

**P6: An Empirical Study of the GitHub code review tool  
Documentation**

**TA: Shirin Pirouzkhah**

**Group Members: Eleonora Pura, Melih Catal**

**Table of Contents**

1. **Dataset Description**
2. **Dataset Sample Scenarios**
3. **Lizard, Function Extraction, and Linking Comments**

## Descriptions:

- **Start\_line:** Refers to the first line that the multi-line comment applies once the changes are applied.
- **Line:** Refers to the last line that the multi-line comment applies to once the changes are applied.
- **Original\_start\_line:** Refers to the first line that the multi-line or single-line comment applies.
- **Original\_line:** Refers to the last line that the multi-line or single-line comment applies.
- **Side:** Can be Left or Right. Left means the left side of the diff split and it points to the base (*old*) code and deletions. Right means the right side of the diff split and it points to the head (*new*) code and insertions.
- **Start\_side:** Can be Left or Right. The difference between Side is, Start\_side is only used when the comment is a multi-line comment.

More information can be found [here](#)

## When to have which data?

### Single Line Comment:

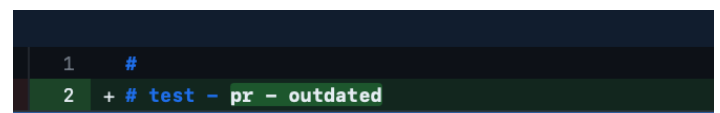
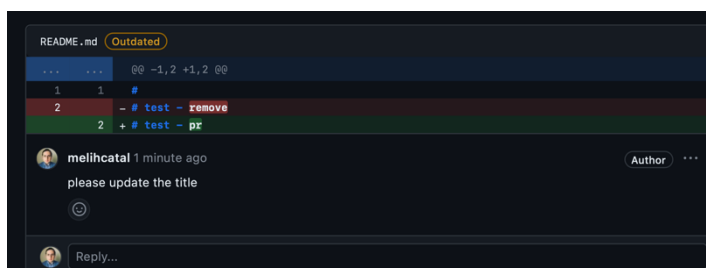
1. original\_line (always)
2. side (always)
3. if not outdated:
  - a. line

### Multiline comment:

1. original\_line (always)
2. side (always)
3. original\_start\_line (if multi-line)
4. start\_side (if multi-line)
5. if not outdated:
  - a. start\_line
  - b. line

**What does outdated mean?** Outdated refers to a comment or code that is no longer valid or applicable due to the subsequent changes made to the codebase.

Example of an outdated scenario:



- Content\_before: test – remove
- Content\_while: test – pr
- Content\_after: test – pr – outdated

Here, the comment is outdated because it was originally referencing line 2 (test-pr) and requesting a change. However, the contributor later decided to modify the code at that line, causing the comment to be outdated. Nevertheless, if there were no direct changes made to that line, but rather code added/removed above it, the comment would not be considered outdated since the code at that line would remain the same but only shifted.

## Dataset Sample Scenarios

### Multi-line comment – Deletion (LEFT side)

```
project                apache/accumulo
url                   https://github.com/apache/accumulo/pull/1505
discussion             https://github.com/apache/accumulo/pull/1505#d...
pull_number           1505
pull_id               373787270
filename              server/base/src/main/java/org/apache/accumulo/...
start_line            NaN
original_start_line    187.0
start_side            LEFT
side                  LEFT
line                  NaN
original_line          189
message               Would it make more sense to keep the replaceme...
owner_id              11872539
user_id               1280725
created_at             2020-02-11T18:01:48Z
updated_at             2020-02-11T18:58:57Z
commit_before          29bdb7024a2ca48a43828b26e2e0c5e28ef21c4e
commit_while           29bdb7024a2ca48a43828b26e2e0c5e28ef21c4e
commit_after           ddd792f022b665cdee83fded2fc2b3dec146b673
url_source_before      https://raw.githubusercontent.com/apache/accum...
url_source_while       https://raw.githubusercontent.com/apache/accum...
url_source_after       https://raw.githubusercontent.com/apache/accum...
file_content_before    /*\n * Licensed to the Apache Software Foundat...
file_content_while     /*\n * Licensed to the Apache Software Foundat...
file_content_after     /*\n * Licensed to the Apache Software Foundat...
Name: 0, dtype: object
```

|                     |      |
|---------------------|------|
| Start_line          | -    |
| Line                | -    |
| Original_start_line | 187  |
| Original_line       | 189  |
| Start_side          | Left |
| Side                | Left |


Here we understand that we have a multi-line comment since the original\_start\_line and original\_line columns are not null. (Or we have both start\_side and side columns not null) Also, the comment refers to deletion since the side is Left. Finally, since the start\_line and line columns are empty we can infer that this comment is outdated. In the following screenshot, we can see the original review.

server/base/src/main/java/org/apache/accumulo/server/fs/VolumeUtil.java
Outdated
Hide resolved

```

187 - VolumeManager vm, KeyExtent extent, TabletFiles tabletFiles, boolean replicate)
188 - throws IOException {
189 - List<Pair<Path,Path>> replacements = ServerConstants.getVolumeReplacements();

```


**ctubbsii** on Feb 11, 2020
Member
...

Would it make more sense to keep the replacements assignment in this method, but short-circuit the method with a `if (replacements.isEmpty()) { return; }` line?

It seems like this is a smaller change and would keep the logic all together that handles the updates, rather than place the special case of no replacements outside the method.

Source: [https://github.com/apache/accumulo/pull/1505#discussion\\_r377803422](https://github.com/apache/accumulo/pull/1505#discussion_r377803422)

## Multi-line Comment – Insertion (side: RIGHT) - Outdated

```

.. project                                apache/accumulo
url                                     https://github.com/apache/accumulo/pull/1614
discussion                             https://github.com/apache/accumulo/pull/1614#d...
pull_number                             1614
pull_id                                420303336
filename                               core/src/main/java/org/apache/accumulo/core/cl...
start_line                             NaN
original_start_line                     591.0
start_side                             RIGHT
side                                   RIGHT
line                                   NaN
original_line                           596
message                                I would wrap `e` so that stack traces are not ...
owner_id                               11872539
user_id                                1268739
created_at                             2020-05-21T21:43:46Z
updated_at                             2020-06-04T19:21:57Z
commit_before                          13ad1b33bc3e92669db8a523e1e3a87025326d7a
commit_while                           2d7f5049dd0c35b9bea9099d68264018a69ffc16
commit_after                           6319d9e408350cd15570eb9a93de613e5d4e730e
url_source_before                      https://raw.githubusercontent.com/apache/accum...
url_source_while                       https://raw.githubusercontent.com/apache/accum...
url_source_after                       https://raw.githubusercontent.com/apache/accum...
file_content_before                    /*\n * Licensed to the Apache Software Foundat...
file_content_while                     /*\n * Licensed to the Apache Software Foundat...
file_content_after                     /*\n * Licensed to the Apache Software Foundat...
Name: 0, dtype: object

```

|                     |       |
|---------------------|-------|
| Start_line          | -     |
| Line                | -     |
| Original_start_line | 591   |
| Original_line       | 596   |
| Start_side          | Right |
| Side                | Right |


Here, the comment is a multi-line (both start\_side and side columns are not null and original\_start\_line exists) comment. The comment starts at line 591 and ends at line 596. It refers to an insertion since the side is Right. Again, it's an outdated comment since the start\_line and line columns are null.

core/src/main/java/org/apache/accumulo/core/clientImpl/bulk/BulkImport.java
Outdated
Hide resolved

```

591 +     Throwable t = e.getCause();
592 +     if (t instanceof IllegalArgumentException) {
593 +         throw (IllegalArgumentException) t;
594 +     } else
595 +         throw new RuntimeException(t);
580 596     }

```


**keith-turner** on May 22, 2020
Contributor
...

I would wrap `e` so that stack traces are not lost and anyone getting the exceptions can trace the full code path from their code to the background thread.

Source: [https://github.com/apache/accumulo/pull/1614#discussion\\_r428932975](https://github.com/apache/accumulo/pull/1614#discussion_r428932975)

## Multi-line Comment – Not Outdated

```
project                apache/accumulo
url                   https://github.com/apache/accumulo/pull/1605
discussion             https://github.com/apache/accumulo/pull/1605#d...
pull_number           1605
pull_id               414994311
filename              core/src/main/java/org/apache/accumulo/core/cl...
start_line            32.0
original_start_line    30.0
start_side            RIGHT
side                 RIGHT
line                  34.0
original_line         32
message               I don't necessarily see the second reason to b...
owner_id              1268739
user_id               1280725
created_at             2020-05-08T04:32:04Z
updated_at             2020-06-11T16:27:47Z
commit_before         a506923dda6ec92bee07ff86ab8ff3e6a980d0ac
commit_while           bd206be8ed50ebf1fee76dc8cf1e9820ef8ca5b8
commit_after           8d024d8ca898987c50a421288fc9eeca6427de98
url_source_before      https://raw.githubusercontent.com/apache/accum...
url_source_while       https://raw.githubusercontent.com/apache/accum...
url_source_after       https://raw.githubusercontent.com/apache/accum...
file_content_before    /*\n * Licensed to the Apache Software Foundat...
file_content_while     /*\n * Licensed to the Apache Software Foundat...
file_content_after     /*\n * Licensed to the Apache Software Foundat...
```

|                     |       |
|---------------------|-------|
| Start_line          | 32    |
| Line                | 34    |
| Original_start_line | 30    |
| Original_line       | 32    |
| Start_side          | Right |
| Side                | Right |

This is a multi-line comment for the reasons explained above. It's not outdated since the start\_line and line columns are not empty. But there are differences between original\_start\_line which is 30 and start\_line, which is 32 and between original\_line, 32 and line, 34. Here start\_line and line refer to the lines after the changes. This means that initially the comment referred to the range 30-32, but after the changes 2 lines were added above this range, making the new range 32-34. The screenshots below should make it clearer. The same comment refers to the same text (which starts with "not support the...") but the lines are shifted.

```
core/src/main/java/org/apache/accumulo/core/client/admin/CompactionStrategyConfig.java
30 + * not support the new compaction execution model. Second, they bind selection
31 + * output file configuration into a single entity when users need to configure
32 + * independently. Third, they use internal Accumulo types and ensuring their st
```

ctubbsii on May 8, 2020

I don't necessarily see the second reason to be a problem. Binding these into a single entity, can be nice for modularizing the user code to make it more reusable, and maintainable in a separate user-controlled repo. Under the new paradigm, what's the best way for users to combine their overall compaction strategy so they can maintain it separately, and just drop it in when needed? Would they just serialize the `CompactionConfig` in some way?

File\_content\_while

```
32 + * not support the new compaction execution model. Second, they bind selection and
33 + * output file configuration into a single entity when users need to configure these
34 + * independently. Third, they use internal Accumulo types and ensuring their stability
```

ctubbsii marked this conversation as resolved.

ctubbsii on May 8, 2020

I don't necessarily see the second reason to be a problem. Binding these into a single entity, can be nice for modularizing the user code to make it more reusable, and maintainable in a separate user-controlled repo. Under the new paradigm, what's the best way for users to combine their overall compaction strategy so they can maintain it separately, and just drop it in when needed? Would they just serialize the `CompactionConfig` in some way?

File\_content\_after

Source: [https://github.com/apache/accumulo/pull/1605#discussion\\_r421931808](https://github.com/apache/accumulo/pull/1605#discussion_r421931808)

## Single-line Comment – Insertion (Right) – Outdated

```
.. project                                apache/accumulo
url                                https://github.com/apache/accumulo/pull/1622
discussion                        https://github.com/apache/accumulo/pull/1622#d...
pull_number                        1622
pull_id                            428010670
filename                        core/src/main/java/org/apache/accumulo/core/co...
start_line                        NaN
original_start_line                NaN
start_side                        NaN
side                              RIGHT
line                              NaN
original_line                      461
message                          The default of true maintains the current beha...
owner_id                          347158
user_id                          1268739
created_at                        2020-06-04T18:34:13Z
updated_at                        2020-06-05T12:44:03Z
commit_before                    dbfab01a39e7bc6ac92f8641fc4b1c0f0c22039f
commit_while                     dbfab01a39e7bc6ac92f8641fc4b1c0f0c22039f
commit_after                     4d6991afc9c9589b829c2b9677a47e85586cc5aa
url_source_before                https://raw.githubusercontent.com/apache/accum...
url_source_while                 https://raw.githubusercontent.com/apache/accum...
url_source_after                 https://raw.githubusercontent.com/apache/accum...
file_content_before              /*\n * Licensed to the Apache Software Foundat...
file_content_while               /*\n * Licensed to the Apache Software Foundat...
file_content_after               /*\n * Licensed to the Apache Software Foundat...
Name: 0, dtype: object
```

|                     |       |
|---------------------|-------|
| Start_line          | -     |
| Line                | -     |
| Original_start_line | -     |
| Original_line       | 461   |
| Start_side          | -     |
| Side                | Right |

Here we can understand that the comment is a single-line comment since the original\_start\_line column is missing and also start\_side is missing. Again, it's an insertion since the side is Right. Since this is a single-line comment original\_line refers to both the start and the end of the comment line.

core/src/main/java/org/apache/accumulo/core/conf/Property.java Outdated Hide resolved


... @@ -452,6 +458,9 @@

452 458 "The time to wait for a tablet server to process a bulk import request."),

453 459 TSERV\_MINTHREADS("tserver.server.threads.minimum", "20", PropertyType.COUNT,

454 460 "The minimum number of threads to use to handle incoming requests."),

461 + TSERV\_MINTHREADS\_ALLOW\_TIMEOUT("tserver.server.thread.timeout.allowed", "true",

 keith-turner on Jun 4, 2020 Contributor

The default of true maintains the current behavior, however that behavior seemed problematic in the issue you opened. Makes me wonder if the default should be false.

Source: [https://github.com/apache/accumulo/pull/1622#discussion\\_r435467765](https://github.com/apache/accumulo/pull/1622#discussion_r435467765)

## Single-line Comment – Insertion – Not Outdated

```
.. project                apache/accumulo
url                https://github.com/apache/accumulo/pull/1614
discussion         https://github.com/apache/accumulo/pull/1614#d...
pull_number        1614
pull_id            420303336
filename           core/src/main/java/org/apache/accumulo/core/cl...
start_line         NaN
original_start_line NaN
start_side         NaN
side              RIGHT
line              144.0
original_line      144
message            I couldn't find a cleaner way to get a single ...
owner_id           11872539
user_id           11872539
created_at         2020-05-21T18:57:26Z
updated_at         2020-06-04T19:21:57Z
commit_before      13ad1b33bc3e92669db8a523e1e3a87025326d7a
commit_while       2d7f5049dd0c35b9bea9099d68264018a69ffc16
commit_after       6319d9e408350cd15570eb9a93de613e5d4e730e
url_source_before  https://raw.githubusercontent.com/apache/accum...
url_source_while   https://raw.githubusercontent.com/apache/accum...
url_source_after   https://raw.githubusercontent.com/apache/accum...
file_content_before /*\n * Licensed to the Apache Software Foundat...
file_content_while  /*\n * Licensed to the Apache Software Foundat...
file_content_after  /*\n * Licensed to the Apache Software Foundat...
Name: 0, dtype: object
```

|                     |       |
|---------------------|-------|
| Start_line          | -     |
| Line                | 144   |
| Original_start_line | -     |
| Original_line       | 144   |
| Start_side          | -     |
| Side                | Right |

In this case, the comment is a single line since the original\_start\_line and start\_side columns are null. It's not outdated since the line column is not null.

 **milleruntime** commented on May 21, 2020 [View reviewed changes](#)

```
core/src/main/java/org/apache/accumulo/core/clientImpl/bulk/BulkImport.java
141 +      maxTablets = Integer.parseInt(prop.getValue());
142 +      break;
143 +    }
144 +  }
```

 **milleruntime** on May 21, 2020 Contributor Author ...  
I couldn't find a cleaner way to get a single table property from the client side...

Source: <https://github.com/apache/accumulo/pull/1614#pullrequestreview-416408481>

## Single-line Comment – Insertion – Not Outdated – Shift


```
... project                apache/accumulo
url                https://github.com/apache/accumulo/pull/1614
discussion         https://github.com/apache/accumulo/pull/1614#d...
pull_number        1614
pull_id            420303336
filename           test/src/main/java/org/apache/accumulo/test/fu...
start_line         NaN
original_start_line NaN
start_side         NaN
side              RIGHT
line               207.0
original_line      206
message            Not sure if this is possible, but it would be ...
owner_id           11872539
user_id            1268739
created_at         2020-05-28T16:07:24Z
updated_at         2020-06-04T19:21:57Z
commit_before      13ad1b33bc3e92669db8a523e1e3a87025326d7a
commit_while       3951d98aeffc20fa4318d14d03ef3f0c8be56876
commit_after       6319d9e408350cd15570eb9a93de613e5d4e730e
url_source_before  https://raw.githubusercontent.com/apache/accum...
url_source_while   https://raw.githubusercontent.com/apache/accum...
url_source_after   https://raw.githubusercontent.com/apache/accum...
file_content_before /*\n * Licensed to the Apache Software Foundat...
file_content_while  /*\n * Licensed to the Apache Software Foundat...
file_content_after  /*\n * Licensed to the Apache Software Foundat...
Name: 0, dtype: object
```

|                     |       |
|---------------------|-------|
| Start_line          | -     |
| Line                | 207   |
| Original_start_line | -     |
| Original_line       | 206   |
| Start_side          | -     |
| Side                | Right |

Here the comment is a single-line comment because of the reasons described above. It's not outdated since the line column is not null. However, the original\_line and line columns are different. This is because an insertion above line 206 which caused a shift by 1 line.

```
test/src/main/java/org/apache/accumulo/test/functional/BulkNewIT.java
```

```
203 + assertTrue("Wrong exception: " + c, c instanceof ExecutionException);
204 + assertTrue("Wrong exception: " + c.getCause(),
205 +           c.getCause() instanceof IllegalArgumentException);
206 +
```


 **keith-turner** on May 28, 2020 Contributor ...

Not sure if this is possible, but it would be really nice to confirm that exception message contains the offending file name. Whenever someone runs into this error message, knowing which file caused the problem will be extremely helpful to them.

If the test does not do this, I would also recommend creating multiple files. One that exceeds the limit and few that do not. Want to ensure in this case the troublesome file is listed in the message.

```
206 + assertTrue("Bad File not in exception: " + msg, msg.contains("bad-file.rf"));
207 +
```

milleruntime marked this conversation as resolved.

 **keith-turner** on May 28, 2020 Contributor ...

Not sure if this is possible, but it would be really nice to confirm that exception message contains the offending file name. Whenever someone runs into this error message, knowing which file caused the problem will be extremely helpful to them.

If the test does not do this, I would also recommend creating multiple files. One that exceeds the limit and few that do not. Want to ensure in this case the troublesome file is listed in the message.

Source: [https://github.com/apache/accumulo/pull/1614#discussion\\_r431953512](https://github.com/apache/accumulo/pull/1614#discussion_r431953512)



## Lizard, Function Extraction and Linking Comments

We are using Lizard to extract methods of the Java files. To do that we are using `file_content_while` and `file_content_after` columns of the dataset. Method information received from Lizard, contains the function start line and end line data.

In the following screenshots, we can see that the original method starts at line 188 and ends at line 202. This is exactly what Lizard returns to us. Once we have this critical information, we can easily link the comments with the methods since the raw GitHub data contains the lines that comments refer to. We only need to check if the comment is in the range of function lines.

```
187  @Test
188  public void testMaxTablets() throws Exception {
189      String maxTablets = "0";
190      try (AccumuloClient client = Accumulo.newClient().from(getClientProps()).build()) {
191          maxTablets = client.instanceOperations().getSystemConfiguration()
192              .get(Property.MASTER_BULK_MAX_TABLETS.getKey());
193          client.instanceOperations().setProperty(Property.MASTER_BULK_MAX_TABLETS.getKey(), "1");
194          testBulkFile(offline:false, usePlan:false);
195          fail("Expected IllegalArgumentException for " + Property.MASTER_BULK_MAX_TABLETS);
196      } catch (IllegalArgumentException e) {} finally {
197          try (AccumuloClient client = Accumulo.newClient().from(getClientProps()).build()) {
198              client.instanceOperations().setProperty(Property.MASTER_BULK_MAX_TABLETS.getKey(),
199                  maxTablets);
200          }
201      }
202  }
```

*Original Code*

```
[
  {
    "name": "BulkNewIT::testMaxTablets",
    "long_name": "BulkNewIT::testMaxTablets()",
    "start_line": 188,
    "end_line": 202,
    "body": " public void testMaxTablets() throws Exception {\n    String maxTablets = \"0\";\n    try (AccumuloClient client = Accumulo.newClient()..."
  }
]
```

*Lizard Extraction Data*

```
def comment_in_method_body(comment_start_line, comment_end_line, function_start_line, function_end_line):
    comment_range = set(range(comment_start_line, comment_end_line+1))
    function_range = set(range(function_start_line, function_end_line+1))

    if comment_range.issubset(function_range):
        return True
    else:
        return False
```

*Comment Linking*