storage (object, block, file). Ceph being highly reliable, easy to manage and free, possesses power to transform company's IT infrastructure and ability to manage vast amounts of data. Ceph's extraordinary scalability has provided clients with accessibility to petabytes to exabytes of data. Moreover basic enterprise storage features including: replication (or erasure coding), snapshots, thin provisioning, auto-tiering (ability to shift data between flash and hard drives), self-healing capabilities has resulted in allowing it to be a reliable big data storage platform. This article explores these salient features Ceph as well as provides study of Ceph architecture and its uniqueness in comparison to few of the existing Large Scale Systems.

Keywords: Ceph, scalability, storage, exabytes

https://github.com/rahulraghatate/sp17-i524/paper1/S17-IR-2026/report.pdf

INTRODUCTION

Developing a distributed file system(DFS) in today's world of exponentially growing data is not a handy job. However, if the right issues are addressed and resolved, it becomes immensely valuable and foundation stone in any business success. Although most of the DFS offer a similar set of features, they also can provide unique features which allow them to stand distinct.

For IT Decision Makers, inadequate storage infrastructure stand out to be the fourth out of the top ten pain points. It's interesting to know that 74% of IT decision makers are worried about their organization's ability to cope with an increasing volume of data, and 70% believe that their current storage systems will not be able to handle next generation workloads [1]. Hence, Enterprises in due struggle to manage the explosive growth of data while remaining agile and cost competitive are turning to cloud technology to store their data.

As a self-healing, self-managing platform with no single point of failure, Red Hat Ceph Storage significantly lowers the cost of storing enterprise data in the cloud and helps enterprises manage their exponential data growth in an automated fashion [1].

Ceph is open-source storage platform providing highly scalable object, block as well as file-based storage. It is a unified, distributed storage system designed for excellent performance, reliability and scalability [2].

Ceph has emerged as one of the best storage ecosystem which initially began as a PhD research project in storage systems by Sage Weil at the University of California, Santa Cruz (UCSC). The name Ceph comes from Cephalopod, a class of mollusks.

ARCHITECTURE

The Ceph architecture consists of four subsystems:

- File System Clients
- Cluster of metadata servers(MDS)
- RADOS which includes Monitor Services and object storage devices(OSDs)
- Data distribution system using CRUSH

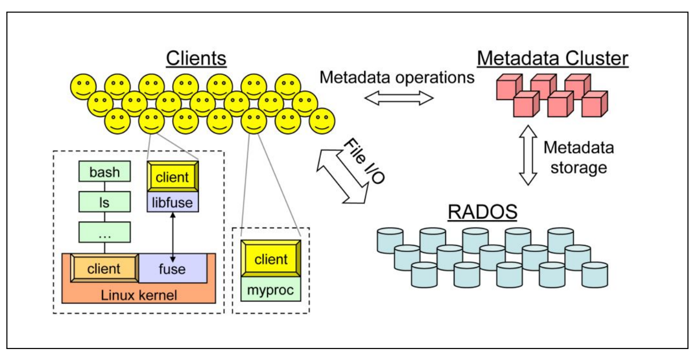

Fig. 1. System Layout of Ceph [3]

.

Client Operation

Ceph depends upon Ceph Clients and Ceph OSD Daemons having knowledge of the cluster topology, which is inclusive of 5 maps collectively referred to as the "Cluster Map".The Ceph Client is the user of the Ceph file system. The Ceph client runs on each host executing application code and exposes a file system interface to applications. Ceph has a user-level client as well as a kernel client. The user-level client is either linked directly to the application or used via FUSE. The kernel client allows the Ceph file system to be mounted. Each client maintains its own file data cache, independent of the kernel page or buffer caches, making it accessible to applications that link to the client directly [3].

The Ceph metadata server(MDS)

Ceph provides a cluster of metadata servers which continually load-balances itself using dynamic subtree partitioning [4]. The responsibility for managing the namespace hierarchy is adaptively and intelligently distributed among tens or even hundreds of metadata servers. The key to the MDS cluster's adaptability is that Ceph metadata items are very small and can be moved around quickly. To enable failure recovery, the MDS journals metadata updates to OSDs. The mapping of metadata servers to namespace is performed in Ceph using dynamic subtree partitioning, which allows Ceph to adapt to changing workloads (migrating namespaces between metadata servers) while preserving locality for performance. Rebalancing of the MDS at even extreme workload changes is usually accomplished within a few seconds. Clients are notified of relevant partition updates whenever they communicate with the MDS [5].

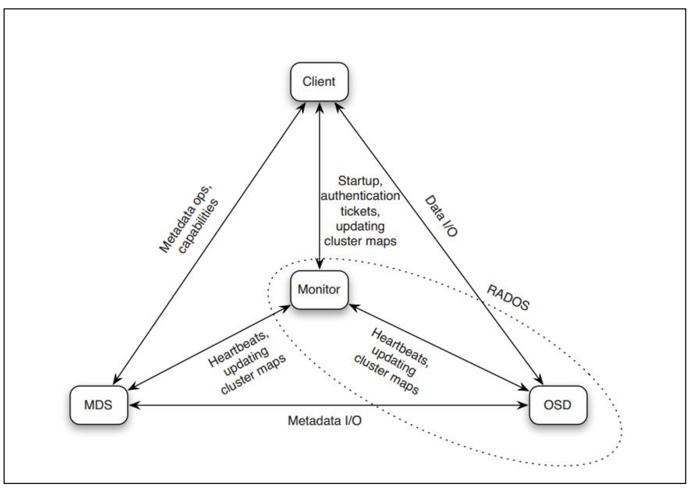

Fig. 2. Ceph components interaction [6].

Reliable Autonomic Distributed Object Storage (RADOS)

From a bird view, object storage cluster made of hundreds of thousands of OSDs as a single logical object store and namespace to the Ceph clients and metadata servers. Ceph's RADOS achieves linearity in both capacity and aggregated performance by delegating management of object replication, cluster expansion, failure detection and recovery to OSDs in a distributed fashion. RADOS can also be used as a stand-alone system.

Unlike other parallel file systems, replication is managed by OSDs instead of clients, which shifts replication bandwidth overhead to the OSD cluster, simplifies the client protocol, and provides fully consistent semantics in mixed read/write workloads. RADOS manages the replication of data using a variant of primary-copy replication and replicas are stored in placement groups which includes a primary OSD which serializes all requests to the placement group.

Writes are applied in two phases and this approach separates writing for the purpose of sharing with other clients from writing for the purpose of durability and makes sharing data very fast. Ceph's failure detection and recovery are fully distributed. The monitor service is only used to update the master copy of the cluster map. OSDs communicates the cluster map updates using epidemic-style propagation that has bounded overhead. This procedure is used to respond to all cluster map updates, whether due to OSD failure, cluster contraction, or expansion. OSDs always collaborate to realize the data distribution specified in the latest cluster map while preserving consistency of read/write access [6].

Data Distribution System

The small size of metadata items in the MDS and the compactness of cluster maps in RADOS are enabled by CRUSH (Controlled Replication Under Scalable Hashing) [7]. Ceph uses this hash function to calculate the placement of data instead of using allocation tables, which can grow very large and unwieldy. CRUSH is part of the cluster map and behaves like a consistent hashing function in that failure, removal, and addition of nodes result in near-minimal object migration to re-establish near-uniform distribution. CRUSH maps a placement group ID to an ordered list of OSDs, using a hierarchically structured cluster map and placement rules as additional input. Any list output by CRUSH meets the constraints specified by placement rules preventing two replicas being placed in same failure domain [3]. Knowledge of failure domains is important for overall data safety of very large storage systems where correlated failures are common.

SALIENT FEATURES

1. Ceph Clients include several service interfaces. These include:

    - Block Devices: The Ceph Block Device (a.k.a., RBD) service provides resizable, thin-provisioned block devices with snapshotting and cloning.
    - Object Storage: The Ceph Object Storage (a.k.a., RGW) service provides RESTful APIs with interfaces that are compatible with Amazon S3 and OpenStack Swift.
    - Filesystem: The Ceph Filesystem (CephFS) service provides a POSIX compliant filesystem usable with mount or as a filesytem in user space (FUSE).

2. Scalability and high availability: In traditional architectures there is single point of entry to a complex subsystem. This imposes a limit to both performance and scalability, while introducing a single point of failure. Ceph eliminates the centralized gateway using CRUSH algorithm to enable clients to interact with Ceph OSD Daemons directly. Ceph OSD Daemons create object replicas on other Ceph Nodes to ensure data safety and Ceph Monitors provide high availability [8].

3. Network Security: Ceph provides its cephx authentication system to authenticate users and daemons. Cephx uses

shared secret keys for mutual authentication, such that both parties can prove to each other they have a copy of the key without actually revealing it.

4. Dyanamic cluster management: Ceph uses CRUSH which enables modern cloud storage infrastructures to place data, re-balance the cluster and recover from faults dynamically. The Ceph storage system supports the notion of 'Pools', which are logical partitions for storing objects.

5. Smart Daemons enable hyperscale [9]: The ability of Ceph Clients, Ceph Monitors and Ceph OSD Daemons to interact with each other allows Ceph OSD Daemons to utilize the CPU and RAM of the Ceph nodes perform task easily that can bog down a centralized server.Leveraging this computing power leads to several major benefits:

   - OSDs Service Clients Directly: Ceph Clients can maintain a session when they need to, and with a certain Ceph OSD Daemon instead of a centralized server.
   - OSD Membership and Status: The Ceph OSD Daemon status reflects whether it is running and able to service Ceph Client requests. Ceph also empowers OSD Daemons with ability to check each other's heartbeats and report back to the Ceph Monitor reliving their burden.
   - Data Scrubbing: Ceph OSD Daemons insures data integrity by scrubbing placement groups.Light scrubbing (daily) catches bugs or filesystem errors. Deep scrubbing (weekly) finds bad sectors on a drive that weren't apparent in a light scrub.
   - Replication:Like Ceph Clients, Ceph OSD Daemons use the CRUSH algorithm, but the Ceph OSD Daemon uses it to compute where replicas of objects should be stored (and for rebalancing).

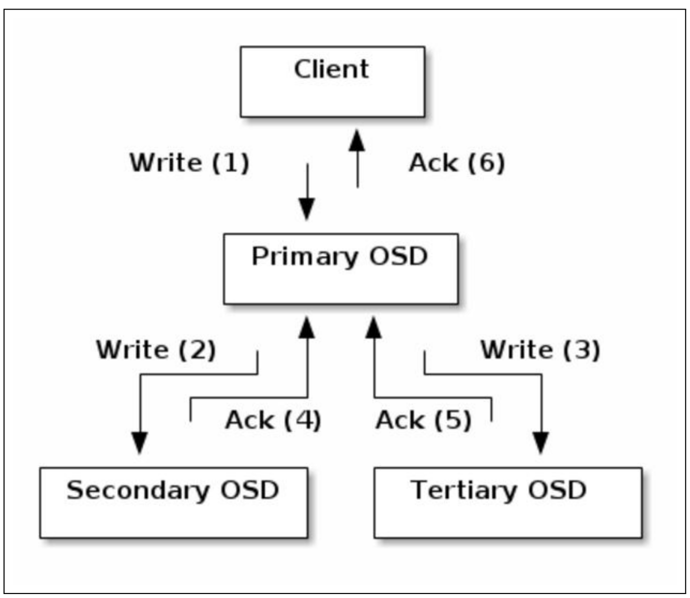

Fig. 3. Replication Process [9].

6. Ceph's provides a native interface to the Ceph Storage Cluster via librados, and a number of service interfaces built on top of librados.

RELATED WORK

Ceph scalability provide high-performance access to a small set of files by tens of thousands of cooperating clients in contrast to Largescale systems like OceanStore [10] and Farsite [11] which fails due to bottlenecks in subsystems such as name lookup. Ceph proves more reliable over other parallel file and storage systems such as Vesta [12], Galley [13], PVFS [14], and Swift [15] due to their lack of strong support for scalable metadata access or robust data distribution. These systems also typically suffer from block allocation issues: blocks are either allocated centrally or via a lock-based mechanism, preventing them from scaling well for thousands of write requests.

USE CASES

Ceph is being used in wide range of applications [16]. Few of them are listed below:

1. Red Hat Ceph Storage team worked with WDLabs and SuperMicro and built and tested a 504 node Ceph cluster with 4 PB of raw storage using these WDlabs Micro-Servers. [17].

2. Cloud Infrastructure for Microbial Bioinformatics (CLIMB) has selected and implemented Red Hat Ceph Storage for their large-scale extensive research needs [18].

3. Yahoo's deployment of the community version of Ceph software for its Flickr and Mail applications on its Cloud Object Store (COS) [19].

4. Red Hat Ceph Storage on Dell PowerEdge server

5. Red Hat Ceph Storage on Intel processors and SSDS

USEFUL RESOURCES

Ceph installation manual [20], provides Installation and Deployment guide which is excellent resource as starter kit. Tutorial on Ceph Deployment by Alan Johnson[21] , is a good tutorial about Ceph deployment.

CONCLUSION

Ceph provides unique solution for the three critical challenges of large scale storage systems—scalability, performance, and reliability. CRUSH and RADOS provides Ceph with improved data safety, ability to manage data replication, failure detection and recovery, low-level disk allocation, scheduling, and data migration without encumbering any central server(s). Ceph's metadata management architecture addresses one of the most vexing problems in highly scalable storage of providing a single uniform directory hierarchy obeying POSIX semantics [3]. Thus, Ceph has proven to be one stop solution for the large-scale storage system in today's Big Data World.

ACKNOWLEDGEMENTS

This work was done as part of the course "I524: Big Data and Open Source Software Projects" at Indiana University during Spring 2017. Many thanks to Professor Gregor von Laszewski and Prof. Geoffrey Fox at Indiana University Bloomington for their academic as well as professional guidance. We would also like to thank Associate Instructors for their help and support during the course.

REFERENCES

[1] Red Hat,Inc., *Red Hat Ceph Storage*, Red Hat,Inc., accessed: 2017-2-26. [Online]. Available: https://www.redhat.com/en/resources/red-hat-ceph-storage-datasheet

[2] Red Hat, Inc., "Ceph homepage-ceph," Web Page, Red Hat, Inc., 2017, accessed: 2017-2-26. [Online]. Available: https://ceph.com/

[3] S. A. Weil, S. A. Brandt, E. L. Miller, D. D. E. Long, and C. Maltzahn, "Ceph: A scalable, high-performance distributed file system," in *Proceedings of the 7th symposium on Operating systems design and implementation*, ser. OSDI '06. Berkeley, CA, USA: USENIX Association, Nov. 2006, pp. 307–320, accessed: 2017-1-26. [Online]. Available: https://www.usenix.org/legacy/event/osdi06/tech/full_papers/weil/weil.pdf

[4] S. A. Weil, K. T. Pollack, S. A. Brandt, and E. L. Miller, "Dynamic metadata management for petabyte-scale file systems," in *Proceedings of the 2004 ACM/IEEE Conference on Supercomputing*, ser. SC '04. Washington, DC, USA: IEEE Computer Society, 2004, pp. 4–, accessed: 2017-2-26. [Online]. Available: https://doi.org/10.1109/SC.2004.22

[5] M. Jones, "Ceph: A linux petabyte-scale distributed file system," Jun 2010, accessed: 2017-2-26. [Online]. Available: https://www.ibm.com/developerworks/library/l-ceph/

[6] C. Maltzahn, E. Molina-Estolano, A. Khurana, A. J. Nelson, S. A. Brandt, and S. Weil, "Ceph as a scalable alternative to the hadoop distributed file system," *login: The USENIX Magazine*, vol. 35, pp. 38–49, 2010, accessed: 2017-2-26.

[7] R. Latham, N. Miller, R. Ross, P. Carns, Mathematics, and C. U. Computer Science, "A next-generation parallel file system for linux cluster." *LinuxWorld Mag.*, vol. 2, Jan 2004, accessed: 2017-2-26.

[8] Red Hat, Inc., "Ceph homepage-ceph," Web Page, Red Hat, Inc., 2017, accessed: 2017-2-26. [Online]. Available: http://docs.ceph.com/docs/master/architecture/#scalability-and-high-availability

[9] Red Hat,Inc., "Ceph homepage-ceph," Web Page, Red Hat, Inc., 2017, accessed: 2017-2-26. [Online]. Available: http://docs.ceph.com/docs/master/architecture/#smart-daemons-enable-hyperscale

[10] J. Kubiatowicz, D. Bindel, Y. Chen, S. Czerwinski, P. Eaton, D. Geels, R. Gummadi, S. Rhea, H. Weatherspoon, W. Weimer, C. Wells, and B. Zhao, "Oceanstore: An architecture for global-scale persistent storage," *SIGPLAN Not.*, vol. 35, no. 11, pp. 190–201, Nov. 2000, accessed: 2017-2-26. [Online]. Available: http://doi.acm.org/10.1145/356989.357007

[11] A. Adya, B. Bolosky, M. Castro, R. Chaiken, G. Cermak, J. J. Douceur, J. Howell, J. Lorch, M. Theimer, and R. a. Wattenhofer, "Farsite: Federated, available, and reliable storage for an incompletely trusted environment," in *Proceedings of the 5th Symposium on Operating Systems Design and Implementation (OSDI)*. Boston, MA: USENIX, December 2002, p. 1–14, accessed: 2017-2-26. [Online]. Available: https://www.usenix.org/legacy/events/osdi02/tech/full_papers/adya/adya.pdf

[12] P. F. Corbett and D. G. Feitelson, "The vesta parallel file system," *ACM Trans. Comput. Syst.*, vol. 14, no. 3, pp. 225–264, Aug. 1996, accessed: 2017-2-26. [Online]. Available: http://doi.acm.org/10.1145/233557.233558

[13] N. Nieuwejaar and D. Kotz, "The galley parallel file system," in *Proceedings of the 10th International Conference on Supercomputing*, ser. ICS '96. New York, NY, USA: ACM, 1996, pp. 374–381, accessed: 2017-2-26. [Online]. Available: http://doi.acm.org/10.1145/237578.237639

[14] R. Latham, N. Miller, R. Ross, P. Carns *et al.*, "A next-generation parallel file system for linux cluster." *LinuxWorld Mag.*, vol. 2, no. ANL/MCS/JA-48544, 2004, accessed: 2017-2-26.

[15] L.-F. Cabrera and D. D. Long, *Swift: Using distributed disk striping to provide high I/O data rates*. University of California, Santa Cruz, Computer Research Laboratory, 1991, vol. 8523, accessed: 2017-2-26. [Online]. Available: https://www.researchgate.net/profile/Darrell_Long2/publication/2752148_Swift_Using_Distributed_Disk_Striping_to_Provide_High_IO_Data_Rates/links/09e415060773f188dd000000.pdf

[16] Red Hat, Inc., "Ceph use cases-ceph," Web Page, Red Hat, Inc., 2017, accessed: 2017-2-26. [Online]. Available: http://docs.ceph.com/use-cases/

[17] Red Hat,Inc., "First large scale ceph storage microserver cluster unveiled," Web Page, Red Hat, Inc., Oct. 2016, accessed: 2017-2-26. [Online]. Available: http://ceph.com/community/500-osd-ceph-cluster/

[18] Red Hat ,Inc., "Climb supports research collaboration with red hat ceph storage," Web Page, Red Hat, Inc., Oct. 2016, accessed: 2017-2-26. [Online]. Available: https://www.redhat.com/en/resources/climb-case-study

[19] N. P.P.S, S. Samal, and S. Nanniyur, "Yahoo cloud object store - object storage at exabyte scale |yahoo engineering," Web Page, Yahoo Engineering, Apr. 2015, accessed: 2017-2-26. [Online]. Available: https://yahooeng.tumblr.com/post/116391291701/yahoo-cloud-object-store-object-storage-at

[20] Red Hat, Inc., "Installation (manual)- ceph documentation," Web Page, Red Hat, Inc., 2017, accessed: 2017-2-26. [Online]. Available: http://docs.ceph.com/docs/master/install/

[21] Alan Johnson, "Ceph - hands-on guide |aj's data storage tutorials," Web Page, accessed: 2017-2-26. [Online]. Available: https://alanxelsys.com/ceph-hands-on-guide/