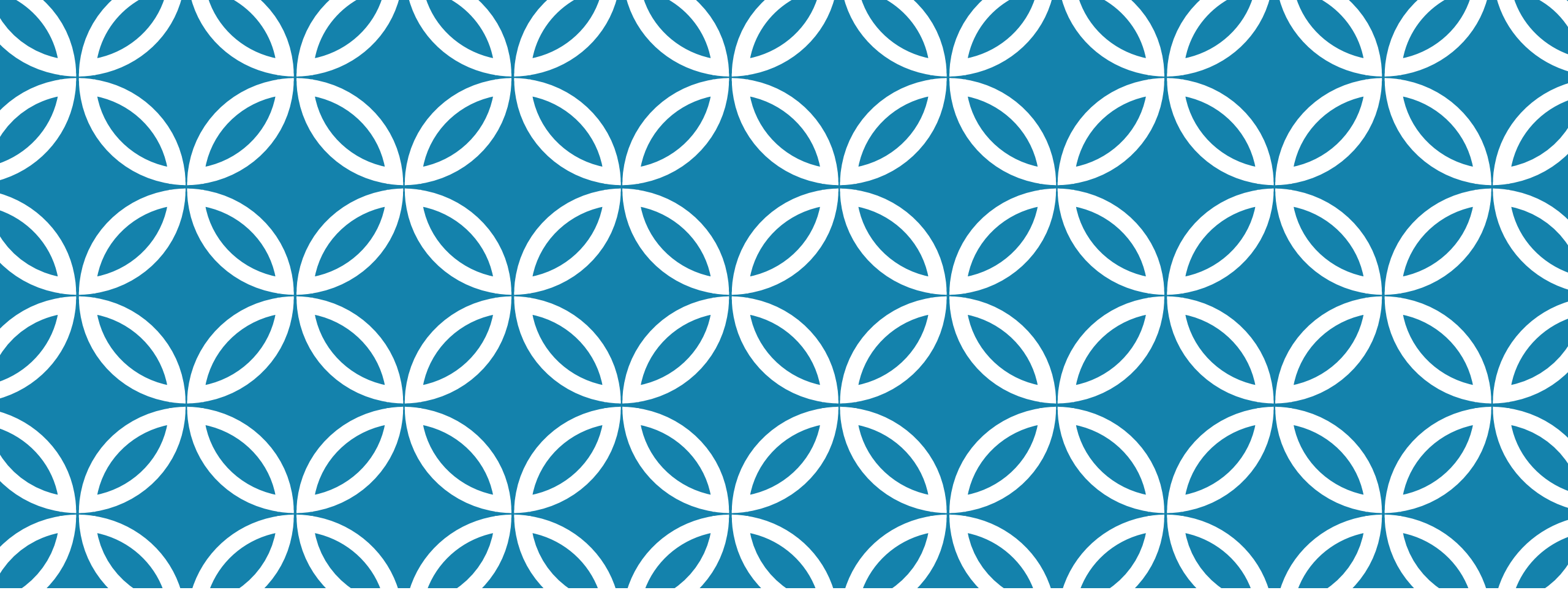

"CASTOR ON CEPH" TESTS FROM THE CEPH POINT OF VIEW

D. van der Ster

9 May 2016

CEPH "ERIN" CLUSTER

18 machines: Open Compute Platform

- 2x Xeon E5-2640, 64GB RAM, 30x 6TB HGST

```
GLOBAL:
    SIZE       AVAIL      RAW USED     %RAW USED
    2930T      2295T          634T         21.65
```

Ceph jewel v10.2.0: Cluster installed with ~v10.1.0, upgraded to stable release.

"New" Challenges:

- 30 OSDs with only 64GB RAM
- Maximize throughput with Erasure Coding

Started with clean ceph.conf to ensure we're not carrying forward obsolete hammer tunings.

LOW MEMORY CONFIGURATION

Highest Ceph memory usage happens with large EC pools, during recovery/backfilling

We provoked some backfilling an pushed some OSD servers into swapping: some OSDs had >4GB RSS.

Re-enabled the small osdmap cache tuning:

- This dropped the memory, but still room for more.

Tested new "async" messenger type, which uses a thread pool to handle all peer connections, (instead of 2 threads per peer)

- `[global] ms type = async`
- Drops #threads in `ceph-osd` from a few thousand to a couple hundred.
- Drops memory usage substantially, and no more tcmalloc problems.
- Just beware that async isn't yet the default – It may have a few small bugs e.g. pgs stuck peering.

Currently never exceed ~900MB per OSD process.

```
[global]
  osd map message max = 10
  ms type = async
[osd]
  osd map cache size = 20
  osd map max advance = 15
  osd map share max epochs = 10
  osd pg epoch persisted max stale = 15
```

THROUGHPUT WITH EC: BLUESTORE TO THE RESCUE?

I was curious to try bluestore, new 2x faster object store backend.

- Setup 1 out of 18 servers with ceph-disk prepare --bluestore …
- Server had lower loadavg, and iotop showed fewer writes than the FileStore servers. (no double write penalty)

But after a few hours we had object inconsistencies.

- Sent a bug: http://tracker.ceph.com/issues/15590

So this is unfortunately not yet usable in production.

- Stable bluestore planned for Kraken release.

THROUGHPUT: REPLICATION VS. EC

Ran some internal all-to-all *rados bench* tests with 1-2-3-rep and various EC pools configurations:

- 1 replica: ~11GB/s
- 2 replicas: ~4.8GB/s
- 3 replicas: ~4GB/s
- EC 2+1: ~5GB/s
- More EC stripes decreases performance.
- Currently running 8+3 ISA, getting ~3-4GB/s internally.
- This helps a lot: `filestore max sync interval = 60`

Important to run *long* tests. Takes ~6 hours to achieve flat performance with a newly created pool.

MY FAVOURITE TOPIC: SCRUBBING

Jewel has a reworked op queue: scrub IO used to happen in dedicated "disk" threads, now it is scheduled with client/recovery IOs.

- See: `osd scrub priority = 1`

I had hoped this removes the need for all scrub tuning (scrub sleep, etc…), but without the scrub tuning we have very long slow requests during a stress test.

Current scrub tuning – very conservative/slow scrubbing:

```
osd scrub chunk max = 1
osd scrub chunk min = 1
osdscrub priority = 1
osd scrub sleep = 0.1
```

Good news is the scrub timing randomization is working: no more thundering herd of scrub IOs.