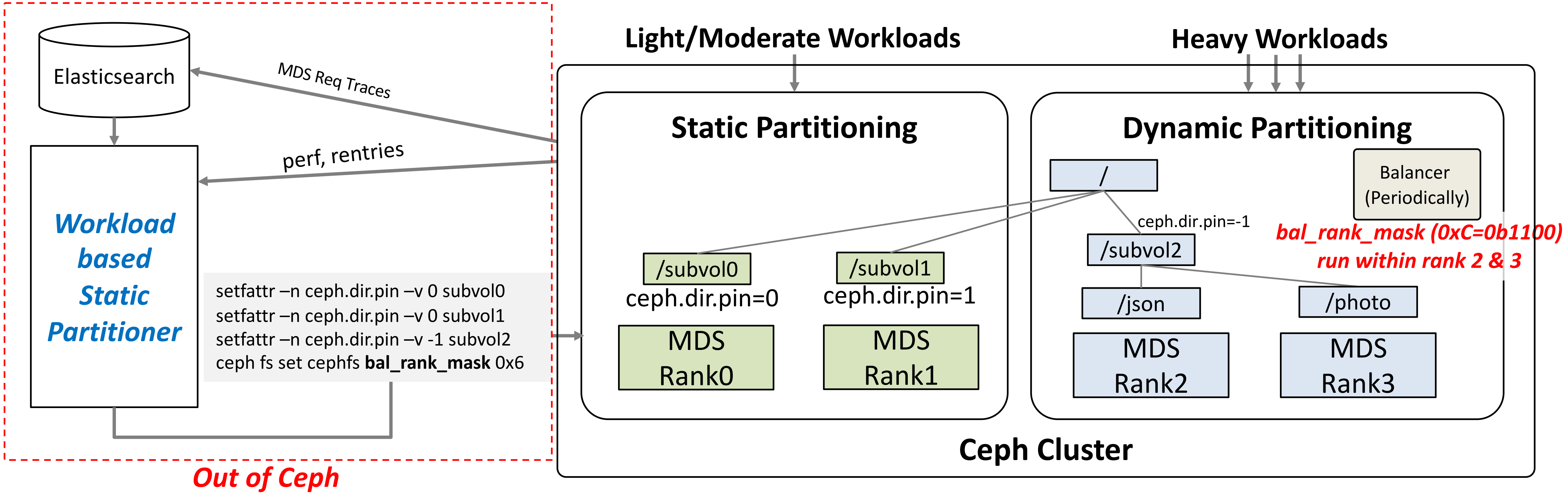
A New MDS Partitioner for CephFS

LINE

Yongseok Oh

Workload Based Static Partitioner with bal_rank_mask



```
setfattr –n ceph.dir.pin –v 0 subvol0
setfattr –n ceph.dir.pin –v 0 subvol1
setfattr –n ceph.dir.pin –v -1 subvol2
ceph fs set cephfs bal_rank_mask 0x6
```

- This idea was presented at **Cephalocon2023**
- ***Workload based static partitioner*** pins subvolumes
  - Workload calculation based on working set, rentries, and performance
  - Rarely or manually conducted if loads are uneven or latencies get higher
  - Make subvolumes involving heavy workloads managed by MDS balancer with bal_rank_mask

Technical Issues

- Our in-house partitioner is useful for performance
  - Compared to simple pinning, it distributes subdirs based on workloads
  - However, it is unavailable as open source
  - It needs to be revised and reimplemented for Ceph community
- bal_rank_mask needs to be enhanced
  - It can isolate unpinned large subtrees within certain MDS ranks from pinned subtrees
  - But, migrating large subdirs incur metadata movements
  - per subdir rank mask will be useful

A New MDS Partitioner

- rank mask option per subdir as a virtual extended attribute
  - A target subdir is dynamically within certain MDS ranks (e.g., rank0 and 1)

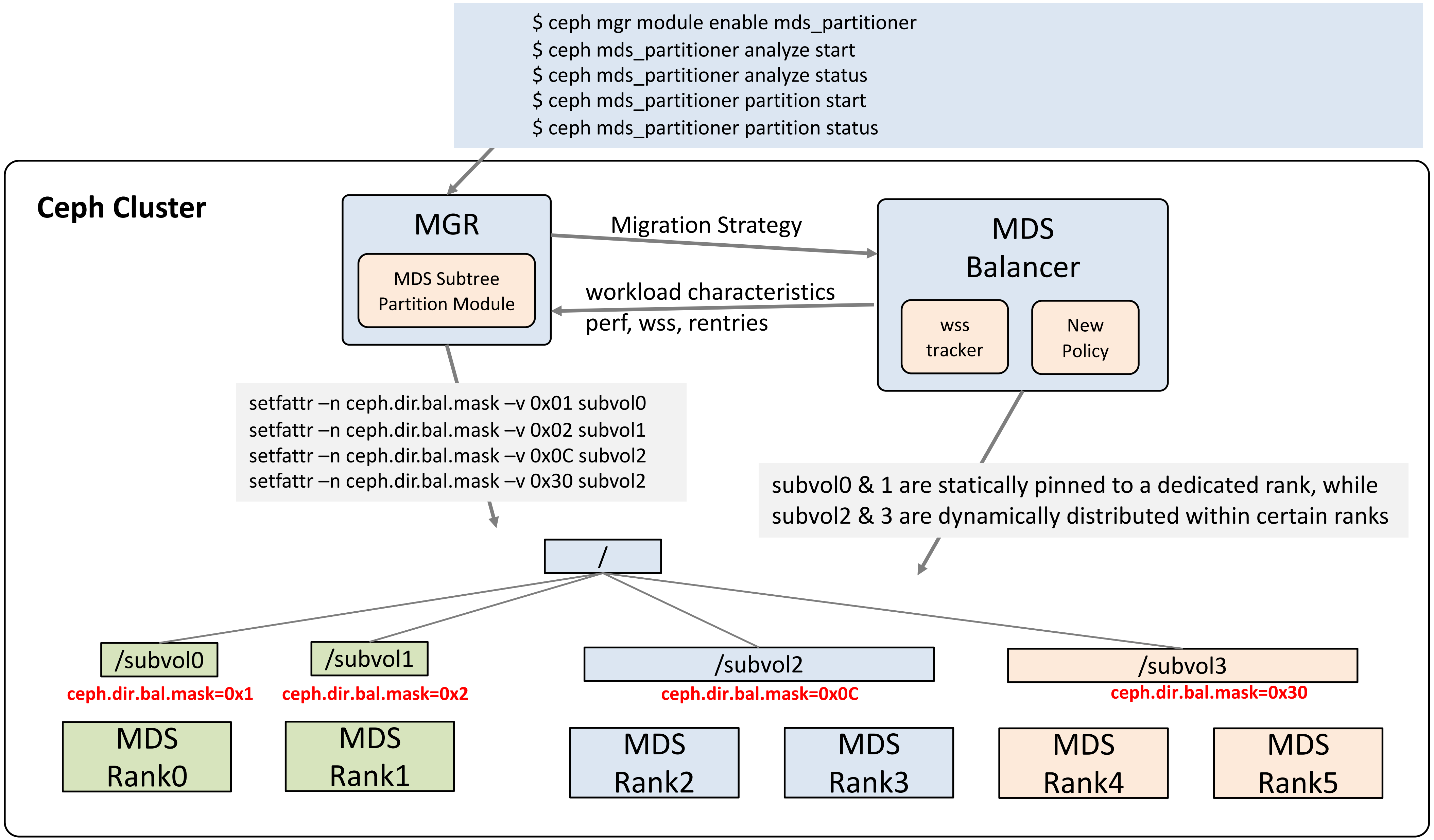
```
setfattr –n ceph.dir.bal.mask –v 0x3 /cephfs/home/yongseok
```

- MDS Subtree Partition Module in MGR

```
ceph mgr module enable mds_partitioner
ceph mds_partitioner analyze start   # analyze client workloads ontained from MDSs
ceph mds_partitioner analyze status # report analysis results and recommend optimal the number of MDSs
ceph mds_partitioner partition start # start partitioning
ceph mds_partitioner partition status # report partitioning status
```

- MDS Balancer modifications
  - Working set size tracker
  - Migrate subdirs based on ceph.dir.bal.mask values of subdirs
  - Minimize MDS slow requests

Overall Architecture

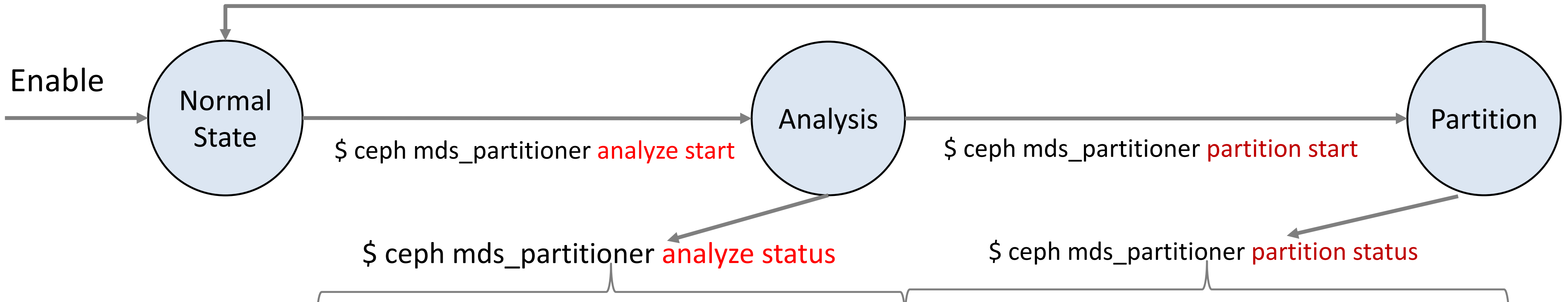


```
$ ceph mgr module enable mds_partitioner
$ ceph mds_partitioner analyze start
$ ceph mds_partitioner analyze status
$ ceph mds_partitioner partition start
$ ceph mds_partitioner partition status
```

```
setfattr –n ceph.dir.bal.mask –v 0x01 subvol0
setfattr –n ceph.dir.bal.mask –v 0x02 subvol1
setfattr –n ceph.dir.bal.mask –v 0x0C subvol2
setfattr –n ceph.dir.bal.mask –v 0x30 subvol2
```

subvol0 & 1 are statically pinned to a dedicated rank, while subvol2 & 3 are dynamically distributed within certain ranks

Example of Operation Flow



```
$ ceph mds_partitioner analyze start
$ ceph mds_partitioner partition start
$ ceph mds_partitioner analyze status
$ ceph mds_partitioner partition status
```

Name	wss	reqs	rentries	workload	Current Ranks	New Ranks	Migration Progress	Migration Status
Subvol1	1,000,000	1,203,030	50,000,000	339	0	0,1	20%	In-progress
Subvol2	700,000	500,000	1,000,000	137	1	2	0%	Ready
subvol3	100,000	5,000	5,000,000	21	2	2	100%	Done
subvol4	3,000	20,000	70,000	3	2	2	100%	Done
Total	1,803,000	1,728,030	56,070,000	500				

wss: working set size
reqs: requests
rentries: files + dirs

workload = (working_set_size / total_working_set *2
+ requests / total_requests * 2
+ rentries / total_rentries) * 100