Online Merge

HBase@Ali

zjusch@163.com

Why Merge

- Region hole or region overlap, can’t be fix by hbck
- Region become empty because of TTL and not reasonable Rowkey design
- Region is always empty or very small because of presplit when create table
- Too many empty or small regions would reduce the system performance(e.g. mslab)

Current Merge Tool

- org.apache.hadoop.hbase.util.Merge Should shutdown HBase cluster
- org.apache.hadoop.hbase.util.HMerge should disable table and can’t specify regions to merge
- Both is offline merge, and if Exception is thrown in the process of merging, both are not able to redo, causing a dirty data

Motivation of Online Merge

- For online system, shutdown cluster or disable table won’t be accept
- Even if we could accept shutdown cluster sometime, it is a short time, do lots of merge work would increase maintenance work

Target of Online Merge

- When merging, only two regions are offline(not serving by any regionserver)
- Merge is a transaction
- Merge can be canceled if it is unavailable(e.g. one of the region become a parent region), region will online again if it should so
- Once we do the merge work in the HDFS, we could redo it until successful if it throws exception or abort or server restart, but couldn't be rolled back
- Merge Transaction is executed on the HMaster

Implement Logic of Online Merge

- For example, merge regionA and regionB into regionC
- 1.Offline the two regions A and B
- 2.Merge the two regions in the HDFS(Create regionC's directory, move regionA's and regionB's file to regionC's directory, delete regionA's and regionB's directory)
- 3.Add the merged regionC to .META.
- 4.Assign the merged regionC

Trouble in Implement

- 1.Region not served by any regionserver != Region is offline(It would be online again through ServerShutdownHandler or cluster restart or enabling table)
- 2. Region is offline == Region belongs to disabled table
- 3.Region is offline after region is split
- 4.Be able to redo the merge after error or abort or restart by any cases

Framework of Online Merge

- Use zookeeper to record the transaction journal state, make redo easier
- Use zookeeper to send/receive merge request
- Merge transaction is executed on the master
- Support calling merge request through API or shell tool

Merge Components

ZK Node
(/hbase/merge)

Record Merge Journal State
/hbase/merge/T#A#B#true
/hbase/merge/T#C#D#false
T=table name; A,B,C,D = region 's encoded name;
true,false=whether a force merge (A non-force merge will only merge adjacent regions)
TransactionName=T#A#B#true , represent the only merge transaction, Its data on ZK represent the transaction journal state:
1. *CREATE_MERGE;* 2. *OFFLINE_REGION*;
3.EXECUTE_MERGING;
4.COMPLETE_MERGING/CANCEL_MERGING;

Merge Manager

1.Reveive merge request through watching ZK;
2.Generate and submit a merge request
3.Manage merge transaction and merging regions

MergeTransaction

1.A Merge Request execute a MergeTransaction in Thread-Pool
2. MergeTransaction do the whole process of merge

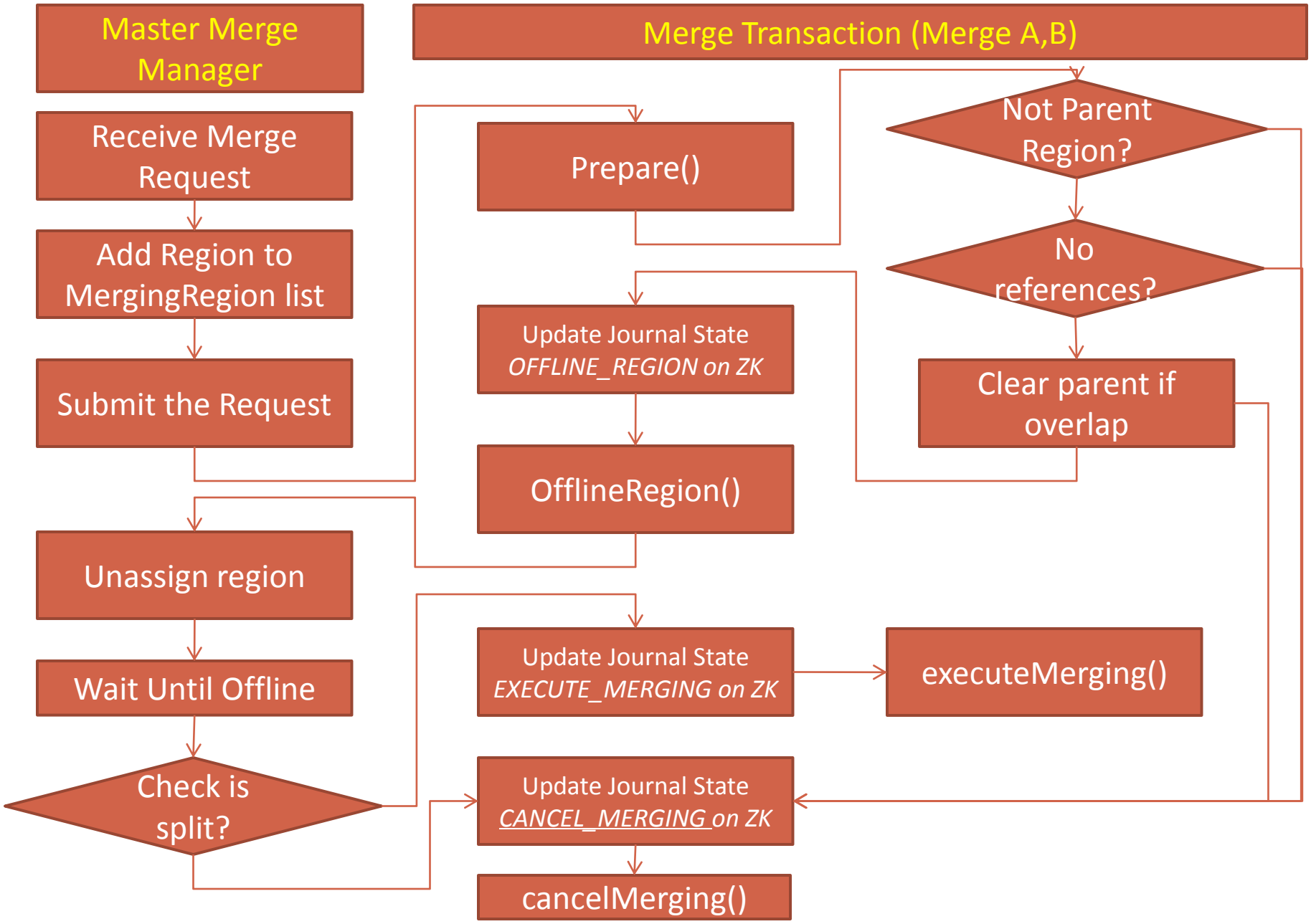
Some Notes

- Region in merging = region is disabled, so it won't be assigned again
- MergeManager construct merging region list before assign regions when master start
- A non-force merge will skip the merge request if two regions are not adjacent
- Two specified regions will be merged into a region with fixed region name
- Merge Transaction could be executed in several threads, but concurrent merge for one region won't happen

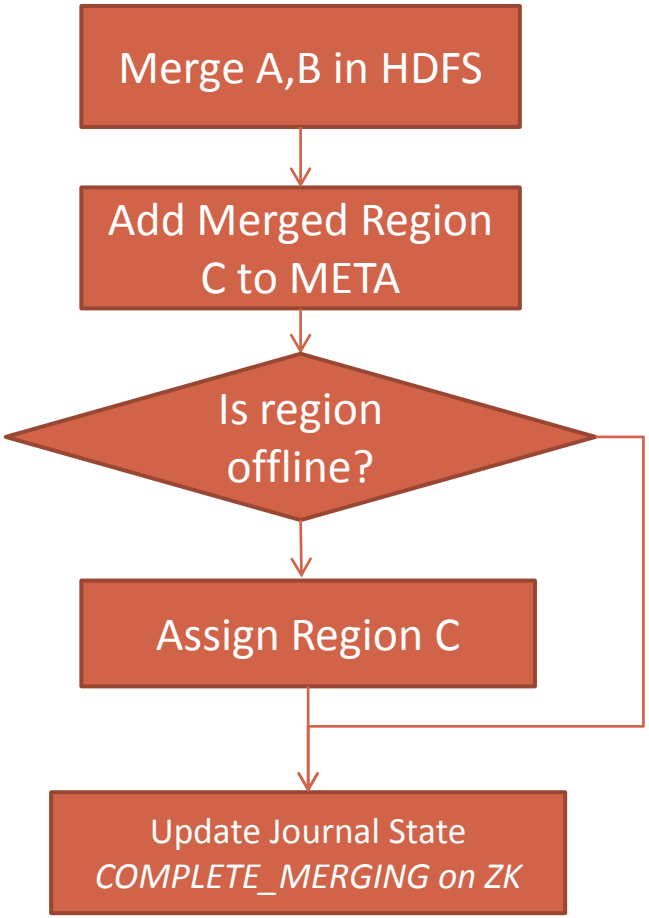
Overall Process of Merge

- MergeManager receive a merge request or continue last remaining merge requests when master restart via ZK
- According to transaction journal state(get from the ZK node's data if a redo merge), start merging work
- 1. *CREATE_MERGE;*
  - prepare()(Check this merge is available)
- 2. *OFFLINE_REGION*;
  - offlineRegion()
- 3.EXECUTE_MERGING;
  - executeMerging()(Merge the fileS in HDFS, Add merged region to META)
- 4.COMPLETE_MERGING/CANCEL_MERGING
  - completeMerging()/ cancelMerging()(Remove useless data)

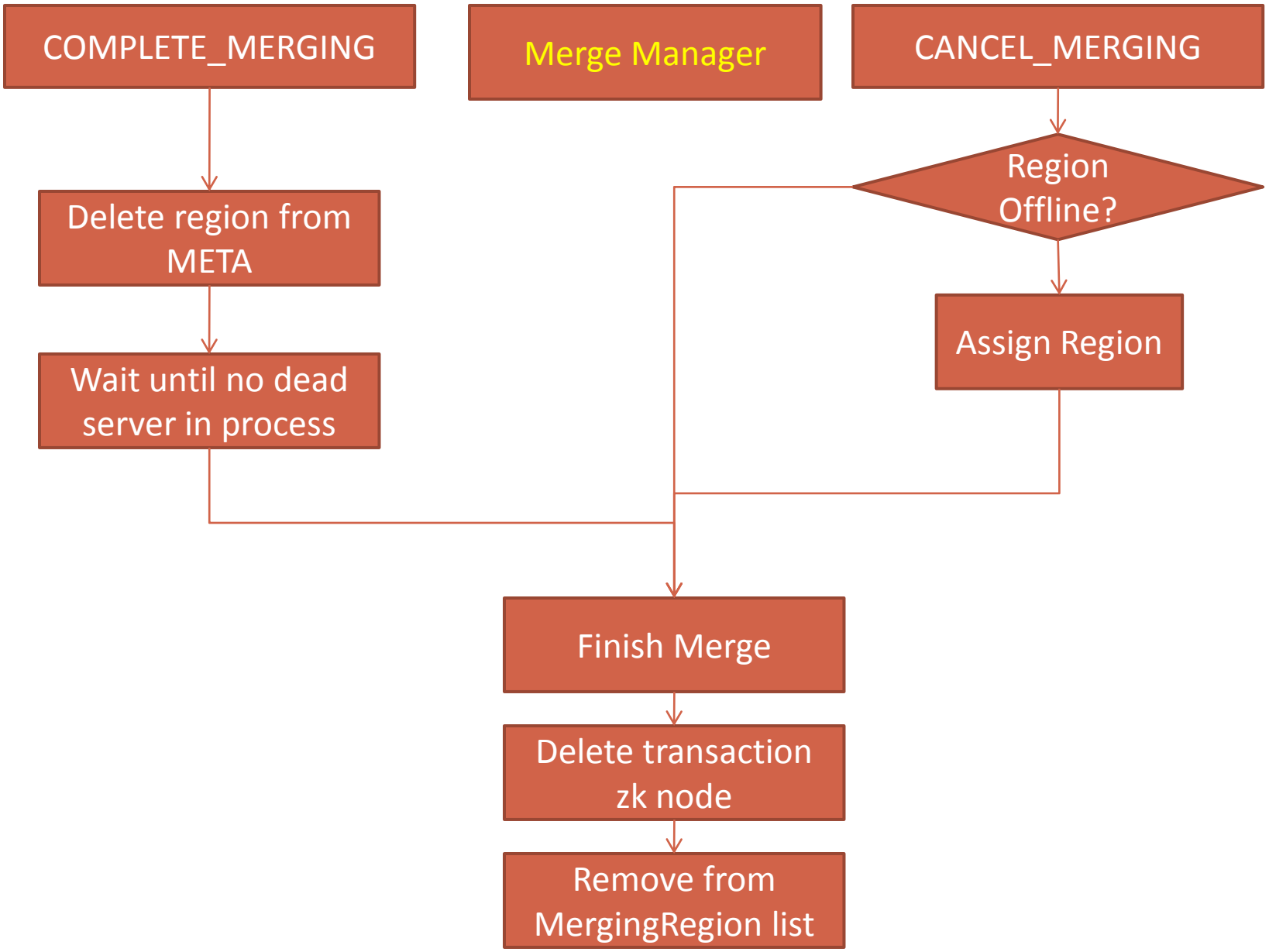
Step1&2:Prepare&Offline Region



Step3:Execute Merge Action



Step4:Complete/Cancel Merge



How To Use

- New conf
  - hbase.master.thread.merge merge Thred Num, 1 as default
- Send merge request
  - Tool: Usage: bin/hbase org.apache.hadoop.hbase.util.OnlineMerge [-force] [-async] [-show] <table-name> <region-encodedname-1> <region-encodedname-2>
  - API: static void MergeManager# createMergeRequest(ZooKeeperWatcher zkw, String tableName, String regionAEncodedName, String regionBEncodedName, boolean force)

Change In Core Code

- Only less change in the master
  - Remove regions in merge from assign list when master start
  - Remove regions in merge from assign list when process ServerShutdownHandler
  - Remove regions in merge from assign list when enabling table
  - Unassign region again if region in merging when handle opened region event
  - Don’t assign region if region in merging when handle closed region event