WHITE PAPER

Intel Solutions for Ceph Deployments

Basic Configuration Guidelines of Intel® Components by Common Ceph Use Cases

Anjaneya (Reddy) Chagam
Intel Corporation
Dan Ferber
Intel Corporation
David J. Leone
Intel Corporation
Orlando Moreno
Intel Corporation
Yaguang Wang
Intel Corporation
Yuan (Jack) Zhang
Intel Corporation
Jian Zhang
Intel Corporation
Yi Zou
Intel Corporation
Mark W. Henderson
Intel Corporation

Introduction

Not all Ceph storage solutions are equal, and understanding your workload and capacity requirements is essential in designing a Ceph solution. Ceph lets organizations deliver object storage, block storage, or file system storage through a unified and distributed cluster. These cluster solutions are optimized for each of their requirements through the design process. The design process starts with the IOPS or Bandwidth required, storage capacity needed, and then drill-down on architecture and component selection that will drive to the desired combination of performance and costs, as shown in Figure 1.

Different workload types require distinct approaches to storage infrastructure. For example, relational database management system (RDBMS) workloads require IOPS-and-latency-optimized storage in order commit transaction and avoids locks, while an object archive might require capacity optimization. Video streaming, for example, requires a sequential streaming bandwidth optimized solution. Which is different than a bandwidth optimized solution that you might use for backup because video can't have gaps in its transmission.

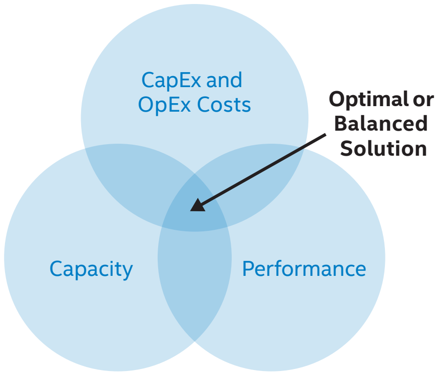


Figure 1. Different storage workloads and demanded capacity require balancing factors as selection of component, cluster organization, and Ceph parameters adopted.

Table of Contents

Intended Use of this White Paper

This paper is for use in guiding Intel customers to appropriate Intel-based solutions.

This paper should supplement, and not replace published reference architectures and solutions from Intel storage partners. More complete Intel Solutions Reference Architectures (SRA's) should always supersede any guidelines here as practical, as SRA's are more rigorous and complete recipes.

For Intel Solutions Reference Architectures, see **http://intel.com/storage**.

For Intel partner reference architectures see the ISV and/or OEM storage web pages.

Feedback, suggestions, and corrections to this paper should go to **mark.w.hendreson@intel.com**.

Usually, we can classify storage workload based on these three main types:

CLUSTER OPTIMIZATION CRITERIA	PROPERTIES	EXAMPLE USES
IOPS-optimized	• Highest IOPS • Lowest cost per IOPS • Usually 3x replication • Single node is less than or equal to 10% of the cluster (for fault tolerance)	• RDBMS virtualized • Typically block storage
Bandwidth-optimized (aka Throughput)	• Highest Bandwidth • Lowest cost per given unit of Bandwidth • Highest Bandwidth per BTU • Highest Bandwidth per watt • Usually 3x replication for higher read throughput • Single node is less than or equal to 10% of the cluster (for fault tolerance)	• Block or Object storage • Video streaming
Capacity-optimized	• Lowest cost per TB • Lowest BTU per TB • Lowest watt per TB • Highest density per TB • Erasure coding common for maximizing usable capacity • Single node is less than or equal to 15% of cluster (for fault tolerance)	• Typically Object storage • Cold/Archive storage

Table 1. Ceph cluster optimization criteria

Beside workloads characterization, a further step on cluster definition with direct impact on cost and performance is the storage capacity required. As an overall rule of thumb, in a range of Terabytes (TB) and required IOPS-optimized storage, servers with dozen 2.5" SSDs (SAS/SATA) or 4x NVMe SSD will probably provide a good IOPS/$. While on the other side, a capacity-optimized in Petabyte (PT) range, nodes with 60 or even more 3.5" HDDs currently deliver the best CapEx TB/$, however advances in flash storage technology, with lower failure rates and longer amortization cycles, are challenging HDDs from a OpEx perspective. Everything in between will require a balance on nodes with fewer capacity but with higher CPU and network throughput with a balanced selection of SSD and HDD to achieve the lowest CapEx and OpEx to a given workload.

Ceph Enabled by Intel® Hardware and Software Configurations

Ceph is an open source highly scalable and redundant server-based storage product which provides for object, block and file system storage in a single storage cluster. Ceph runs on high-volume Intel-based hardware in bare metal or virtualized configurations. Its popularity comes from the flexibility to pick hardware configurations from different storage vendors based on workload requirements as well as support wide variety of applications using unified access with block, file and object interfaces. Ceph provides flexibility to consume block storage using upstreamed Linux kernel drivers and user mode QEMU/libvirt interfaces. It can scale to hundreds of storage servers and thousands of storage clients.

Ceph Community

For more information on Ceph, see http://ceph.com. To see some of the many commercial and educational organizations using Ceph, see their Ceph Days presentations http://www.slideshare.net/Inktank_Ceph, For OpenStack's use of Ceph and usage information, see http://www.openstack.org/surveys/ And for Intel storage solutions, see http://www.intel.com/storage

Ceph developers at Intel contribute a significant number of performance based enhancements and features to the upstream Ceph community. In fact Intel has often been second in its number of Ceph upstream contributions only to Redhat. Intel also hosts several Ceph Days meetings around the world, and in 2015 hosted the first community face to face hackathon for Ceph developers, focused on performance topics.

This white paper describes the current most common block and object use cases for Ceph. For each use case, a typical Intel based hardware configuration for the Ceph storage cluster is illustrated and any public performance for the configuration is referenced. In some cases there may be a secondary configuration listed for consideration, with different characteristics. That said, this paper and the configurations contained within are not reference architectures. They are not recipes that are guaranteed to produce certain performance levels. Rather they are guideline configurations based on our and the community's experience. If you require set performance levels or price/performance levels, you'll want to work with one of many Intel partners who provide Ceph solutions. Or you will want to look at detailed reference architectures available at http://www.intel.com/storage or at one of the many Ceph solution provider web sites.

Elements Not Covered

- Note that guidelines are not given for client nodes as these vary widely
- Use cases and configs for Ceph file (Cephfs)

Ceph Pool Types and Relation to Hardware and Use Cases

Ceph has three pool types; 1) replicated pools, 2) erasure coded pools, and 3) cache tier pools. The selection of pool types is independent of the guided hardware configuration and replicated and erasure coded storage types were mentioned in the introduction. However, performance data does depend on how the pool used has been set up as far as pool type. You will often see the block use cases here show data for replicated pools and object use cases use erasure coded pools, respectively as they are optimal for each type of storage.

Cache tier pools are not in general use as their performance is still being optimized, so you will not see data on cache tiers with any of these guided configurations. Likewise, file use cases will not be covered as those are in the early stages for Ceph production use and not as prevalent as block and object.

Ceph Use Cases Covered

A subset of the most common block and object use cases will be covered in this paper. To see more details, refer to http://pad.Ceph.com/p/hack-athon_2015-08.

As you read this paper and review the use cases, configurations, and performance we discuss with you—you can think of two basic high level Ceph storage node hardware configurations, as follows:

- A standard Ceph configuration. These are Intel® Xeon® processor D or Intel® Xeon® processor E5 nodes with Intel® SSD Data Center Family for PCIe devices flash storage (SATA or PCIe NVM SSD) for journaling and HDDs as data drives. Intel® Cache Acceleration Software (CAS) can be added for additional performance.
- A high performance Ceph configuration. These are Intel Xeon E5 nodes with Intel Flash NVM / PCIe SSD for journaling and Intel SATA SSD as data drives. Intel® CAS can be added for additional performance.

Ceph storage pool types such as replicated and erasure coded pools are layered on top of these hardware configurations, and will be discussed in the use cases.

Block Use Cases

For block, the most common use cases are:

- Virtual Desktop Hosting – VDI or Virtual Desktop Images, or similar
- Database or similar
- General Use – meaning general block storage for applications

The I/O characteristics of VDI, Database, and General block patterns are described below.

Object Use Cases

For object, the most common use cases are:

- Digital Video Recording (DVR) or similar
- Video on Demand (VOD) or similar
- Backup or similar

The I/O characteristics of DVR, VOD, and backup are described below.

NAME	USE	POOL CONFIG/ REPLICATION	CLUSTER PRE-CONDITION
General	General raw synthetic performance	3x replication	60% full
VDI	Virtual desktop environment	3x replication	60% full
VirtInfr	Virtual clients	3x replication/snapshot	60% full
OLTP	RDBMS performance	3x replication	60% full
NoSQL	Cassandra, etc	3x replication	60% full

Table 2. The I/O Characteristics of DVR, VOD, and Backup

Intel Hardware and Software Components

The following Intel hardware is covered in these guideline systems:

- Intel® Xeon® Processor Family
- Intel® Solid-State Drives
- Intel® Ethernet Gigabit Server Adapters

The following Intel software is covered in these guideline systems:

- Intel® Storage Acceleration Library
- Intel® Cache Acceleration Software

The following Intel-provided community tools are covered as well:

- Virtual Storage Manager
- Ce-Tune, UI based Ceph benchmarking and analysis
- COSBENCH for object based benchmarking

The examples shown below are one of many ways to implement a particular solution and your results may vary. Intel bases solutions are available from your preferred system provider.

General Guidelines

The following is true for all of the use cases we discuss.

Ceph Version

Each Ceph release generally provides increased performance, reliability, and functionality. We recommend deploying a long term support (LTS) release. At the time of this paper's writing, the latest LTF release is the “hammer” or 0.94 release. We anticipate ongoing improvements with later releases of both hardware (CPU, Flash SSDs/NVMe) and software (Ceph, ISA-L, SPDK, etc).

Ceph and Linux OS Tuning Parameters

There is a Ceph Tunings Guide maintained by one of Intel's Ceph development teams. The recommendations in this guide are generally valid for all use cases unless noted otherwise in the use case description. See Appendix A for an overview of the various tunings—the complete document is available under NDA.

NUMA configuration

A single CEPH node NUMA configuration is shown on the below picture. The CPU supports hyperthreading which is enabled in the BIOS. The logical cores are numbered 0-5, 12-17 on the CPU 0 and 6-11, 18-23 on CPU 1.

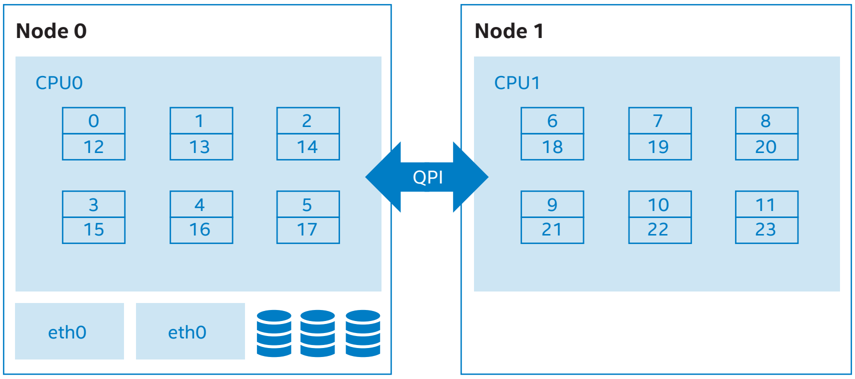


Figure 2. Single CEPH Node NUMA Configuration

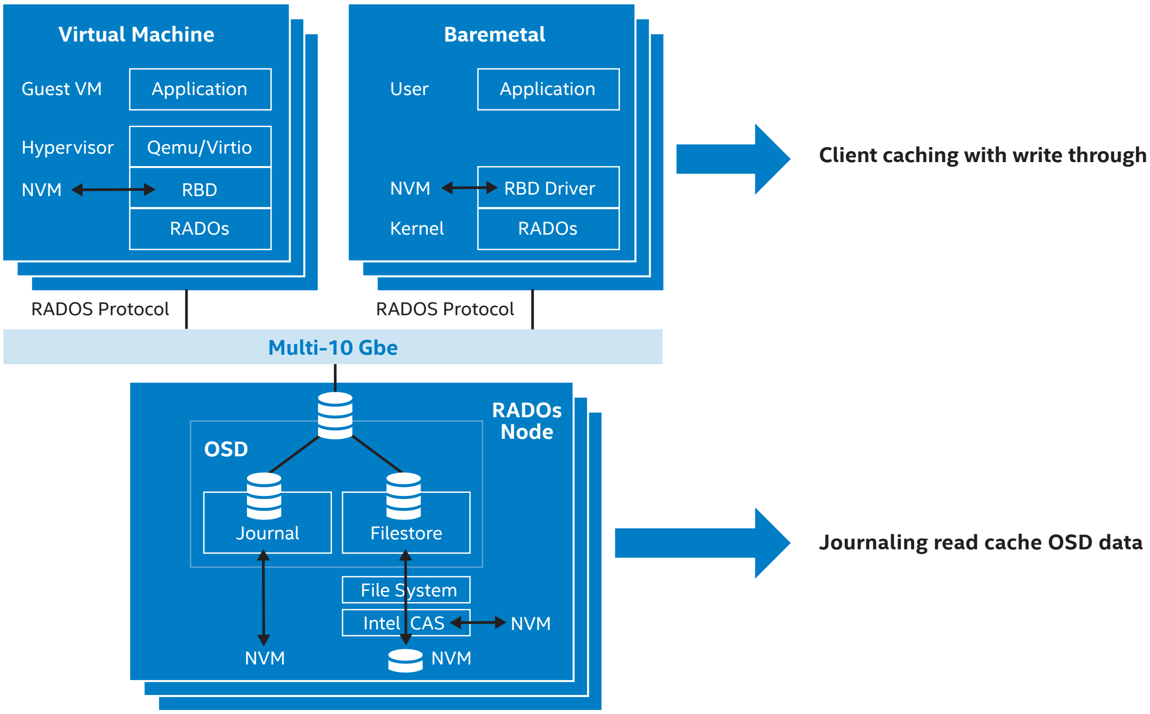


Figure 3. Top View on Where SSD/NVM are Used at Ceph Infrastructure

The network cards and the HBA providing storage connectivity are connected to the NUMA node 0 which is the optimal configuration for the processes running on the logical cores 0-5 and 12-17.

In order to optimize hardware access speed for the CEPH processes the Ceph startup script needs to be modified to force the Ceph-osd process to run on CPU0. This is detailed in Apendix B, but generally done with the following command:

```
setaffinity=" numactl --mem-
bind=0 --cpunodebind=0 "
```

This also releases resources on CPU1 and makes it available to potentially run Velocix middleware services (running in a real cDVR configuration) and makes the configuration more realistic.

Intel SSD/NVM Technology – General Observations with Ceph

The section covers the general use SSD/NVM technology and recommendations for storage node configurations based on performance and cost.

SSD usage scenarios for Ceph

Independent SSD pools

Deploy SSDs across different server/ nodes into an OSD SSD pool, separated from HDD pool. The SSD pool is then dedicated for high performance applications.

Several performance/capacity pools can be coexist in same Ceph cluster, three recommended configurations are listed below in Figure 5.

Three storage node configurations

Good/Standard configuration

SSDs as journal and caching drive, HDDs for OSD data, ratio is below:

It is highly recommended to use PCIe*/ NVMe SSD for high performance and low latency because: 1)NVMe technology is architected and optimized from group up for Non-volatile Memory (NVM)/SSD and PCIe places storage closer to CPU for lower latency, 2) NVMe is transforming data center from SATA/SAS to PCIe interface.

1. **PCIe/NVMe** SSD:HDD, **1:12**, Intel PCIe/NVMe P3700 as journal drive
2. **SATA** SSD: HDD, **1:4**, Intel SATA S3700 as Journal drive

Use Intel CAS with hint-based I/O Classification, allocation, and prioritization technology along with Intel® Differentiated Storage Services (DSS) to identify and cache those storage elements that increase performance. Because of the distributed nature of Ceph, caching and Journal can be on same SSD drive(s).

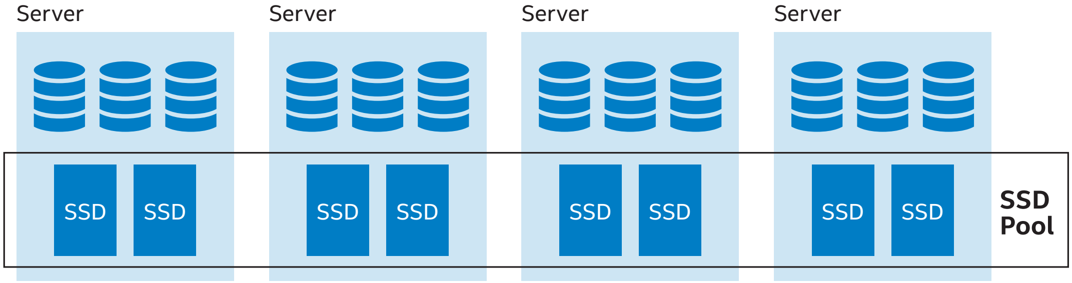


Figure 4. Independent SSD Pools

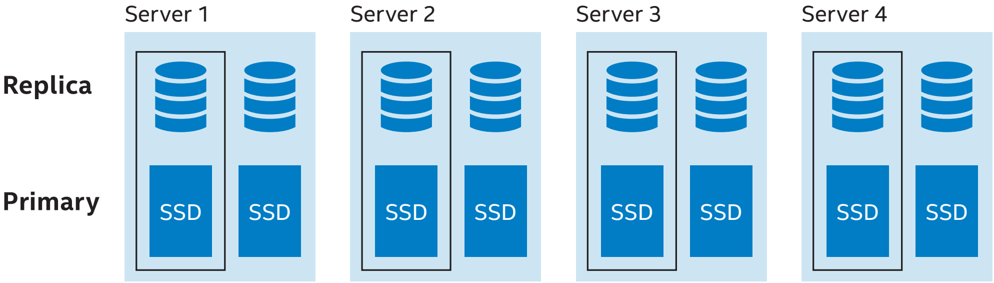


Figure 5. Recommended Configurations

CEPH STORAGE NODE - GOOD	
CPU	Intel® Xeon® Processor E5-2650v3
NIC	10GbE
Drives	1x 1.6 TB P3700 12x 4 TB HDDs (1:12 ratio) (P3700 as Journal and caching)
Software	Intel CAS RSTe/MD4.3 (optional)

Table 3. Good Configuration

CEPH STORAGE NODE - BETTER	
CPU	Intel® Xeon® Processor E5-2690
Memory	128 GB
NIC	Dual 10GbE
Drives	1x 800 GB P3700 4x 1.6 TB S3510 (P3700 as Journal and caching)
Software	Intel CAS

Table 4. Better Configuration

CEPH STORAGE NODE - BEST	
CPU	Intel® Xeon® Processor E5-2699v3
Memory	>=128 GB
NIC	2x 40GbE 4x dual 10GbE
Drives	4-6x 2 TB P3700

Table 5. Best Configuration

This configuration targets for high capacity storage with high throughput performance.

Note that in Ceph configurations the failure domain is the entire storage node. So having a single SSD (either SATA or PCIe based) does not represent a single point of failure for the Ceph system. The redundancy is across hardware systems. This is direct contrast to non-scale-out systems where the failure domain is typically contained within a single system. The advantage to the scale-out failure domain is that it can be leveraged for RAS, scalability, site level replication, EC and continuous operations when nodes are being upgraded.

Advanced configuration:
PCIe/NVMe SSD for Journal and Large capacity low cost SATA SSD as OSD data drives, this configuration is for those use cases/applications that need higher performance especially IOPS and SLAs with medium storage capacity requirements, and this is best cost effective configuration.

Best performance configuration:
Use all NVMe/PCIe SSDs, this configuration is the best performance storage solution for those use cases/applications that need highest performance and low latency.

Example: 4 to 6x P3700 2 TB SSDs as OSDs

SSD Selection guide

Enterprise SSD vs. Client SSD

SSDs are not all same, even within the same SSD vendor, they may have different SSD production lines. In general, Client SSDs and Enterprise SSDs are tested and specified differently on performance, reliability, and endurance. For example, client SSD is specified on 8 hour/day usage with less data integration requirement while an Enterprise SSD is expected to be in use 24 hours/day with end to end data integration and power protection etc., For additional details refer to the industry standard JEDC for SSD's reliability, endurance requirements.

Drive Writes Per Day

Unlike HDDs, SSDs are a consumable resource and can only be written a large, but finite, number of times. They are sized and have performance characteristics to be consumed in a way that is consistent with their warranty and performance expectations. The term "Drive Writes Per Day" DWPD is a measure/specification of the industry for SSD endurance, and it is the amount of data that can be written each day based on JEDC workload, which is 4K block size random writes, and based on this DWPD, how many years SSD can continue to function correctly.

For example, Intel P3700 800 GB is 10 DWPD for 5 years, this allows host to write total of 8 TB per day for 5 years.

Below in table 6 are typical DWPD. It is for reference only, it is highly recommended to evaluate/measure endurance before production as well as monitoring SSD endurance through SMART indicator at production, for example, Intel SSD provides SMART attributes E2, E3, and E4 for off-line endurance evaluations as well as E9, E8 as real-time/production life/endurance indicator.

Intel Data Center SSDs

Intel DC SSDs are classified in 3 categories based on performance, endurance, and cost.

Intel's PCIe/NVMe SSD starts with "P," example P3700. The Intel SATA SSD starts with "S," example, S3700. Both form factors are available in all three of the endurance levels.

Refer to Intel Data Center SSD family products and additional information on Intel SSDs including Intel Optane and 3D NAND technologies.

SSD SMART attributes

SMART attributes in table 7 and 8 are used for monitoring SSD health status, such as endurance indicator, error log, host read/write, NAND read/write etc, these can be managed/monitored by external management software, such as Intel® Virtual Storage Manager (VSM).

DRIVE TYPE	DRIVE WRITES PER DAY (DWPD)	WARRANTY (YEARS)	COMMENTS
High Endurance (P3700 or S37xx)	10	5	High intensive random writes
Medium Endurance (P3600 or S36xx)	3	5	Balanced reads and writes
Standard Endurance (P3500 or S35xx)	0.3	5	High intensive read workloads

Table 6. Intel Data Center SSDs

BYTE	# OF BYTES	DETAILS
0	1	Critical Warning: These bits if set, flag various warning sources Bit 0: Available Spare is below Threshold Bit 1: Temperature has exceeded Threshold Bit 2: Reliability is degraded due to excessive media or internal errors Bit 3: Media is placed in Read-Only Mode Bit 4: Volatile Memory Backup System has failed (e.g.: PCI capacitor test failure) Bits 5-7: Reserved
1	2	Temperature: Overall Device current temperature in Kelvin
3	1	Available Spare: Contains a normalized percentage (0 to 100%) of the remaining spare capacity
4	1	Available Spare Threshold
5	1	Percentage Used Estimate
21	16	Data Units Read
48	16	Data Units Written
64	16	Host Read Commands
80	16	Host Write Commands
96	16	Controller Busy Time
112	16	Power Cycles
128	16	Power On Hours
144	16	Unsafe Shutdowns
160	16	Media Errors
176	16	Number of Error Information Log Entries

Table 7. Example, NVMe Health Log Table (Log Page Identifier 02h)

BYTE	# OF BYTES	ATTRIBUTE	DESCRIPTION
0	1	AB (Program Fail Count)	Raw value: shows total count of program fails. Normalized value: beginning at 100, shows the percent remaining of allowable program fails.
3	1	Normalized Value	
5	6	Current Raw Value	
12	1	AC (Erase Fail Count)	Raw value: shows total count of erase fails. Normalized value: beginning at 100, shows the percent remaining of allowable erase fails. Raw value: Bytes 1-0: Min. erase cycle Bytes 3-2: Max. erase cycle Bytes 5-4: Avg. erase cycles Normalized value: decrements from 100 to 0.
15	1	Normalized Value	
17	6	Current Raw Value	
24	1	AD (Wear Leveling Count)	
27	1	Normalized Value	
29	6	Current Raw Value	
36	1	B8 (End-to-End Error Detection Count)	Raw value: reports number of End-to-End detected and corrected errors by hardware. Normalized value: always 100. Raw value: shows total number of encountered interface cyclic redundancy check (CRC) errors. Normalized value: always 100. Raw value: measures the wear seen by the SSD (since reset of the workload timer, attribute E4h), as a percentage of the maximum rated cycles. Divide the raw value by 1024 to derive the percentage with 3 decimal points. Normalized value: always 100.
39	1	Normalized Value	
41	6	Current Raw Value	
48	1	C7 (CRC Error Count)	
51	1	Normalized Value	
53	6	Current Raw Value	
60	1	E2 (Timed Workload, Media Wear)	
63	1	Normalized Value	
65	6	Current Raw Value	
72	1	E3 (Timed Workload, Host Reads %)	Raw value: shows the percentage of I/O operations that are read operations (since reset of the workload timer, attribute E4h). Reported as integer percentage from 0 to 100. Normalized value: always 100.
75	1	Normalized Value	
77	6	Current Raw Value	
84	1	E4 (Timed Workload, Timer)	Raw value: measures the elapsed time (number of minutes since starting this workload timer). Normalized value: always 100. Raw value: reports Percent Throttle Status and Count of events. Byte 0: Throttle status reported as integer percentage. Bytes 1-4: Throttling event count. Number of times thermal throttle has activated. Preserved over power cycles. Byte 5: Reserved. Normalized value: always 100.
87	1	Normalized Value	
89	6	Current Raw Value	
96	1	EA (Thermal Throttle Status)	

Table 8. Example, Intel NVMe/PCIe SSD Additional SMART Attributes

Ceph Block Storage – Virtual Desktop Hosting Use Case 1

The focus for this use case is 4K block random IOPS. Different customers use various percentages of read and write for their simulated VM hosted IOPS use cases. Likewise, each customer has a different number of IOPS in mind that should be provided to each VM user. CERN, in a presentation at the Open Stack Summit Vancouver in the spring of 2015, said they default IOPS they provide each VM user is 100.[1]

Most often, this use case is after IOPS at the lowest cost. This contrasts to some database or custom IOPS use cases where the highest IOPS are desired, and a higher cost would be a trade-off for the higher IOPS. For highest 4K random IOPS, see our database use case and the guideline configuration noted there.

In this guideline system for VM hosting IOPS, we looked at the performance of 100 percent 4K random read, and also 100 percent random write. The writes and reads are as seen from VMs running FIO I/O tests on remote client nodes.

We assumed that 100 IOPS per VM was what we wanted Ceph to provide, and so we looked at how many VMs this guideline configuration could support before delivered IOPS performance fell.

First we'll describe the guideline hardware configuration, and then we'll show the performance we saw.

Guideline Hardware Configuration

(See the general software configuration and tuning sections for software configuration information.)

Ceph Storage Cluster

- 4 Server Minimum
- 10 OSDs per server
- 2 Intel SATA SSDs per server
  - Five OSD journals per SSD
- Monitors on OSD hosts (3 monitors minimum)
- Top of Rack Switch

Network – Top of Rack

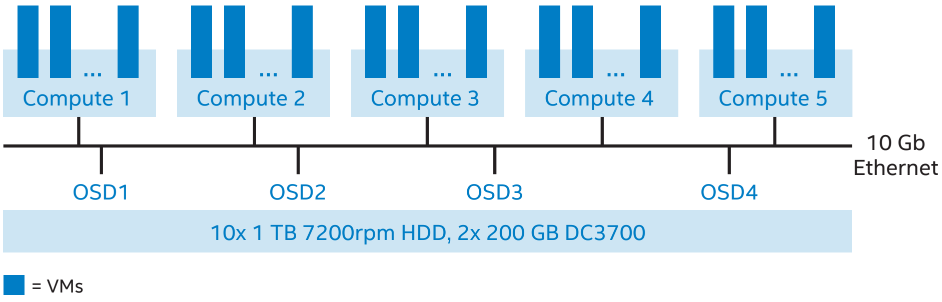
- 1 x 10GbE public (public network client-facing) switch
- Optional 1x 1GbE switch - management
- Optional 1 x 10GbE private (cluster network cluster-facing) network – cluster data
- Optional switch for IPMI

Each Ceph Storage Node

- Processor
  - 1 x Intel® Xeon® Processor E3-1200 v2 CPU
- Memory
  - 16 GB (or 32 GB if you are also running the Ceph monitor on this node)
- HDD
  - JBOD disk controller (8 ports) + chipset SATA (4 ports)
  - Data disks: 10 3 GB Enterprise SATA 3.5"
- Journal SSDs
  - 2 x 100 GB Intel DC S3700
- Network
  - 1 x 10GbE Intel® 82599ES port for public network data
  - Optional 1 x 10GbE Intel® 82599ES port for cluster network data
  - Optional 1GbE Intel Intel® 82574L port for management
- Management
  - Optional IPMI port

The above configuration is one that Intel put together and tested with the Ceph Firefly* release. Since then there have been Ceph releases and Intel has introduced PCIe SSD as well as Intel Xeon Processor D family. It is expected that performance will increase with the later releases and technology, and will be reflected in updated versions of this document.

The currently tested configuration is pictured below:



- 10GbE Network
- Intel® Xeon® Processor E3 server for Ceph cluster
  - 16 GB memory (each node)
  - 10x 1 TB SATA HDD for data through LSI9205 HBA (JBOD) each parted into one partition of OSD daemon
  - 2 x SSD for journal directly connected with SATA controller, 20 GB for each OSD (3, 3, 4)
- 5 nodes client cluster
  - VM host systems

Figure 6. Tested Configuration

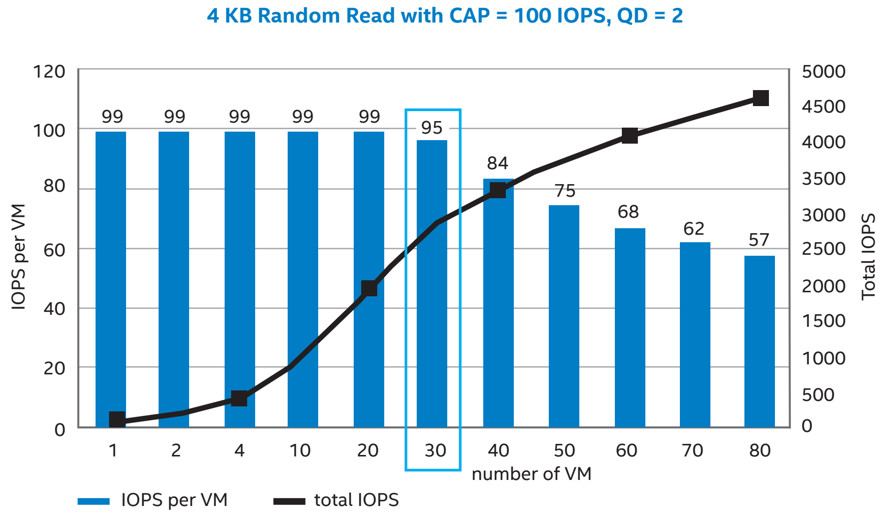


Figure 7. 40-HDD Array-4K Random Read: 4500 IOPS 112 IOPS Per HDD

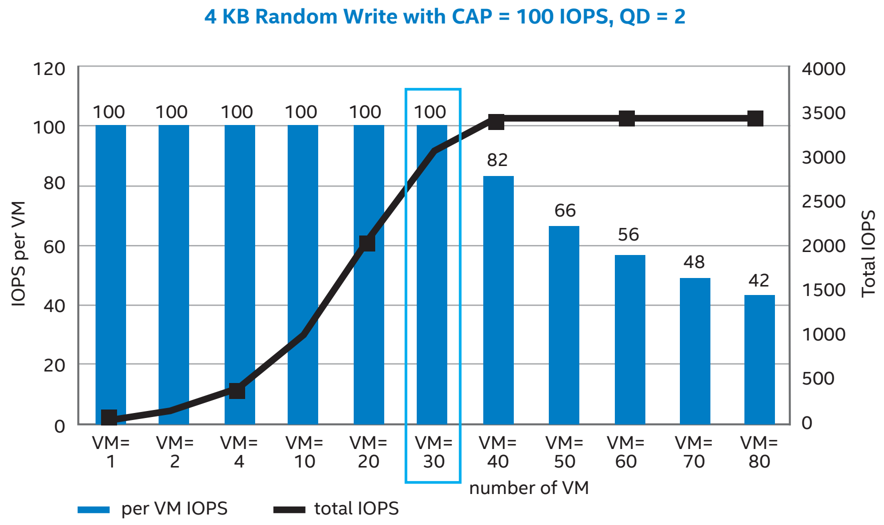


Figure 8. 40-HDD Array-4K Random Read: 3358 IOPS 112 IOPS (with 2X replication) 84 IOPS Per HDD

The rough calculation for how much space to allow for a journal is here:

http://docs.Ceph.com/docs/v0.94/rados/configuration/osd-config-ref/

Without performance optimization, Ceph stores the journal on the same disk as the Ceph OSD Daemon data. A Ceph implementation optimized for performance may use a separate disk to store journal data (e.g., a solid state drive delivers high performance journaling).

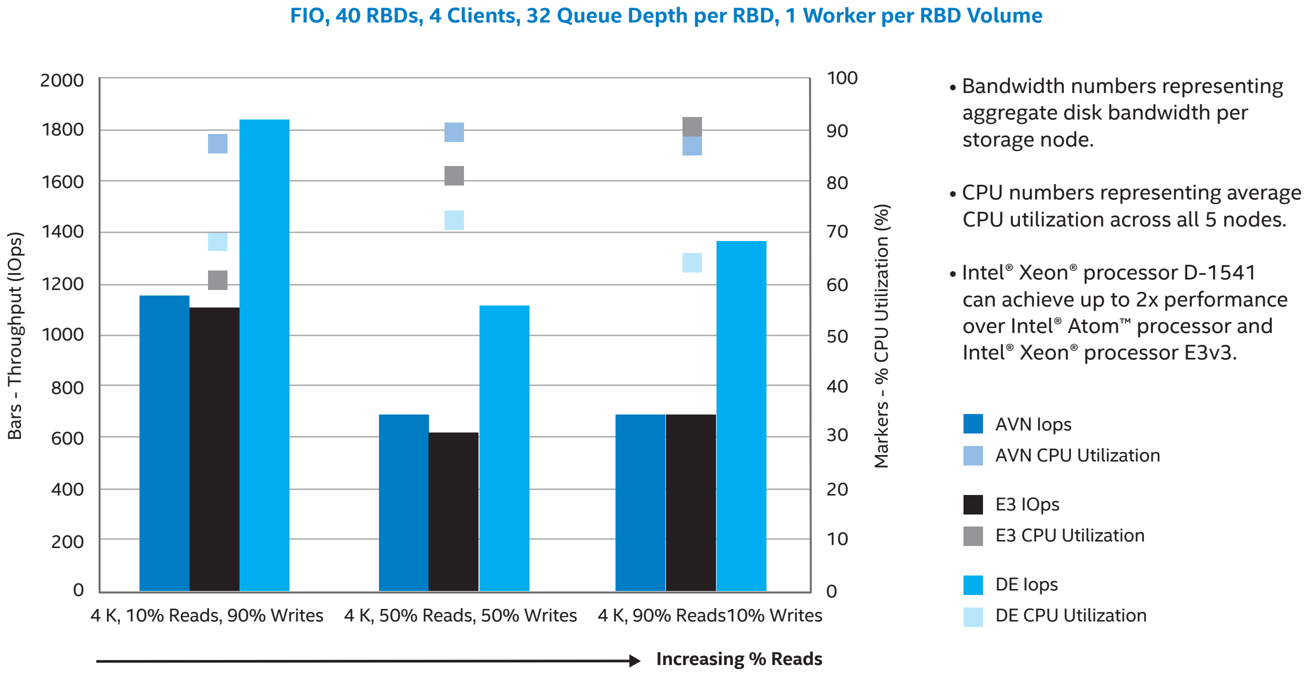
Ceph's default `osd journal size` is 0, so you will need to set this in your ceph.config file. A journal size should find the product of the filestore max sync interval and the expected throughput, and multiple the product by two (2), to be able to commit one journal while the other one continues to log.

```
osd journal size = {2 *
(expected throughput * filestore
max sync interval)}
```

The expected throughput number should include the expected disk throughput (i.e., sustained data transfer rate), and network throughput. For example, a 7200 RPM disk will likely have approximately 100 MB/s. Taking the min( ) of the disk and the network throughput should provide a reasonable expected throughput. Some users just start off with a 10 GB journal size. For example:

```
osd journal size = 10000
```

By default the sync interval is 5 seconds. So we want the journal to be able to hold two times all the data that a drive could hold between sync intervals, for the current data and the previous sync interval. Assuming a SATA HDD can do at best 100 MB/sec, that'd be just 1 GB. Obviously the calculation changes when using flash media. Given the high speed of flash the limiting factor for the journal sizing is most likely not the flash drive but the



- Bandwidth numbers representing aggregate disk bandwidth per storage node.
- CPU numbers representing average CPU utilization across all 5 nodes.
- Intel® Xeon® processor D-1541 can achieve up to 2x performance over Intel® Atom™ processor and Intel® Xeon® processor E3v3.

Figure 9. Intel® Xeon® Processor D Performance

ability of the rest of the system to support the speeds. (Network, processor, memory, or other limitations.)

Use Case 2 Intel PCIe SSD Instead of SATA SSD as Ceph Journals

The typical and common recommendation for Ceph journal is to use SSD, and SSD here is SATA based SSD, typical use is 1:5 HDD ratio, two SATA SSDs journals with 10 HDDs in each node is quite poplar. This is currently driven by common 12 and 24 drive bay systems. As flash memory evolves this ratio is likely to change to reflect actual performance requirements verses common packaging offerings.

Intel as an industry leader has released new, higher speed flash technology called NVMe. This can be thought of as a SSD sitting on an OSD PCIe bus. When compared to a traditional SATA SSD the NVMe has much lower latency due to the elimination of several intermediate I/O functions, higher bandwidth being on the PCIe bus, and greater potential capacity due to an increased form factor over a 2.5" drive layout. This increases IOPS and lowers latency such that an NVMe flash device is capable of outperforming two to four SATA SSD devices for a given underlying memory technology.

For an all PCIe SSD configuration see our database use case.

When to Consider Intel® Xeon® Processor D Instead of Intel® Xeon® E3 Processor

The new Intel® Xeon® Processor D makes an attractive option to use as the processor over Intel® Xeon® E3 processor in each storage server for standard use cases where SSD are used as journals and HDD as backing storage. As you will see from the chart below, Intel® Xeon® processor D-1541 can achieve up to 2.5x of a performance improvement over Intel® Xeon® processor E3v3. The integration of the Intel Xeon processer D can deliver performance at a lower cost.

See Appendix B for the configuration and tuning parameters used in this Intel Xeon processor D chart.

Use Case 3 Ceph Block Storage – SQL Database and High IOPS

In this section, we will outline the optimal configuration for a Ceph cluster to support a low latency high throughput workload for databases. While any configuration of Ceph cluster can provide block storage usable by any application, there are some configuration considerations to be made based on the intended primary workload. For example, an archive workload may require only that the network bandwidth be sufficient for large block data transfers, while a VDI workload may have a high transaction rate or small block size, but latency consistency within a narrow range may not be as important.

A relational database workload (often referred to as OLTP) is typically a small transfer size workload (4k-8k per I/O) and very random in nature. The mix of reads and writes is typically skewed more to reads, with 70 or 80 percent of transactions being reads. Storage transaction latency becomes important with a relational database workload, because each database transaction may contain multiple serialized storage transactions.

In order to achieve the best performance and lowest latency for these kinds of workloads, all-flash—specifi-

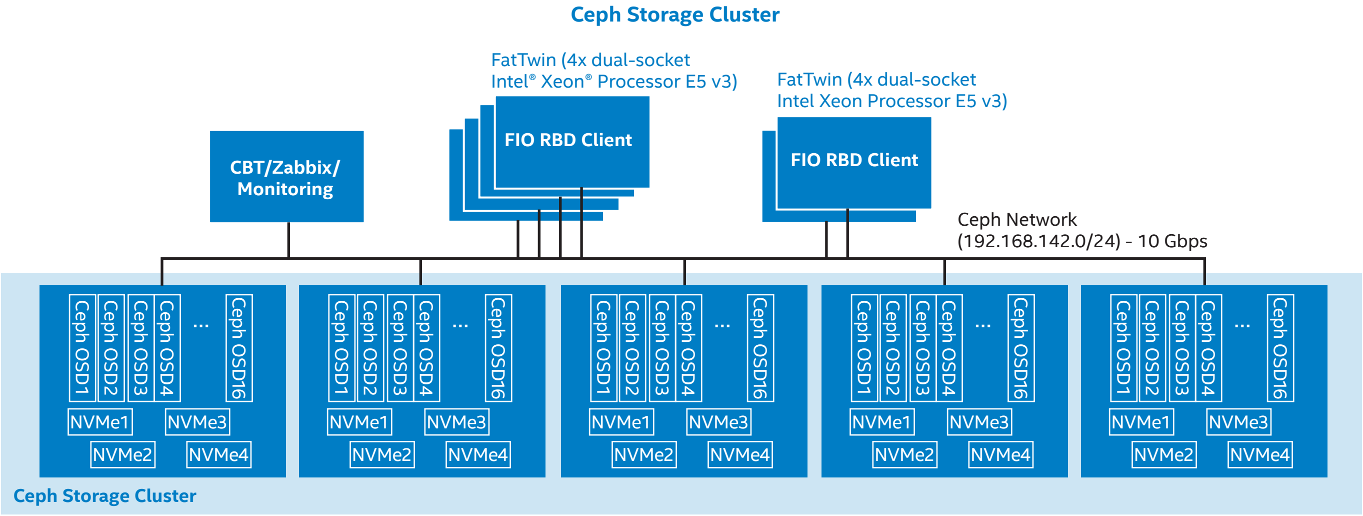


Figure 10. Dense 5U All-flash Ceph Cluster and Clients

WORKLOAD PATTERN	MAX IOPS	LATENCY
4K 100% Random Reads (4.8 TB Dataset)	1.15 M	1 ms
4K 100% Random Writes (4.8 TB Dataset)	200 K	3 ms
4K 70%/30% Read/Write OLTP Mix (4.8 TB Dataset)	452 K	3 ms

Table 9. In the described configuration, the following maximum performance is obtained

cally PCIe-based flash devices using the NVMe architecture—are utilized. To achieve the highest utilization possible of the flash, high core-count CPUs have been found to increase the level of parallelism and performance within the storage node. This is because Ceph OSDs contain many active threads, and more cores in the system allows more processes to run simultaneously increasing transactional throughput.

The cluster consists of five 1U All-flash systems, each with two Intel Xeon E5-2699 v3 18-core CPUs for a total usable core count of 54, (with Hyper-Thread enabled, 36 CPUs are utilized by the OS), 128 GB of DDR4, and four Intel P3700 800 GB NVMe Flash devices. There are four 10GbE network ports available per node, in this configuration a single 10GbE link is used for public and private Ceph networks. The targeted workload (small block transactional) is able to achieve > 1 M random IOPS with the single 10GbE node connection, saving switch ports. Multiple ports could be employed in order to boost sequential throughput if required. The installed software is Linux, the CentOS 7.1 distribution, along with Ceph version 0.94.3 (Hammer release).

The dense 5-node cluster only occupies 5U of rack space and contains 16 TB of raw capacity. Using up to 10 2.5" SFF NVMe slots per node, up to 40 TB of flash could be installed using the P3700 800 GB NVMe, or more if a higher capacity flash device was used. The cluster is also scale-able dynamically and concurrently using Ceph's automatic data distribution scheme. Other advanced features such as thin provisioning, snap shots, cache tiering, and OpenStack integration are supported within Ceph.

One configuration tactic employed for this workload is "multi-partitioning" of the NVMe devices. In this scenario, the NVMe devices are partitioned into four OSD devices, with four journal partitions each. In this way, Ceph runs 16 OSDs per node on top of the 4 physical devices. By using multiple OSD partitions, lock contention within a single OSD process is reduced, resulting in lower latency at all queue depths and much higher maximum throughput. While this introduces the concept that Ceph is storing data for multiple OSDs on the same physical device, data durability is maintained within Ceph by the default crush map which distributes replicas outside of the same single node. If desired, an additional Crush map level for "device" could be created.

A two-replica pool is used to house the RBDs (block device volumes). The reason for employing two copies of

IODepth Scaling - Latency cs IOPS - Read, Write, and 70/30 4K Random Mix
5 nodes, 60 OSDs, Intel® Xeon® Processor E5 2699 v3 Dual Socket/
128 GB Ram/10GbE Ceph0.94.3 with JEMalloc

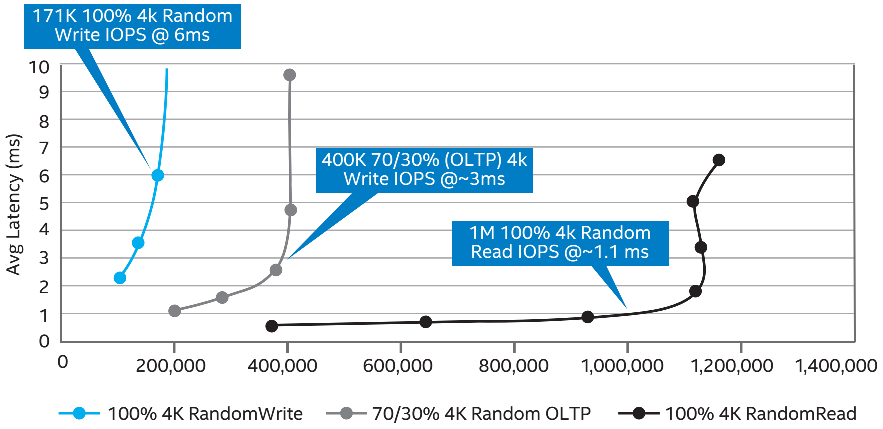


Figure 11. IODepth Scaling

data instead of three is due to the much higher endurance and reliability provided by flash and NVMe. Also shorter rebuild times due to smaller capacities and higher speeds, all of which contributes to less data exposure in the event of failures. Flash devices are much less likely to fail than mechanical spinning drives, and the endurance of the data center class DC P3700 devices is in excess of 10 full drive writes per day. The physical hardware configuration is also more robust, as there is only the PCIe connection from the CPU to the device in the storage path, as opposed to SAS/SATA devices that must go through a PCIe based HBA, and then through a SAS/SATA cable connection with multiple connectors.

Beyond just maximum transactional throughput, it is also important to look at the "useful performance"—that is, the maximum performance at a given latency required for the workload. The following diagram shows the performance in IOPS compared to the latency for the same three workload mixtures.

As shown in Figure 11, the performance of the "Database" mixed workload with 4K random reads and writes in a 70%/30% mixture reaches 400,000 IOPS at 3ms of average latency. The ability to randomly read and write small amounts of data at a low single-digit latency is important for performance critical database workloads.

In summary, this section describes the hardware and software configuration for a low-latency Ceph cluster that provides a high level of transactional performance suitable for relational databases. The high level characteristics include: Data center class Intel DC P3700 NVMe devices, high core count Intel Xeon processor E5 2699v3 processors, and high density 1U servers. The Ceph software is configured to use multiple partitions on a single NVMe device in order to boost throughput and reduce latency. The result is a 5U scale-able storage cluster that can support reads at over 1M IOPS at 1ms of latency, up to 200K IOPS of writes, and over 400K IOPS 70%/30% mixed read/write performance at 4ms of latency.

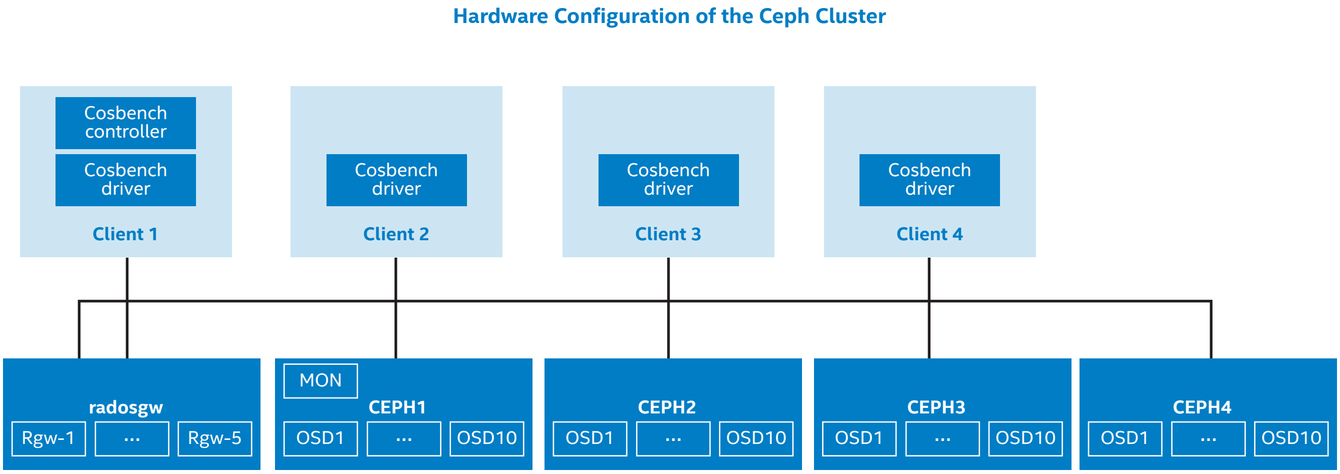


Figure 12. Hardware Configuration

Example #4 Ceph Object Storage

In this section, we will outline the optimized configurations for Ceph object storage and a brief performance overview of a typical Ceph object cluster. Object storage is an emerging technology that is different from traditional file systems (e.g., NFS) or block device systems (e.g., iSCSI). Amazon S3 and Openstack* swift are well-known object storage solutions. In this section, we use COSbench as the benchmarking tool. COSBench is developed and maintained by Intel, which is designed to measure the performance of Cloud Object Storage services by multi object interfaces including OpenStack* Swift and Amplidata v2.3, 2.5 and 3.1, as well as custom adaptors.

Ceph Object Gateway (Rados Gateway, RGW) is an object storage interface built on top of librados to provide applications with a RESTful gateway to Ceph Storage Clusters. When we referring to "object interface performance of Ceph cluster," there actually have two ways to perform the performance test: a. adopting one or more radosgw nodes as gateway(s) to access the Ceph cluster; or b. put/get objects directly through librados library. We adopted the first method for the performance testing in the following part. The tests cover two different scenario: small objects (128 KB) and large objects (10 MB).

Figure 12 describes the hardware configuration of the Ceph cluster.

There are four Ceph-osd nodes and one radosgw node in the cluster. Each Ceph-osd node has 10 Seagate 3.5" 3 TB 7200rpm HDD as OSD device and two Intel S3500 400 GB 2.5" SSD working as journal device. The radosgw node will be more CPU intensive, so this node is equipped with 2 x Intel Xeon processor E5-2699 v3 @ 2.30 GHz, and 64 GB DDR3 memory.

For client side, we used four nodes as COSBench driver and one of these also acts as COSBench controller. All four COSBench driver will generate a random layout object request and send to radosgw node. All nodes are connected by 10GbE Intel® 82599ES port.

HDD CONFIGURATION	Readahead 2048, writecache on
HAPROXY CONFIGURATION	Listen on 5 ports: 7850 – 7854

CEPH OPTIMIZED TUNING	mount omap of each osd to a SSD partition
	turn down all debug log in Ceph.conf
	
	[global]
	mon_pg_warn_max_per_osd = 1500
	ms_dispatch_throttle_bytes = 1048576000
	objecter_inflight_op_bytes = 1048576000
	objecter_inflight_ops = 10240
	throttler_perf_counter = false
	[osd]
	osd_op_threads = 20
	filestore_queue_max_ops = 500
	filestore_queue_max_bytes = 1048576000
	filestore_queue_committing_max_ops = 500
	filestore_queue_committing_max_bytes = 1048576000
	journal_max_write_entries = 1000
	journal_queue_max_ops = 3000
	journal_max_write_bytes = 1048576000
	journal_queue_max_bytes = 1048576000
	filestore_max_sync_interval = 10
	filestore_merge_threshold = 20
	filestore_split_multiple = 2
	osd_enable_op_tracker = false
	filestore_wbthrottle_enable = false

Table 10. Ceph Cluster System Configuration

WORKLOAD PATTERN	OBJECT SIZE	CONTAINERS
Small object	128 KB	Random(1,100)
Large object	10 MB	Random(1,100)
WORKLOAD PATTERN (CONT.)	**OBJECTS**	**WORKER**
Small object (continued)	Random(1,100)	320
Large object (continued)	Random(1,100)	320

Table 11. Workload Pattern

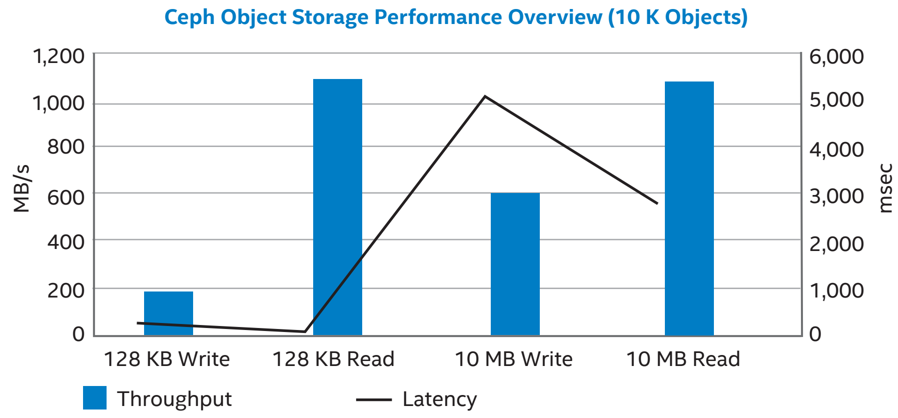


Figure 13. Ceph Cluster Object Performance Overview

For the Ceph cluster configuration, below performance based on the latest release, Infernalis (9.2.0), replica size is two. Detailed configuration is shown at Table 10 and 11.

Figure 13 shows Ceph cluster object performance tested by small/large object size, with optimized Ceph tuning.

As we can see, read performance with small object or large object all hit RadosGW node NIC maximum throughput. But there is some issue with small objects (128k Write) write performance, which is suspected to be caused by the RadosGW implementation, which will need further investigation.

Example #5 Cloud Digital Video Recording (Cloud DVR)

Cloud DVR and U.S. Law

Due to the evolution of copyright laws, the U.S. DVR market is restricted to individual ownership and single copies of broadcast content. This started as recording shows on home VCRs with the Fair Use laws, which has now been applied to Digital Video Recorders. However with the emergence of cloud computing and scale out storage, it is becoming an increasingly popular architecture to store and play back TV shows over the internet. This also provides flexibility to download individual consumer recordings over any internet connection, wherever they are. Based on legal and country specific roles, content can either be a shared copy or private copy when stored in the cloud. In the U.S. Digital Rights Management and Fair Use laws dictate that every individual who records content on a Cloud DVR have a separate copy, which cannot be duplicated for data protection or any other reason. This technically includes any deduplication of identical content for storage efficiencies. To complete the picture, most content ~85% on a DVR (cloud or stand-alone) is not actually viewed. So you have a streaming object store cloud environment that is write dominate—more like archive, backup, or digital video surveillance.

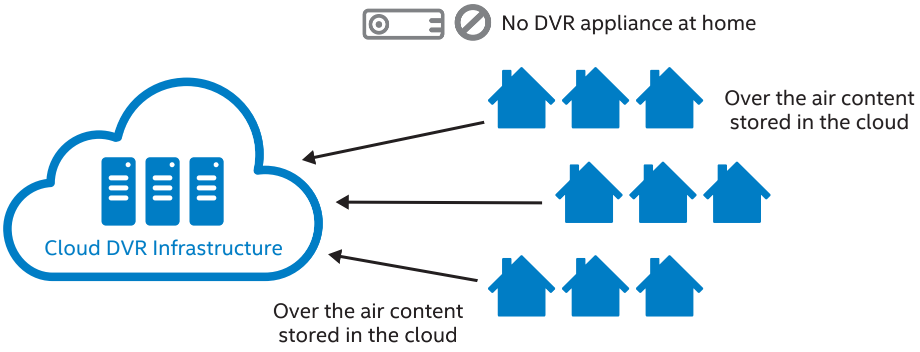

No DVR appliance at home
Over the air content stored in the cloud
Cloud DVR Infrastructure
Over the air content stored in the cloud

Figure 14. Cloud DVR Infrastructure

OPERATION TYPE	BIT RATES	% OBJECTS
8-14 sec Subscriber Segment Reads	4 MB	80
	2.5 MB	10
	1.9 MB	5
	1.25 MB	5
8-14 sec Subscriber Segment Writes	4 MB	25
	2.5 MB	25
	1.9 MB	25
	1.25 MB	25

Table 12. Typical mix of subscriber read and write object sizes and mix

This individual copy has other implications for space reclamation as content rolls off. (i.e., “I only want to pay to keep the last week of a show in my DVR.”) Because of the restriction of only one copy, to remove content out of an object storage one has to rewrite each of the recorded segments which are usually stored as a few seconds of content. So while this doesn’t impact the bandwidth into and out of the cloud, it does significantly impact the cloud DVR storage system as the objects have to be over-written before they are returned as free space.

Storage DVR content in the cloud provides the following advantages:

- Reduced storage cost as the content can be shared across different account holders internally with unique external views (outside the U.S.)
- Cloud DVR can take advantage of storage innovations like erasure coding to achieve space savings
- Ability to access content from any device within large geographic areas as the content is stored in the cloud
- Additional data protection though data center practices as the content can be shared among multiple servers or data centers in the cloud. DVR appliances at home can fail which results in content loss
- Additional revenue opportunities for service providers by bundling services like targeted advertisements

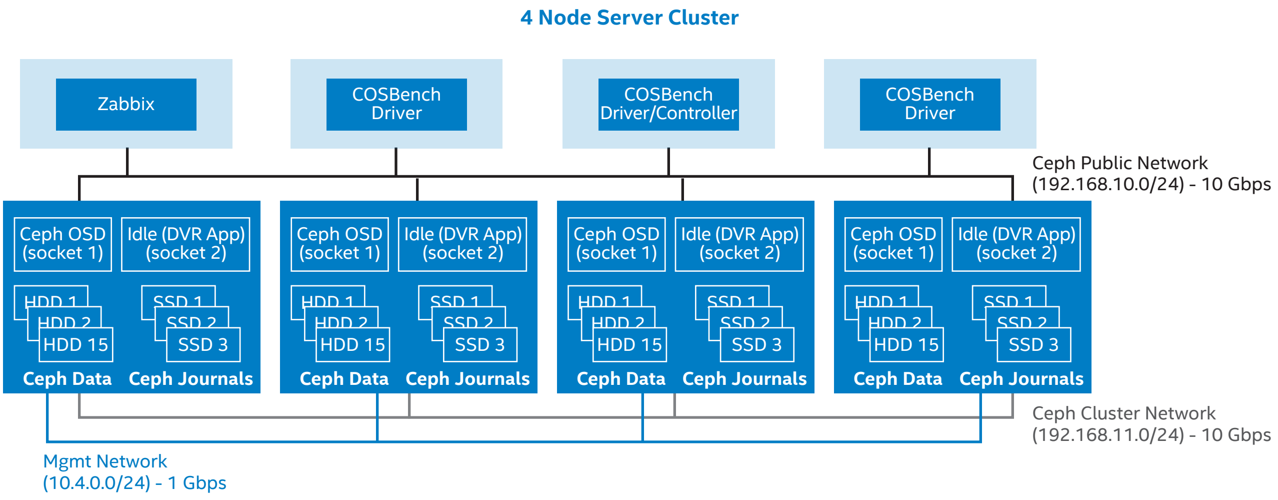

4 Node Server Cluster
Zabbix
COSBench Driver
COSBench Driver/Controller
COSBench Driver
Ceph Public Network (192.168.10.0/24) - 10 Gbps
Ceph OSD (socket 1)
Idle (DVR App) (socket 2)
HDD 1
HDD 2
HDD 15
SSD 1
SSD 2
SSD 3
Ceph Data
Ceph Journals
Ceph Cluster Network (192.168.11.0/24) - 10 Gbps
Mgmt Network (10.4.0.0/24) - 1 Gbps

Figure 15. 4 Node Server Cluster Used To Test DVR scenarios

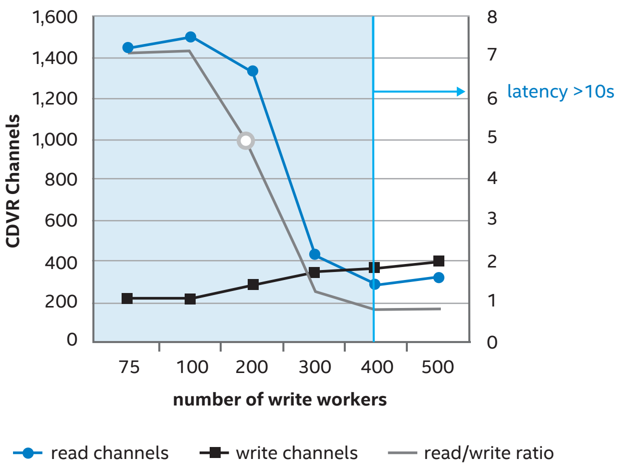


Figure 16. 100 Readers

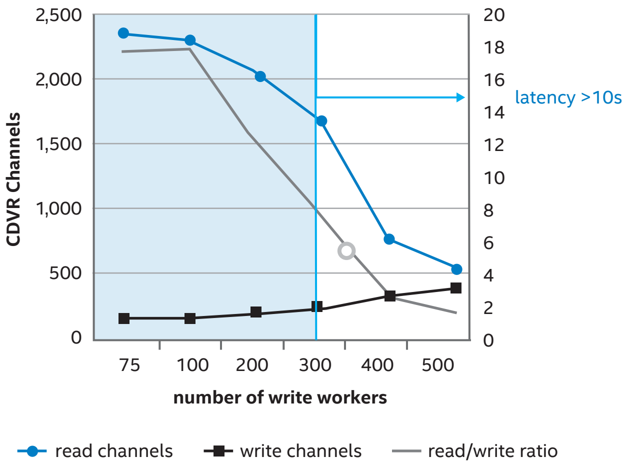


Figure 17. 200 Readers

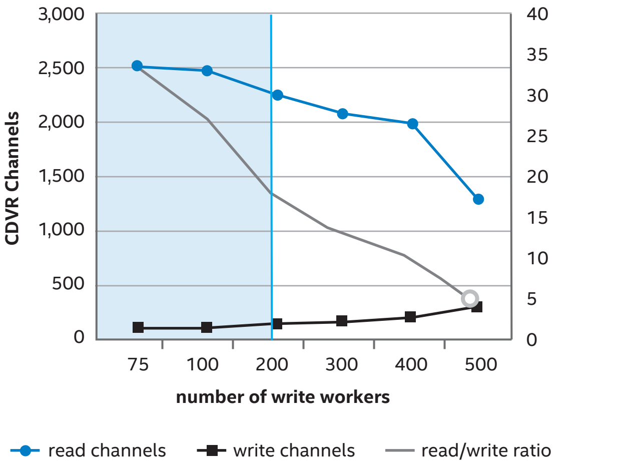


Figure 18. 300 Readers

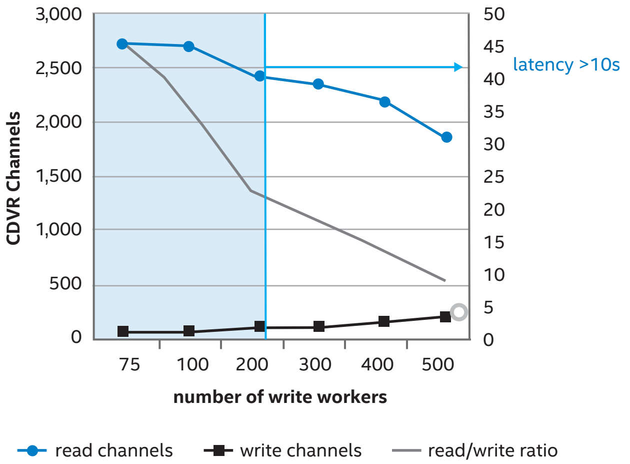


Figure 19. 400 Readers

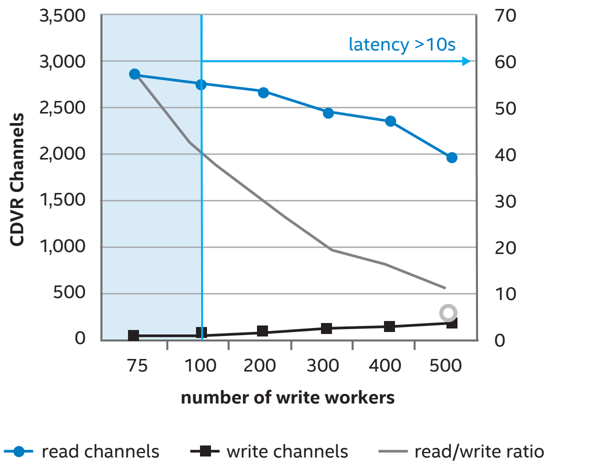


Figure 20. 500 Readers

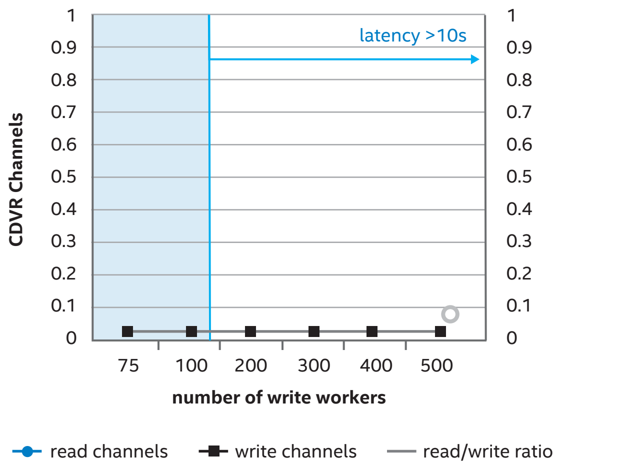


Figure 21. 600 Readers

Ceph Cloud DVR Lab Testing

Ceph is becoming an increasingly popular choice to provide scale out storage for storing cloud DVR content. Ceph provides erasure coded storage with optimized ISA-L EC plug-in for accelerating erasure coding for reading and writing DVR content.

Private copy use-case is a write heavy workload as the content need to be stored per user even if it is same over the air content for other subscribers. Shared copy use-case is a read heavy workload as the content is shared so writes tend to be lesser compared to reads. Media content is stored on per segment basis in the cloud and it normally tends to be 8-14 seconds play back time.

The following four node server cluster is used to test various DVR scenarios. This is a hyper converged configuration where only one socket is used to run Ceph storage services. Each dual socket storage node contains Intel Xeon processor E5-2620 v3 @ 2.40 GHz, 24 cores w/ HT, 96 GB, Cache 15360 KB, 8 Channel SATA 6 Gb/sec SAS low profile 600 MBps PCIe 3.0 x8, DCS3700 400 GB SSDs. Ceph journals are stored on SSD drives. One SSD is used to store five Hard Disk Drive journals. COSBench is used to simulate subscriber segment read and write operations using three servers.

Figures 16-21 show a different mix of subscriber readers and writes and latency which is used to arrive at optimal mix of read and write operations for shared copy and private copy Cloud DVR use-cases.

Example #6 Backup/Archive

Backup/Archive Introduction

Ceph object storage coupled with erasure coded pools can be used to archive and backup objects efficiently. Backups range from small objects like pictures and images to very large objects that are several Giga Bytes in size. Erasure coding is a data protection method that can safe guard against node, rack, and data center failures to protect data. The following picture outlines encoding operation that takes the original object, encodes it into segments and distributes the encoded segments across the cluster to achieve data protection goals. Similarly decoding process reconstructs original object by taking shards and decoding to create original object (Figure 22).

The Advantages of Intel Accelerated Erasure Coding Algorithms

Intel EC Acceleration is available in Ceph as part of the standard release. Simply include the library and enjoy the faster throughput. Erasure coding (EC) algorithms can be configured to provide equal or better data durability than triple RAID data redundancy while using up to 50 percent less storage. Calculation based on internal Intel measurements on usable capacity of 320 drives totaling 960 TB of raw capacity with no single point of failure, compared to a 3-way RAID setup; EC uses a configurable scheme so numbers vary but a common one is for every 14 drives you get to store 10 drives worth of data so 14n/10n = 1.4x vs. 3x for typical tri-replication. Features and benefits may require an enabled system and third party hardware, software, or services. Performance varies depending up your specific configuration. Consult your system provider. For more information go to http://www.intel.com/performance

Intel Cache Acceleration Software with Ceph

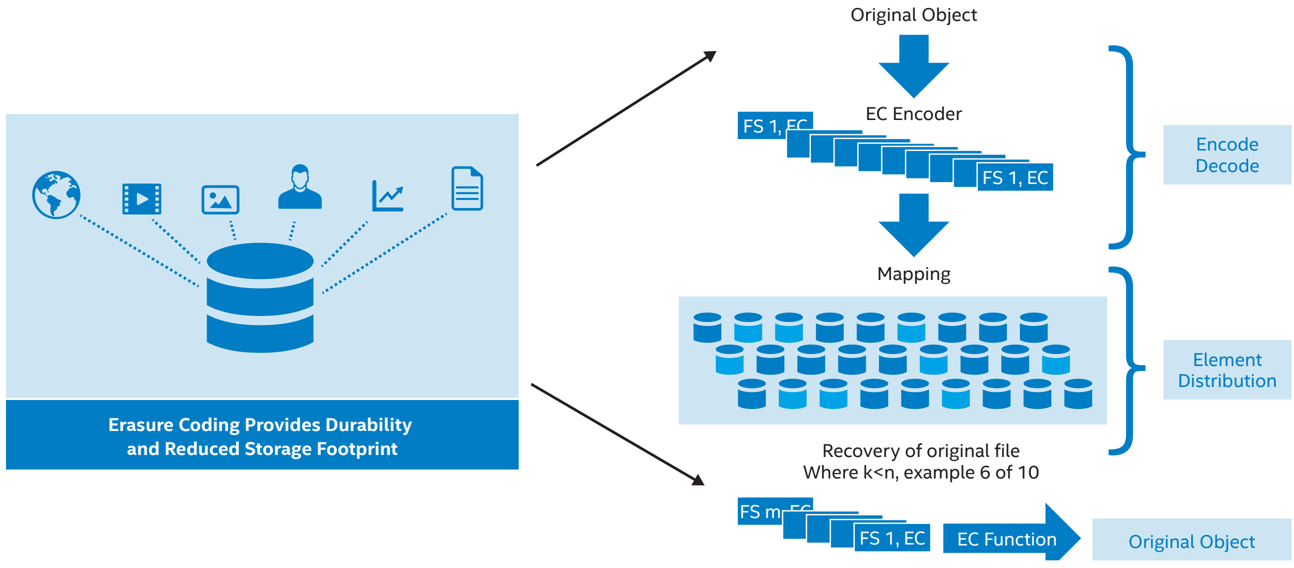


Figure 22. Cache Acceleration Software

Example #7: Intel Cache Acceleration Software with Ceph

In cases where moving to 100 percent SSD storage is impractical due to cost or capacity, as may be the case with large-scale Ceph deployments, Intel® Cache Acceleration Software (CAS) provides a mid-level performance, lower cost alternative.

Intel CAS is a file-level for Windows* and a block-level for Linux* software caching solution. When paired with an Intel® DC Series PCIe NVMe SSD, Intel CAS delivers excellent performance improvements to your existing HDD backing store at a fraction of the cost of replacing the HDDs 100 percent with SSDs.

Intel CAS for Linux is implemented as a loadable kernel module and user-space administration tool. It is a truly "drop-in" solution that requires no changes to your OS, apps, or hardware infrastructure.

Intel CAS caches your hottest data on a fast SSD for high throughput, low latency delivery. Traditionally, like most other caching solutions, Intel CAS had cached all block storage accesses using a Least Recently Used (LRU) eviction algorithm to keep the hottest data in the cache and evict the oldest unused data. However, Intel CAS v3.0 for Linux delivers a new feature called **I/O Classification**, which is based on Intel® Differentiated Storage Services (DSS) Technology invented by Intel Labs. This new I/O Classification feature enables classification, prioritization, and selective allocation of I/O at a finer granularity than the traditional LRU caching algorithm, giving the user more flexibility and control than ever before over what gets cached and what stays in the cache. Ultimately, this results in improved cache performance (which translates to higher throughput and lower latency) with the added benefit of reduced cache capacity requirements.

Intel CAS exposes the following I/O Classes, each of which can be enabled/disabled for caching and given its own priority for eviction through a user editable configuration file. See Table 13.

Over the past year, Intel has worked with Yahoo to improve their Ceph performance for Yahoo Mail,* Flickr,* and Tumblr.* Yahoo's typical workload uses a Ceph object store implementation with XFS as the underlying file system, and using 8+3 Erasure Coding (EC 8+3) (instead of 3x replication, to maximize HDD capacity availability) to store Yahoo Mail,* Flickr,* and Tumblr* attachments, photos, and videos.

The Problem (everything counts in large amounts)

In this implementation, the file system of the individual Ceph storage node normally contains 100s of millions of small files. Retrieving one of these files required gathering the 8+3 erasure coded pieces to reassemble them into the original object. In order to get each piece, that piece must first be found on the file system by tracing through the XFS inodes to find that file on the disk hosting the corresponding XFS file system. Because there are 100s of millions of files on the disk, this results in having to trace through 4-6 inode blocks before loading the small (few block) file. The direct impacts are a latency that is up to 6x longer than it takes to load the small file itself, and throughput that is one-sixth the overall potential. To make things even worse,

CAS DSS IO CLASSES
Unclassified
Metadata *(Superblock, GroupDesc, BlockBitmap, InodeBitmap, Inode, IndirectBlk, Directory, Journal, Extent, Xattr)*
<=4KiB
<=16KiB
<=64KiB
<=256KiB
<=1MiB
<=4MiB
<=16MiB
<=64MiB
<=256MiB
<=1GiB
>1GiB
O_DIRECT
Misc

Table 13. User Editable Configuration File

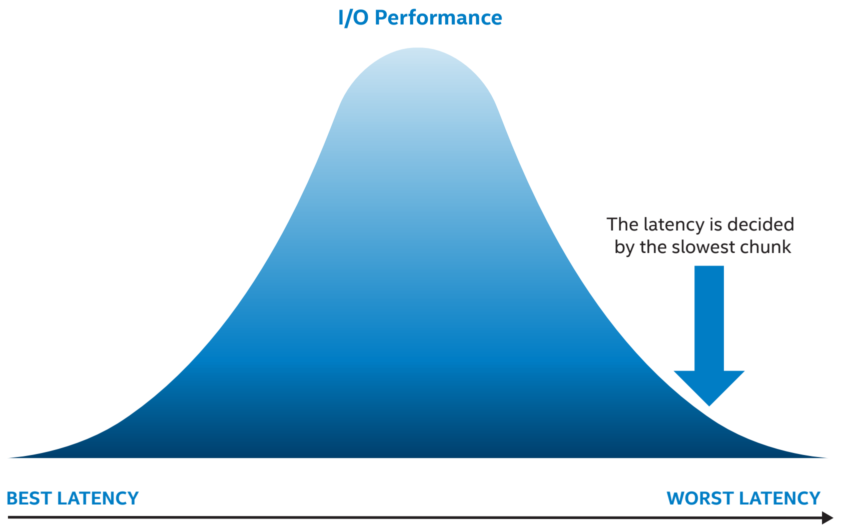


Figure 23. I/O Performance

the overall latency for reconstructing the object is dependent on the worst case (tail) latency of retrieving the first eight of the 11 EC pieces.

Yahoo's SLA with their customers was such that they were over-provisioning by 3X in order to meet the performance expectations. And these are not small numbers. When Flickr wants 1 PB of storage Yahoo would have to provision 3 PB, all running in parallel, to reach the required level of throughput.

Solution: Do things differently with Intel Cache Acceleration Software featuring Intel® Differentiated Storage Services Technology

The CEPH solution Intel designed for Yahoo was to move all of the XFS file system metadata into a cache using Intel® Differentiated Storage Services (DSS) Technology and Intel® CAS. The inodes are re-used over and over when retrieving files, while the files themselves may be less frequently used and not as useful to cache. The major performance improvements are achieved by caching the file system metadata alone. This is where Intel CAS I/O Classification excels. Intel CAS allows the user to choose to cache metadata while not caching file data, thus achieving the best performance with the smallest possible cache.

By caching the file system metadata, all of the inode accesses are served at PCIe NVMe throughput and latency, leading to overall throughput improvement and latency reduction.

The configuration details:

Hardware Configuration:

- **SERVER:** HP ProLiant DL180 G6 ySPEC 39.5
- **CPU:** 2x Intel® Xeon® processor X5650 2.67 GHz (HT enabled, total 12 cores, 24 threads)
- **CHIPSET:** Intel® 5520 IOH-36D B3 (Tylersburg)
- **RAM:** 48 GB 1333 MHz DDR3
  - 12 x 4 GB PC3-10600 DDR3-1333 ECC Registered CL9 2Rx4
- **HDD:** 10 * 8 TB 7200 RPM SATA HDDs
- **SSD:** 1 * 1.6 TB Intel P3600 SSD
- **NET:**
  - 2 * HP NC362i/Intel 82576 Gigabit
  - 2 * Intel® 82599EB 10Gbe
- **OS:**
  - RHEL 6.5, kernel 3.10.0-123.4.4.el7

Ceph Configuration:

- Ceph Giant v87.1
- 1 admin node
- 2 monitor nodes
- 8 OSD nodes, each with 10 * 8 TB Enterprise-class SATA HDDs and 1 * 1.6 TB Intel DC P3600 SSD.
  - NOTE: The SSD is used for both journaling and caching. The drive is partitioned to have a 1.5 TB partition for caching and 10 * 10 GB partitions for journaling (one 10 GB partition per OSD).
- EC 8+3

Benchmarking:

To benchmark performance we took GET and PUT performance samples using rest-bench. Samples were taken with and without caching at different levels of cluster data loading (fullness). The benchmarking process is was as follows:

1. Clear page and disk caches in between each step below
2. Fill cluster 10% with caching disabled (PUT)
3. Fill cluster 10% with caching enabled (PUT_cache)
4. Read (GET) test with caching enabled (GET_cache)
5. Read test with caching disabled (GET)
6. Repeat with GET test points at cluster 50% full and 70% full
7. Compare non-cached to cached performance.

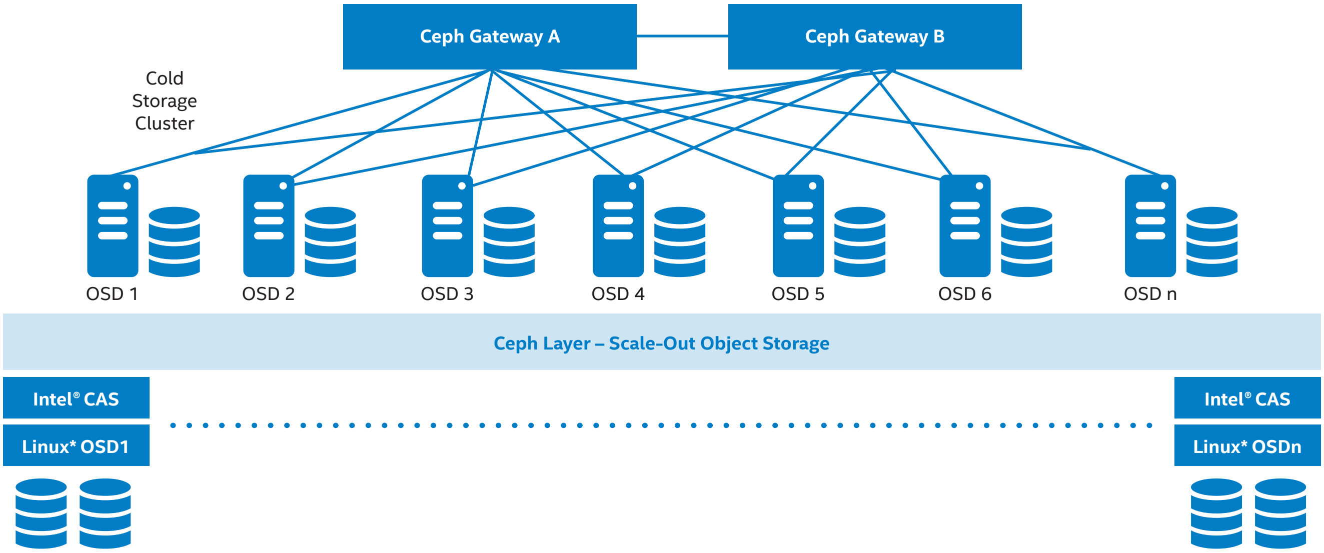


Figure 24. Ceph Configuration

Results:

Using Intel CAS Software with I/O Classification paired with an Intel DC P3600 SSD cache that is just 2 percent of the size of the backing store, Yahoo has achieved:

- 200% GET throughput increase
- 50% GET latency reduction
- 100% PUT throughput increase
- 30% PUT latency reduction
- ½ the # of clusters needed to meet SLA performance requirements (as a result of the GET throughput increase and latency reductions, Yahoo can now meet the SLA level throughput with half the previously necessary amount of overprovisioning = fewer HDDs and servers needed).

As can be seen in Figure 25, using Intel CAS featuring Intel DSS resulted in a reduction of latency AND an increase in utilization as the number of requests scales. (RPS = requests per second) As the utilization grows the CAS solution doesn't suffer from increased overhead of longer and longer head seeks inherent with HDDs used in random access applications.

For writes the results were even more dramatic, (Figure 26), as the all HDD solution would time out 30 percent of the time.

This solution translates to real-world CapEx savings (over-provisioning), OpEx savings (reduced power, space, and cooling), and improved scalability planning (performance and predictability).

In summary, if you're looking for a mid-level performance, low cost Ceph performance adder, check out Intel® Cache Acceleration Software, Intel® featuring Intel® Differentiated Storage Services (DSS) Technology and Intel PCIe NVMe SSDs.

To learn more, contact your Intel Representative, or go to http://www.intel.com/cas

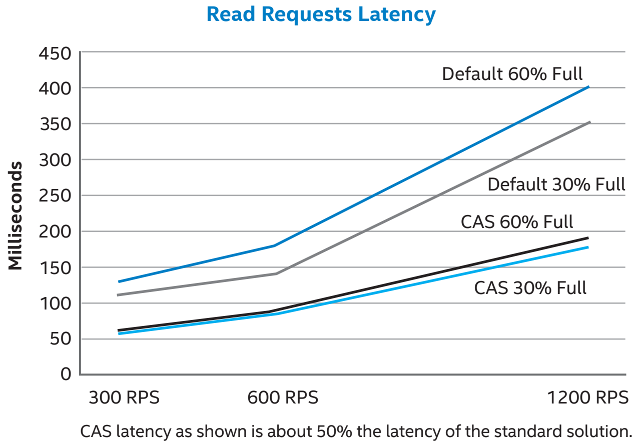


Figure 25. Read Requests Latency

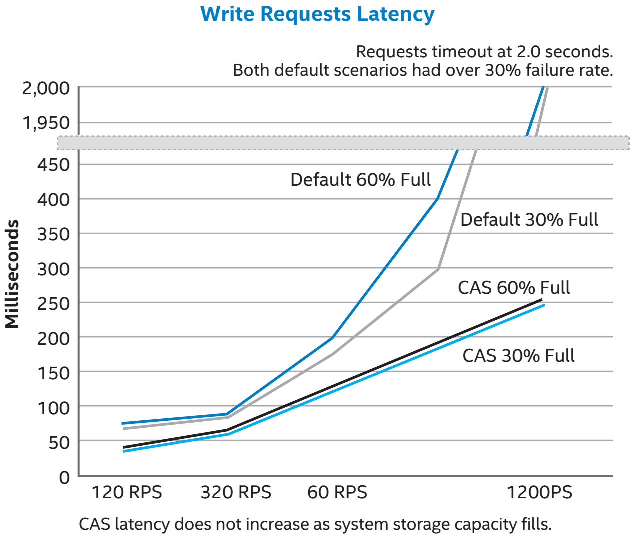


Figure 26. Write Requests Latency

Differentiated Storage Services – Next Steps

In the previous section's discussion of large scale Ceph object store used by Yahoo, we have shown the potential of applying the Intel® Differentiated Storage Services (DSS) technology feature from Intel® Cache Acceleration Software and Intel PCIe NVMe SSDs. Ceph, moreover, is capable of providing block level as well as file level storage services, making it one of the most appealing storage service solutions.

However, I/O characteristics from applications using Ceph may be very different from what Ceph storage nodes actually see, such as in Yahoo's case. Similarly, in case of using Ceph RADOS Block Device (RBD) for block storage for virtual machines (VM), I/O characteristics from an application inside one VM may be very different from what Ceph storage nodes actually see, furthermore, may also be very different from the same application inside another VM. Traditional storage caching may not perform very well due to lack of understanding of I/O from applications, even with the investment of large capacity SSDs as caching devices. Meanwhile, almost all Cloud Service Providers (CSPs) are seeking better ways of (1) achieving Service Level Objects (SLOs) and, (2) creating differentiation in Service Level Agreements (SLAs). The former allows CSPs minimize the risk of violating customers' SLA and the latter allows CSPs to build flexible pricing models for customers.

Intel Labs is leading the research in cloud based storage services to help bring out the fullness of the versatility in Ceph. More particularly, Intel Labs is actively exploring Ceph RADOS Block Devices (RBD) in QEMU. The individual I/O from any particular VM, with the help of DSS I/O hinting mechanism, can carry its own I/O class information. This per VM I/O hinting allows more advanced tuning on the storage caching policy on a per VM basis, or even per VM on a per VM host basis. This potentially allows Ceph, while providing storage services without having to know actual applications, to take full advantage of high-performing Intel® PCIe NVMe SSDs through DSS caching policies tuned based on per VM I/O hints. On the other hand, in the use case like Yahoo, Intel Labs is also actively looking into native object based I/O hinting. For example, customers may prefer faster access of pictures to documents form the same objet store where Ceph is providing service through Ceph RADOW Gateway (RGW). Furthermore, DSS I/O hints available to Ceph, either from block or object storage services, become a valuable asset for CSPs to derive more intelligent storage caching policies based on their own definition of SLA differentiations. Intel Labs is also actively looking to cloud DSS for innovative methodologies of helping Ceph users in this regard.

SLAs are the overall agreement, SLOs are the particular performance metrics that make service.

CeTune – UI Based Benchmark Tool for Ceph

As the landscape of storing and retrieving huge amount of data to cloud storage continues to grow and expand, so do the needs to benchmarking, profiling and tuning the cloud storage solutions in a much easier and efficient way. Intel observed in production environments, customers still face numerous challenges to driving the best performance, including how to trouble shoot the bottlenecks, identify the best tuning knobs from the many (500+) parameters and handle the unexpected performance regression between frequent releases. To make this easier Intel developed and open sourced CeTune—the Ceph benchmarking, profiling and tuning tool.

The CeTune framework comprises five distinct components:

1. The Deployer, which can easily deploy a Ceph cluster in minutes
2. The Benchmarker, generates well defined use cases and automatically evaluate the RBD, object, and CephFS performance with various pluggable workloads
3. The Analyzer, monitors the performance of all aspects (system characterization data, workloads throughput, latency) with a single interface and reveals Ceph software stack latency through common visualization GUI based on Lttng and Zipkin
4. The Tuner, dynamically injects args and compares the performance to identify best tuning knobs
5. The Visualizer, automatically presents the data on a web based performance portal

With CeTune, we can evaluate the Ceph performance for every major release, identify performance bottleneck, and publish the results for users/developer reference in a short time (Figures 27 and 28).

CeTune is available for download at: https://github.com/01org/CeTune.

Virtual Storage Manager (VSM) for Managing Ceph Clusters

The survey from OpenStack Summit Vancouver 2015 shows, 44 percent Openstack adopters are using Ceph as block storage option (Figure 29).

But operating Ceph cluster is still complex and painful for storage administrators and operators. To lower the barrier for adoption and accelerate the landing of Ceph based solutions, we developed Virtual Storage Manager (VSM) to fill the operation gaps. Virtual Storage Manager (VSM) is a Ceph management tool, it was published as an open source project on 2014 Nov. OpenStack Paris summit.

Virtual Storage Manager (VSM) consists of two major components: controller and agents.

- VSM Controller runs on dedicated server or server instance, and manage Ceph cluster through VSM agents. Also, if users expect to present storage pool resources for OpenStack use, VSM controller is in charge of the connection to OpenStack cluster.
- VSM agent runs on every Ceph server, it accepts requests from controller and relays server configuration and status information to VSM controller.

Figure 27. CeTune Generated Performance Report

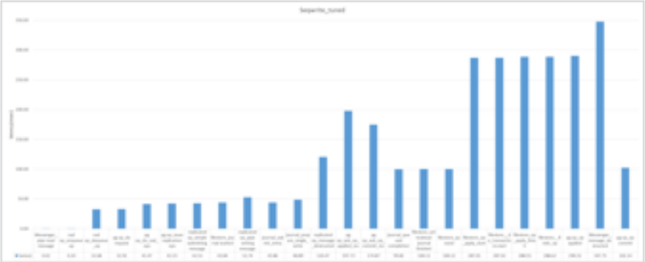


Figure 28. CeTune Generated Latency Breakdown Report

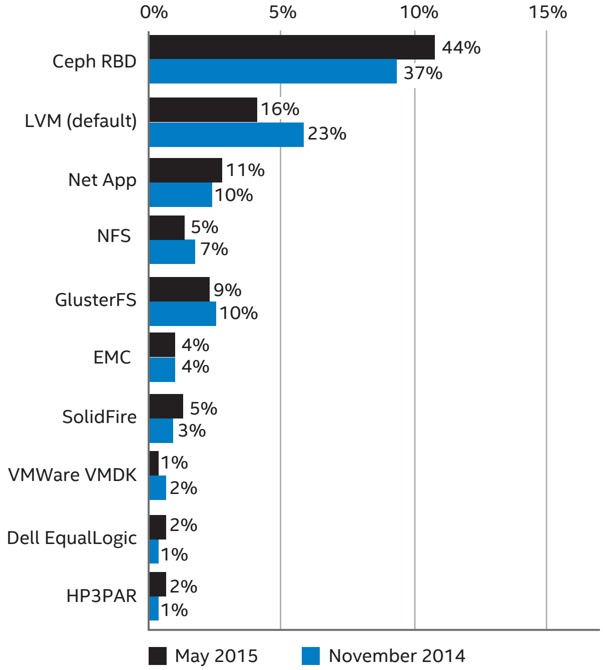


Figure 29. Block Storage Drivers: Production

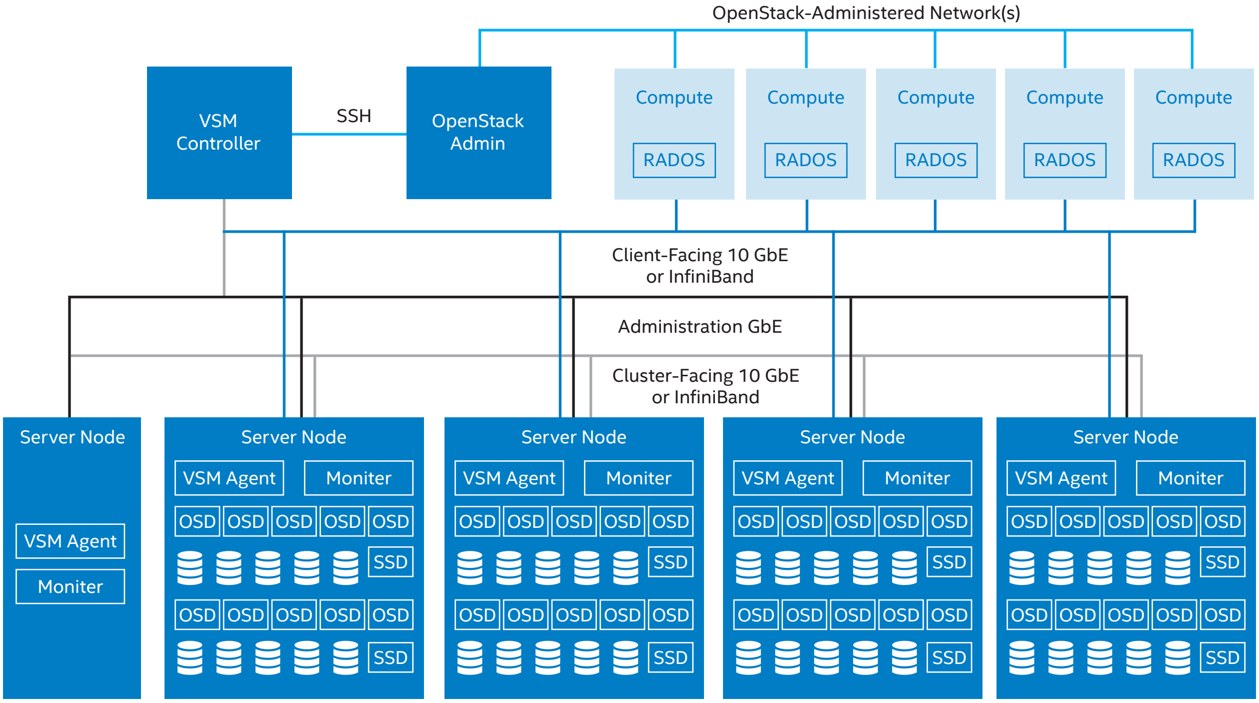
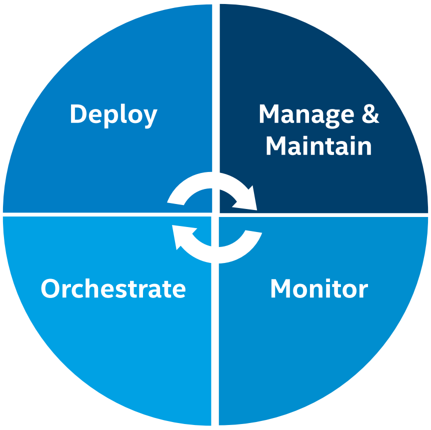


Figure 30. VSM Controller



To support Ceph operation, VSM provides a few features, they are roughly categorized into four areas:

1. Deployment

VSM supports to create Ceph cluster from scratch by itself, what it requires are manifest files to describe the cluster topology. VSM can also manage existing Ceph clusters by importing them.

VSM provides an automatic installer to help early user to deploy VSM itself with one command line.

2. Management and Maintenance

VSM provides features to assist storage administrators for daily operations. One usual task for daily operation is to replace failed disks or servers. VSM could assist and ease those operations and mitigate impacts.

Also VSM can classify different storage media into storage groups, then they could be used to create different storage pools to meet different business use. VSM can create and manage replicated pools, erasure-coded pools and cache-tiered pools, it also supports to assign pool quota, and allows a pool to leverage more than one storage groups, it means to support previously mentioned primary-replica mode.

For a long time system, maintenance is an important aspect, VSM can upgrade itself to newer version, also it supports to upgrade Ceph to specified version.

3. Monitoring

To know cluster status, monitoring cluster health status is vital step. Beside the overall cluster health status, VSM also monitors capacity utilization, which includes cluster level, storage group level, OSD level.

VSM also monitors other resources like servers, devices, pools. For devices, VSM also tried to retrieve disk SMART log to know device status.

Beside resource status, VSM also monitors overall cluster performance like IOPS, Bandwidth, latency, and CPU utilization over time.

4. Integration

Ceph is often integrated with OpenStack cloud platform, and VSM has a feature to present pools to OpenStack. Through it OpenStack could create VM instances on pools with different characteristics to meet different business needs.

To make it easy to integrate VSM with third-party tools and workflows, VSM provides a set of REST API and CLI tools.

The VSM project home is at https://01.org/zh/virtual-storage-manager, the code repository is on https://github.com/01org/virtual-storage-manager, and binary packages can be downloaded from https://github.com/01org/virtual-storage-manager/releases. The community is at http://vsm-discuss.33411.n7.nabble.com/.

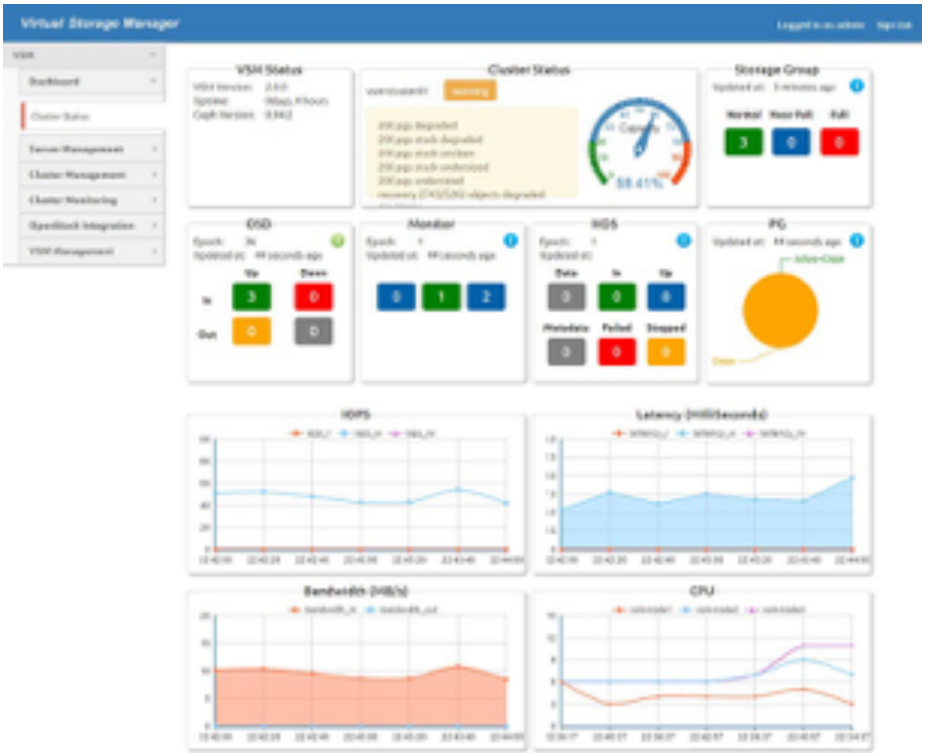


Figure 29. VSM Project Home

Ongoing Work

Intel continues to update Ceph related development, so check for the latest version of this paper located at https://soco.intel.com/docs/DOC-2146636 to get the latest performance and configuration information.

Summary

Cloud workloads and cost are driving the need for scale out storage solutions. Ceph is a popular open source storage software with many significant production deployments already. Backup, archive, virtual block, and media streaming workloads are dominant in the current Ceph production deployments. Intel® Architecture and flash based reference solutions covered in the above sections outline how an end customer can deploy optimized Ceph configurations to meet performance, latency and space efficiency goals with optimal infrastructure.

Appendix A – Recommended Tuning Parameters

The complete current tuning parameters document *Ceph Cookbook: Configuration Guide* is available under Intel NDA. The introduction is included below.

Introduction

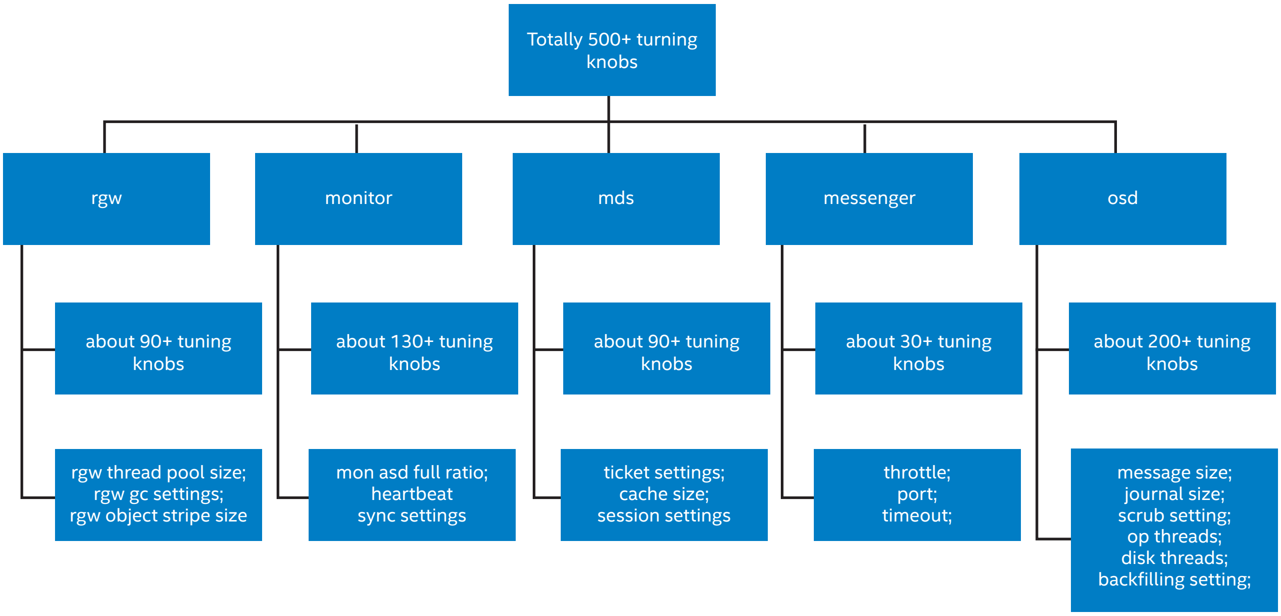
In addition to traditional enterprise-class storage technology, many organizations now have storage needs with varying performance and price requirements. Open-Stack has support for Storage and Block Storage, with many deployment options for each depending on the use case.

Ceph, The Future of Storage,* is a massively scalable, open source, software-defined storage system that runs on commodity hardware. Ceph has been developed for the ground up to deliver object block, and file system storage in a single software platform that is self-managing, self-healing and has no single point of failure. Because of its highly scalable, software defined storage architecture, Ceph is an ideal replacement for legacy storage systems and a powerful storage solution for object and block storage for cloud computing environments.

Ceph has 500+ tuning knobs in total. This document introduces the best known methods on Ceph performance tuning we find in our testing. With some of these tunings the performance has great improvements.

This document assumes that the reader has basic knowledge of Linux operations system and cloud storage infrastructure.

To complete this document, citations of resources from Internet were included. And due to the limit of resource and knowledge, there must be a lot of improvement areas or mistakes in current release. We will keep update the document with new learning or finding identified, and any comments will be appreciated. For more information, check https://github.com/01org/CeTune. There you can find contact information, mail list link, WIKI, and Q&A etc.



Totally 500+ turning knobs

rgw

about 90+ tuning knobs

rgw thread pool size; rgw gc settings; rgw object stripe size

monitor

about 130+ tuning knobs

mon asd full ratio; heartbeat sync settings

mds

about 90+ tuning knobs

ticket settings; cache size; session settings

messenger

about 30+ tuning knobs

throttle; port; timeout;

osd

about 200+ tuning knobs

message size; journal size; scrub setting; op threads; disk threads; backfilling setting;

Appendix B – Intel Xeon Processor D and Ceph Config and Tuning

Ceph Performance Configuration

Ceph version: 0.94.3 “Hammer” with JEMalloc

(# of OSDs * 100) / Size = # of PGs

50 prefilled 25 GB RBD volumes mapped to pool, 10 per client

5:1 OSD/Journal ratio

10 GB journal size

FIO Benchmark Configuration

Version: 2.2.9

I/O Engine: libRBD

Direct: yes

Queue Depth:

32 for 4 KB Random and 1M Sequential I/O

8 for 32 M Sequential I/O

Numjobs: 1

Ramp Time: 30 seconds

Run Time: 300 seconds

Refill buffers: 1

Invalidate: 0

WORKLOAD	OBJECT SIZE	PATTERN
Tiny Workload	4KB	90% Writes 10% Reads 50% Writes 50% Reads 10% Writes 90% Reads
Medium Workload	1MB	90% Writes 10% Reads 50% Writes 50% Reads 10% Writes 90% Reads
Large Workload	32MB	90% Writes 10% Reads 50% Writes 50% Reads 10% Writes 90% Reads

Ceph Configuration [Cont.]

```
[global]
auth_cluster_required = cephx
auth_service_required = cephx
auth_client_required = cephx
filestore_xattr_use_omap = true

debug_default = 0
debug_lockdep = 0/0
debug_context = 0/0
debug_crush = 0/0
debug_buffer = 0/0
debug_timer = 0/0
debug_filer = 0/0
debug_objecter = 0/0
debug_rados = 0/0
debug_rbd = 0/0
debug_journaler = 0/0
debug_objectcatcher = 0/0
debug_client = 0/0
debug_osd = 0/0
debug_optracker = 0/0
debug_objclass = 0/0
debug_filestore = 0/0
debug_journal = 0/0
debug_ms = 0/0
debug_monc = 0/0
debug_tp = 0/0
filestore_op_threads = 8
filestore_max_inline_xattr_size = 254
filestore_max_inline_xattrs = 6
filestore_queue_max_ops = 500
filestore_queue_committing_max_ops = 5000
filestore_merge_threshold = 40
filestore_split_multiple = 10
Journal_max_write_entries = 1000
Journal_queue_max_ops = 3000
Journal_max_write_bytes = 1048576000
osd_mkfs_options_xfs = -f –I size=2048
osd_mount_options_xfs = noatime,largeio,nobar
rier,inode64,allocsize=8M
ods_op_threads = 32
osd_journal_size = 10000
filestore_queue_max_bytes = 1048576000
filestore_queue_committing_max_bytes =
1048576000
journal_queue_max_bytes = 1048576000
filestore_max_sync_interval = 10
filestore_journal_parallel = true
```

Linux Initiator Tuning

Mapped LUNs configuration

```
# echo 1024 > /sys/block/$DEVICE/queue/nr_requests
# blockdev –setra 32768 /dev/sd{}
# echo 128 > /sys/block/sd{}/device/queue_depth
# echo noop > /sys/block/sd{}/queue/scheduler
```

iSCSI setup: Changes to /etc/iscsi/iscsid.conf

```
Node.session.cmds_max=2048
Node.session.queue_depth=128
```

Network configuration

TCP settings: Recommend changes to /etc/systemctl.conf

```
Net.ipv4.tcp_rmem= 10000000 10000000 10000000
Net.ipv4.tcp_wmem= 10000000 10000000 10000000
Net.ipv4.tcp_mem= 10000000 10000000 10000000
Net.core.rmem_default=524287
Net.core.wmem_default=524287
Net.core.rmem_max=524287
Net.core.wmem_max=524287
Net.core.netdev_max_backlog=300000
```

i40e driver settings

```
# service irqbalance stop
# ./scripts/set_irq_affinity <net_dev>
```

Linux LIO Driver Tuning

Parameter values for iSCSI Target Portal Group

```
FirstBurstLength=65536
MaxBurstLength=262144
MaxRecvDatSegmentLength=8192
ImmediateDataYes
InitialR2T=Yes
Default_cmdsn_depth=128
```

CPU	No. of Cores	8 Cores, 16 Threads	8 Cores, 8 Threads	4 Cores, 8 Threads
	CPU Name	Intel® Xeon® processor D 1541	Intel® Atom™ processor C2750	Intel® Xeon® processor E3v3-1265L
	Frequency	2.1 GHz	2.4 GHz	2.5 GHz
MEMORY	Spec	DDR4 2400 MT/s	DDR3 1600 MHz	DDR3 1600 MHz
	Size	32 GB, 2 Memory Channels 2 x 1 GB DIMMs per channel	16 GB, 2 Memory Channels 2 x 8 GB DIMMs per channel	32 GB, 2 Memory Channels 2 x 16 GB DIMMs per channel
STORAGE BACKEND	Drive Configuration	4 TB WD SATA 64 MB cache	3 TB WD SATA 64 MB cache	3 TB WD SATA 64 MB cache
	RPM	7200	7200	7200
	No. of Drives	20 OSDs for storage, 4x SSDs for journaling	10 OSDs for storage, 2x SSDs for journaling	10 OSDs for storage, 2x SSDs for journaling
NETWORK	Bandwidth	20GbE, MTU 9000	10GbE, MTU 9000	10GbE, MTU 9000
OPERATING SYSTEM	Distribution	Ubuntu Server 14.04.2	Ubuntu Server 14.04.2	Ubuntu Server 14.04.2
	Kernel	3.16.0-30-generic kernel	3.16.0-30-generic kernel	3.16.0-30-generic kernel

Ceph Storage Node Configuration

[1] See https://www.openstack.org/summit/vancouver-2015/summit-videos/presentation/Cph-at-cern-a-year-in-the-life-of-a-petabyte-scale-block-storage-service

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more information go to http://www.intel.com/performance

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families.
See http://www.intel.com/performance for details.

Performance tests and ratings are measured using specific computer systems and/or components and reflect the approximate performance of Intel products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance. Buyers should consult other sources of information to evaluate the performance of systems or components they are considering purchasing. For more information on performance tests and on the performance of Intel products, visit www.intel.com/performance/resources/limits.htm or call (U.S.) 1-800-628-8686 or 1-916-356-3104.

No computer system can provide absolute security. Requires an enabled Intel® processor, enabled chipset, firmware and/or software optimized to use the technologies. Consult your system manufacturer and/or software vendor for more information.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can provide absolute security.

All dates and products specified are for planning purposes only and are subject to change without notice.

Relative performance for each benchmark is calculated by taking the actual benchmark result for the first platform tested and assigning it a value of 1.0 as a baseline. Relative performance for the remaining platforms tested was calculated by dividing the actual benchmark result for the baseline platform into each of the specific benchmark results of each of the other platforms and assigning them a relative performance number that correlates with the performance improvements reported.

Information in this document is provided in connection with Intel products. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document. Except as provided in Intel's Terms and Conditions of Sale for such products, Intel assumes no liability whatsoever, and Intel disclaims any express or implied warranty, relating to sale and/or use of Intel products including liability or warranties relating to fitness for a particular purpose, merchantability, or infringement of any patent, copyright or other intellectual property right. Intel products are not intended for use in medical, lifesaving, or life sustaining applications. Intel may make changes to specifications and product descriptions at any time, without notice.