

CEPH DAYS · CHICAGO

Minimizing Data Movement on CRUSH Failure Domain Changes

Rack to row on a real 86 PB, all-NVMe, erasure-coded cluster

Frédéric Nass · Ceph Solutions Architect, CLYSO
Ceph Ambassador France · Ceph Foundation Board Member

A Real Production Cluster

1600

OSDs

Up from 680. All NVMe, pure RADOS, 86 PB raw.

80

Racks

Up from 34, for 160 nodes in total.

2

Nodes per rack

In each of the 80 racks.

One EC 6+2 pool, min_size 7, failure domain **rack**.

Two Concerns

Too Few Hosts per Rack

With only 2 nodes per rack bucket, there are not enough buckets to redistribute data properly when a node fails.

One Rack Down at a Time

EC 6+2 with min_size 7 lets only one rack (2 nodes) go offline while I/O continues. OS and firmware upgrades took ages, and even longer after the extension to 1600 OSDs.



Still from Apollo 13 (1995), Universal Pictures

As a cluster evolves, what you chose back then may no longer fit today's configuration.

So the Operator Asked

- Can we switch our failure domain from rack to row?
- How much data would have to move?
- How much pressure would that put on the cluster and the clients?

PART 1

What Would It Cost?

Simulating the failure domain change before touching the cluster



We Simulated the Failure Domain Change*

From a single **ceph report**, we replayed CRUSH offline and simulated 10 rows of 8 racks.

82.5%

Shards would move

Of the EC 6+2 pool, with a plain rule change.

~23 PiB

Of data to move

Nearly all of the data in the pool.

26.6%

Actually required

The shards that truly violate the new failure domain.

Far too much: three times what the new failure domain actually requires.

** Genchi genbutsu (Toyota): "go and see for yourself"*

Why So Much Movement?

1

CRUSH Draws Again

On a rule change, CRUSH recomputes every PG. With EC (indep), most shards land on different OSDs at the new level.

2

Upmaps Dropped

78.6% of the pool's pg_upmap_items no longer match the new rule: cancelled or pruned. The balancer's work is gone.

3

Data Moves Anyway

Even shards that could legally stay are moved, because nothing considers where the data already is.

Replicated pools (firstn) barely notice; erasure-coded pools (indep) take the full hit.

BTW, Ever Seen This?

Changed the rule, reverted it under norebalance, and PGs are still misplaced?

Same cause.

On the production map: 12.15% of shards (~3.45 PiB) are left misplaced, with zero bytes moved.

PART 2 - THE IDEA

What if, in the same epoch, upmap kept every OSD the new rule still allows and evenly replaced the rest?



How It Works

1

Keep

For each PG, keep the acting OSDs that already satisfy the new failure domain. Most of them do.

2

Replace, Balanced

Swap only the violating shards, picking replacements that keep OSD load even, so the balancer has little left to do.

3

Same Epoch

The monitor stages the pg_upmap_items with the rule change, in the same epoch, before the new osdmap is published.

Every staged placement goes through verify_upmap first: nothing that violates the new rule can be written.

In Ceph: PR #70441, Opt-In

Opt-in and off by default. PR #70441 (tracker #78569) is under review, targeting v22.

```
$ ceph config set mon mon_crush_rule_change_preserve_acting true
$ ceph osd pool set <pool> crush_rule <row-rule>
```

It uses plain pg_upmap_items, nothing new on the wire: it only needs require_min_compat_client luminous, which any cluster using the balancer already has.

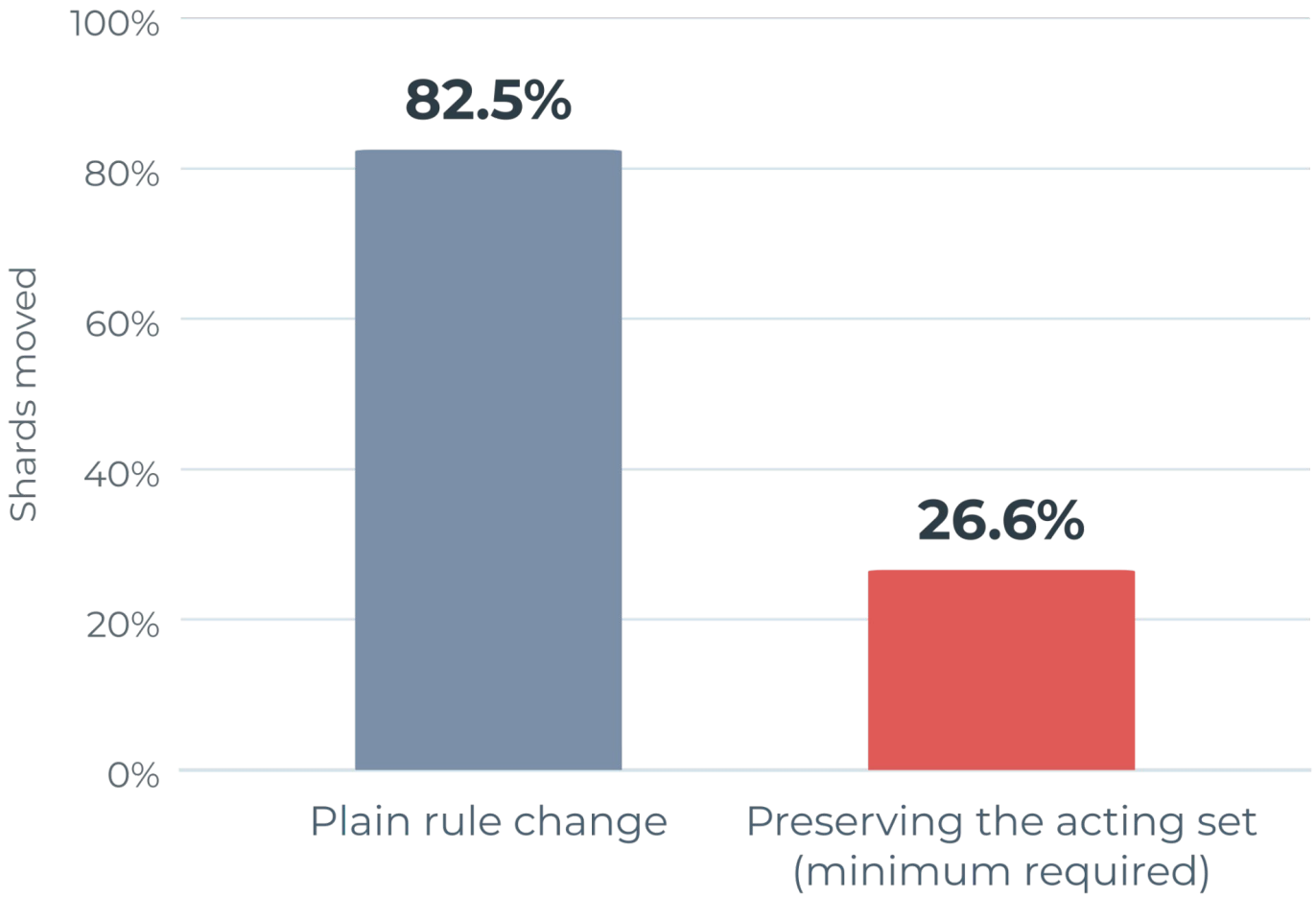
PART 3

Results, Demo and What You Can Use Today

Same cluster, same map, same target rule



Shards Moved on the Rack → Row Switch



The Minimum, and Nothing More

With the option, the switch moves only the shards the new failure domain forces to move.

What It Means for the Operator

3.1x

Less data moved

About 7.5 PiB to move instead of ~23 PiB.

0.7%

Left for the balancer

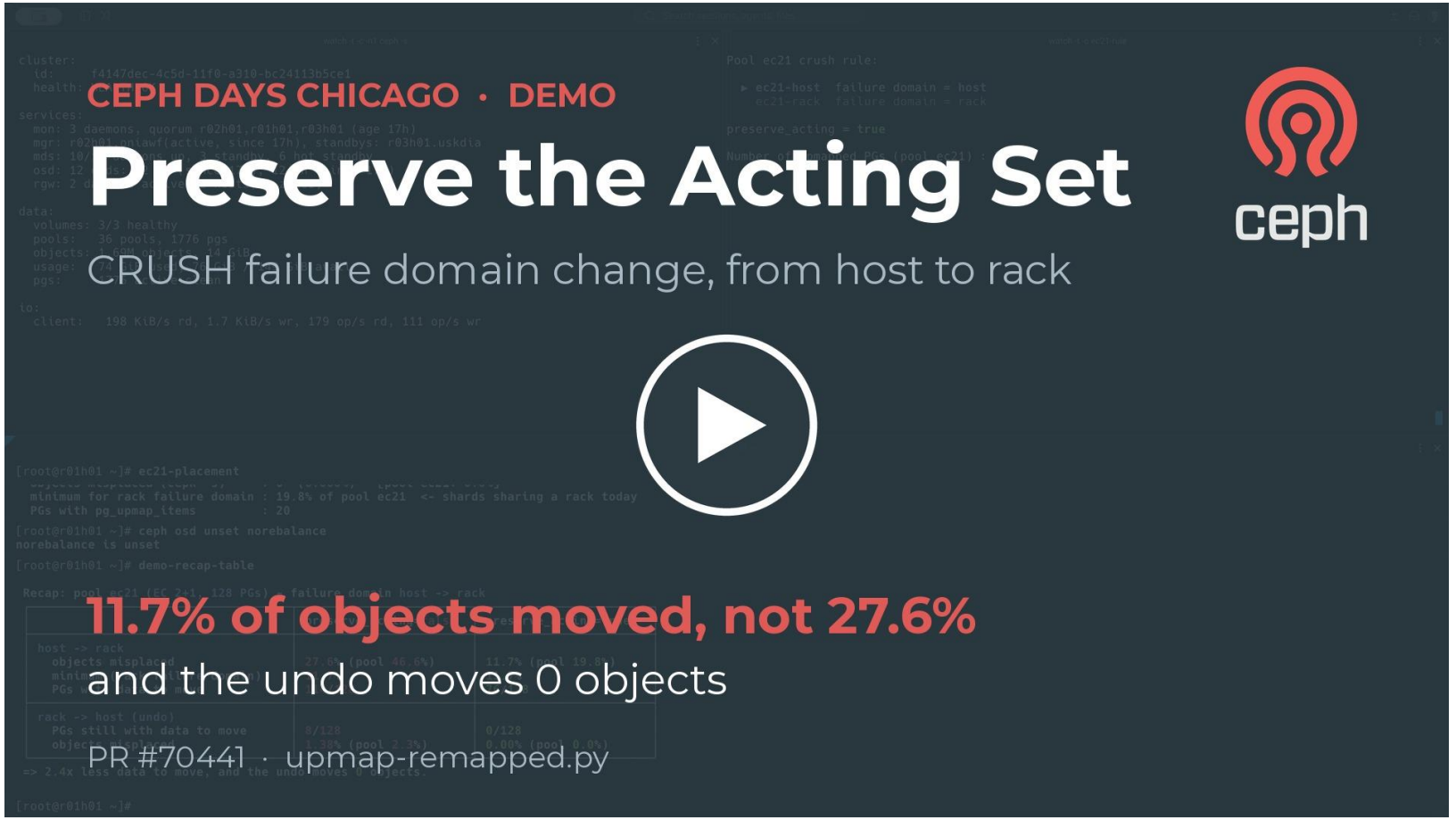
Was 4.5%. OSDs stay as balanced as before the change (spread 1.9% vs 11.2%).

0%

After a revert

Was 12.15% (~3.45 PiB) without the option: try it, roll it back, for free.

Demo: Host → Rack, With and Without the Option



12-OSD test cluster · pool ec21 (EC 2+1, 128 PGs) · failure domain host → rack, then undo

CEPH DAYS CHICAGO · DEMO

Preserve the Acting Set

CRUSH failure domain change, from host to rack

11.7% of objects moved, not 27.6%

and the undo moves 0 objects

PR #70441 · upmap-remapped.py

What You Can Use Today

TODAY

AVAILABLE NOW

upmap-remapped.py

- Same logic as the PR, as a script
- Emits plain pg-upmap-items
- No patched Ceph needed
- Merged in ceph/ceph main: src/tools/contrib/upmap-remapped.py

NEXT

Built Into Ceph

- mon_crush_rule_change_preserve_acting
- Opt-in, off by default
- PR #70441, tracker #78569
- Under review, targeting v22

Give it a try, and share your feedback on PR, tracker, dev@ceph.io or ceph-users@ceph.io

Thank You!

Frédéric Nass · Ceph Solutions Architect, CLYSO

Ceph Ambassador France · Ceph Foundation Board Member ·
github.com/frednass



PR: **github.com/ceph/ceph/pull/70441**

Tracker: **tracker.ceph.com/issues/78569**

Script: **upmap-remapped.py**

Feedback: **dev@ceph.io**