Item 23
git_comments:

git_commits:

1. **summary:** Hive 22255 : Hive don't trigger Major Compaction automatically if table contains only base files (Raj Kumar Singh, reviewed by Karen Coppage)
   message: Hive 22255 : Hive don't trigger Major Compaction automatically if table contains only base files (Raj Kumar Singh, reviewed by Karen Coppage) Closes (#1085)

github_issues:

github_issues_comments:

github_pulls:

1. **title:** Hive 22255 : Hive don't trigger Major Compaction automatically if table contains only base files
   body: Creating the PR to fix the multi base-files seems in the table directory after Insert overwrite operations. As a fix, I will be checking doing changes in the initiator thread where it will check for the compaction eligibility. 1. Initiator thread will check for the if there are no delta and obsolete files containing the base then it will put the table in compaction queue. 2. TxnStore will be having one more method to request cleanup.

github_pulls_comments:

1. @klcopp I have created the test case, pushing the table/part into the compaction queue will help Worker#isEnoughToCompact to mark it clean
2. @rajkrrsingh Thanks for creating the test case! [the isEnoughToCompact check will not mark it clean if there are obsolete directories. markCompacted puts it into "ready for cleaning" state] (https://github.com/apache/hive/blob/master/ql/src/java/org/apache/hadoop/hive/ql/txn/compactor/Worker.java#L501-L507) I think taking this route is much simpler and less error-prone than expanding the TxnHandler API. If you take it, you'll need to add startWorker() to the test.
3. @rajkrrsingh TestPigHBaseStorageHandler is causing issues on every test rerun, even though the changes that broke it were reverted. I guess the tests don't check out a fresh master branch every time? Anyway, if you close and reopen this PR the tests should pass.

github_pulls_reviews:

1. Why do we need this?
2. Maybe use findAny() instead of count()?
3. Unnecessary parenthesis and also I think dir.getObsolete().size() check is not needed, since multiBase is true
4. Meybe we should move this select out to a different method, since this is the copy of the one used in compact(). Also I would add state cleaning as well. We do not want 2 cleaners running parallel
5. I think, we really want to ignore this, so we would like to return here.
6. Could we do this in a try with resource construct?
7. Again this is very similar that we have in compact(), we might to create a new method for it and reuse.
8. Also using a prepared statement would be nice, I think
9. Can we use preparedstatement here as well?
10. try with resource would be nice here too
11. If there is some exception before in the finally, this unlockInternal will not be called. Isn't this a problem?
12. Further clarified thoughts: - It would be good to have a single method inserting to the COMPACTION_QUEUE table - We should use a preparedstatement for this so JDBC execution could be faster - TxnHandler.compact() should check for only INITIATED/WORKING status - it might still worth tho start a new compaction, even if the cleanup not finished yet - TxnHandler.requestCleanup() should check for INITIATED/WORKING/READY_TO_CLEAN status (the first 2 should not be there anyway), so we do not queue multiple compactions for the same table/partition Thanks, Peter
13. without setting explicitly here, insert query find it null and skip the [here] (https://github.com/rajkrrsingh/hive/blob/3d12959339b22becee0aa986852049b46867f016/standalone-metastore/metastore-server/src/main/java/org/apache/hadoop/hive/metastore/txn/TxnHandler.java#L5288)
14. incorporated the suggested change.
15. incorporated the suggested change.
16. incorporated the suggested change.
17. moved this logline to debug and return from here.
18. incorporated the suggested change.
19. incorporated the suggested change.

20. incorporated the suggested change for stmt and pst. try-resource with dbConn make code clumsy with so many nested try-catch so I skipped it.
21. unlockInternal is mostly applicable for derby so I think it should not create the problem.
22. Nit: I think this is misleading, and unnecessary since we have already logged the values of deltaSize and multiBase.
23. opportunity: add: startWorker(); Assert.assertEquals("ready for cleaning",rsp.getCompacts().get(0).getState());
24. nit: no newline at end of file

jira_issues:

jira_issues_comments: