HDDS Container Balancer

Background

When an existing Ozone cluster is nearly full, we have to add more datanodes into the Ozone cluster, but there are two issues we must face.

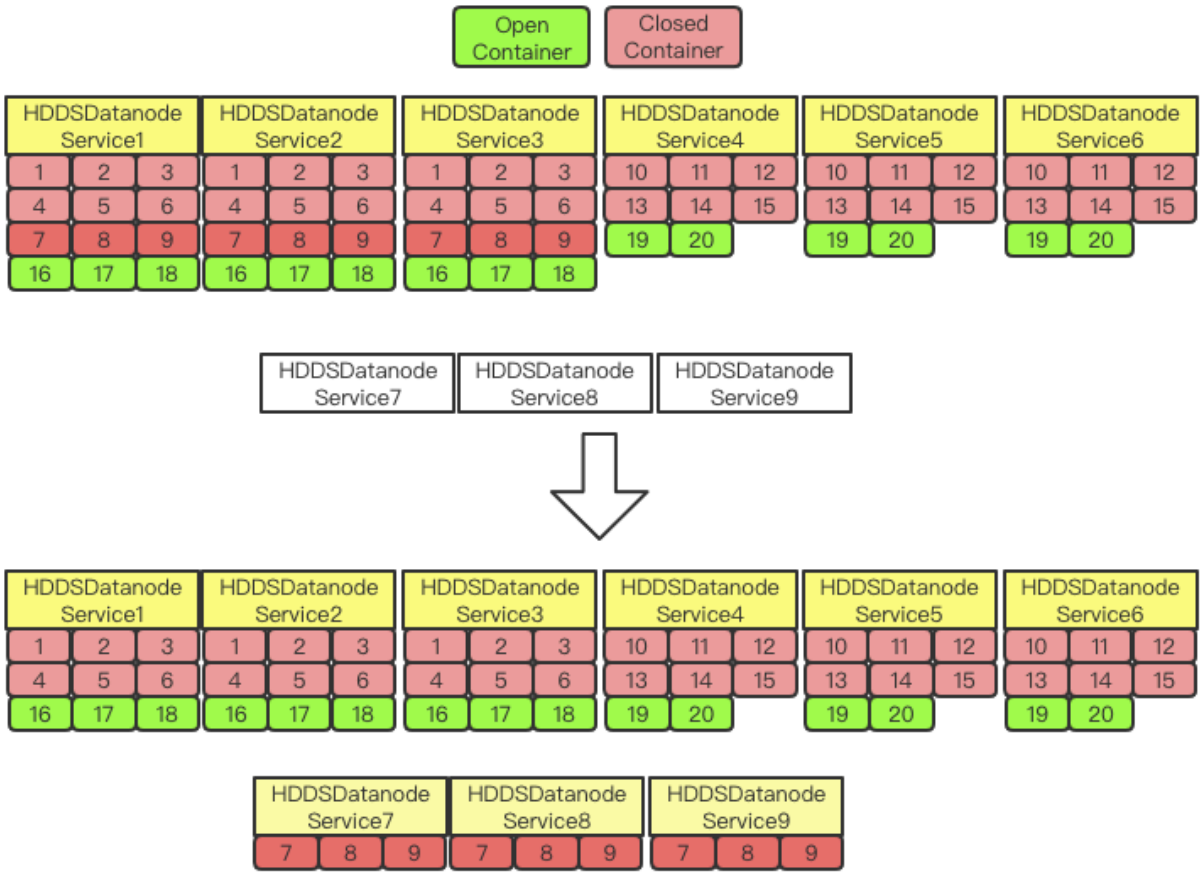
- When new allocated container requests come, SCM should better choose the datanodes in low usage, if not, the performance will get poor.[**Not the goal of this design doc**]
- For read request, the existing datanodes stored lots of blocks, so they are responsible for serving the read request and supply the data stream service, meanwhile, the new coming datanodes can help nothing.

If we have a balancer tool just like hdfs balancer, we can move the block or container from some high usage datanodes to low, I think this is one of necessary tools for Ozone.

Related Jira

HDDS-4656

Approach



Let's say we have 6 existing datanodes in the current Ozone cluster, red block stand for closed container, green block stand for open container, the number in the block means the container id.

Now, we have 3 new datanodes 7, 8, 9 join this Ozone cluster, the storage becomes non balanced, so we start the SCM container balancer to move containers from higher usage datanodes 1, 2, 3 to lower usage datanodes 7, 8, 9.

Balancers only move the closed container, for an open container, we do nothing. Move container operation is a combine operation, for example, move #7 from datanode1 to datanode7, first, copy the #7 to datanode7, then, remove #7 of datanode1.

Balancer is a new program for Ozone cluster, it is a Ozone client.

Main process logic

- Get Datanode usage from SCM

- Make a plan for which datanodes as source and which datanodes as destination
- Move(Copy + Delete) container replica from source to target datanode, and report to SCM.
- Loop until achieved to balance.

Keep container placement satisfy rack policy

I think we have two ways to move container replicas and keep rack policy.

1. Improve the ReplicationManager and let ReplicationManager handle the container replica placement during balance container.
2. Copy or get a datanode topology from SCM, so the balancer can choose the new target as destination without breaking the rack policy.

Administration Commands

As balancer is a tool for administrator of Ozone cluster, so we should provide a batch of command line tools for users to execute balancer.

ozone admin balancer
[-threshold <threshold>]
[-exclude [-f <hosts-file> | <comma-separated list of hosts>]]
[-include [-f <hosts-file> | <comma-separated list of hosts>]]
[-source [-f <hosts-file> | <comma-separated list of hosts>]]
[-asService]

If you want to run Balancer as a long running service, please start Balancer using -asService parameter with daemon-mode.

Independent-program vs service-in-SCM

There would be pros and cons if we put the balancer as a service into the SCM.

Pros.

- It would be easier to use and sync with ReplicationManager for balancing the containers.

- If needed the balancing commands can be provided as part of the admin command line.
- It can use the rack aware placement policy in SCM to provide default balancing features.

Cons

- It would affect the SCM core logic
- Hard to implement and update the balancer related code.
- Make SCM fat if we put lots of features into it while some of these features can be an independent tool.

Follow-up work

- Balancer Metrics & UI
- Speed limitation

Reference

https://issues.apache.org/jira/browse/HDFS-13783

https://issues.apache.org/jira/browse/HDFS-15729

http://hadooptutorial.info/hdfs-rebalance/

https://techtalks.tech/knowledge-base/hdfs-balancer/

https://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-hdfs/HdfsUserGuide.html#Balancer

https://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-hdfs/HDFSCommands.html#balancer