

git_comments:

1. **comment:** clean up
label: code-design
2. * * Consume decoder data and write into {@link VectorSchemaRoot}.
3. * Licensed to the Apache Software Foundation (ASF) under one or more * contributor license agreements. See the NOTICE file distributed with * this work for additional information regarding copyright ownership. * The ASF licenses this file to You under the Apache License, Version 2.0 * (the "License"); you may not use this file except in compliance with * the License. You may obtain a copy of the License at * * <http://www.apache.org/licenses/LICENSE-2.0> * * Unless required by applicable law or agreed to in writing, software * distributed under the License is distributed on an "AS IS" BASIS, * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. * See the License for the specific language governing permissions and * limitations under the License.
4. * * Composite consumer which hold all consumers. * It manages the consume and cleanup process.
5. **comment:** * * Close this consumer when occurs exception to avoid potential leak.
label: code-design

git_commits:

1. **summary:** ARROW-6199: [Java] Avro adapter avoid potential resource leak.
message: ARROW-6199: [Java] Avro adapter avoid potential resource leak. Related to [ARROW-6199] (<https://issues.apache.org/jira/browse/ARROW-6199>). Currently, avro consumer interface has no close API, which may cause resource leak like AvroBytesConsumer#cacheBuffer. To resolve this, make consumer extends AutoCloseable and create CompositeAvroConsumer to encompasses consume and close logic. Closes #5059 from tianchen92/ARROW-6199 and squashes the following commits:
d60d94c48 <tianchen> fix 42f22da7c <tianchen> clear vectors in close 5b91da75f <tianchen> fix comments 3ffc07600 <tianchen> ARROW-6199: Avro adapter avoid potential resource leak. Authored-by: tianchen <niki.lj@alibaba-inc.com> Signed-off-by: Micah Kornfield <emkornfield@gmail.com>
label: code-design

github_issues:**github_issues_comments:****github_pulls:**

1. **title:** ARROW-6199: [Java] Avro adapter avoid potential resource leak.
body: Related to [ARROW-6199] (<https://issues.apache.org/jira/browse/ARROW-6199>). Currently, avro consumer interface has no close API, which may cause resource leak like AvroBytesConsumer#cacheBuffer. To resolve this, make consumer extends AutoCloseable and create CompositeAvroConsumer to encompasses consume and close logic.
label: code-design

github_pulls_comments:

1. # [Codecov] (<https://codecov.io/gh/apache/arrow/pull/5059?src=pr&el=h1>) Report > Merging [#5059] (<https://codecov.io/gh/apache/arrow/pull/5059?src=pr&el=desc>) into [master] (<https://codecov.io/gh/apache/arrow/commit/3cc12abdbac324cb2e6a93fd2da0ff41929b6cac?src=pr&el=desc>) will ****increase**** coverage by `2.11%`. > The diff coverage is `n/a`. [![Impacted file tree graph](<https://codecov.io/gh/apache/arrow/pull/5059/graphs/tree.svg?width=650&token=LpTCFbqVT1&height=150&src=pr>)] (<https://codecov.io/gh/apache/arrow/pull/5059?src=pr&el=tree>) ``diff @@ Coverage Diff @@ ## master #5059 +/- ##
===== + Coverage 87.63% 89.75% +2.11%
===== Files 1014 674 -340 Lines 144947 100382 -44565 Branches 1437 0 -1437 ===== - Hits 127030 90093 -36937 + Misses 17555 10289 -7266 + Partial 362 0 -362 `` | [Impacted Files] (<https://codecov.io/gh/apache/arrow/pull/5059?src=pr&el=tree>) | Coverage Δ | | ---|---|---| | [r/src/recordbatch.cpp] (<https://codecov.io/gh/apache/arrow/pull/5059/diff?src=pr&el=tree#diff->

ci9zcmMvcMvjb3JkYmF0Y2guY3Bw) ||| [r/R/Table.R]
 (https://codecov.io/gh/apache/arrow/pull/5059/diff?src=pr&el=tree#diff-ci9SL1RhYmxlLlI=) |||
 [js/src/util/fn.ts](https://codecov.io/gh/apache/arrow/pull/5059/diff?src=pr&el=tree#diff-anMvc3JjL3V0aWwvZm4udHM=) ||| [go/arrow/array/bufferbuilder.go]
 (https://codecov.io/gh/apache/arrow/pull/5059/diff?src=pr&el=tree#diff-Z28vYXJyb3cvYXJyYXkvYnVmZmVyYnVpbGRlci5nbw==) ||| [r/src/symbols.cpp]
 (https://codecov.io/gh/apache/arrow/pull/5059/diff?src=pr&el=tree#diff-ci9zcmMvc3ltYm9scy5jcHA=) |
 ||| [rust/datafusion/src/execution/projection.rs](https://codecov.io/gh/apache/arrow/pull/5059/diff?src=pr&el=tree#diff-cnVzdC9kYXRhZnVzaW9uL3NyYy9leGVjdXRpb24vcHJvamVjdGlubi5ycw==) ||
 || [rust/datafusion/src/execution/filter.rs](https://codecov.io/gh/apache/arrow/pull/5059/diff?src=pr&el=tree#diff-cnVzdC9kYXRhZnVzaW9uL3NyYy9leGVjdXRpb24vZmlsdGVyLnJz) |||
 [rust/arrow/src/csv/writer.rs](https://codecov.io/gh/apache/arrow/pull/5059/diff?src=pr&el=tree#diff-cnVzdC9hcjVjdy9zcmMvY3N2L3dyaXRlci5ycw==) ||| [rust/datafusion/src/bin/main.rs]
 (https://codecov.io/gh/apache/arrow/pull/5059/diff?src=pr&el=tree#diff-cnVzdC9kYXRhZnVzaW9uL3NyYy9iaW4vbWVpbi5ycw==) ||| [go/arrow/ipc/file/_reader.go]
 (https://codecov.io/gh/apache/arrow/pull/5059/diff?src=pr&el=tree#diff-Z28vYXJyb3cvYXJyb3cvYXJyYXkvYnVmZmVyYnVpbGRlci5nbw==) ||| ... and [332 more]
 (https://codecov.io/gh/apache/arrow/pull/5059/diff?src=pr&el=tree-more) || ----- [Continue to review full report at Codecov](https://codecov.io/gh/apache/arrow/pull/5059?src=pr&el=continue). > **Legend** -
 [Click here to learn more](https://docs.codecov.io/docs/codecov-delta) > `Δ` = absolute <relative>
 (impact)`, `ø` = not affected`, `?` = missing data` > Powered by [Codecov]
 (https://codecov.io/gh/apache/arrow/pull/5059?src=pr&el=footer). Last update [3cc12ab...d60d94c]
 (https://codecov.io/gh/apache/arrow/pull/5059?src=pr&el=lastupdated). Read the [comment docs]
 (https://docs.codecov.io/docs/pull-request-comments).

2. @emkornfield Hi Micah, Please help take a look on this. About avro adapter there are Array, Map, Fixed type left, I would like to start follow-ups(support all types and iterator) as soon as possible to merge it before 0.15 branch cut together with Jdbc iterator. Many thanks :)
3. @emkornfield Please help take a look at this one :)
4. @emkornfield build passed, and I am curious about which time zone you are in?
5. +1 thank you. US/Pacific.

github_pulls_reviews:

1. Is this necessary, reading the javado it just seems like this resets the value but doesn't free any resources? It seems like if anything we should be closing the vectors in the consumers? But maybe that is a decision for when we support reading only a set number of rows?
2. Thanks, you are right, unnecessary here. 1. close should release vectors but only when exception occurs, otherwise vectors should release by users. 2. the same for iterator API later. PR updated now.
3. lets fix in one of the follow-ups. but I think looping should be done outside of the consumer interfaces.
4. Thank you Micah, will fix in next PR. I plan to start two PRs one by one(support all type, iterator API), then I think i could be called a initial version and hope to catch up 0.15 branch cut.

jira_issues:

1. **summary:** [Java] Avro adapter avoid potential resource leak.
description: Currently, avro consumer interface has no close API, which may cause resource leak like {{AvroBytesConsumer#cacheBuffer}}. To resolve this, make consumer extends {{AutoCloseable}} and create {{CompositeAvroConsumer}} to encompasses consume and close logic.

jira_issues_comments:

1. Issue resolved by pull request 5059 [https://github.com/apache/arrow/pull/5059]