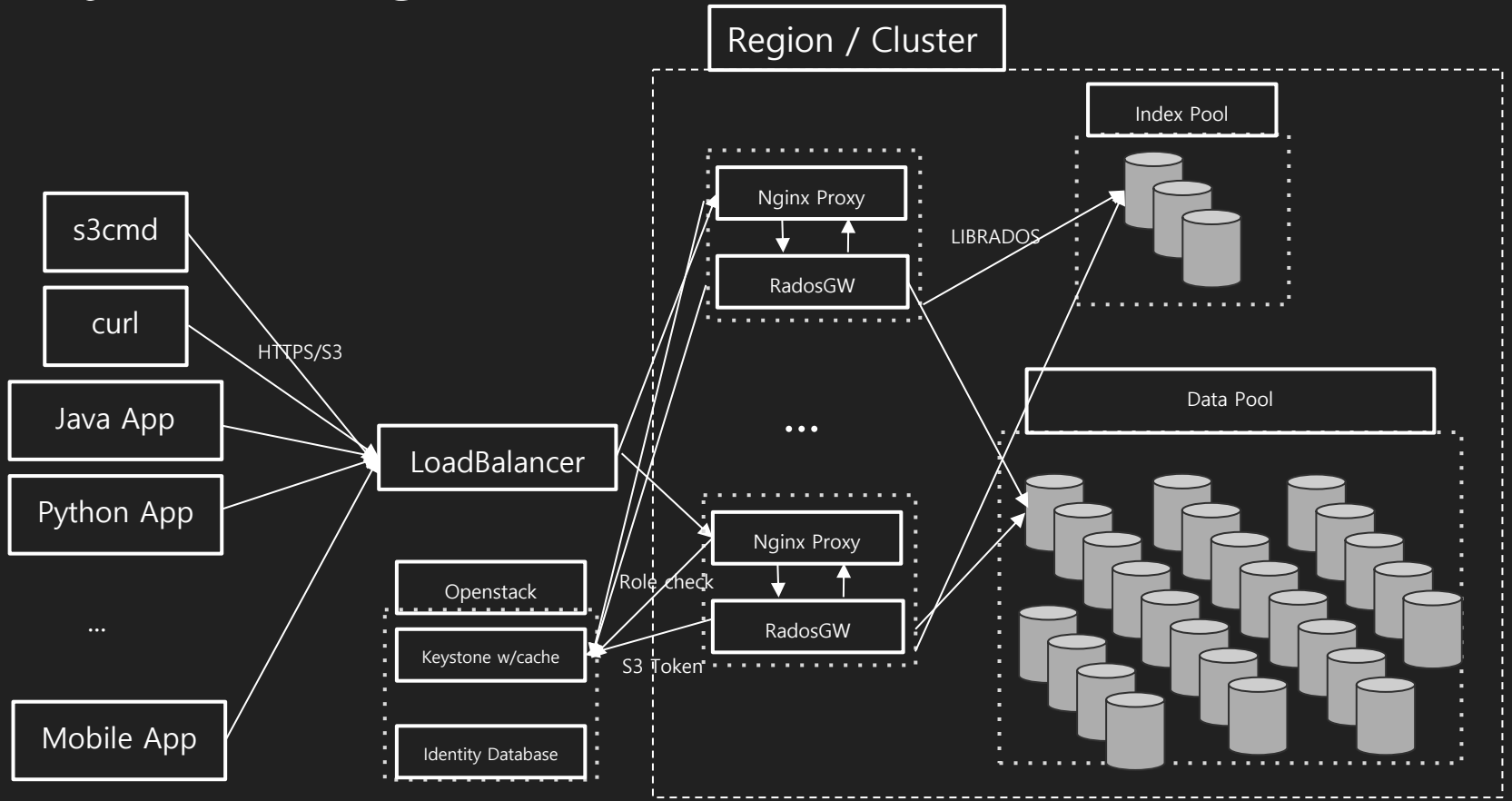
Online migration process of 6 billion objects from HDD-based OSD to SSD-based OSD

Open Infra Community Day Korea 2020
Dongsoo Song, LINE+

Object Storage in LINE



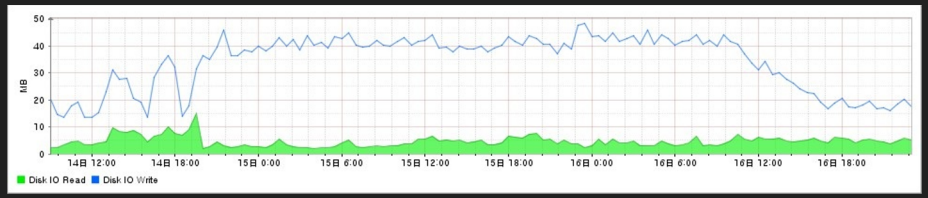
Before Migration

Category	#	Description
Number of Objects	**50 Billions – 125 TB**	•average object size = 25KB •objects per OSD = 104 millions •objects per PG = 610 thousands •PGs per OSD = 170 •Total objects = 50 Billions * 3 replicas = 150 Billions (375 TB)
Storage Type	**File Store**	•Journal on NVMe •RocksDB on xfs •Data on xfs , 100M+ inode / xfs
Storage Media	**NVMe SSD** **HDD**	•Index Pool = NVMe •Data Pool = 7.2RPM SAS HDD
Traffic	**Up - 20 Millions** **Down - 40 Millions**	•Contents - mostly web contents as CDN Origin •CDN traffic - 1~2 Billions / Day
Maintenance	**patch daemon &** **restart daemon**	•20~30 mins to restart a single OSD servers •12 servers * 12 OSDs => 6+ hours to patch daemon

Motivation

- 1 HDD failure
  - 8 hours to recovery by self healing
    (ex, if 1 disk failed, data in failed disk are rebalanced to the rest of 143 disks) → IOs are spread into 143 disks.
  - at least 4 days to backfill data when failed disk is replaced
    (ex. data need to be recovered to 1 disk from 143 disks) ← IOs are concentrated into 1 disk.

	recovery time (30% full) - 50 Millions	**recovery time (60% full) - 100 Millions**
Self Healing	4 hours	8 hours
Replacement	52 hours	104 hours



← backfilling started from 13:15 14/05 ~ until 17:20 16/05
=~ 52 Hours

Initial Plan

Category	Plan	Description
Servers to be replaced	HDD 8TB * 12ea * 12 2U servers ==> SSD 3.8TB * 24ea * 12 2U servers	raw 1PB, usable 340TB 5 billions objects → total 15 billions objects need to be migrated
Estimated migration window	3 months	2 months - backfiling 1 month - decommisson HDD servers
Steps	1. insert 3 SSD servers and backfill objects 2. repeat step#1 3 more times 3. decommission 3 HDD servers 4. repeat step#3 3 more times	expected least service impaction during backfilling

In real world (1)

1 pg backfilling = 600,000 objects are read from HDD + 600,000 objects are removed from HDD + GET/PUT to from HDD

→ cause disk utilization up to 100%

If osd_max_backfills = 1 → estimated 1 year to complete

In real world (2)

osd_max_backfills = 5 → estimated 3 months to complete, but ...

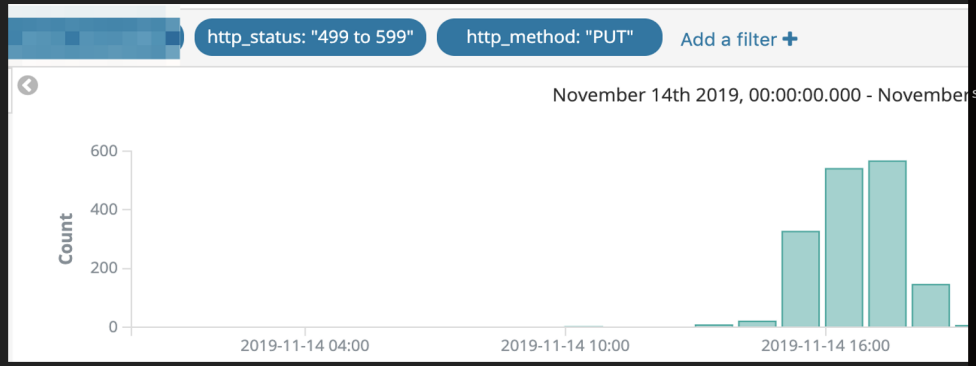
```
cluster:
  id:     2d07cbf3-123b-42ec-a73d-2a7a7a54f021
  health: HEALTH_WARN
          noout,noscrub,nodeep-scrub flag(s) set
          16 large omap objects
          2852437587/15024729363 objects misplaced (18.985%)

services:
  mon: [redacted]
  mgr: [redacted]
  osd: 720 osds: 720 up, 720 in; 3999 remapped pgs
       flags noout,noscrub,nodeep-scrub
  rgw: 12 daemons active

data:
  pools:   13 pools, 13632 pgs
  objects: 5.01G objects, 159TiB
  usage:   552TiB used, 1.53PiB / 2.07PiB avail
  pgs:     2852437587/15024729363 objects misplaced (18.985%)
           9633 active+clean
           3732 active+remapped+backfill_wait
           267  active+remapped+backfilling
```

nvme-nvme-sda--sdb--sdc--sdd--sde--sdf--sdg--sdh--sdi--sdj--sdk--sdl--sdm-													
util	util	util	util	util	util	util	util	util	util	util	util	util	util
7.55	7.44	0.03	2.28	3.72	2.79	2.61	2.44	2.50	2.40	3.08	1.67	17.2	1.07	5.11
72.8	84.2	0	93.8	90.3	99.3	49.4	100	99.9	98.4	99.3	91.4	99.9	99.7	100
81.1	63.1	0	87.2	86.8	99.2	55.1	99.9	99.7	97.4	99.9	96.5	100	99.8	100
51.5	56.5	0	88.8	91.5	99.2	60.4	98.5	98.7	98.9	99.3	94.6	100	99.8	100
63.5	78.6	0	85.4	83.6	99.4	41.7	98.6	99.2	98.3	100	95.4	100	99.9	99.6
74.6	89.4	0	85.8	91.4	99.3	39.4	97.5	98.6	99.3	100	89.5	100	99.4	99.6
67.6	90.7	0	86.3	86.5	98.7	44.0	99.1	99.1	98.2	100	92.1	94.9	99.1	100
94.1	94.0	0	79.7	92.0	99.5	50.1	98.4	99.1	99.3	100	86.5	89.4	99.6	100
97.7	87.6	0	88.3	90.9	99.8	58.6	98.6	99.2	99.1	100	92.2	90.8	99.6	99.9
93.0	87.6	0	88.8	96.0	99.5	47.3	99.1	99.1	96.7	100	93.4	88.5	99.5	99.9
89.4	91.2	0.10	90.5	88.4	99.7	42.1	98.8	99.7	98.3	100	92.1	93.2	99.4	99.9
55.8	46.4	0	97.1	95.6	99.5	65.2	99.9	99.1	99.5	100	100	93.0	100	100
97.4	84.4	0	100	93.5	99.9	64.5	99.0	99.7	98.4	100	96.0	93.5	99.7	100
92.2	89.6	0	99.7	96.0	99.8	75.5	99.4	100	98.8	100	95.1	89.1	99.6	100
94.9	91.4	0	99.9	88.8	99.8	55.5	99.2	100	97.5	100	87.3	97.0	99.7	99.9
80.7	77.7	0	99.9	86.6	99.8	61.6	98.4	100	98.2	100	97.8	100	99.8	100
97.9	84.9	0	100	92.0	100	52.0	98.6	100	99.3	100	98.4	99.9	98.5	99.8
80.9	98.2	0	100	100	99.9	60.2	99.2	100	97.7	99.8	97.8	100	99.9	100
85.2	88.2	0	100	100	100	75.3	99.2	99.8	99.8	100	97.5	99.8	99.7	100
63.4	97.3	0	100	100	99.5	64.1	99.8	99.9	98.4	100	94.7	100	100	99.8
67.0	9.40	0	99.9	100	99.4	98.6	99.8	99.9	98.8	99.7	93.1	100	99.9	98.8
40.6	76.5	0	100	100	98.9	98.7	99.4	100	98.7	99.6	87.2	99.6	99.7	99.2
42.3	49.5	0	99.4	100	99.2	99.2	99.7	100	97.7	99.8	92.4	100	98.0	99.4
69.5	66.1	0	92.3	100	99.7	99.2	99.2	99.9	98.4	98.4	92.2	98.5	99.6	98.9
66.0	69.7	0	93.0	100	99.7	90.9	99.6	100	98.7	99.6	90.2	92.6	99.1	99.4
91.7	92.5	0	93.7	88.3	100	75.2	99.6	99.9	99.1	99.6	93.7	89.4	99.9	100
82.1	77.1	0	99.8	98.7	100	75.0	99.8	99.7	99.4	99.4	95.0	86.1	99.6	99.8
94.3	92.0	0	99.8	92.7	100	66.1	99.5	99.9	97.8	99.9	89.3	93.3	99.7	99.9
80.1	68.5	0	100	92.7	100	74.2	99.5	99.9	98.7	99.7	90.2	91.2	98.2	99.8
85.7	94.8	0	99.8	94.4	100	61.8	99.8	100	97.4	99.6	89.2	94.6	99.6	99.7
93.5	63.9	0	100	90.4	100	62.6	100	100	98.7	99.8	95.9	89.0	100	99.5

In real world (3)

OSD failed to respond health check inspite of alive → flapping osd

⇒ **ceph osd set nodown**




```
cluster:
  id:     2d07cbf3-123b-42ec-a73d-2a7a7a54f021
  health: HEALTH_WARN
          nodown,noout,noscrub,nodeep-scrub flag(s) set
          16 large omap objects
          2716297836/15041207850 objects misplaced (18.059%)
          Degraded data redundancy: 36208250/15041207850 objects degraded (0.241%), 117 pgs degraded, 63 pgs under
sized
          75 slow requests are blocked > 32 sec. Implicated osds 62,80,89,91,94,98,102,124,128,142,145,158,162

services:
  mon:
  mgr:
  osd: 720 osds: 720 up, 720 in; 3916 remapped pgs
       flags nodown,noout,noscrub,nodeep-scrub
  rgw: 12 daemons active

data:
  pools:   13 pools, 13632 pgs
  objects: 5.01G objects, 160TiB
  usage:   556TiB used, 1.53PiB / 2.07PiB avail
  pgs:     36208250/15041207850 objects degraded (0.241%)
           2716297836/15041207850 objects misplaced (18.059%)
           9679 active+clean
           3761 active+remapped+backfill_wait
           65   active+remapped+backfilling
           62   active+undersized+degraded+remapped+backfill_wait
           30   active+recovery_wait+degraded
           23   active+recovery_wait+degraded+remapped
           6    active+recovery_wait
           4    active+recovery_wait+remapped
           1    active+recovery_wait+forced_recovery+degraded
           1    active+undersized+degraded+remapped+backfilling
```

In real world (4)

Super Outage for 2 hours

```
[redacted]~]# ceph osd map default.rgw.users.uid anonymous
osdmap e42025 pool 'default.rgw.users.uid' (10) object 'anonymous' -> pg 10.eac94372 (1
152], p82)
[redacted]~]# ceph osd find 82
{
    "osd": 82,
    "ip": "[redacted]6809/382106",
    "osd_fsid": "7f7aa037-ce94-4ca1-b026-0e90d68a2d4f",
    "crush_location": {
        "host": "[redacted]",
        "rack": "pb01pb02-data",
        "root": "data"
    }
}
```

- **Problem 1**

  All operations should be checked wheather they have proper authorizations.

  Authorization requires its owner's ACL which is stored in a user meta object in the form of key/value paris.

  Once a user's ACL is accessed, it is stored in a cache to serve following requests.

  But in case of an anonymous user, his ACL is not cached, so every requests from anonymous users involve reading ACLs from user meta objects in the default.rgw.users.uid pool every time.

  Unfortunately the default.rgw.users.uid pool was on the HDD based pool

- **Cause**

  Disk util 100% → filestore_op_thread_suicide_timeout (180sec) expired → osd kill himself → restart osd

  → rocksdb journal check on startup (it took around 9 mins)

  → gw objecter_inflight (either 24576 or 100MB contents) threshold was over

  → rgw OP IO threads were throttled and waited until the OP Q was run out.

In real world (5)

Super Outage for 2 hours

- **Problem 2**

  LB periodically sends an operation as an anonymous user(GET / HTTP/1.0), which means every request from LB involves reading ACL from objects.

  In this situation, one of OSDs which has anonymous user meta objects was stalled to respond.

  Because of this, every LB operations were also blocked.

  this cause health check failure of LB and LB would not send traffic to real servers which are radosgw.

  this cause whole cluster is not operational for some time even though radosgw and osd servers are not busy.

- **Cause**

  osd **nodown** flag was set because heavy disk IO may result in osd health check fail (if 2 or more peers report to ceph-mon, then ceph-mon mark the osd down.)

  ← it is recommended to turn on nodown flag under backfilling state with heavy loaded DISK IO and this prevent osd flapping but...

  all the osds were waiting for the troubled osd to answer the health check (the osd was bootstrapped, have no response of health check message but it took 10 minutes due to roscksdb was on HDD and it was pressed by disk IO)

In real world (6)

Patch & Tune

- **anonymous user → pass authorization procedure**
- **move bucket meta pools to SSD based pool**
  - **ceph osd pool set default.rgw.users.uid crush_rule SSD**
- **tune osd thread_suicide_timeout** https://access.redhat.com/solutions/2971581

osd_op_thread_timeout = 90 #default is 15

osd_op_thread_suicide_timeout = 2000 #default is 150

osd_backfill_scan_max = 1 # default 512

osd_backfill_scan_min = 1 # default 64

osd_peering_wq_threads = 4 # default 2

- If recovery thread is hitting the timeout, increase recovery op threads timeout

osd_recovery_thread_timeout = 120 #default is 30

osd_recovery_thread_suicide_timeout = 2000 #default 300

- If filestore op threads are hitting the timeout

filestore_op_thread_timeout = 180 #default is 60

filestore_op_thread_suicide_timeout = 2000 #default is 180

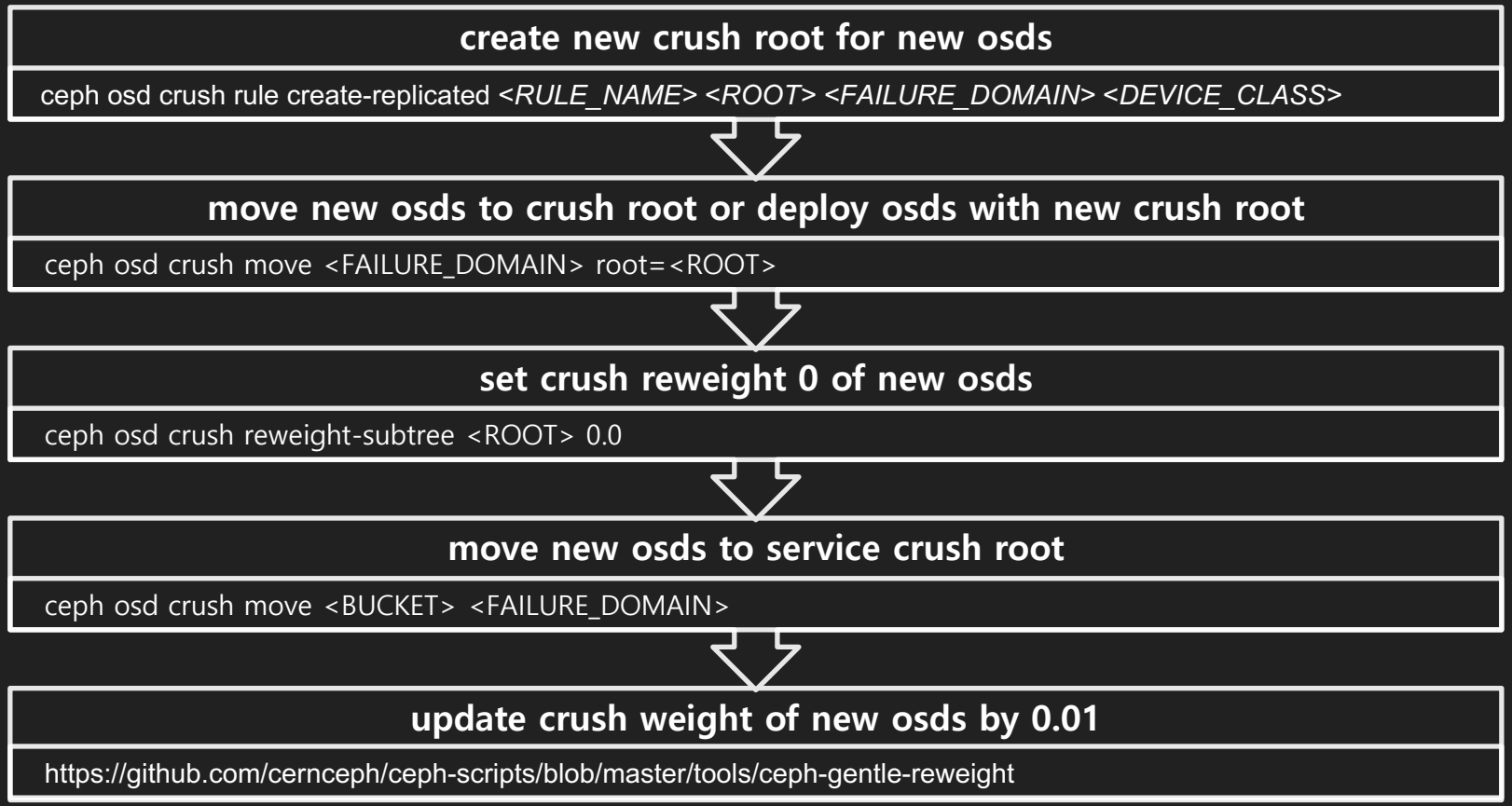
Outage Prevention

- slow down backfilling speed & control backfilling
  - set osd_max_backfills = 1
  - if slow_requests are found, then stop backfilling (set osd_max_backfills = 0)
- make HDD based osd secondary or tertiary if possible
  - ceph osd primary-affinity osd.$id 0
- Prevent osd flapping & prepare disk failure
  - ceph osd set nodown
  - if osd commit suicide due to hight disk utilization or osd die due to HW failure, then 'ceph osd unset nodown'
- gently update crush reweight of new osd by 0.01
  - https://github.com/cernceph/ceph-scripts/blob/master/tools/ceph-gentle-reweight
- patch anonymous user not to check authorization
  - https://github.com/ceph/ceph/pull/34287

Final results - 10 months later

	Before	After
Disks and Servers	HDD 8TB * 12ea * 12 servers	SSD 3.8TB * 24ea * 30 servers
SIZE	raw 1PB	raw 2.5PB
OSDs	144	720
Objects	5 billions	6 billions
Placement groups	8192	16384
Traffic	60 millions / day	140 millions / day
CDN hit	2 billions / day	3.5 billions / day
Rebalance	8 hours	2 hours
Recovery	4.5 days	8 hours
Restart osd	20 minutes	2 minutes

Lesson and Learn - capacity expansion, migration, ...



THANK YOU

Q & A

dongsoo.song@linecorp.com