COMMUNITY
THE ASF CONFERENCE
CODE

Apache Ozone: Balance Data Through Disk Balancer

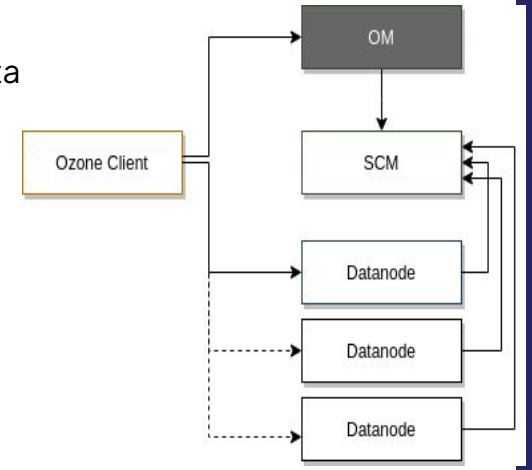
Sammi Chen (Ozone PMC Chair & Hadoop PMC)
Yiyang Zhou (Ozone PMC)

COMMUNITY
THE ASF CONFERENCE
CODE

CONTENTS

1. Motivation
2. Design
3. Benchmark

Apache Ozone Introduction

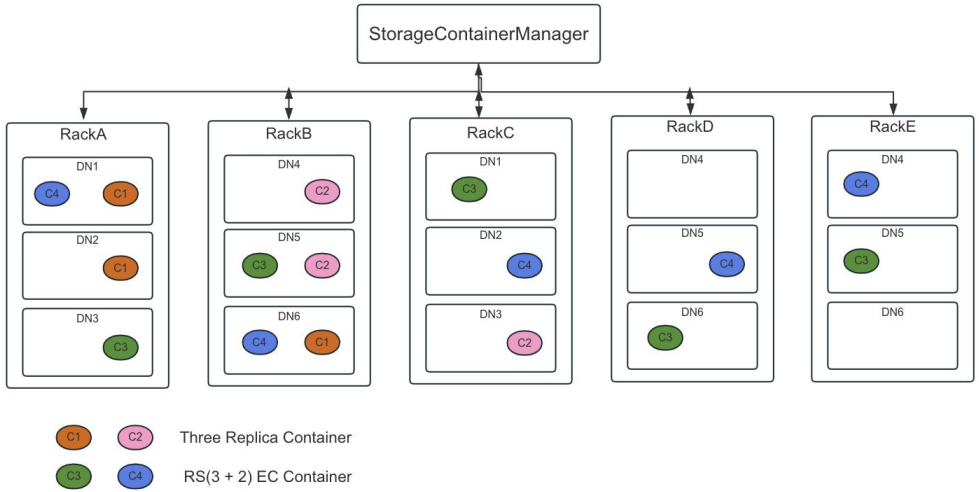


- Apache Ozone is a scalable, redundant, and distributed data storage for big data ecosystem.
- Ozone Manager (OM) manages the namespace (volumes, buckets, keys), thus called the namespace manager.
- Storage Container Manager (SCM) handles block allocation and replication, called the block space manager.
- Containers are the fundamental replication unit of Ozone, they are managed by the SCM service.
- All data are stored on Datanodes (DN).

Background – Cluster Wise Container Distribution

SCM is responsible for choosing datanodes for container allocation

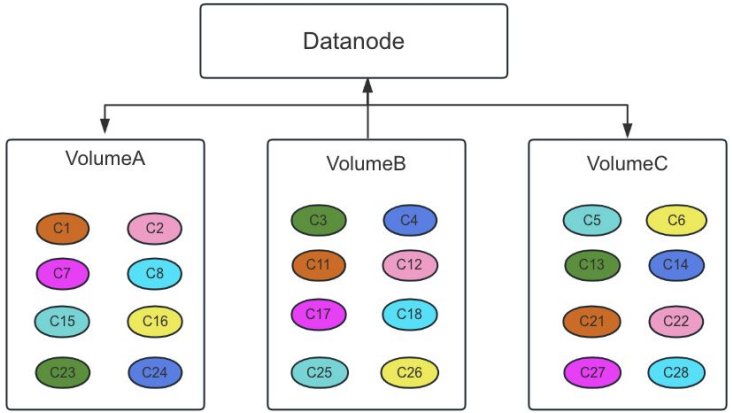
- Three replica container, container replica will spread across one rack
- Erasure coding container, container replica will spread across (data + parity) racks



Background - Datanode Wise Container Distribution

Datanode is responsible for distribute container across volumes

- RoundRobinVolumeChoosingPolicy
- CapacityVolumeChoosingPolicy



Motivation

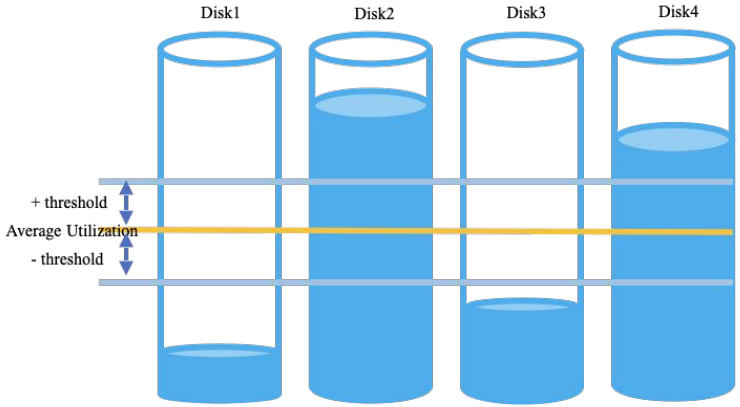
Make sure data evenly spread between disks on a Datanode

Unevenly data spread due to

- New disks added to expand datanode storage space
- Broken old disks replaced with new disks
- Massive block deletion causes disk utilization uneven

Volume Data Density

- TotalCapacity = ∑(diskCapacity)
- TotalFree = ∑(diskFreeSpace)
- AverageUtilization = (TotalCapacity - TotalFree) / TotalCapacity
- VolumeUtilization = (diskCapacity - diskFree) / diskCapacity
- VolumeDensity = VolumeUtilization - AverageUtilization



Data spread uneven when VolumeDensity is larger than threshold

Volume Data Density Example

	Disk 1	Disk 2	Disk 3	Disk 4
capacity	200GB	300GB	350GB	500GB
used	100GB	76GB	300GB	475GB
free	100GB	224GB	50GB	25GB
usedRatio	0.5	0.25	0.85	0.95
volume Data Density	0.2	0.45	-0.15	-0.24

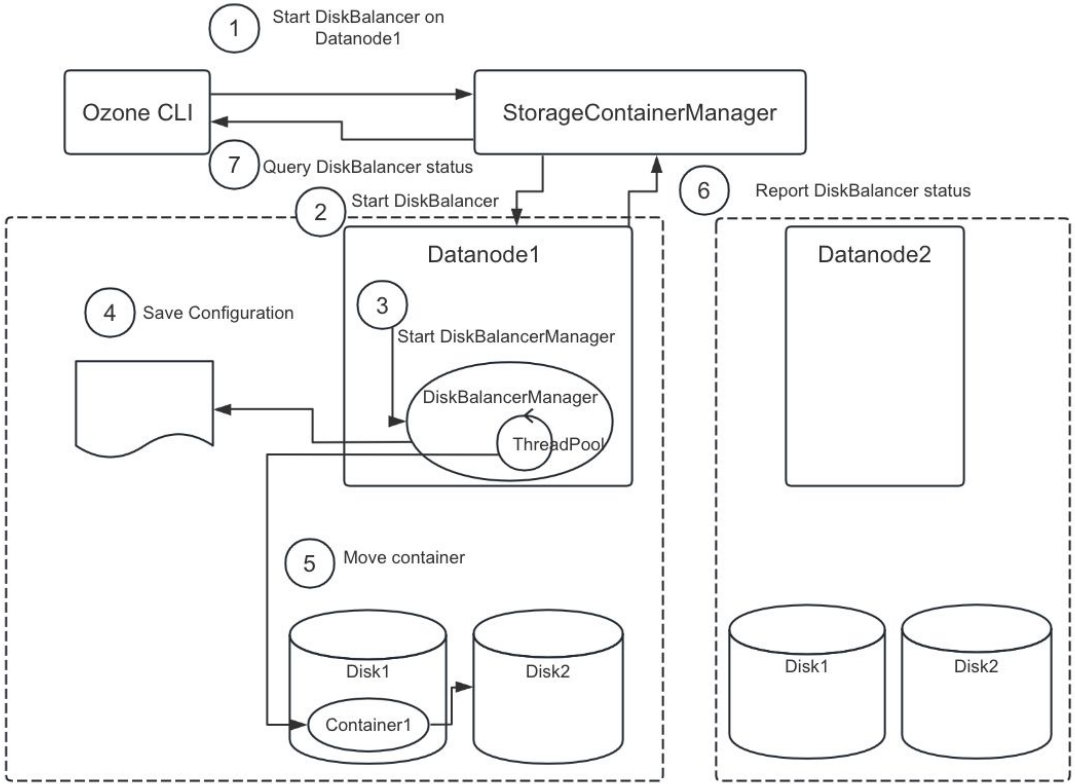
TotalCapacity = 200 + 300 + 350 + 500 = 1350GB

TotalFree = 100 + 224 + 50 + 25 = 399GB

idealUsage = (TotalCapacity - TotalFree) / TotalCapacity = 951 / 1350 = 0.7

densityOfDisk1 = idealUsage - usedRatio Of Disk 1 = 0.7 - 0.5 = 0.2

Process Flow



Configurations

- shouldRun (default : false)
- threshold (default : 10.0%)
- parallelThread (default : 5)
- bandwidth (default : 10MB/sec)
- stopAfterDiskEven (default : true)

COMMUNITY
THE ASF CONFERENCE
CODE

CLI - ozone admin datanode diskbalancer

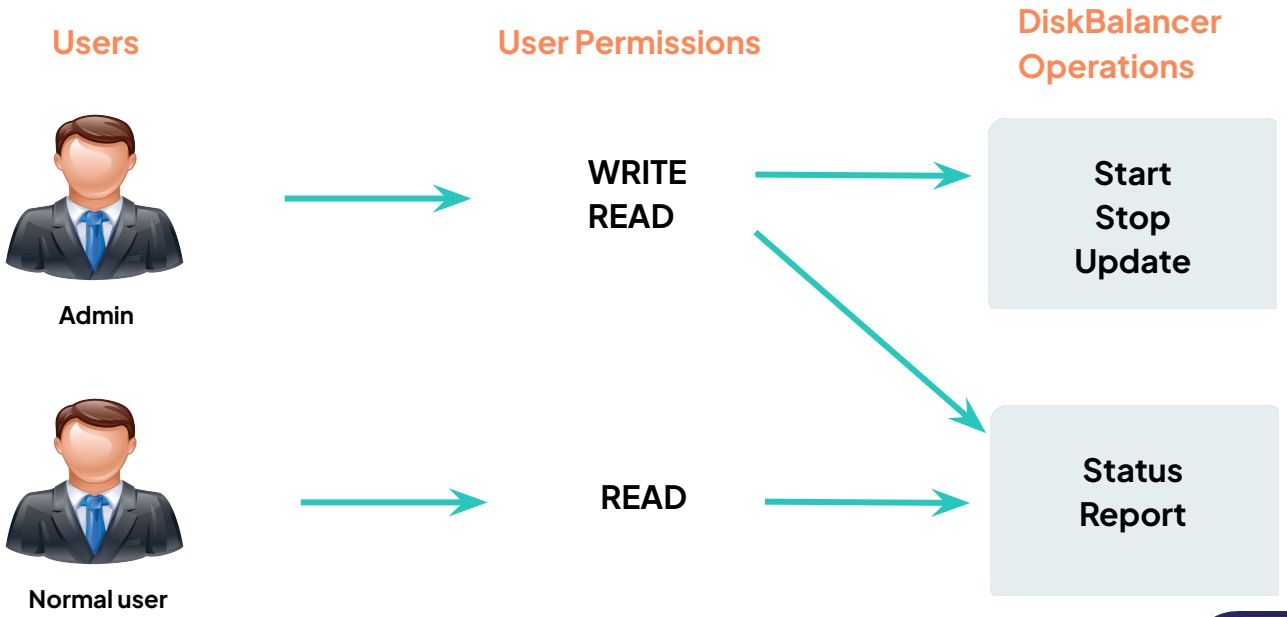
- Start - start all or specified Datanodes
- Stop - stop all or specified Datanodes
- Update - dynamic change configuration on all or specified Datanodes
- Report - report Datanodes volume density
- Status - show current configuration, tasks statistics

DiskBalancer Status

```
sh-5.1$ ozone admin datanode diskbalancer status
Status result:
Datanode                          Status    Threshold(%)  BandwidthInMB  Threads  SuccessMove  FailureMove  BytesMoved(MB)  EstBytesToMove(MB)  EstTimeLeft(min)
ozone-datanode-2.ozone_default    RUNNING   0.0001        10             5        9            0            9386            0                   0
ozone-datanode-1.ozone_default    RUNNING   0.0001        10             5        5            4            5207            1090                2
ozone-datanode-3.ozone_default    RUNNING   0.0001        10             5        6            4            6067            42                  1
ozone-datanode-5.ozone_default    RUNNING   0.0001        10             5        8            2            8207            917                 2
ozone-datanode-4.ozone_default    RUNNING   0.0001        10             5        8            2            8274            131                 1

Note: Estimated time left is calculated based on the estimated bytes to move and the configured disk bandwidth.
```

CLI Permission

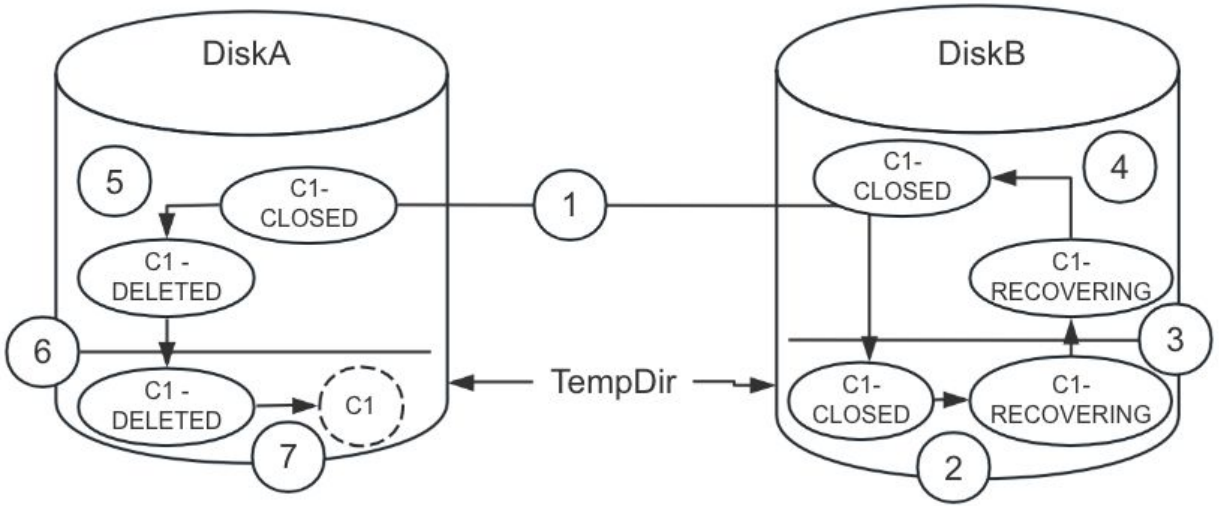


Configuration File

```
$ cat /data/metadata/diskBalancer.info
!!org.apache.hadoop.ozone.container.diskbalancer.DiskBalancerYaml$DiskBalancerInfoYaml {
 bandwidthInMB: 10,
 operationalState: RUNNING,
 parallelThread: 10,
 stopAfterDiskEven: true,
 threshold: 10.0,
 version: 1
}
```

diskBalancer.info under directory defined by “hdds.datanode.disk.balancer.info.dir” or fallback to “ozone.metadata.dirs”

Move Process



Metrcis

- Task execution count
- Task execution time
- Bytes balanced by Task

DiskBalancer Service Metrics


```
}, {
  "name" : "Hadoop:service=HddsDatanode,name=DiskBalancerServiceMetrics",
  "modelerType" : "DiskBalancerServiceMetrics",
  "tag.Context" : "dfs",
  "tag.Hostname" : "c695c629e1a3",
  "FailureCount" : 2,
  "IdleLoopExceedsBandwidthCount" : 1,
  "IdleLoopNoAvailableVolumePairCount" : 0,
  "MoveFailureTimeNumOps" : 2,
  "MoveFailureTimeAvgTime" : 0.0,
  "MoveSuccessTimeNumOps" : 8,
  "MoveSuccessTimeAvgTime" : 0.0,
  "RunningLoopCount" : 3,
  "SuccessBytes" : 8675917824,
  "SuccessCount" : 8
```

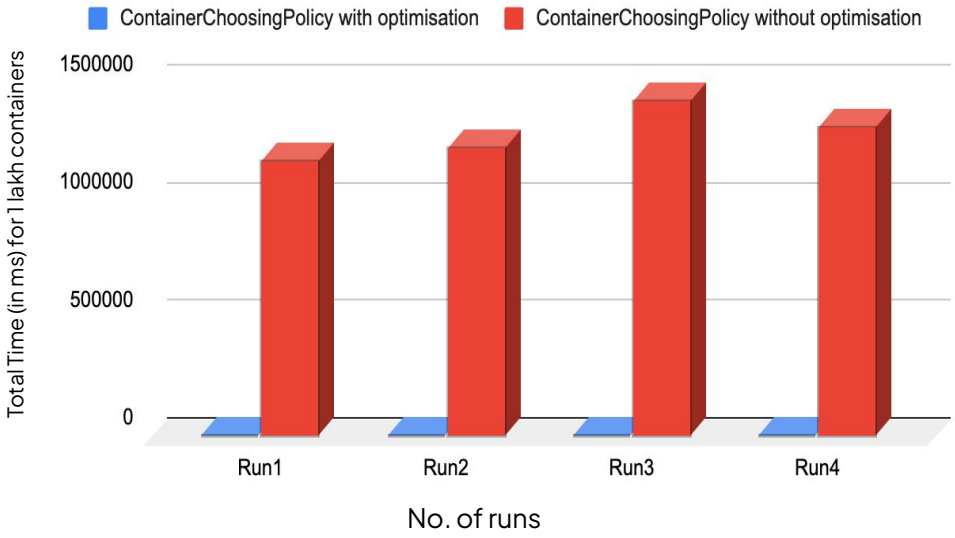
Default Configurations

- hdds.datanode.disk.balancer.should.run.default false
- hdds.datanode.disk.balancer.volume.density.threshold 10
- hdds.datanode.disk.balancer.parallel.thread 5
- hdds.datanode.disk.balancer.max.disk.throughputInMBPerSec 10
- hdds.datanode.disk.balancer.info.dir default empty, fall back to ozone.metadata.dirs
- hdds.datanode.disk.balancer.service.interval 60s
- hdds.datanode.disk.balancer.service.timeout 300s
- hdds.datanode.disk.balancer.volume.choosing.policy
  org.apache.hadoop.ozone.container.diskbalancer.policy.DefaultVolumeChoosingPolicy
- hdds.datanode.disk.balancer.container.choosing.policy
  org.apache.hadoop.ozone.container.diskbalancer.policy.DefaultContainerChoosingPolicy
- hdds.datanode.disk.balancer.stop.after.disk.even true

Micro-Bechmark

ContainerChoosingPolicy decides which container to move from an over-utilized disk

Performance Result



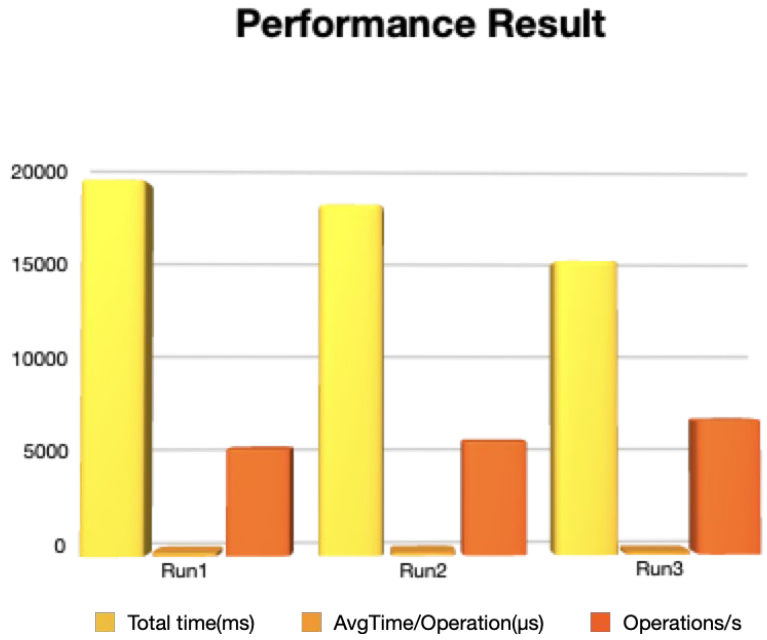
Total Volumes : 20
Total Containers : 100000

Total time to choose one container without optimization : **11 ms**

Total time to choose one container with optimization : **0.02 ms**

Micro-Bechmark

VolumeChoosingPolicy decides source volume, and destination volume of a container to move



Total Volumes : 20
Concurrency(threads) : 10
Total Operations: 100000

Average time to choose one volume pari:
0.17 ms

Throughput of choose volume : **5926 ops/s**

Reference Links

- Disk Balancer Feature
- Container Balancer Feature

COMMUNITY
THE ASF CONFERENCE
CODE

Thanks

sammichen@apache.org
yiyang0203@apache.org