

Item 267

git_comments:

git_commits:

1. **summary:** [HUDI-1233] Deltastreamer Kafka consumption delay reporting indicators (#2074)
message: [HUDI-1233] Deltastreamer Kafka consumption delay reporting indicators (#2074)

github_issues:

github_issues_comments:

github_pulls:

1. **title:** [HUDI-1233] Deltastreamer Kafka consumption delay reporting indicators
body: ## *Tips* - *Thank you very much for contributing to Apache Hudi.* - *Please review <https://hudi.apache.org/contributing.html> before opening a pull request.* ## What is the purpose of the pull request Currently hudi-deltastreamer does not report the indicator of Kafka data consumption delay, I suggest that this function can be added ## Committer checklist - [] Has a corresponding JIRA in PR title & commit - [] Commit message is descriptive of the change - [] CI is green - [] Necessary doc changes done or have another open PR - [] For large changes, please consider breaking it into sub-tasks under an umbrella JIRA.

github_pulls_comments:

1. ![1](https://user-images.githubusercontent.com/25769285/92343761-32515200-f0f7-11ea-8229-014d267e5021.png)
2. @yanghua @vinothchandar Please help with suggestions
3. @liujinhui1994 The CI build failed, please check it.
4. @liujinhui1994 You should fix the conflicts and let the Travis pass.
5. @yanghua please help review
6. @wangxianghu Can you help to review this PR firstly?
7. @liujinhui1994 When you are ready, ping me, please.
8. Please help review @yanghua

github_pulls_reviews:

1. Why two empty lines?
2. **body:** Replace class name with `XXX.class.getName()` should be better
label: code-design
3. @liujinhui1994 thanks for your contribution, I left some comments you can consider.
4. OK, all right
5. It seems we can use the same metric in one `sync` loop. how about initializing it at the begging(in `DeltaSync`) and deliver it to the source constructor?
6. **body:** This is also a way. Currently non-kafka souce has not used metic, so I made a judgment and only passed metic when kafka souce
label: code-design
7. Move this check into `UtilHelpers#createSource`? use different constructor according to source type this can avoid checking two times
8. **body:** Why change this indent?
label: code-design
9. **body:** The first indent is correct?
label: code-design
10. `delayOffectCalculation` -> `delayOffsetCalculation`?

jira_issues:

1. **summary:** deltastreamer Kafka consumption delay reporting indicators
description: currently hudi-deltastreamer does not report the indicator of Kafka data consumption delay, I suggest that this function can be added

jira_issues_comments:

1. yes. we also need to revisit if there is a different mechanism we can use for spark kafka reading, that will get this easily? we can use this Jira to flesh out a plan.
2. Currently I use the result of `consumer.endOffsets-checkpointOffsets` as the Kafka delay [~vinoth]
3. Implemented via master branch: `20b9b399c980270afdd60cc5fbce783c700a8376`