MEP 16 -- Compaction

Current state: Under Discussion

ISSUE: https://github.com/milvus-io/milvus/issues/9530

PRs:

Keywords: datacoord, segment, compaction

Released:

Summary

Milvus needs a compaction mechanism to merge small segments and remove deleted rows to save disk space.

Motivation

There are many ways to generate small segments:

1. DataCoord will auto flush a segment when it is opened for a long time(eg. 24hours)
2. Users may call flush manually

And deleted rows should be removed after they are not used anymore.

So we have 2 targets:

1. Merge small segments to improve query efficiency
2. Remove deleted rows to save disk space

Public Interfaces

We will add a compaction interface in sdk to start a compaction.

Design Details

Some preset conditions:

1. We do compaction at channel&partiton level. Because a segment is generated at channel&partiton level.
2. Delta log and insert log is at segment level.
3. Delta log and insert log in time-travel range should be saved.
4. Segment has a max size(limited by memory size).

We divide compaction task to 2 phases.

We merge insert and delta logs in the first phase:

1. Considering time-travel, we only merge segments outside the time-travel range.
2. When to trigger a compaction:
    a. After a flush
    b. The time interval from the last compaction is greater than max_ compaction_ interval
    c. call compaction manually
3. How to choose segments:
    a. The size of all delta logs is bigger than max_delete_binlog_size
    b. deleted rows / total rows >= compaction_delta_binlog_ratio

We merge segments in the second phase:

1. The time period of time travel may be very long, such as dozens of days, so it is still necessary to merge small segments within the scope of time travel.
2. When to trigger a compaction:
    a. After a segment flush, if the total number of segments less than 1/2*max_segment_size at channel&partition level exceeds the compaction_ segment_ num_ threshold.
    b. The time interval from the last compaction is greater than max_ compaction_ interval
    c. call compaction manually
3. How to choose segments: greedy algorithm‘

Some details

1. Only merge flushed segments
2. We choose the max dml position of merged segments as the dml position of the new generated segment.

Test Plan