

git_comments:

1. 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.
2. 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.
3. flake8: noqa out of order import to avoid circular deps
4. not None check here to ensure MagicMock aren't treated as None
5. Object hasn't been created yet, can't refresh.
6. type: ignore

git_commits:

1. **summary:** Python: Support creating tables (#1216)
message: Python: Support creating tables (#1216)

github_issues:

github_issues_comments:

github_pulls:

1. **title:** [Python] create table support
body: First step in write support for python. Adds support & tests for create on: * `HiveTables` * `FilesystemTables`

github_pulls_comments:

1. Should be merged after #1214
2. I merged #1214. Could you please rebase this one? Thanks!
3. > I merged #1214. Could you please rebase this one? Thanks! :+1: rebased and ready for review!
4. Hey @danielcweeks & @rdblue anyone have some time to take a peek at this?
5. @rymurr, I should be able to review it this week and hopefully tomorrow.
6. @rymurr, overall this looks good to me. We might want to make it more pythonic in areas, like relying on `if reference` instead of `if reference is not None`, but those are minor. Have you tested this in practice by actually connecting to a Hive Metastore? It looks like the tests only mock the behavior. We might want to repurpose some of the Java test code to run a metastore for the duration of python testing from a script.
7. thanks for the review @rdblue! I have addressed your comments and made the code more pythonic. For some odd reason I thought keeping python close to Java was a good idea when I first raised the PR :) As for the metastore tests: adding an integration test with HMS has proved slightly more annoying than I hoped so we can either leave this open while I work on it or I can add it as part of another PR later
8. @rymurr, thanks for the update. I think this is fine to commit without the integration tests, as long as someone has tested it by hand to make sure it actually works to create tables in Hive. It works, right?
9. > It works, right? This comment made my morning :)
10. > > It works, right? > > This comment made my morning :) Thanks @jacques-n :rofl: and yes @rdblue it is working :-)

11. Everything looks good to me. There are a couple of minor things, like still having the Hive types defined (I'm not sure if you intended to remove those). Tests are passing except what looks like a flaky test in Java.
12. fixed the build error, should be good to go now :-)

github_pulls_reviews:

1. In ``load``, the number of items returned is checked. Should that be done here as well?
2. I'm getting a warning about PEP 8 and limiting lines to 120 chars. Is that something we want to do in Python? Seems to make sense to me to limit lines to 120.
3. Minor: if we add types, the docs are better. For example, changing this to ``table_identifier: str`` made it so that documentation for ``str.rsplit`` is found, at least in PyCharm. We should probably start adding types where it makes sense, like in the base class ``Tables``. Not something we need to do in this commit, but it really helps reviewing.
4. I agree with using `table_identifier` here. Should we also rename it in ``load`` since the location is the table identifier for FS tables?
5. Couldn't this just be ``if self.current_metadata_location``? I think both ``None`` and ``````` evaluate to ``False`` in python if statements.
6. I think you can probably embed this method in ``unlock``. I think the ``doUnlock`` method in Java is only there because of the use of ``Optional`` for a lock id.
7. Nit: I think ``timedOut`` would be a better name?
8. Since this is identical to the fall through case, can this be ``pass`` instead?
9. agreed, fixed
10. I agree, I have added types in places where this PR has made changes. Hopefully we can build full type coverage over time rather than having someone slog through the code base and add them individually
11. yes, fixed
12. yeah, its strange it wasn't caught by flake8. Fixed now
13. correct, fixed
14. yup, fixed
15. agreed, fixed
16. agreed
17. I'm not sure I understand this comment. Will a mock object evaluate to ``False`` in an if?
18. I think this does need to be the metadata file location and not the table location.
19. just wasn't mocked correctly, fixed now and removed the ``is not None`` check
20. I don't really like the idea of exposing hive types through Iceberg types natively. Seems like we should externalize this and have a separate hive<->iceberg type mapping utility class/function.
21. ``metadataFileLocation`` is the Java name, ``location`` is the Python name. Same value though.
22. That seems confusing to me. How do you get the table's base location in Python?
23. agreed. thats a good point. Thats fixed.
24. ahh, i see now. It appears that the Java and Python impls are a bit out of sync. Python doesn't use ``file`` at all, which is where java takes its location from. Will look at figuring out what happened there and update the PR this eve or tomorrow
25. should be good to go now @rdblue, we had missed a property in ``TableMetadata`` in python
26. I don't see this ``hive_name`` property used anywhere. Did you mean to remove these changes from types?
27. Should we check that ``location is None`` to ensure that a user doesn't pass both a table identifier (location) and a different location?
28. This doesn't agree with what is enforced. I like that ``location`` must always be ``None`` in the code so it would be good to state that here. It would also be easier to read since it avoids "should be not None".
29. yup, apologies. Changed my mind halfway through impl. Fixed now

jira_issues:

jira_issues_comments: