Item 48
git_comments:

1. Decode a copy of the accumulator when necessary.
2. Encode a version of the accumulator if it is in multiple windows. The combineFn is able to mutate the accumulator instance and this could lead to incorrect results if the same instance is merged across multiple windows so we decode a new instance as needed. This prevents issues during merging of accumulators.

git_commits:

1. **summary:** Merge pull request #13061: [BEAM-11050] Duplicate accumulator if it appears in multiple windows.
   message: Merge pull request #13061: [BEAM-11050] Duplicate accumulator if it appears in multiple windows.

github_issues:

github_issues_comments:

github_pulls:

1. **title:** [BEAM-11050] Duplicate accumulator if it appears in multiple windows.
   body: Accumulators can be mutated during merging by the combine fn so we must ensure that we use a unique instance of the accumulator per window. ---------- Thank you for your contribution! Follow this checklist to help us incorporate your contribution quickly and easily: - [ ] [**Choose reviewer(s)**](https://beam.apache.org/contribute/#make-your-change) and mention them in a comment (`R: @username`). - [ ] Format the pull request title like `[BEAM-XXX] Fixes bug in ApproximateQuantiles`, where you replace `BEAM-XXX` with the appropriate JIRA issue, if applicable. This will automatically link the pull request to the issue. - [ ] Update `CHANGES.md` with noteworthy changes. - [ ] If this contribution is large, please file an Apache [Individual Contributor License Agreement](https://www.apache.org/licenses/icla.pdf). See the [Contributor Guide](https://beam.apache.org/contribute) for more tips on [how to make review process smoother](https://beam.apache.org/contribute/#make-reviewers-job-easier). Post-Commit Tests Status (on master branch) ---------- ---------- Lang | SDK | Dataflow | Flink | Samza | Spark | Twister2 --- | --- | --- | --- | --- | --- | --- Go | [![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Go/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Go/lastCompletedBuild/) | --- | [![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Go_VR_Flink/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Go_VR_Flink/lastCompletedBuild/) | --- | [![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Go_VR_Spark/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Go_VR_Spark/lastCompletedBuild/) | --- Java | [![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Java/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Java/lastCompletedBuild/) | [![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_Dataflow/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_Dataflow/lastCompletedBuild/)
[![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_Dataflow_Java11/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_Dataflow_Java11/lastCompletedBuild/) | [![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_Flink/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_Flink/lastCompletedBuild/)
[![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_Flink_Java11/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_Flink_Java11/lastCompletedBuild/)
[![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Java_PVR_Flink_Batch/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Java_PVR_Flink_Batch/lastCompletedBuild/)
[![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Java_PVR_Flink_Streaming/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Java_PVR_Flink_Streaming/lastCompletedBuild/) | [![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_Samza/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_Samza/lastCompletedBuild/) | [![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_Spark/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_Spark/lastCompletedBuild/)
[![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Java_PVR_Spark_Batch/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Java_PVR_Spark_Batch/lastCompletedBuild/)
[![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_SparkStructuredStreaming/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_SparkStructuredStreaming/lastCompletedBuild/) | [![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_Twister2/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Java_ValidatesRunner_Twister2/lastCompletedBuild/) Python | [![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Python36/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Python36/lastCompletedBuild/)
[![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Python37/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Python37/lastCompletedBuild/)
[![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Python38/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Python38/lastCompletedBuild/) | [![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Py_VR_Dataflow/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Py_VR_Dataflow/lastCompletedBuild/)
[![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Py_VR_Dataflow_V2/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Py_VR_Dataflow_V2/lastCompletedBuild/)
[![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Py_ValCont/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Py_ValCont/lastCompletedBuild/) | [![Build Status](https://ci-beam.apache.org/job/beam_PreCommit_Python_PVR_Flink_Cron/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PreCommit_Python_PVR_Flink_Cron/lastCompletedBuild/)
[![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Python_VR_Flink/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Python_VR_Flink/lastCompletedBuild/) | --- | [![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_Python_VR_Spark/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_Python_VR_Spark/lastCompletedBuild/) | --- XLang | [![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_XVR_Direct/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_XVR_Direct/lastCompletedBuild/) | --- | [![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_XVR_Flink/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_XVR_Flink/lastCompletedBuild/) | --- | [![Build Status](https://ci-beam.apache.org/job/beam_PostCommit_XVR_Spark/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PostCommit_XVR_Spark/lastCompletedBuild/) | --- Pre-Commit Tests Status (on master branch) ---------- ---------- --- |Java | Python | Go | Website | Whitespace | Typescript --- | --- | --- | --- | --- | --- | --- Non-portable | [![Build Status](https://ci-beam.apache.org/job/beam_PreCommit_Java_Cron/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PreCommit_Java_Cron/lastCompletedBuild/) | [![Build Status](https://ci-beam.apache.org/job/beam_PreCommit_Python_Cron/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PreCommit_Python_Cron/lastCompletedBuild/)
[![Build Status](https://ci-beam.apache.org/job/beam_PreCommit_PythonLint_Cron/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PreCommit_PythonLint_Cron/lastCompletedBuild/)
[![Build Status](https://ci-beam.apache.org/job/beam_PreCommit_PythonDocker_Cron/badge/icon)](https://ci-beam.apache.org/job/beam_PreCommit_PythonDocker_Cron/lastCompletedBuild/)
[![Build Status](https://ci-

beam.apache.org/job/beam_PreCommit_PythonDocs_Cron/badge/icon)](https://ci-beam.apache.org/job/beam_PreCommit_PythonDocs_Cron/lastCompletedBuild/) | [![Build Status](https://ci-beam.apache.org/job/beam_PreCommit_Go_Cron/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PreCommit_Go_Cron/lastCompletedBuild/) | [![Build Status](https://ci-beam.apache.org/job/beam_PreCommit_Website_Cron/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PreCommit_Website_Cron/lastCompletedBuild/) | [![Build Status](https://ci-beam.apache.org/job/beam_PreCommit_Whitespace_Cron/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PreCommit_Whitespace_Cron/lastCompletedBuild/) | [![Build Status](https://ci-beam.apache.org/job/beam_PreCommit_Typescript_Cron/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PreCommit_Typescript_Cron/lastCompletedBuild/) Portable | --- | [![Build Status](https://ci-beam.apache.org/job/beam_PreCommit_Portable_Python_Cron/lastCompletedBuild/badge/icon)](https://ci-beam.apache.org/job/beam_PreCommit_Portable_Python_Cron/lastCompletedBuild/) | --- | --- | --- See [.test-infra/jenkins/README](https://github.com/apache/beam/blob/master/.test-infra/jenkins/README.md) for trigger phrase, status and link of all Jenkins jobs. GitHub Actions Tests Status (on master branch) ---------- [![Build python source distribution and wheels](https://github.com/apache/beam/workflows/Build%20python%20source%20distribution%20and%20wheels/badge.svg?branch=master&event=schedule)](https://github.com/apache/beam/actions?query=workflow%3A%22Build+python+source+distribution+and+wheels%22+branch%3Amaster+event%3Aschedule) [![Python tests](https://github.com/apache/beam/workflows/Python%20tests/badge.svg?branch=master&event=schedule)](https://github.com/apache/beam/actions?query=workflow%3A%22Python+Tests%22+branch%3Amaster+event%3Aschedule) [![Java tests](https://github.com/apache/beam/workflows/Java%20Tests/badge.svg?branch=master&event=schedule)](https://github.com/apache/beam/actions?query=workflow%3A%22Java+Tests%22+branch%3Amaster+event%3Aschedule) See [CI.md](https://github.com/apache/beam/blob/master/CI.md) for more information about GitHub Actions CI.

github_pulls_comments:

1. R: @iemejia
2. Run Spark StructuredStreaming ValidatesRunner
3. Run Java PreCommit
4. Run Java PreCommit
5. **body:** This LGTM but I prefer that @echauchot takes a look before merging because he has been optimizing this code for a while so better to make him aware of the issue and the minor performance hit of the extra encoding needed.
   label: code-design
6. **body:** I would prefer taking the fix and then further optimizing for performance as the implementation I suggested only duplicates when a value is in multiple windows which is uncommon in practice.
   label: code-design
7. **body:** Good point, I suppose if @echauchot has a suggestion or a better way to do this we can improve it in the future, at least this fixes the breakage on tests and it produces correct results.
   label: code-design
8. @lukecwik I don't see why this change is necessary because of 2 reasons: 1. all the validates runner tests including multiple window (eg. sliding windows) already passed. 2. when I wrote this code, I already took some safety mesures about the modification of the (first only) accumulator during the `combineFn.mergeAccumulator` by creating a new first accumulator for each merged window see initial code below and the comment in the code: ``` // merge the accumulators for each mergedWindow ... for (Map.Entry<W, List<Tuple2<AccumT, Instant>>> entry : mergedWindowToAccumulators.entrySet()) { ... // we need to create the first accumulator because combineFn.mergerAccumulators can modify the // first accumulator AccumT first = combineFn.createAccumulator(); Iterable<AccumT> accumulatorsToMerge = Iterables.concat(Collections.singleton(first), accumsAndInstantsForMergedWindow.stream() .map(x -> x._1()) .collect(Collectors.toList())); ... combineFn.mergeAccumulators(accumulatorsToMerge), ... } ```
9. @echauchot The VR tests were breaking on this (I don't know why, maybe the tests were improved). That's the reason why Luke did this PR, it was needed at least for correctness. You can reproduce this by reverting this PR and running the tests: ``` git revert 6264b47afd51d33d95d6c04a2106b4208a89ca41 ./gradlew :runners:spark:validatesStructuredStreamingRunnerBatch ``` produces ``` org.apache.beam.sdk.transforms.CombineTest$WindowingTests > testSlidingWindowsCombine FAILED java.lang.AssertionError at CombineTest.java:1156 Caused by: java.lang.AssertionError at MatcherAssert.java:18 ``` Something odd I noticed is that if you run the single test instance it passes so I am not sure if there is some interleaving issue with other tests. The VR suite of the Structured Streaming Runner has been broken since September 10 also because of this issue and BEAM-11023 too. http://104.154.241.245/d/8N6LVeCmk/post-commits-status-dashboard?refresh=30s&orgId=1
10. thanks @iemejia for the context. Strange `org.apache.beam.sdk.transforms.CombineTest$WindowingTests > testSlidingWindowsCombine ` was passing. So I would prefer to figure out what was changed in the between in the Combine translation. https://issues.apache.org/jira/browse/BEAM-11023 refers to groupByKey so it is unrelated. But I just checked it was passing when the runner was merged to master. I guess we need to dig into the history of commits rather

github_pulls_reviews:

jira_issues:

1. **summary:** AggregatorCombiner reuses mutable accumT across multiple merges leading to incorrect results
   description: Example failure:
   [https://scans.gradle.com/s/lsf5y44b36pyc/tests/:runners:spark:validatesStructuredStreamingRunnerBatch/org.apache.beam.sdk.transforms.CombineTest$Windowing
   The test passes occassionaly and it depends on the order of merge/reduce steps that Spark does. A good run: {noformat} LCWIK merge accum1: [] accum2: [TimestampedValueInMultipleWindows{value=[c], timestamp=1970-01-01T00:00:00.003Z, windows=[[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), [1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), [1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z)], pane=PaneInfo{isFirst=true, timing=EARLY, index=0}}] rval: [TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}] LCWIK merge accum1: [] accum2: [TimestampedValueInMultipleWindows{value=[b], timestamp=1970-01-01T00:00:00.002Z, windows=[[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), [1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), [1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z)], pane=PaneInfo{isFirst=true, timing=EARLY, index=0}}] rval: [TimestampedValueInSingleWindow{value=[b], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] LCWIK merge accum1: [] accum2: [TimestampedValueInMultipleWindows{value=[a], timestamp=1970-01-01T00:00:00.001Z, windows=[[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), [1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), [1969-12-31T23:59:59.999Z..1970-01-01T00:00:00.002Z)], pane=PaneInfo{isFirst=true, timing=EARLY, index=0}}] rval: [TimestampedValueInSingleWindow{value=[a], timestamp=1970-01-01T00:00:00.001Z, window=[1969-12-31T23:59:59.999Z..1970-01-01T00:00:00.002Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] LCWIK reduce value: TimestampedValueInMultipleWindows{value=KV{null, b}, timestamp=1970-01-01T00:00:00.002Z, windows=[[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), [1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), [1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z)], pane=PaneInfo{isFirst=true, timing=EARLY, index=0}} accum: [] result: [TimestampedValueInSingleWindow{value=[b], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] LCWIK reduce value:

TimestampedValueInMultipleWindows{value=KV{null, a}, timestamp=1970-01-01T00:00:00.001Z, windows=[[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), [1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), [1969-12-31T23:59:59.999Z..1970-01-01T00:00:00.002Z)], pane=PaneInfo{isFirst=true, timing=EARLY, index=0}} accum: [] result: [TimestampedValueInSingleWindow{value=[a], timestamp=1970-01-01T00:00:00.001Z, window=[1969-12-31T23:59:59.999Z..1970-01-01T00:00:00.002Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] LCWIK reduce value: TimestampedValueInMultipleWindows{value=KV{null, c}, timestamp=1970-01-01T00:00:00.003Z, windows=[[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), [1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), [1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z)], pane=PaneInfo{isFirst=true, timing=EARLY, index=0}} accum: [] result: [TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}] LCWIK merge accum1: [] accum2: [TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}] rval: [TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}] LCWIK merge accum1: [TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}] accum2: [TimestampedValueInSingleWindow{value=[b, c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b, c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] rval: [TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b, c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b, c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] LCWIK merge accum1: [TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b, c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b, c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] accum2: [TimestampedValueInSingleWindow{value=[a], timestamp=1970-01-01T00:00:00.001Z, window=[1969-12-31T23:59:59.999Z..1970-01-01T00:00:00.002Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a, b, c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a, b], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] rval: [TimestampedValueInSingleWindow{value=[a], timestamp=1970-01-01T00:00:00.001Z, window=[1969-12-31T23:59:59.999Z..1970-01-01T00:00:00.002Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a, b, c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b, c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a, b], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] {noformat} Bad run: {noformat} LCWIK merge accum1: [] accum2: [TimestampedValueInMultipleWindows{value=[c], timestamp=1970-01-01T00:00:00.003Z, windows=[[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), [1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), [1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z)], pane=PaneInfo{isFirst=true, timing=EARLY, index=0}}] rval: [TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}] LCWIK merge accum1: [] accum2: [TimestampedValueInMultipleWindows{value=[b], timestamp=1970-01-01T00:00:00.002Z, windows=[[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), [1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), [1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z)], pane=PaneInfo{isFirst=true, timing=EARLY, index=0}}] rval: [TimestampedValueInSingleWindow{value=[b], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] LCWIK reduce value: TimestampedValueInMultipleWindows{value=KV{null, c}, timestamp=1970-01-01T00:00:00.003Z, windows=[[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), [1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), [1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z)], pane=PaneInfo{isFirst=true, timing=EARLY, index=0}} accum: [] result: [TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}] LCWIK reduce value: TimestampedValueInMultipleWindows{value=KV{null, b}, timestamp=1970-01-01T00:00:00.002Z, windows=[[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), [1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), [1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z)], pane=PaneInfo{isFirst=true, timing=EARLY, index=0}} accum: [] result: [TimestampedValueInSingleWindow{value=[b], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] LCWIK merge accum1: [TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}] accum2: [TimestampedValueInMultipleWindows{value=[a, c], timestamp=1970-01-01T00:00:00.001Z, windows=[[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), [1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), [1969-12-31T23:59:59.999Z..1970-01-01T00:00:00.002Z)], pane=PaneInfo{isFirst=true, timing=EARLY, index=0}}] rval: [TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.001Z, window=[1969-12-31T23:59:59.999Z..1970-01-01T00:00:00.002Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING},

TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] LCWIK reduce value: TimestampedValueInMultipleWindows{value=KV{null, a}, timestamp=1970-01-01T00:00:00.001Z, windows=[[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), [1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), [1969-12-31T23:59:59.999Z..1970-01-01T00:00:00.002Z)], pane=PaneInfo{isFirst=true, timing=EARLY, index=0}} accum: [TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}] result: [TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.001Z, window=[1969-12-31T23:59:59.999Z..1970-01-01T00:00:00.002Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] LCWIK merge accum1: [] accum2: [TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.001Z, window=[1969-12-31T23:59:59.999Z..1970-01-01T00:00:00.002Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] rval: [TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.001Z, window=[1969-12-31T23:59:59.999Z..1970-01-01T00:00:00.002Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] LCWIK merge accum1: [TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.001Z, window=[1969-12-31T23:59:59.999Z..1970-01-01T00:00:00.002Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] accum2: [TimestampedValueInSingleWindow{value=[b, a, c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b, c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b, a, c], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] rval: [TimestampedValueInSingleWindow{value=[a, c], timestamp=1970-01-01T00:00:00.001Z, window=[1969-12-31T23:59:59.999Z..1970-01-01T00:00:00.002Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[c], timestamp=1970-01-01T00:00:00.005Z, window=[1970-01-01T00:00:00.003Z..1970-01-01T00:00:00.006Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b, a, c], timestamp=1970-01-01T00:00:00.003Z, window=[1970-01-01T00:00:00.001Z..1970-01-01T00:00:00.004Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b, c], timestamp=1970-01-01T00:00:00.004Z, window=[1970-01-01T00:00:00.002Z..1970-01-01T00:00:00.005Z), pane=PaneInfo.NO_FIRING}, TimestampedValueInSingleWindow{value=[b, a, c], timestamp=1970-01-01T00:00:00.002Z, window=[1970-01-01T00:00:00.000Z..1970-01-01T00:00:00.003Z), pane=PaneInfo.NO_FIRING}] {noformat}

jira_issues_comments:

1. Turns out the issue is that the accumulator is being reused across windows and the combine fn is mutating it so those mutations are appearing in the wrong window.