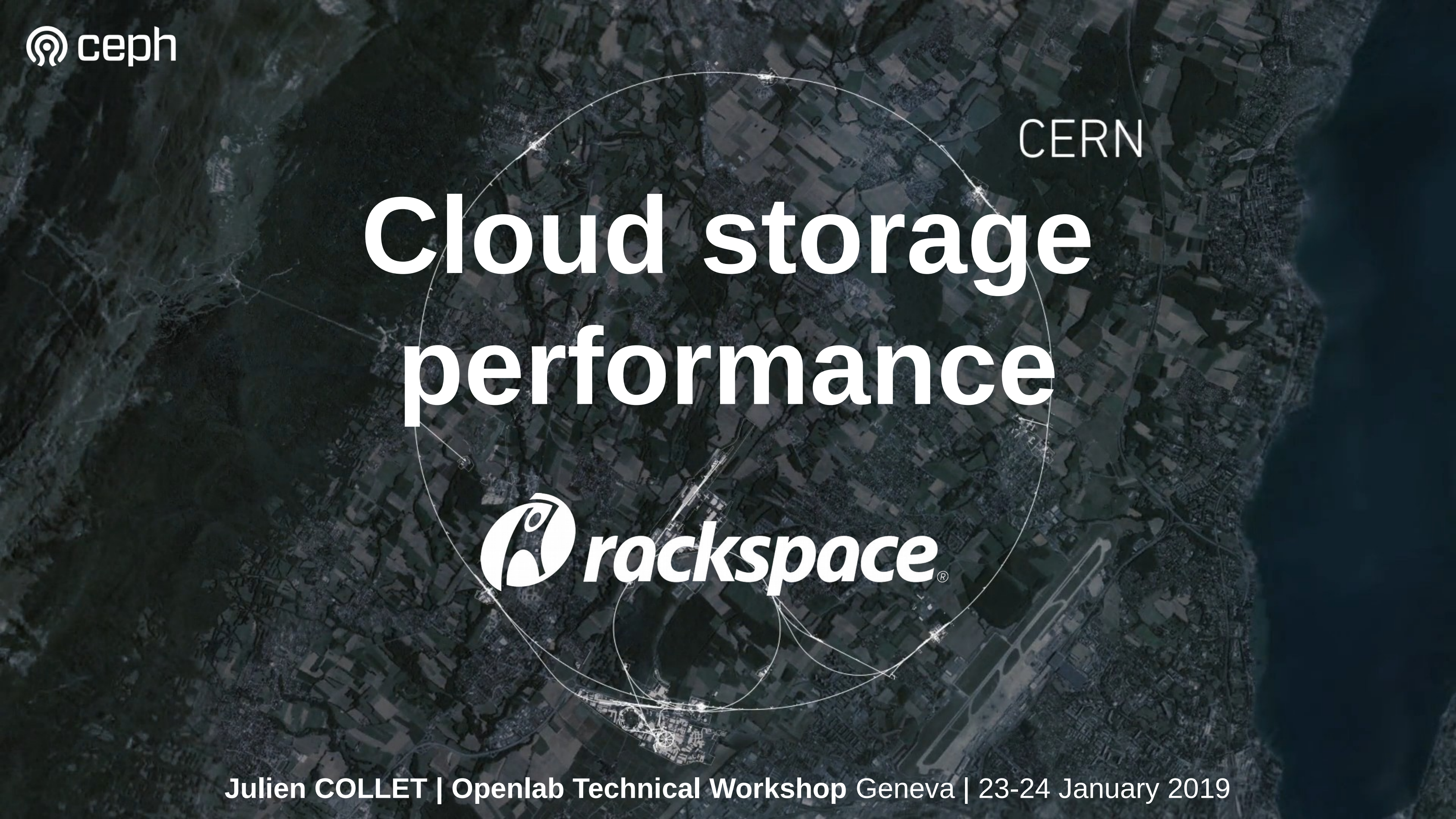
CERN



ceph

CERN

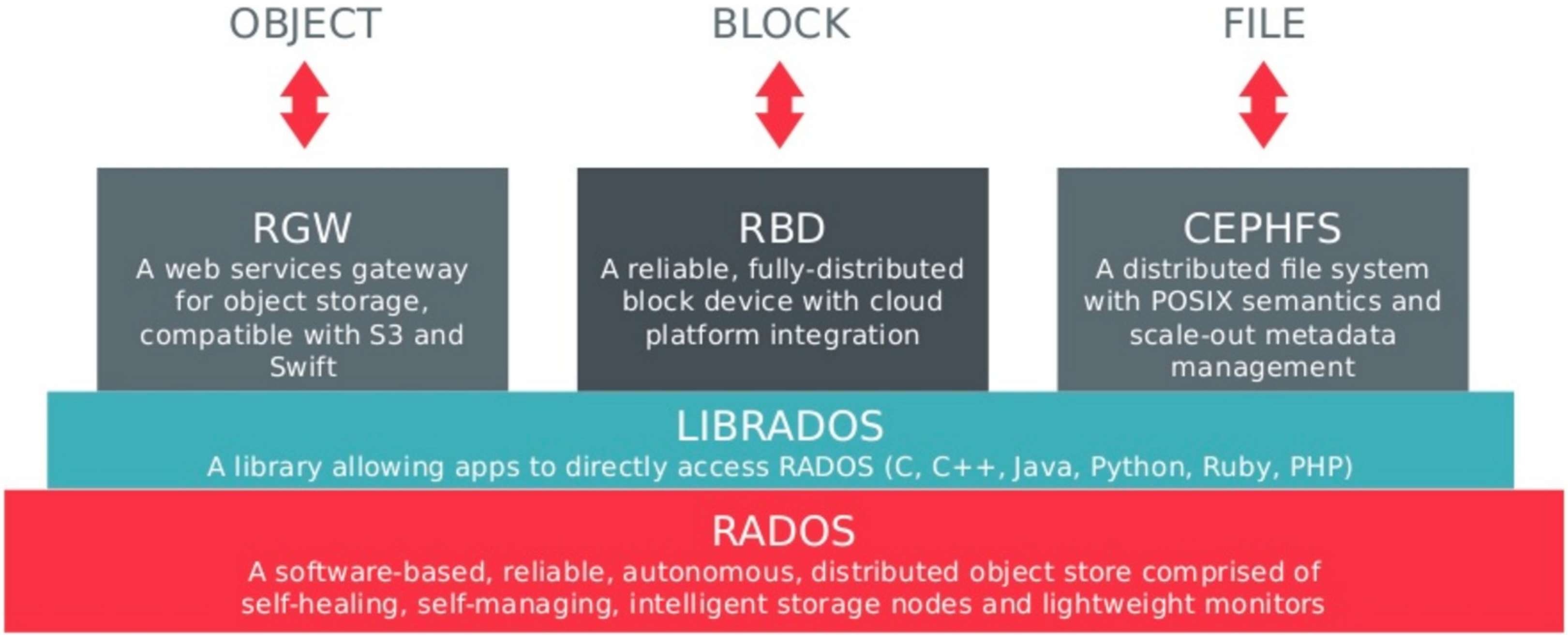
# Cloud storage performance

rackspace®

**Julien COLLET | Openlab Technical Workshop** Geneva | 23-24 January 2019

# A bit of context...

# Ceph open-source storage system



OBJECT

BLOCK

FILE

**RGW**
A web services gateway for object storage, compatible with S3 and Swift

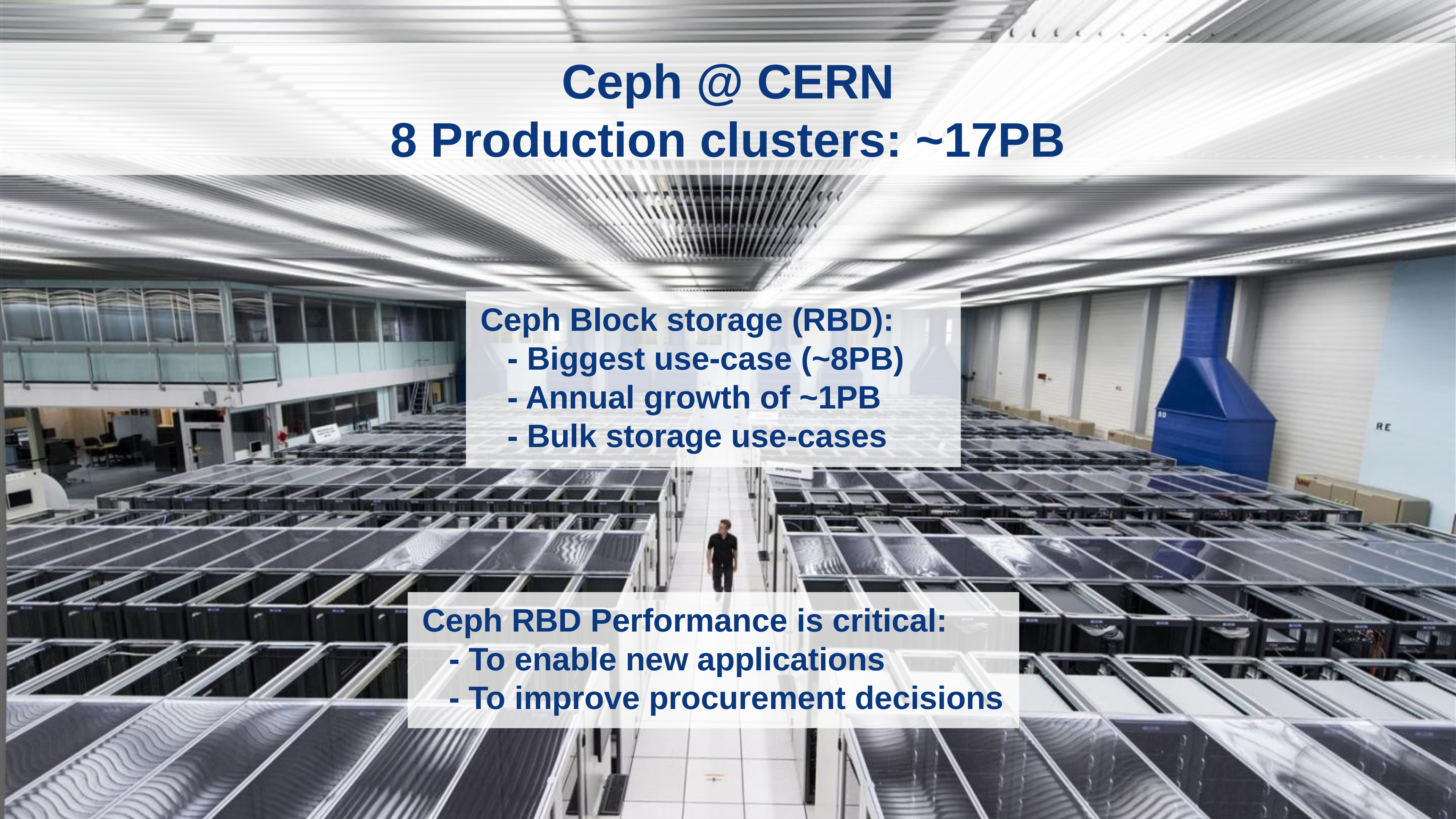
**RBD**
A reliable, fully-distributed block device with cloud platform integration

**CEPHFS**
A distributed file system with POSIX semantics and scale-out metadata management

**LIBRADOS**
A library allowing apps to directly access RADOS (C, C++, Java, Python, Ruby, PHP)

**RADOS**
A software-based, reliable, autonomous, distributed object store comprised of self-healing, self-managing, intelligent storage nodes and lightweight monitors

# Ceph @ CERN
# 8 Production clusters: ~17PB



Ceph Block storage (RBD):

- Biggest use-case (~8PB)
- Annual growth of ~1PB
- Bulk storage use-cases

Ceph RBD Performance is critical:

- To enable new applications
- To improve procurement decisions

# Project overview

- Development of a benchmarking suite to understand performance variations between layers of Ceph
- Evaluation of a new all-flash hyperconverged cluster to tackle new CERN use-cases on top of Ceph
- Implementation of a tool to help operators understand workload behavior
- Contributions to the Ceph project in collaboration with upstream team

# Benchmarking ceph/rbd

# Benchmarking a ceph cluster

*Overview:*

- Raw disk baseline performance: dd/fio
- Ceph storage level performance: rados bench
- Block device performance: fio (librbd) and rbd bench

- Metrics: IOPS, Disk IO and CPU utilizations, latency

- Single-node cluster to avoid network latency impacts on performance

# Benchmarking a ceph cluster

## *RADOS-level performance*

Starting point: raw fio results give 85.1 kIOPS (SSD) and 232 IOPS (HDD), what is RADOS performance ? (4k random sync writes)

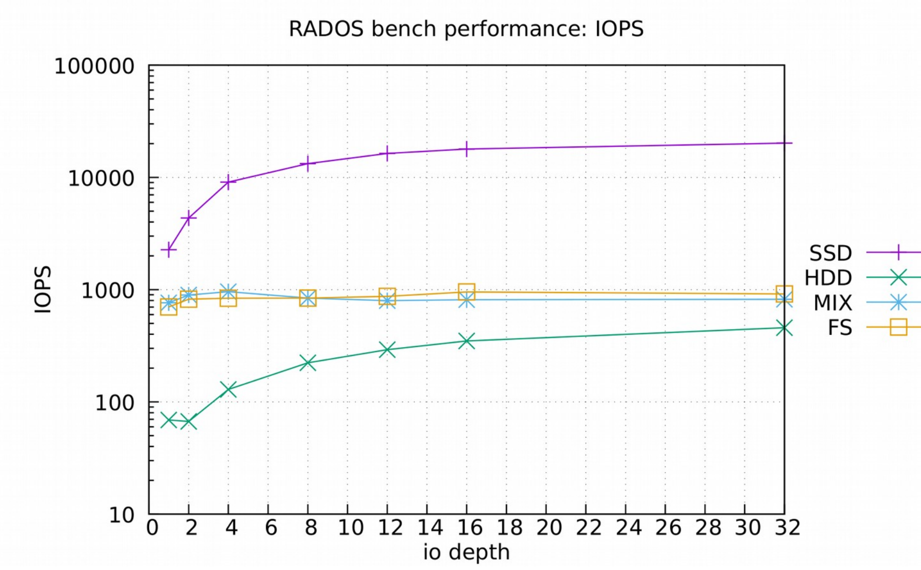
*SSD: bluestore ssd*

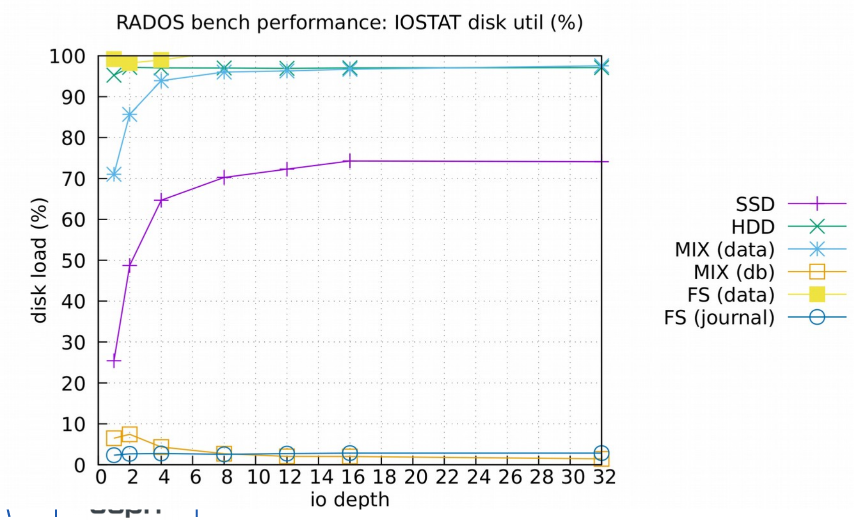
*HDD: bluestore ssd*

*MIX: bluestore data:hdd db:ssd*

*FS: filestore data:hdd journal:ssd*

Mixed configurations: SSD not-stressed, HDD remaining the bottleneck with a 100% I/O util

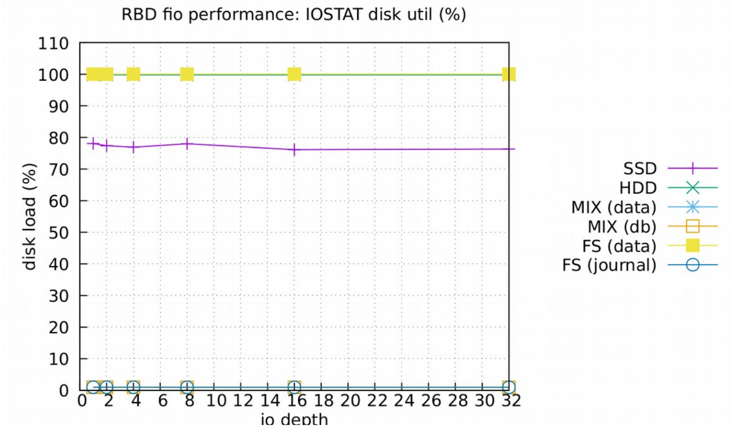


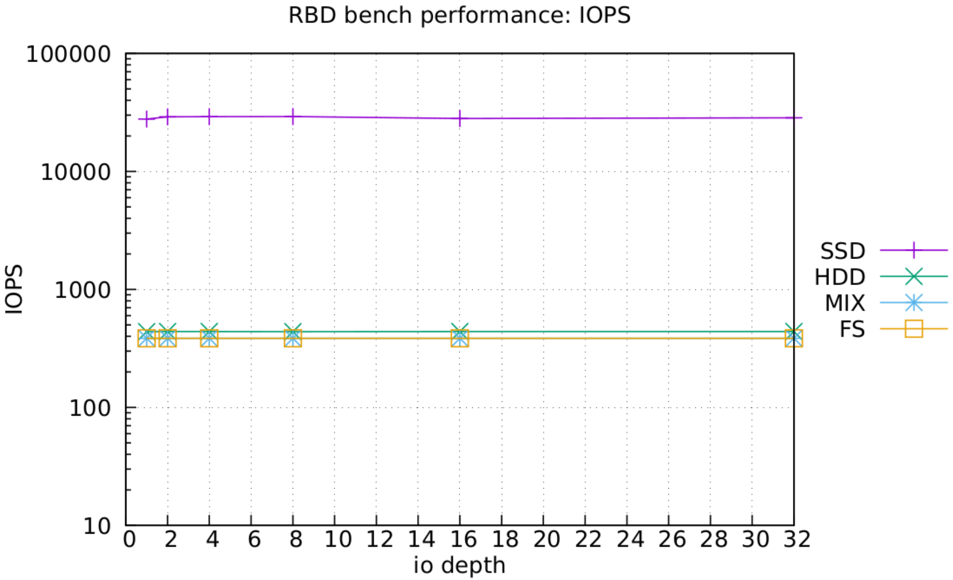


# Benchmarking a ceph cluster

## *RBD-level performance*

Starting point: raw fio results give 85.1 kIOPS (SSD) and 232 IOPS (HDD), what is RBD performance ?   (4k random sync writes)





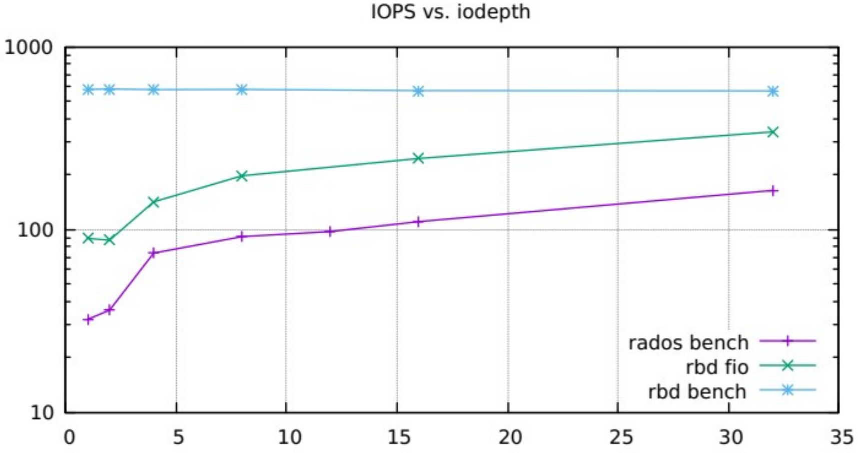
- Mixed configurations roughly equivalent, unable to stress the SSD…
- As SSDs are never I/O bound, recommended to use partitioning (multiple OSDs per SSDs)

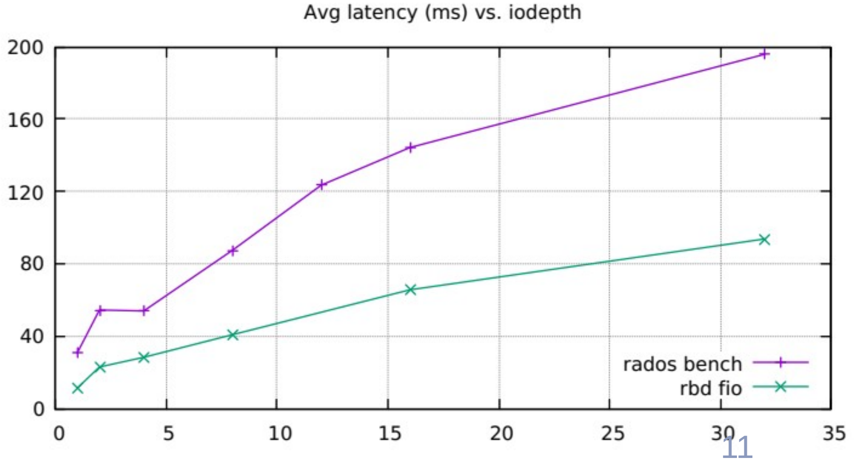
# Client-side caching

*Leveraging dm-cache*

- dm-cache enables linux kernel’s device-mapper to use faster devices (e.g. flash) to act as a cache for HDDs

- Slightly better performance at the RBD bench level compared to standard client configuration, but not a silver bullet either: *dm-writecache?*

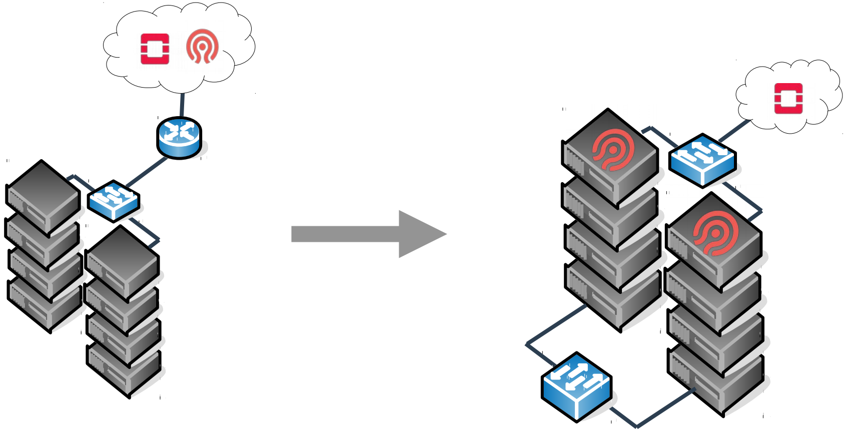




# Hyperconverged clusters

# Hyperconverged OpenStack+RBD

- Cluster: 22 nodes
  - 16-core Xeon (SMT disabled)
  - 128GB of memory
  - 16x 960GB SSDs



- Configuration
  - Memory: 64GB for the VMs, 32GB for Ceph, rest for overheads
  - 14 OSDs per node

Plan

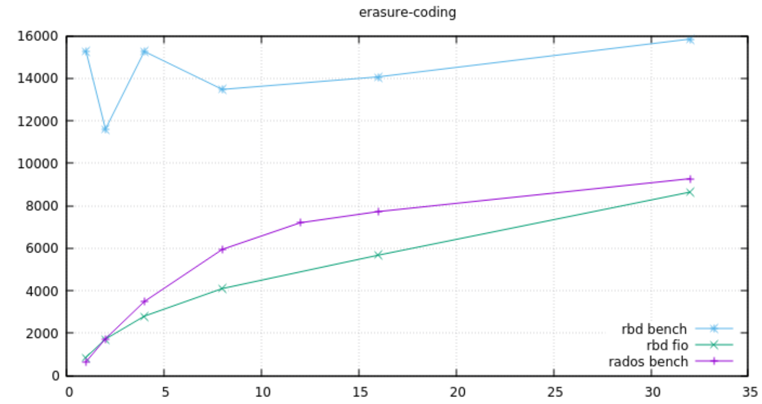
- Build it
- Internal perf tests
- Disaster tests
- Develop ops procedures
- Invite early adopters

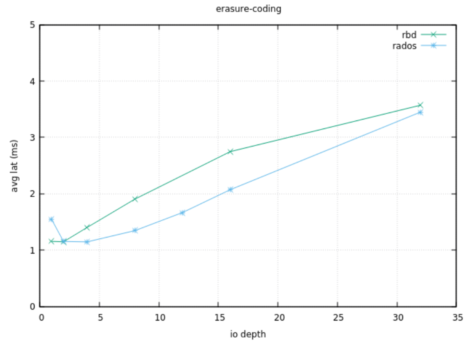
CERN | ceph | rackspace.

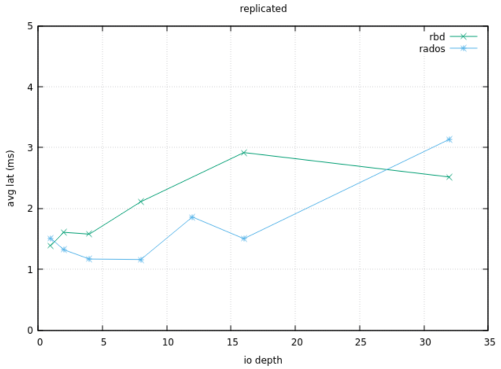
# Hyperconverged OpenStack+RBD

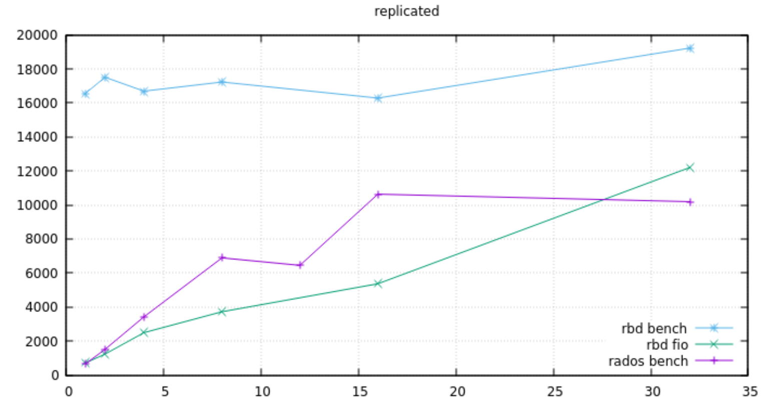
## *OSD performance*

- Benchmarking an all-flash test node to bring Ceph/rbd to other CERN use-cases
- High IOPS + Low latency
- Outperforms by far client-side caching alternatives



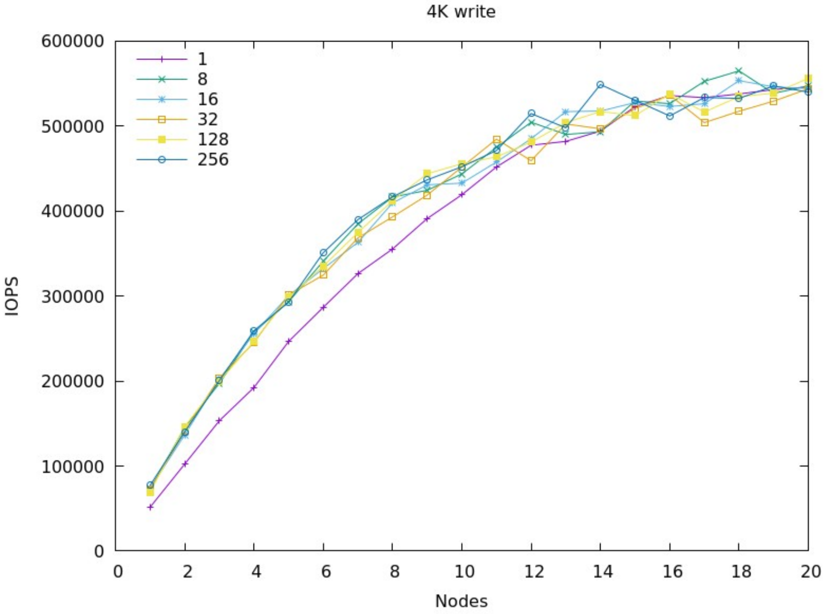






# Hyperconverged OpenStack+RBD

*Cluster performance: 20-node all-flash system*



- Up to 500k IOPS on best-case scenario (sequencial 4K read, replicated 3x)
- Peak bandwidth up to 10GBs w/1MB objects
- Erasure-coding for free? Decent performance on ec-enabled pools (4+2)
- In progress: run application-specific benchmarks to validate its use for new use-cases: databases...

# Assessing omap performance

*Hyperconverged vs. Mixed configuration*

- omap performance is critical to S3 bucket indexes

- Set-up:

  Dedicated rados pool

  1x object

  Script executed from the same network switch

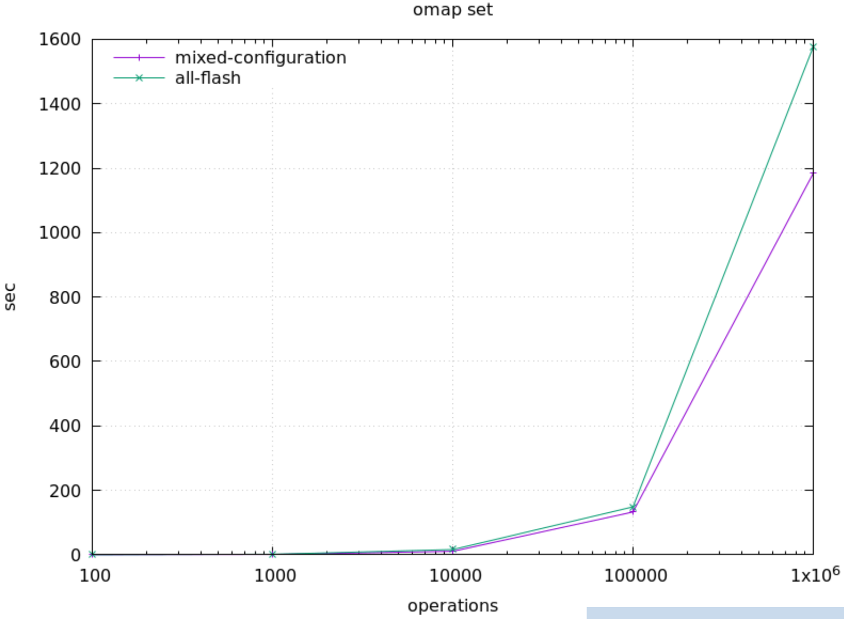
  key-value pairs: average size of the key-value of our real workload

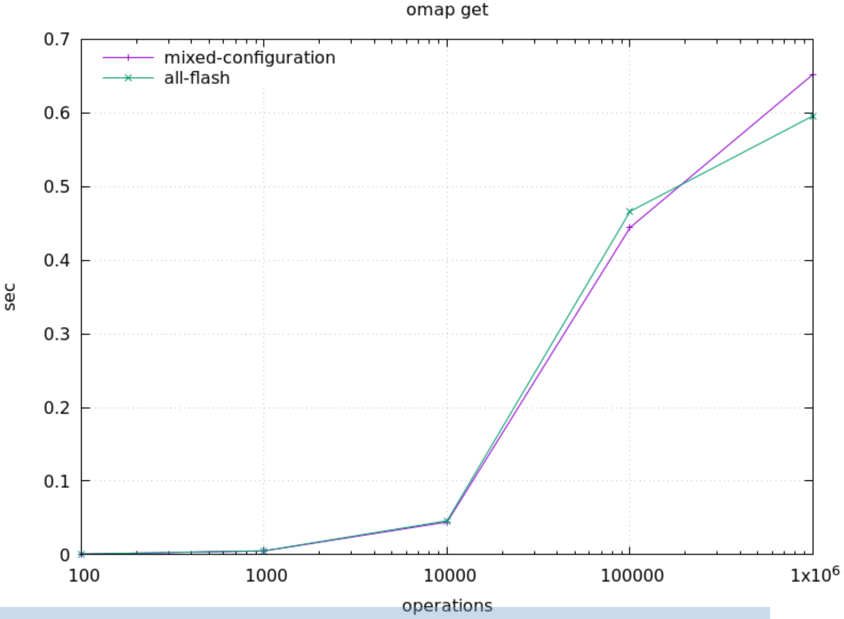
  All-flash vs mixed-configuration setup

**Are all-flash clusters worth it ?**

# Assessing omap performance

*Hyperconverged vs. Mixed configuration*





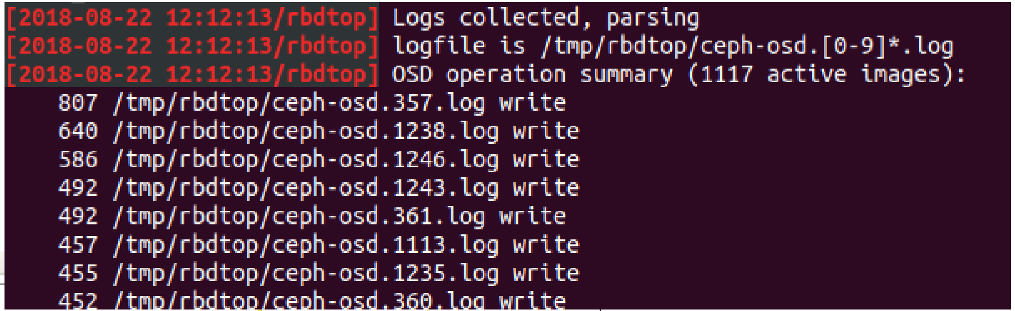
**HDD+SSD shows performance of same order as SSDs for this S3 use-case**

# Monitoring ceph/rbd performance

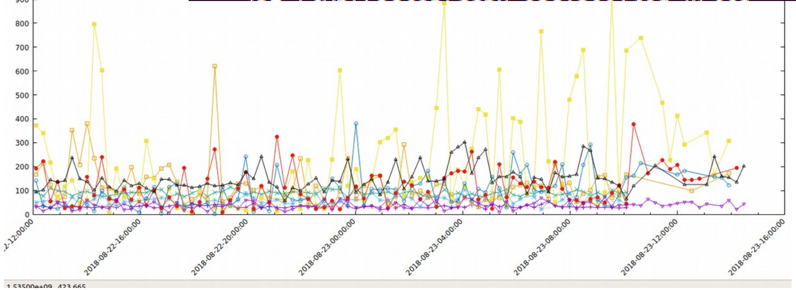
# Ceph/rbd top v1

*Identifying busiest OSDs/RBD Images*

- List all clients on the host
- Activate logging for the clients for a given time
- Extract read/write
- Sort/Filter by most active
- Generate a report/plot



```
[2018-08-22 12:12:13/rbdtop] Logs collected, parsing
[2018-08-22 12:12:13/rbdtop] logfile is /tmp/rbdtop/ceph-osd.[0-9]*.log
[2018-08-22 12:12:13/rbdtop] OSD operation summary (1117 active images):
    807 /tmp/rbdtop/ceph-osd.357.log write
    640 /tmp/rbdtop/ceph-osd.1238.log write
    586 /tmp/rbdtop/ceph-osd.1246.log write
    492 /tmp/rbdtop/ceph-osd.1243.log write
    492 /tmp/rbdtop/ceph-osd.361.log write
    457 /tmp/rbdtop/ceph-osd.1113.log write
    455 /tmp/rbdtop/ceph-osd.1235.log write
    452 /tmp/rbdtop/ceph-osd.360.log write
```



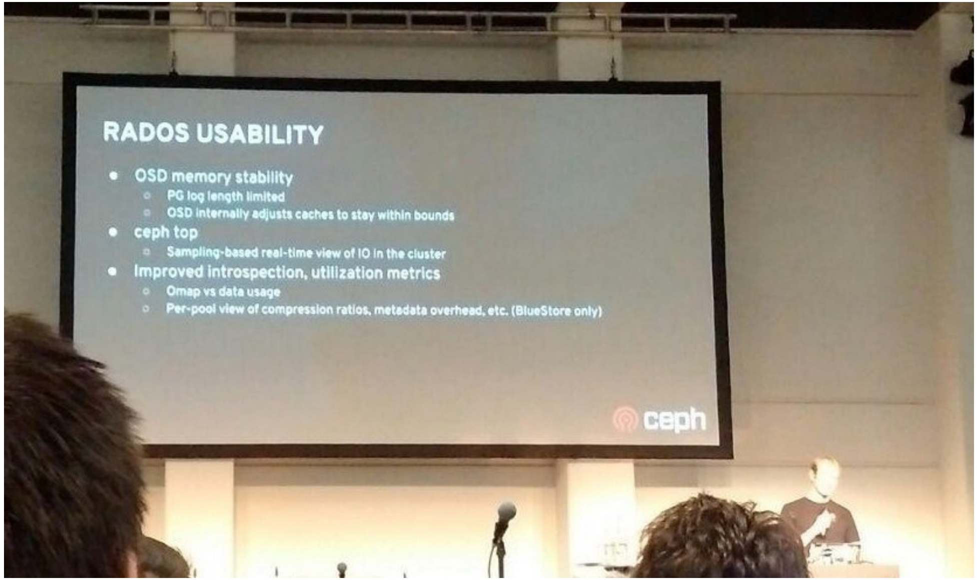
# Ceph/rbd top v2

*Integrating top as a Ceph feature (work-in-progress)*

- Feature announced at Ceph Day Berlin on 13.11.2018

- Will help operators to identify “hot” clients/images

- Still work-in-progress



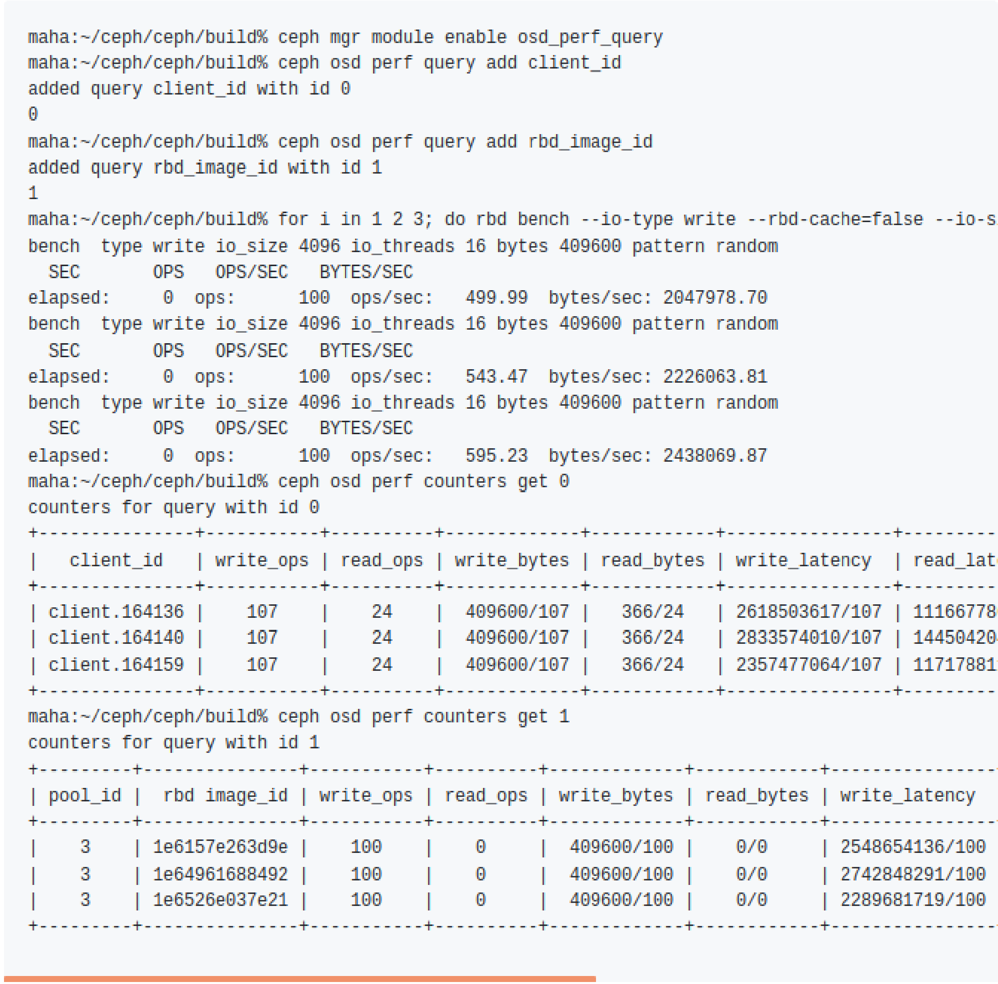
RADOS USABILITY

- OSD memory stability
  - PG log length limited
  - OSD internally adjusts caches to stay within bounds
- ceph top
  - Sampling-based real-time view of IO in the cluster
- Improved introspection, utilization metrics
  - Omap vs data usage
  - Per-pool view of compression ratios, metadata overhead, etc. (BlueStore only)

# Ceph/rbd top v2

*Integrating top as a Ceph feature*

- Mgr module issuing requests to OSDs to collect perf metrics
- Python interface to add/remove requests and get query results
- Group by object prefix (rbd image name)



```
maha:~/ceph/ceph/build% ceph mgr module enable osd_perf_query
maha:~/ceph/ceph/build% ceph osd perf query add client_id
added query client_id with id 0
0
maha:~/ceph/ceph/build% ceph osd perf query add rbd_image_id
added query rbd_image_id with id 1
1
maha:~/ceph/ceph/build% for i in 1 2 3; do rbd bench --io-type write --rbd-cache=false --io-s
bench  type write io_size 4096 io_threads 16 bytes 409600 pattern random
  SEC       OPS   OPS/SEC   BYTES/SEC
elapsed:     0  ops:      100  ops/sec:   499.99  bytes/sec: 2047978.70
bench  type write io_size 4096 io_threads 16 bytes 409600 pattern random
  SEC       OPS   OPS/SEC   BYTES/SEC
elapsed:     0  ops:      100  ops/sec:   543.47  bytes/sec: 2226063.81
bench  type write io_size 4096 io_threads 16 bytes 409600 pattern random
  SEC       OPS   OPS/SEC   BYTES/SEC
elapsed:     0  ops:      100  ops/sec:   595.23  bytes/sec: 2438069.87
maha:~/ceph/ceph/build% ceph osd perf counters get 0
counters for query with id 0
+----------------+-----------+----------+-------------+------------+----------------+--------
|   client_id    | write_ops | read_ops | write_bytes | read_bytes | write_latency  | read_lat
+----------------+-----------+----------+-------------+------------+----------------+--------
| client.164136 |    107    |    24    |  409600/107 |   366/24   | 2618503617/107 | 11166778
| client.164140 |    107    |    24    |  409600/107 |   366/24   | 2833574010/107 | 14450420
| client.164159 |    107    |    24    |  409600/107 |   366/24   | 2357477064/107 | 11717881
+----------------+-----------+----------+-------------+------------+----------------+--------
maha:~/ceph/ceph/build% ceph osd perf counters get 1
counters for query with id 1
+---------+--------------+-----------+----------+-------------+------------+-------------
| pool_id |  rbd image_id | write_ops | read_ops | write_bytes | read_bytes | write_latency
+---------+--------------+-----------+----------+-------------+------------+-------------
|    3    | 1e6157e263d9e |    100    |    0     |  409600/100 |    0/0     | 2548654136/100
|    3    | 1e64961688492 |    100    |    0     |  409600/100 |    0/0     | 2742848291/100
|    3    | 1e6526e037e21 |    100    |    0     |  409600/100 |    0/0     | 2289681719/100
+---------+--------------+-----------+----------+-------------+------------+-------------
```

# Next steps...

# Conclusions

- Hypervisor-side writeback cache is complex, still far in the future
- Small amounts of flash can greatly improve performance of hdd-only ceph clusters (rbd, omap, ...)
- All-flash, hyperconverged clusters are the best solution for IOPS critical applications
- Some developments (“top”) can enable operators to identify bottlenecks and tune the storage systems

# Future work

*Benchmarking and performance evaluation*

- Continuous improvement of the benchmarking suite
- Validation of upcoming all-flash architecture using real-life use-cases (database applications, low-latency analysis...)

*Monitoring of workload behavior*

- Finalizing the implementation of the built-in ceph top tool

CERN