
CERN

Ceph Ops Team Update

Dan van der Ster, CERN IT Storage Group
daniel.vanderster@cern.ch

Ceph HEP Monthly, 5 September 2016

New Tools

- https://github.com/cernceph/ceph-scripts

- **ceph-gentle-reweight**
  - Gradually add or remove OSDs from a cluster.
- **ceph_osds_in_bucket.py**
  - Module to find OSDs in a CRUSH bucket.
- **crush-reweight-by-utilization**
  - Updated to reweight OSDs in a CRUSH bucket.
- **ceph-leader**
  - Tool which exits 0 if the current machine is the mon leader. (Useful for crons)

Ceph Hardware Replacement

- Need to replace 960-3TB OSDs with 1152 new 6TB drives
- **How not to do it…** add new OSDs and remove old OSDs all at once
  - Would lead to massive re-peering, re-balancing, unacceptable IO latency.
- **How to do it:** gradually add new & remove old OSDs
  - How quickly? OSD-by-OSD, server-by-server, rack-by-rack? Tweaking weights as we go?
- Considerations:
  - We want to reuse the low OSD id's (implies add/remove/add/remove/… loop)
  - We don't want to have to babysit (need to automate the process)
  - **We want to move rgw pools to another cluster!**

Hardware Replacement: Done

- Completed during July-August. No problems, transparent to users.

- Procedure:
  - Add 1 new rack. Create new OSDs with

```
osd crush initial weight = 0
osd crush update on start = true
```

  - Drain 2-3 old racks. Make sure to set this, in case OSDs restart during draining.

```
osd crush update on start = false
```

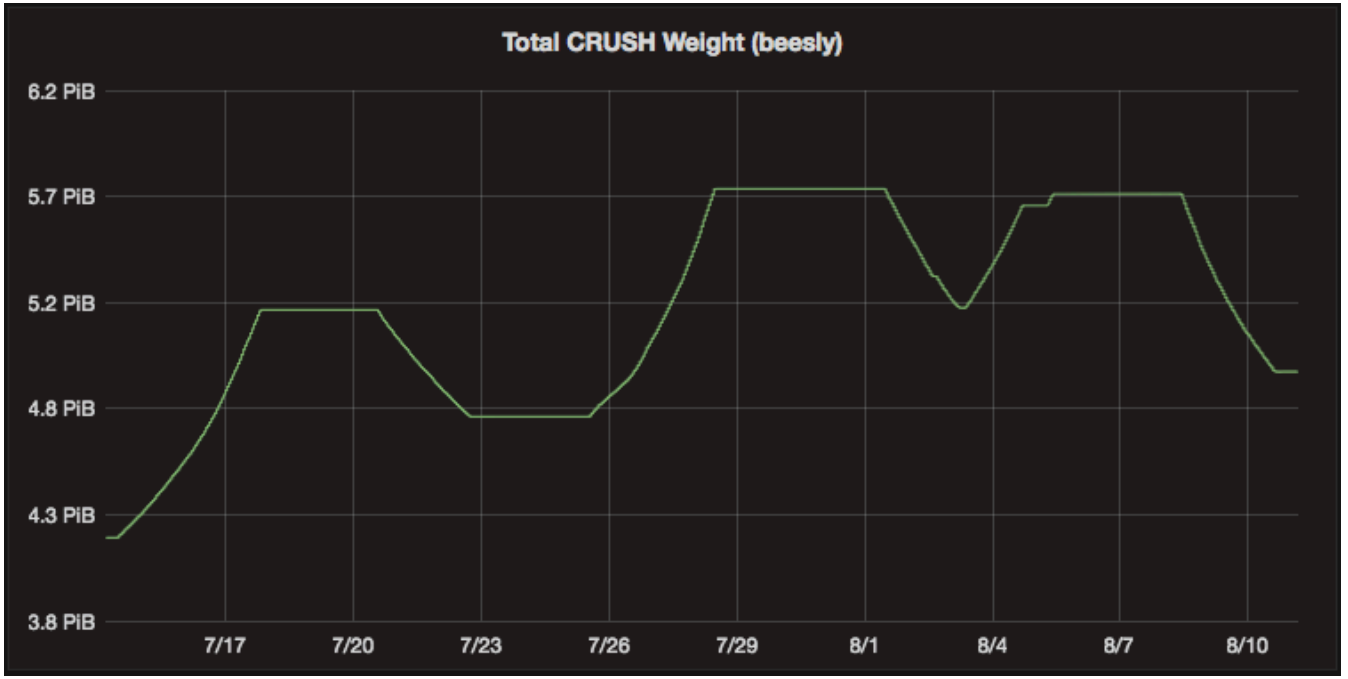
  - Repeat.

- **ceph-gentle-reweight** used to add and remove.
  - Adding a list of OSDs:

```
ceph-gentle-reweight -o osd.102,osd.103,… -l 15 -b 50 -d +0.01 -t 5.46
```

  - Draining a list of OSDs:

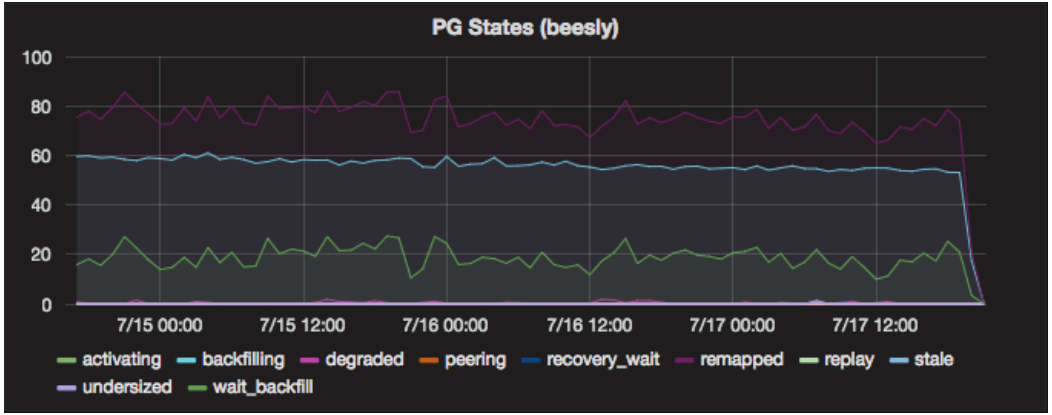
```
ceph-gentle-reweight -o osd.202,osd.203,… -l 15 -b 50 -d -0.01 -t 0
```

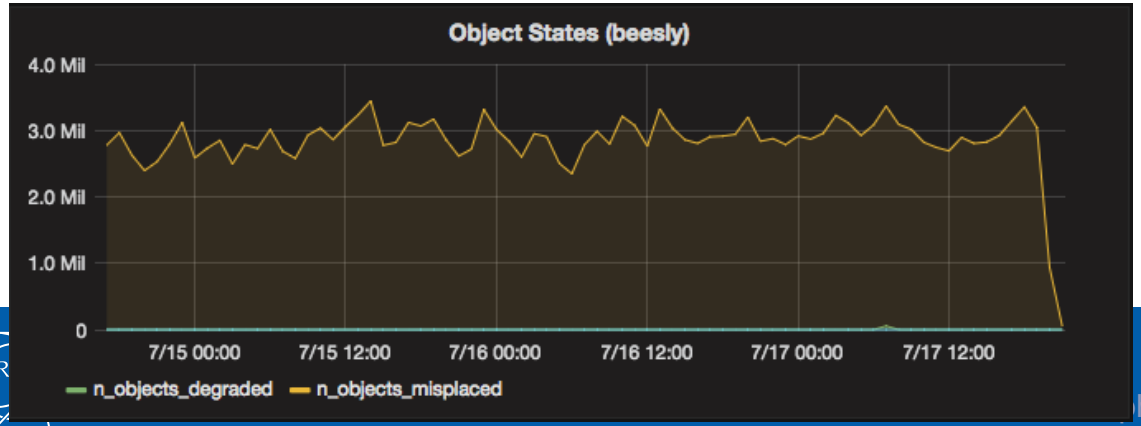


- This plot shows the total capacity changing as we added/drained OSDs.

PG and Object States

`ceph-gentle-reweight` script kept 50-60 PGs consistently backfilling

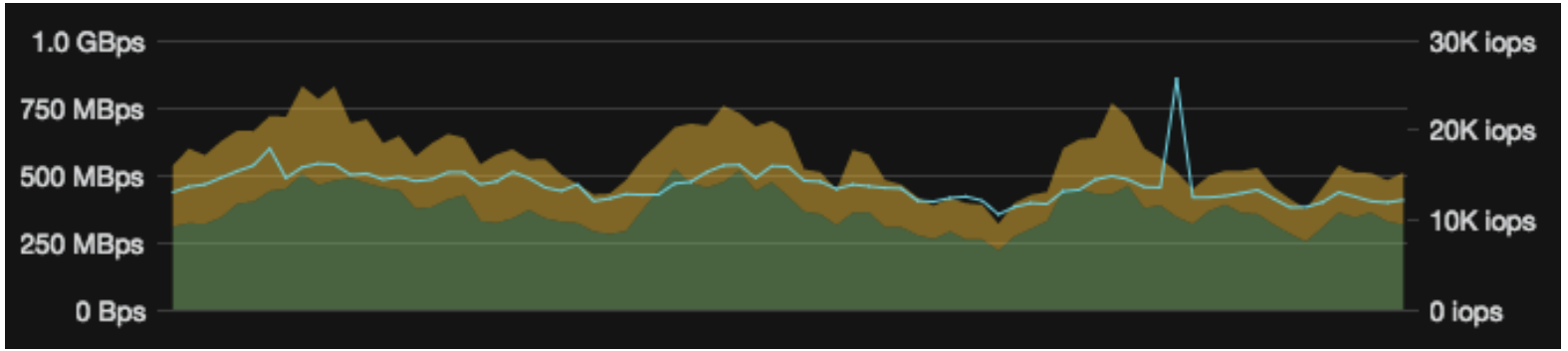


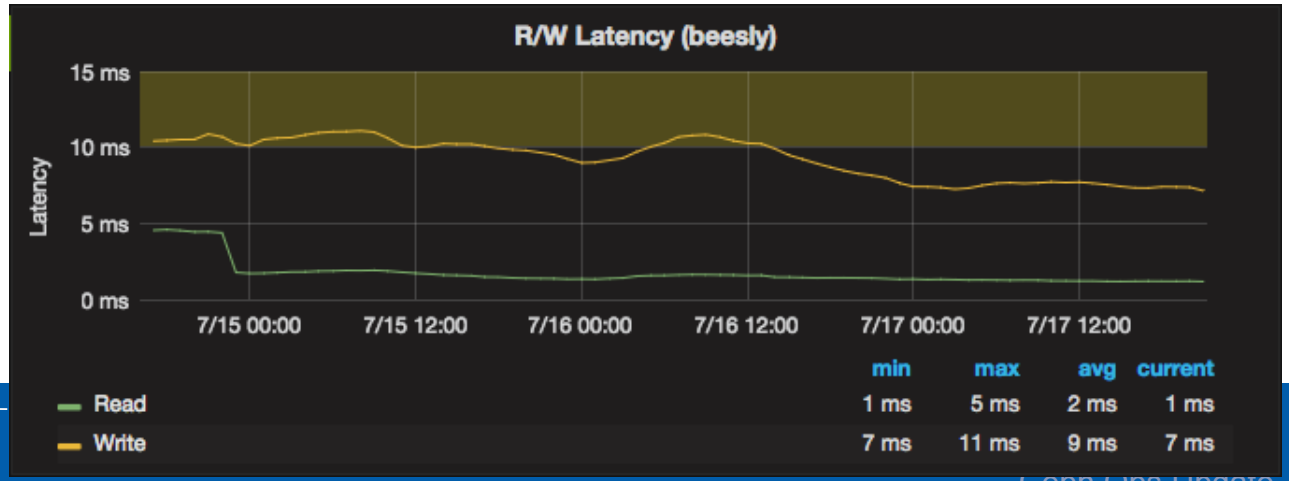


2-3 million objects misplaced

<100 degraded objects

Activity during replacement campaign

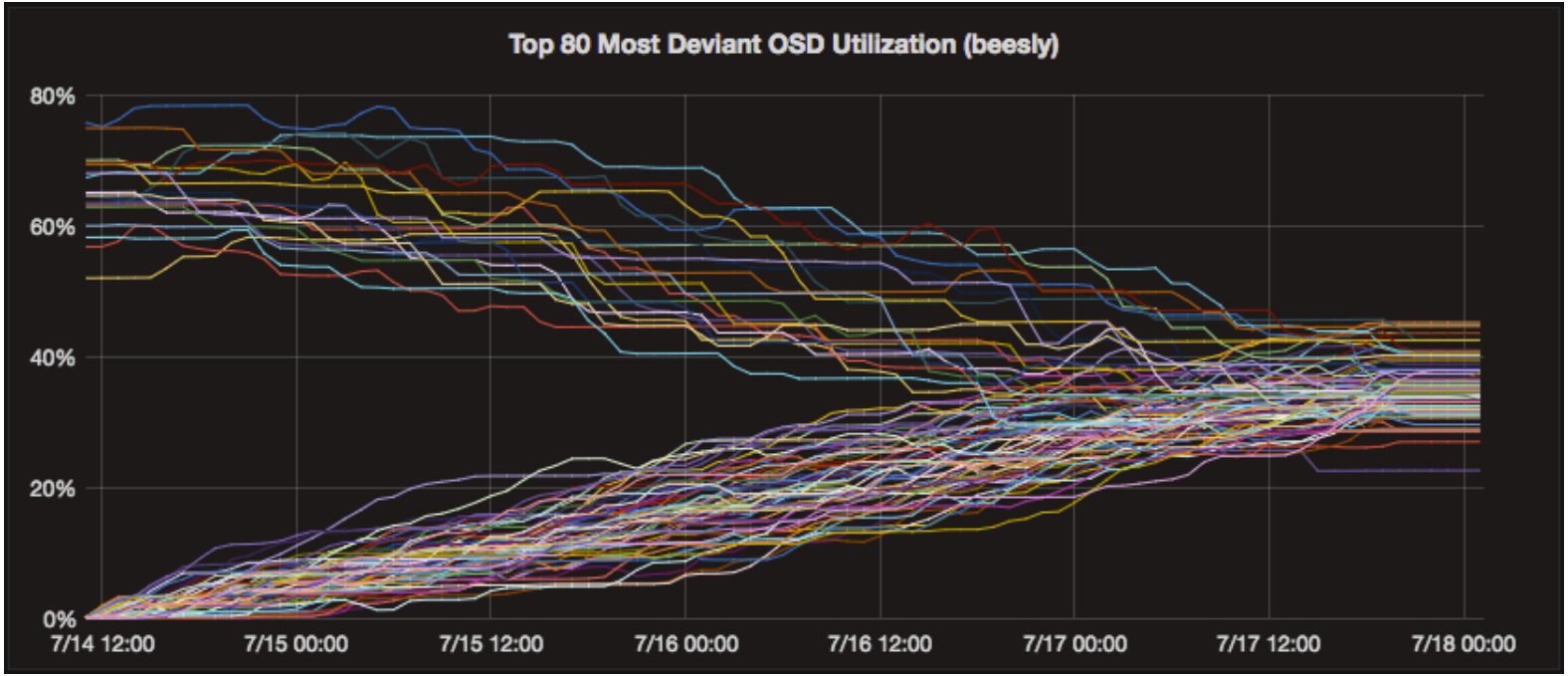




15’000 user IOPS

Write latency stayed under ~10ms

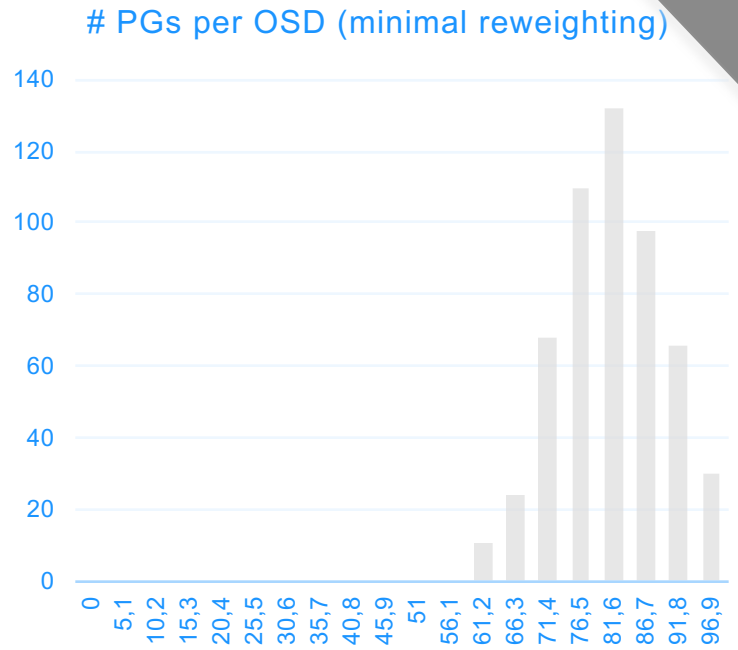
Cool plot showing filling/draining OSDs



Balancing OSD data

From Ceph Day

- We often want to fill a cluster:
  - Imagine not being able to use 10% of a 10PB cluster !!
- Hammer 0.94.7 and Jewel have a new *gradual* (test-)reweight-by-utilization feature
  - This is a good workaround, but it decreases the flexibility of the OSD tree
  - Proactive reweighting of an empty cluster is much more effective than fixing things later.



Data balancing: Done?

- It turns out that jewel's *reweight-by-utilization* feature still has some limitations:
  - No support for clusters with many root CRUSH buckets or rooms
    - Different CRUSH roots have different avg utilizations, which confuses the reweight tool.
  - In general, it behaves badly on clusters which are uneven by design:
    - E.g. 3 server cluster, with one server smaller than the other two.

- So our `crush-reweight-by-utilization` tool has been updated to support irregular clusters.
  - Example, to reweight only the OSDs within the room named *asdf*, do

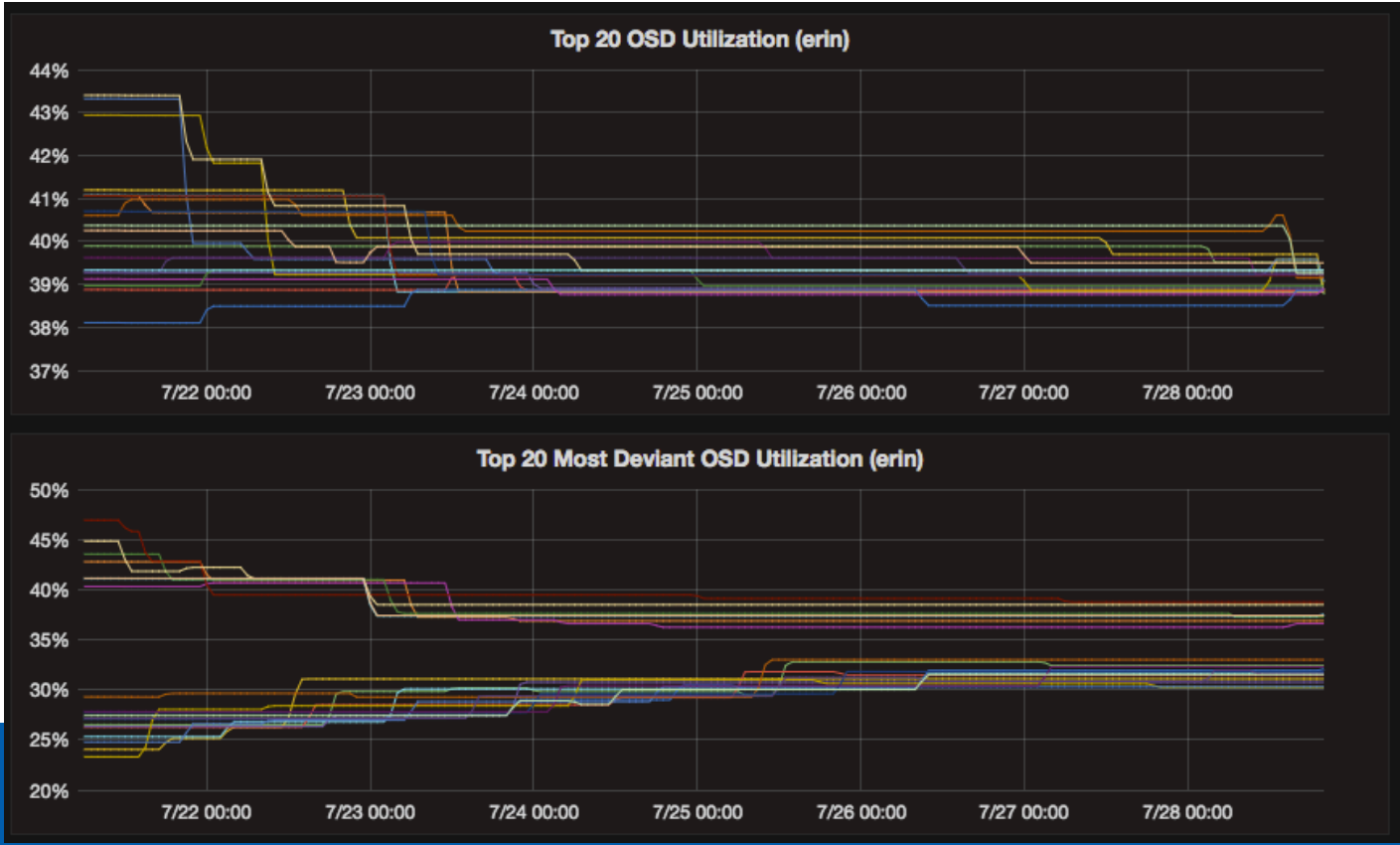
```
crush-update-by-utilization --bucket=asdf
```

Automating the reweights

- We're now running `crush-reweight-by-utilization` in a cron on all mons:

```
41 */2 * * * ceph-leader && ceph health | grep -q HEALTH_OK &&
crush-reweight-by-utilization.py --overload=115 --doit --really
```

Squeezing the OSD util. distribution




CERN