Optimizing CephFS Deployments with High Performance NVMe SSD Technology

[TUT1138]

David Byte
Sr. Technology Strategist, Alliances

Karl Vietmeier
Introductions

David Byte
Karl Vietmeier
Agenda

Project Goal

Hardware Environment

Testing methodology & Tools

Interesting Results

Final Recommendations
Intel, the Intel logo, Intel Optane, Xeon, and others are trademarks of Intel Corporation in the U.S. and/or other countries.

© Intel Corporation.

*Other names and brands may be claimed as the property of others.
SUSE, Intel, & Ceph
Intel’s Ceph contribution timeline

- **2014**
  - Giant
  - New Key/Value Store Backend (rocksdb)
  - Cache-tiering with SSDs (Write support)
  - Virtual Storage Manager (VSM) Open Sourced

- **2015**
  - Hammer
  - Cache-tiering with SSDs (Read support)
  - Erasure Coding support with ISA-L
  - RADOS I/O Hinting (35% better EC Write performance)

- **2016-2018**
  - Infernalis
  - Bluestore w/ SPDK and PMDK backends
  - RADOS I/O Hinting
  - Bluestore Flash Optimizations
  - CeTune Open Sourced

- **2019**
  - Jewel
  - Ceph Client Optimizations (for Hyperconverged, Telco)
  - Luminous and beyond
  - DPDK/SPDK Based Async OSD (improvements for NVMe)
  - Client-side Persistent Cache (read-only, crash-consistent WB, RWL)
  - Bluestore, RGW Compression, Encryption (w/ ISA-L, QAT backends)

* Right Edge of box indicates GA release date
The Project
Goal

Identify tuning options and their impact on performance for CephFS under real world scenarios for an all flash environment

Test w/P4800 devices in several various roles
- RocksDB/WAL on P4800
- RocksDB/WAL on SSDs
- RocksDB/WAL on SSDs + Intel CAS
Hardware Environment

10 OSD nodes
● 1x Intel Gold 6142
● 96 TB RAM
● 12 Intel® SSD DC S4500
● 1 P4800 Optane
● 1 Mellanox ConnectX-4 100Gb

3 MON/MDS/MGR
● 1x Intel Silver 4110
● 32GB RAM
● 1 Mellanox ConnectX-4 100Gb

Loadgen Nodes
● 2x MON type nodes
● 15x HP BL460G6
  ○ Dual Xeon 5560
  ○ 48GB RAM
  ○ Dual 10GbE bonded mode-6
Intel Assistance

- Hardware
- Knowledge and their tuning recommendations
- Open line of communication
A Bit on the Intel Storage Tech Utilized
Intel® Optane™ DC SSDs Operate Differently than NAND

Every NAND SSD
- Read and write in **PAGES**
- ERASE only in **BLOCKS**

Negative performance impact

Intel® Optane™ dc SSD p4800x
- Media is bit addressable - **NO GARBAGE TO COLLECT!**

Improved performance

Permission to use garbage truck icon made by monkik from www.flaticon.com with reference
Intel® Optane™ DC SSDs Overview

Building Blocks

High Endurance

More responsive under load
Improved latency

Predictably fast service
QoS

Breakthrough performance
IOPS

End-User Value

![Graph showing latency comparison between HDD, NAND SSDs (SAS/SATA), NAND SSDs (NVMe*), and Intel® Optane™ DC SSDs (NVMe*)](image)

1 End User Value Source – Intel-tested: Average read latency measured at queue depth 1 during 4k random write workload. Measured using FIO 3.1. Common Configuration - Intel 2U Server System, OS CentOS 7.5, kernel 4.17.6-1.el7.x86_64, CPU 2 x Intel® Xeon® 6154 Gold @ 3.0GHz (18 cores), RAM 256GB DDR4 @ 2666MHz. Configuration – Intel® Optane™ SSD DC P4800X 375GB and Intel® SSD DC P4600 1.6TB. Latency – Average read latency measured at QD1 during 4K Random Write operations using FIO 3.1. Intel Microcode: 0x2000043; System BIOS: 00.01.0013; ME Firmware: 04.00.04.294; BMC Firmware: 1.43.91f76955, FRUSDR: 1.43. SSDs tested were commercially available at time of test. The benchmark results may need to be revised as additional testing is conducted.

Performance results are based on testing as of July 24, 2018 and may not reflect all publicly available security updates. See configuration disclosure for details. No product can be absolutely secure. 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 complete information visit [www.intel.com/benchmarks](http://www.intel.com/benchmarks). * Other names and brands may be claimed as the property of others.
Caching: Intel® Optane™ SSD dc p4800x, the ideal caching solution.

- **lower & more consistent latency**
  - Average Read Latency under Random Write Workload

- **higher endurance**
  - Drive Writes Per Day (DWPD)

- **more efficient**
  - Cache as a % of Storage Capacity

Low latency + high endurance = greater SDS system efficiency

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 complete information visit [www.intel.com/benchmarks](http://www.intel.com/benchmarks).

1. Source – Intel tested: Average read latency measured at queue depth 1 during 4k random write workload. Measured using FIO 3.1. Common Configuration - Intel 2U Server System, OS CentOS 7.5, kernel 4.17.6-1.el7.x86_64, CPU 2 x Intel® Xeon® 6154 Gold @ 3.0GHz (18 cores), RAM 256GB DDR4 @ 2666MHz. Configuration – Intel® Optane™ SSD DC P4800X 375GB and Intel® SSD DC P4600 1.6TB. Latency – Average read latency measured at QD1 during 4k Random Write operations using FIO 3.1. Intel Microcode: 0x2000043; System BIOS: 0.0.0.0.013; ME Firmware: 04.00.04.294; BMC Firmware: 1.43.9175955; FRUSDR: 1.43. SSDs tested were commercially available at time of test. The benchmark results may need to be revised as additional testing is conducted. Performance results are based on testing as of July 24, 2018 and may not reflect all publicly available security updates. See configuration disclosure for details. No product can be absolutely secure.


Optimize IOPS/$ with Intel® Optane™ DC SSDs

All-Flash (SATA) Ceph* Cluster
- CPU: Intel® Xeon® Platinum 8180 Processor
- Capacity/RocksDB*/WAL: Intel® SSD DC S4500 (SATA)

All-Flash (SATA) Ceph* Cluster with Intel® Optane™ SSD accelerating metadata
- CPU: Intel® Xeon® Platinum 8180 Processor
- Metadata/RocksDB*/WAL/Cache: Intel® Optane™ SSD DC P4800X
- Capacity: Intel® SSD DC S4500 (SATA)

*Other names and brands may be claimed as the property of others.
Details, Details, Details
Testing Tools

Cluster build scripts: github.com/dmbyte/SES-Scripts/clusterbuilder

fio

Scripts for benchmarking: github.com/dmbyte/benchmaster
Configuration

SLES12 SP3

Inbox Mellanox driver -> Mellanox driver

BIOS set to performance mode (lock CPUs at max GHz)
The Process

Iterative testing process building each successive step on previous

Start w/ baselines and do basic tuning

Move down to 1x to look for clear latency reducers/IOPS improvements w/specific option tuning

Discuss findings along the way with Intel and SUSE Ceph Devs

Take fresh measurements after tuning for 1x-3x replica

Move to 17 clients
Hiccups

Had an issue where fio was timing out on update messages. Result are some lost test results in early runs.

- Resolved with “--eta=never” on fio command line
- Inadvertently dropped blk-mq from kernel command line in step 11
## Baseline Numbers

<table>
<thead>
<tr>
<th></th>
<th>Write bw</th>
<th>Write iops</th>
<th>Write avg lat</th>
<th>Write max lat</th>
<th>Read bw</th>
<th>Read iops</th>
<th>Read avg lat</th>
<th>Read max lat</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBD 4k mixed</td>
<td>21</td>
<td>5589.37353</td>
<td>3.22968876</td>
<td>409</td>
<td>87</td>
<td>22375.8525</td>
<td>0.62090812</td>
<td>432</td>
</tr>
<tr>
<td>CephFS 4k rand read</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>188</td>
<td>48377.0192</td>
<td>0.32913839</td>
</tr>
<tr>
<td>CephFS 4k seq write</td>
<td>64</td>
<td>16477.395</td>
<td>0.96865061</td>
<td>126</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Latency target of 10ms for which 99.99999% of I/O in a 5s window had to fall inside
Initial Tuning

1. Disable spectre/meltdown remediations
2. Enable blk-mq

Kernel command line additions: spectre_v2=off,pti=off
scsi_mod.use_blk_mq=1

SUSE TID about Spectre/Meltdown on SUSE Enterprise Storage: https://www.suse.com/support/kb/doc/?id=7023480
## Results After Initial Tuning

<table>
<thead>
<tr>
<th></th>
<th>Write bw</th>
<th>Write iops</th>
<th>Write avg lat</th>
<th>Write max lat</th>
<th>Read bw</th>
<th>Read iops</th>
<th>Read avg lat</th>
<th>Read max lat</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBD 4k mixed</td>
<td>21</td>
<td>5382.73215</td>
<td>4.2043349</td>
<td>295</td>
<td>84</td>
<td>21561.5695</td>
<td>0.43237217</td>
<td>281</td>
</tr>
<tr>
<td>CephFS 4k Mixed</td>
<td>48</td>
<td>12321.0973</td>
<td>1.40914326</td>
<td>119</td>
<td>192</td>
<td>49378.3513</td>
<td>0.29386291</td>
<td>19</td>
</tr>
<tr>
<td>CephFS 4k Rand Read</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>380</td>
<td>97285.0791</td>
<td>0.32751933</td>
<td>14</td>
</tr>
<tr>
<td>CephFS 4k Rand Write</td>
<td>141</td>
<td>36257.5327</td>
<td>0.88203342</td>
<td>125</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CephFS 4k Seq Read</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>385</td>
<td>98566.8125</td>
<td>0.32815892</td>
<td>16</td>
</tr>
<tr>
<td>CephFS 4k Seq Write</td>
<td>128</td>
<td>32975.3585</td>
<td>0.96844091</td>
<td>125</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Highlights:**
- CephFS 4k Random Read: 188 -> 380
- CephFS 4k Sequential Write: 64 -> 128

Average latencies did not change appreciably.
# Drop Latency Targets for fio

## Highlights:

Every category rose while maintaining reasonable average latencies

CephFS 4k mixed w=21(4.2) -> 203(5.4) & r=84 (.43ms) -> 816(3.53)

95 percentile average latency between 2 test nodes 9.8ms

CephFS 4k Rand Read 380(.32) ->1040(3.8)

<table>
<thead>
<tr>
<th></th>
<th>Write bw</th>
<th>Write iops</th>
<th>Write avg lat</th>
<th>Write max lat</th>
<th>Read bw</th>
<th>Read iops</th>
<th>Read avg lat</th>
<th>Read max lat</th>
</tr>
</thead>
<tbody>
<tr>
<td>CephFS 4k mixed</td>
<td>203</td>
<td>52214</td>
<td>5.44046165</td>
<td>136</td>
<td>816</td>
<td>208988</td>
<td>3.537089307</td>
<td>124</td>
</tr>
<tr>
<td>CephFS 4k Rand Read</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1040</td>
<td>266274</td>
<td>3.844578346</td>
</tr>
<tr>
<td>CephFS 4k Rand Write</td>
<td>721</td>
<td>184741</td>
<td>5.532163551</td>
<td>155</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CephFS 4k Seq Read</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1127</td>
<td>288628</td>
<td>3.548652131</td>
</tr>
<tr>
<td>CephFS 4k Seq Write</td>
<td>745</td>
<td>190805</td>
<td>5.421591009</td>
<td>178</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Continued Testing

After initial tuning, none of the changes made significant, across the board impact

Hypothesis: 2 test nodes were maxed out. Changed NIC driver and experienced increased throughput per node of 44Gb/s vs 30Gb/s tested with iperf3
NIC Tuning

#!/bin/bash
salt '*' cmd.run 'setpci -s 5b:00.0 68.w=5936'
salt '*' cmd.run 'setpci -s 5b:00.1 68.w=5936'
salt '*' cmd.run 'ip link set bond0 mtu 9000'
salt '*' cmd.run 'for j in `cat /sys/class/net/bond0/bonding/slaves`;do LOCAL_CPUS=`cat /sys/class/net/$j/device/local_cpus`;echo $LOCAL_CPUS > /sys/class/net/$j/queues/rx-0/rps_cpus;done'
salt '*' cmd.run 'ethtool -G eth4 rx 8192 tx 8192'
salt '*' cmd.run 'ethtool -G eth5 rx 8192 tx 8192'
salt '*' cmd.run 'sysctl -w net.ipv4.tcp_sack=0'
salt '*' cmd.run 'sysctl -w net.core.rmem_max=2147483647'
salt '*' cmd.run 'sysctl -w net.core.wmem_max=2147483647'
salt '*' cmd.run 'sysctl -w net.core.somaxconn=2048'
salt '*' cmd.run 'sysctl -w net.ipv4.tcp_low_latency=1'
salt '*' cmd.run 'sysctl -w net.ipv4.tcp_fastopen=1'
salt '*' cmd.run 'sysctl -w net.ipv4.tcp_rmem="10240 87380 2147483647"'
salt '*' cmd.run 'sysctl -w net.ipv4.tcp_wmem="10240 87380 2147483647"'
salt '*' cmd.run 'sysctl -w net.ipv4.tcp_timestamps=0'
salt '*' cmd.run 'sysctl -w net.core.netdev_max_backlog=250000'
salt '*' cmd.run 'mlnx_tune -p HIGH_THROUGHPUT'
Network Tuning Results

<table>
<thead>
<tr>
<th></th>
<th>Write bw</th>
<th>Write iops</th>
<th>Write avg lat</th>
<th>Write max lat</th>
<th>Read bw</th>
<th>Read iops</th>
<th>Read avg lat</th>
<th>Read max lat</th>
</tr>
</thead>
<tbody>
<tr>
<td>CephFS 4k mixed</td>
<td>225</td>
<td>57735.56892</td>
<td>5.177062304</td>
<td>135</td>
<td>902</td>
<td>231152.2841</td>
<td>3.028517366</td>
<td>120</td>
</tr>
<tr>
<td>CephFS 4k Rand Read</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1151</td>
<td>294677.3218</td>
<td>3.4019127</td>
</tr>
<tr>
<td>CephFS 4k Rand Write</td>
<td>686</td>
<td>175643.7346</td>
<td>5.828067818</td>
<td>148</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CephFS 4k Seq Read</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1258</td>
<td>322236.5802</td>
<td>3.117764669</td>
<td>94</td>
</tr>
<tr>
<td>CephFS 4k Seq Write</td>
<td>754</td>
<td>193223.2902</td>
<td>5.308887288</td>
<td>151</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Negligible benefits
Surmised that we are fighting not enough load generators at this point. Increase from 2x @ 100GB to 2x@100GB
## 17 Loadgen Node Results

<table>
<thead>
<tr>
<th></th>
<th>Write bw</th>
<th>Write iops</th>
<th>Write avg lat</th>
<th>Write max lat</th>
<th>Read bw</th>
<th>Read iops</th>
<th>Read avg lat</th>
<th>Read max lat</th>
</tr>
</thead>
<tbody>
<tr>
<td>CephFS 4k mixed</td>
<td>294</td>
<td>75350.9777</td>
<td>84.9389699</td>
<td>12385</td>
<td>1179</td>
<td>301952.9174</td>
<td>7.597368489</td>
<td>11802</td>
</tr>
<tr>
<td>CephFS 4k Rand Read</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4158</td>
<td>1064590.205</td>
<td>7.831963847</td>
<td>6992</td>
</tr>
<tr>
<td>CephFS 4k Rand Write</td>
<td>422</td>
<td>108254.8212</td>
<td>80.38623681</td>
<td>2595</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CephFS 4k Seq Read</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4630</td>
<td>1185434.883</td>
<td>7.332546306</td>
<td>13459</td>
</tr>
<tr>
<td>CephFS 4k Seq Write</td>
<td>380</td>
<td>97363.16688</td>
<td>89.525632</td>
<td>2678</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Significant Increase for Reads *** Over 1 Million IOPS *****
- 4k Random Read (1151MiB/s -> 4158 MiB/s)
- 4k Sequential Read (1258->4630)

**Significant Decrease for Writes due to saturation**
- 4k Random Write (688 MiB/s -> 422 MiB/s)
- Average Latency went from 5ms -> 80ms
Change up the config

No RocksDB or WAL on P4800

SSD Only
Random Read is the only case where a significant gain wasn’t seen
Many of the other 4k tests saw gains approaching 2x. Probable cause is a write amplification and chokepoints in OSD code
## 64k Performance

<table>
<thead>
<tr>
<th></th>
<th>Write bw</th>
<th>Write iops</th>
<th>Write avg lat</th>
<th>Write max lat</th>
<th>Read bw</th>
<th>Read iops</th>
<th>Read avg lat</th>
<th>Read max lat</th>
</tr>
</thead>
<tbody>
<tr>
<td>CephFS 64k mixed</td>
<td>471</td>
<td>7547.891356</td>
<td>282.3424996</td>
<td>51979</td>
<td>1896</td>
<td>30336.53719</td>
<td>311.228771</td>
<td>52381</td>
</tr>
<tr>
<td>CephFS 64k Rand Read</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1631</td>
<td>26106.9962</td>
<td>357.9005411</td>
</tr>
<tr>
<td>CephFS 64k Rand Write</td>
<td>7730</td>
<td>123689.4811</td>
<td>70.32053268</td>
<td>7823</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CephFS 64k Seq Read</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1973</td>
<td>31566.55734</td>
<td>275.5793086</td>
</tr>
<tr>
<td>CephFS 64k Seq Write</td>
<td>7562</td>
<td>120982.1464</td>
<td>71.90007369</td>
<td>4422</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
# 1MB Performance

<table>
<thead>
<tr>
<th></th>
<th>Write bw</th>
<th>Write iops</th>
<th>Write avg lat</th>
<th>Write max lat</th>
<th>Read bw</th>
<th>Read iops</th>
<th>Read avg lat</th>
<th>Read max lat</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CephFS 1MB</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq Read</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3994</td>
<td>3990.463307</td>
<td>2155.816468</td>
<td>212036</td>
</tr>
<tr>
<td><strong>CephFS 1MB</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq Write</td>
<td>12390</td>
<td>12386.01587</td>
<td>668.8836366</td>
<td>107648</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Write bw</td>
<td>Write iops</td>
<td>Write avg lat</td>
<td>Write max lat</td>
<td>Read bw</td>
<td>Read iops</td>
<td>Read avg lat</td>
<td>Read max lat</td>
</tr>
<tr>
<td>------------------</td>
<td>----------</td>
<td>-----------------</td>
<td>----------------</td>
<td>---------------</td>
<td>----------</td>
<td>----------------</td>
<td>----------------</td>
<td>---------------</td>
</tr>
<tr>
<td>CephFS 4k</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mixed</td>
<td>342</td>
<td>87701.89278</td>
<td>52.73225511</td>
<td>1505</td>
<td>1372</td>
<td>351381.4842</td>
<td>11.51741075</td>
<td>1470</td>
</tr>
<tr>
<td>Rand Read</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2459</td>
<td>629722.5313</td>
<td>13.65634488</td>
</tr>
<tr>
<td>Rand Write</td>
<td>472</td>
<td>121017.0002</td>
<td>71.9806716</td>
<td>393</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Seq Read</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3210</td>
<td>821908.4713</td>
<td>10.51756716</td>
</tr>
<tr>
<td>Seq Write</td>
<td>406</td>
<td>104173.5814</td>
<td>84.38886591</td>
<td>1134</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>Write bw</td>
<td>Write iops</td>
<td>Write avg lat</td>
<td>Read bw</td>
<td>Read iops</td>
<td>Read avg lat</td>
<td>Read IOPS/Dev</td>
<td>Read MB/dev</td>
</tr>
<tr>
<td>----------------</td>
<td>----------</td>
<td>------------</td>
<td>---------------</td>
<td>---------</td>
<td>-----------</td>
<td>--------------</td>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Rep-cephfs-4k-mixed Offload</td>
<td>294</td>
<td>75350</td>
<td>84</td>
<td>1179</td>
<td>301952</td>
<td>7</td>
<td>2516</td>
<td>9</td>
</tr>
<tr>
<td>Rep-cephfs-4k-mixed SSD</td>
<td>566</td>
<td>144933</td>
<td>35</td>
<td>2267</td>
<td>580466</td>
<td>6</td>
<td>4837</td>
<td>18</td>
</tr>
<tr>
<td>Rep-cephfs-4k-mixed P4800</td>
<td>342</td>
<td>87701</td>
<td>52</td>
<td>1372</td>
<td>351381</td>
<td>11</td>
<td>35138</td>
<td>137</td>
</tr>
<tr>
<td>Rep-cephfs-4k-randread Offload</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4158</td>
<td>1064590</td>
<td>7</td>
<td>8871</td>
<td>34</td>
</tr>
<tr>
<td>Rep-cephfs-4k-randread SSD</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3585</td>
<td>917982</td>
<td>9</td>
<td>7649</td>
<td>29</td>
</tr>
<tr>
<td>Rep-cephfs-4k-randread P4800</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2459</td>
<td>629722</td>
<td>13</td>
<td>62972</td>
<td>245</td>
</tr>
<tr>
<td>Rep-cephfs-4k-randwrite Offload</td>
<td>422</td>
<td>108254</td>
<td>80</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Rep-cephfs-4k-randwrite SSD</td>
<td>852</td>
<td>218111</td>
<td>39</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Rep-cephfs-4k-randwrite P4800</td>
<td>472</td>
<td>121017</td>
<td>71</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Rep-cephfs-4k-seqread Offload</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4630</td>
<td>1185434</td>
<td>7</td>
<td>9878</td>
<td>38</td>
</tr>
<tr>
<td>Rep-cephfs-4k-seqread SSD</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4999</td>
<td>1279983</td>
<td>6</td>
<td>10666</td>
<td>41</td>
</tr>
<tr>
<td>Rep-cephfs-4k-seqread P4800</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3210</td>
<td>821908</td>
<td>10</td>
<td>82190</td>
<td>321</td>
</tr>
<tr>
<td>Rep-cephfs-4k-seqwrite Offload</td>
<td>380</td>
<td>97363</td>
<td>89</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Rep-cephfs-4k-seqwrite SSD</td>
<td>981</td>
<td>251206</td>
<td>34</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Rep-cephfs-4k-seqwrite P4800</td>
<td>406</td>
<td>104173</td>
<td>84</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
What about 2x Replica?

Depends on your tolerance for failure and data loss

If you absolutely want to do it, I’ve done a little performance work.
<table>
<thead>
<tr>
<th>testname</th>
<th>writebw</th>
<th>writeiops</th>
<th>writeavglat</th>
<th>readbw</th>
<th>readiops</th>
<th>readavglat</th>
</tr>
</thead>
<tbody>
<tr>
<td>rep-cephfs-1M-seqread</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3994</td>
<td>3990</td>
<td>2155.816468</td>
</tr>
<tr>
<td>rep-cephfs-1M-seqread</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4253</td>
<td>4248</td>
<td>1994.763213</td>
</tr>
<tr>
<td>rep-cephfs-1M-seqwrite</td>
<td>12390</td>
<td>12386</td>
<td>668.8836366</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>rep-cephfs-1M-seqwrite</td>
<td>15927</td>
<td>15922</td>
<td>539.6763443</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>rep-cephfs-4k-mixed</td>
<td>566</td>
<td>144933</td>
<td>35.45755848</td>
<td>2267</td>
<td>580466</td>
<td>6.141105364</td>
</tr>
<tr>
<td>rep-cephfs-4k-mixed</td>
<td>780</td>
<td>199930</td>
<td>22.79429772</td>
<td>3129</td>
<td>801039</td>
<td>5.17158598</td>
</tr>
<tr>
<td>rep-cephfs-4k-randread</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3585</td>
<td>917982</td>
<td>9.254214539</td>
</tr>
<tr>
<td>rep-cephfs-4k-randread</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2971</td>
<td>760735</td>
<td>11.44746839</td>
</tr>
<tr>
<td>rep-cephfs-4k-randwrite</td>
<td>852</td>
<td>218111</td>
<td>39.89544242</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>rep-cephfs-4k-randwrite</td>
<td>1344</td>
<td>344230</td>
<td>25.28227172</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>rep-cephfs-4k-seqread</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4999</td>
<td>1279983</td>
<td>6.922590204</td>
</tr>
<tr>
<td>rep-cephfs-4k-seqread</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4935</td>
<td>1263406</td>
<td>6.869140772</td>
</tr>
<tr>
<td>rep-cephfs-4k-seqwrite</td>
<td>981</td>
<td>251206</td>
<td>34.60790575</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>rep-cephfs-4k-seqwrite</td>
<td>1517</td>
<td>388572</td>
<td>22.40924006</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Pink=3x
Purple=2x
Thoughts on Architecting Clusters

With QLC Intel Drives becoming available, the $/GB is approaching that of spinning media.

With many servers supporting U.2 and SATA, it makes sense to have a cluster with a mix of SSD and Optane where a small, higher performing pool is needed.

For now, spinning disk can still make up the most dense storage tier, but perhaps, not for much longer.

Ceph continues to improve OSD performance.
Future Goal

Do some testing with P4800 as RocksDB/WAL and iCAS for SSD and spinners with RBD as well.

Capture benchmark information related to workloads such as rendering, virtual machines, etc

Repeat some testing on SES 6