Bulk Load Replication in HBASE

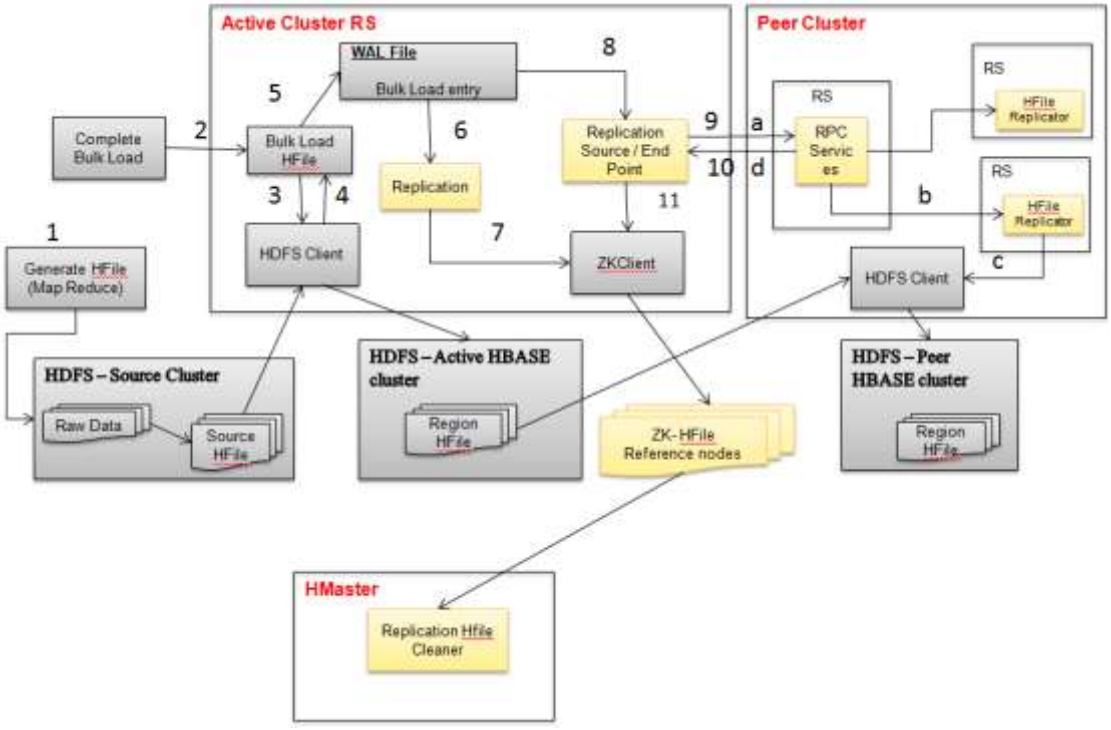
Ashish Singhi & Bhupendra Kumar Jain
Huawei Technologies
August 2015
JIRA: HBASE-13153

1 Requirement Scenario

Hbase Replication currently doesn't replicate the bulk loaded hfile data.

Currently replication happens thru replaying the WAL entries from Active cluster to peer cluster. Since Bulk Load process by-passes the WAL edits, so bulk load data will not be replicated.

2 High Level Solution



- HDFS- Source cluster: Contains RAW data and generated hfiles
- HDFS- Active HBASE cluster: HDFS cluster used by active HBASE cluster
- HDFS- Peer HBASE cluster: HDFS cluster used by Peer HBASE cluster

Replication module will be enhanced to support bulk loaded hfiles also. The bulk load events are already captured as wal entry in the wal file (refer HBASE-11567). The wal entry for bulk load event contains the loaded hfile info. If this feature is enabled, Replication scope will be added for these wal entries.

Active Hbase RS will also send these wal entries to peer cluster. Peer cluster will

i) Read these wal entries

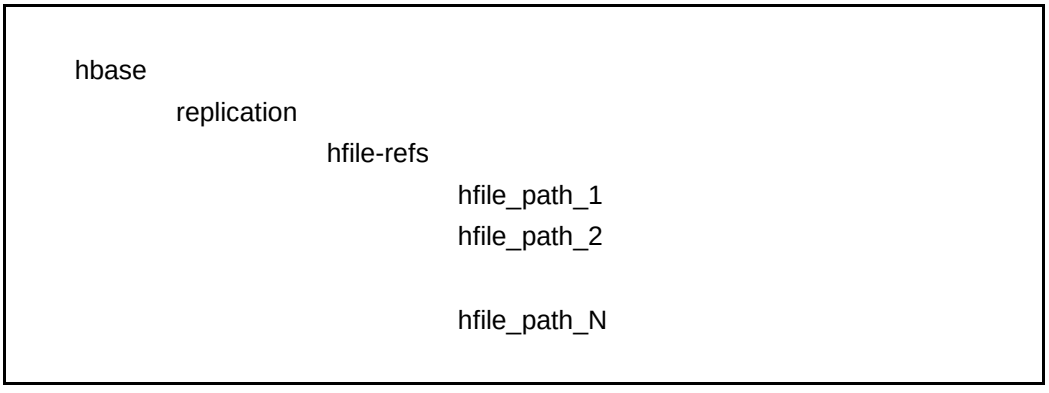
ii) Construct the hfile path pointing to source hdfs cluster and

iii) Load these hfiles [Load mechanism will be similar as complete bulk load]

Cyclic Replication detection: Source cluster ids will also be persisted as part of bulk load wal entry. This source cluster ids will be used for cycle detection. [Same as existing design]

Replication Hfile Cleaner: If compaction/merge /split happen, then the hfile will be moved to archive folder. This will be cleaned up by Hfile cleaner periodically. To avoid this before replication completes, replication module will maintain these bulk loaded hfile paths into ZK. Cleaner will not clean the hfiles till its entry is found in ZK. Once the hfile is replicated successfully, the ZK entry will be deleted by replication module.

ZK Replication hfile references



Other approaches to avoid hfile cleanup:

a) Maintain the hfile reference in a new system table. [Can be thought along with HBASE-10295]

b) Master cleaner thread, sends the call to each RS to fetch the bulk loaded hfile paths from its wals. [Reading wal on each cleanup call might be heavy, So ignored this option]

Failover Handling

All failover scenarios are already handled by existing replication module. Same will hold good for hfile replication also.

3 Constraints and Limitations

- If data in Visibility Labels table is different in active and peer cluster, then Visibility expressions will fail during scan in peer cluster.

  Example:

  Active cluster visibility table has entry such as: SECRET 1

  Peer cluster visibility table has entry such as: SECRET 2

  Bulk loaded Hfile of Active cluster will contain the value as "1", this hfile will be directly copied to Peer cluster during replication, So peer cluster hfile will still have entry as "1". During Scan there will not be any matching entry corresponding to "1" in Peer cluster Visibility Tables. So scan will fail to parse this visibility expression.
- Peer cluster will require the Read permission for active HDFS cluster.
- Peer cluster must have Compression codec library used in source cluster for hfile compression.

4 Performance Scenarios

- Hfile Data Transfer: To reduce the amount of data sent over the network, source hfile should be compressed.
- Table Split: In case, Source and Peer cluster have different split points for table regions, then hfile will be split before loading into peer cluster. It will slow down the replication process. To avoid this, Source and Peer cluster table should have same

split points.

- Bandwidth control: The RPC call from hbase contains only hfile Paths. So there is no need to control the data limit in hbase. The main network data traffic is generated by copy of hfiles from one HDFS cluster to another.

5 Backward Compatibility

- No changes in existing public APIs.

6 Security

- Peer cluster will require Read permission for active HDFS cluster.
- Other security configuration remains same as existing.
- Audit Logs: Audit logs of Peer cluster RS will contain the replication audit info.

7 Metrics

All below metrics are at active RS level.

- Number of bulk load entries shipped

All below metrics are at peer RS level.

- Number of bulk load entries applied

8 Configuration

- hbase.bulkload.replication.enabled=false/true [Default : false]

9 Interfaces (API)

No external interfaces