Item 295
**git_comments:**

**git_commits:**

1. **summary:** Merge pull request #224 from dave2718/master
   **message:** Merge pull request #224 from dave2718/master Changing read and write methods in ParquetInputSplit so that they can de...

**github_issues:**

**github_issues_comments:**

**github_pulls:**

1. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
   **body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
2. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
   **body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
   **label:** documentation
3. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
   **body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
4. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
   **body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
5. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
   **body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
6. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
   **body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
7. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …

**body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?

8. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
**body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
9. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
**body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
10. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
**body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
11. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
**body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
12. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
**body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
13. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
**body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
14. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
**body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
15. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
**body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. -

Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?

16. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
    **body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
17. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
    **body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
18. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
    **body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
19. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
    **body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
20. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
    **body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
    **label:** documentation
21. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
    **body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
22. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
    **body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
23. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
    **body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s.

So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
**label:** code-design

24. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
**body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
25. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
**body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
26. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
**body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?
27. **title:** PARQUET-315: Add PARQUET_1_0 and non-repeated data performance tests …
**body:** Here's a list of changes I did on the parquet-benchmarks module: - Add variable to test PARQUET_1_0 format - Add variable to generate random data for the tests - Move part of the DataGenerator.generate() code to constructors so that we measure time when when writing data only. - Add annotation to display AverageTime. The Throughput mode was always displaying less than 1 op/s. So I thought time was a better mode here. @nezihyigitbasi @danielcweeks You did a good job starting the parquet-benchmarks module. Could you please review these changes?

**github_pulls_comments:**

1. @spena thanks for taking the time to create this PR. Can you also update the README of the benchmark module?
2. **body:** @spena Added a few comments. Once you update the README and address the comments, I will be happy to give it a try. Thanks again!
**label:** documentation
3. Thanks @nezihyigitbasi for the feedback. I fixed the issues you mentioned. You will be able to generate data this time.
4. Thanks @nezihyigitbasi I added the changes based on your feedback. I liked your explanation about random data, so I replaced the old paragraph with yours. I tried to figured out why the benchmark cannot be executed when compiling with hadoop-2, but I did not know why. So, I just replace the README to use the command with out the profile. One thing. The read-benchmark.sh tests will fail when using parquetVersion=v2 and randomData=true due to the issue in PARQUET-152. If the patch is applied, then all tests run fine.
5. @spena To keep the fixes to the existing benchmark module separate I created a new PR: https://github.com/apache/parquet-mr/pull/226 Yep I verified that your change makes the read-benchmark go through.
6. @spena I only added a few minor comments to the README. Other than that LGTM. @rdblue I have provided feedback to @spena and verified his changes. When he addresses my final comments this patch LGTM. It's all yours now.
7. Thanks @nezihyigitbasi. I added the minor changes you mentioned. Once #226 is committed, I will merge the code to my branch, and check it is applied correctly.
8. @nezihyigitbasi @rdblue I updated the branch with the latest changes of master (includes #226). No more work to do here.
9. Thanks @spena, I'll take a look at this today.
10. @rdblue did you want to merge this? @spena do you need to merge master

**github_pulls_reviews:**

1. please add a newline here
2. newline
3. can you use specific imports instead of wildcards?
4. why only average time? I think we can do `Mode.ALL` to get as many metrics as possible, which may provide useful info.
5. Is there a reason that you updated to `1.9.3` instead of the latest `1.10` release?
6. Is `randomData` optional? When I only specify the version to generate some data (with a command that looks like `.... org.apache.parquet.benchmarks.DataGenerator generate v2`) I get an exception ``` Java Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 2 at org.apache.parquet.benchmarks.DataGenerator.main(DataGenerator.java:188) ```
7. Seems like to generate data I _have to_ specify `-randomData`, then why is that a cmd line flag at all? I tried a few command line args to see how it behaves - `java -cp ${SCRIPT_PATH}/target/parquet-benchmarks.jar org.apache.parquet.benchmarks.DataGenerator generate` ``` Generating test data ---> This shouldn't be printed btw. Usage: generate VERSION [-randomData] Options: VERSION Use a specific parquet file version to generate data. - v1 for PARQUET_1_0 version. - v2 for PARQUET_2_0 version. - randomData This flag specifies if random data will be used to generate data. By default, only 1 random row is generated, and repeated across the file. ``` - `java -cp ${SCRIPT_PATH}/target/parquet-benchmarks.jar org.apache.parquet.benchmarks.DataGenerator generate v2` ``` Generating test data ---> This shouldn't be printed btw. Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 2 at org.apache.parquet.benchmarks.DataGenerator.main(DataGenerator.java:188) ``` - `java -cp ${SCRIPT_PATH}/target/parquet-benchmarks.jar org.apache.parquet.benchmarks.DataGenerator generate v2 -randomData` `This one runs successfully`
8. newline
9. **body:** A user following the README will get the following error: ``` [/Users/nyigitbasi/other_workspace/parquet-mr] >> ./parquet-benchmarks/run.sh -wi 1 -i 1 -f 1 -p parquetVersion=v2 -p randomData=true bash: write-benchmark.sh: No such file or directory ```
   **label:** documentation
10. `read-benchmarks.sh` -> `read-benchmark.sh` `write-benchmarks.sh` -> `write-benchmark.sh`
11. I don't know what has changed in the mean time, but with the `hadoop-2` profile now the generated parquet-benchmarks jar doesn't contain `/META-INF/BenchmarkList` and it fails ``` [/Users/nyigitbasi/other_workspace/parquet-mr/parquet-benchmarks] >> ./run.sh -wi 1 -i 1 -f 1 -p parquetVersion=v2 -p randomData=true Starting WRITE benchmarks Exception in thread "main" java.lang.RuntimeException: ERROR: Unable to find the resource: /META-INF/BenchmarkList at org.openjdk.jmh.runner.AbstractResourceReader.getReaders(AbstractResourceReader.java:96) at org.openjdk.jmh.runner.BenchmarkList.find(BenchmarkList.java:104) at org.openjdk.jmh.runner.Runner.internalRun(Runner.java:228) at org.openjdk.jmh.runner.Runner.run(Runner.java:178) at org.openjdk.jmh.Main.main(Main.java:66) ``` A build without specifying any profile creates a valid jar though. So build command should be ``` mvn --projects parquet-benchmarks -amd -DskipTests -Denforcer.skip=true clean package ```
12. **body:** Would be nice to rephrase this & clarify a little bit. How about something like: ``` While creating a test Parquet file of N records, by default all benchmarks use a single record that's repeated N times throughout the file. To have a larger number of random records in this test file you can specify `randomData=true` -- with this flag 100K random records are generated and used repeatedly while creating the test file. ```
    **label:** code-design
13. `... and using random data` --> `... and use random data`
14. `command` --> `commands`
15. `command` --> `commands`

**jira_issues:**

1. **summary:** Add PARQUET_1_0 and non-repeated data performance tests to parquet-benchmarks
   **description:** The current parquet-benchmarks module run some performance tests between different block & page sizes for PARQUET_2_0 version only. We should run some tests with PARQUET_1_0 version as well in order to get a view about new parquet version enhancements, and be able to catch possible overheads early by comparing with the old file format. Also, this module uses repeated data to

benchmark the settings. We should also use random data to get different results about how current and new encodings work with real world data.

2. **summary:** Add PARQUET_1_0 and non-repeated data performance tests to parquet-benchmarks
   **description:** The current parquet-benchmarks module run some performance tests between different block & page sizes for PARQUET_2_0 version only. We should run some tests with PARQUET_1_0 version as well in order to get a view about new parquet version enhancements, and be able to catch possible overheads early by comparing with the old file format. Also, this module uses repeated data to benchmark the settings. We should also use random data to get different results about how current and new encodings work with real world data.
   **label:** code-design
3. **summary:** Add PARQUET_1_0 and non-repeated data performance tests to parquet-benchmarks
   **description:** The current parquet-benchmarks module run some performance tests between different block & page sizes for PARQUET_2_0 version only. We should run some tests with PARQUET_1_0 version as well in order to get a view about new parquet version enhancements, and be able to catch possible overheads early by comparing with the old file format. Also, this module uses repeated data to benchmark the settings. We should also use random data to get different results about how current and new encodings work with real world data.

**jira_issues_comments:**

1. Add link to pull request https://github.com/apache/parquet-mr/pull/224