Item 196
git_comments:

1. * * Rule will split the Python {@link FlinkLogicalTableFunctionScan} which includes java calls into a * {@link FlinkLogicalCalc} which will be the left input of the new {@link FlinkLogicalCorrelate} * and a new {@link FlinkLogicalTableFunctionScan} without java calls.
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. add the fields of the extracted java rex calls.
4. extract correlate output RexNode.
5. add the fields of the primitive left input.
6. create a new java calc
7. extract java funcs.
8. 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.
9. * * Rule will split the Python {@link FlinkLogicalTableFunctionScan} which includes java calls into a * {@link FlinkLogicalCalc} which will be the left input of the new {@link FlinkLogicalCorrelate} * and a new {@link FlinkLogicalTableFunctionScan} without java calls.
10. * 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.
11. add the fields of the extracted java rex calls.
12. extract correlate output RexNode.
13. add the fields of the primitive left input.
14. create a new java calc
15. extract java funcs.

git_commits:

1. **summary:** [FLINK-16007][table-planner][table-planner-blink][python] Add PythonCorrelateSplitRule to push down the Java Calls contained in Python Correlate node
   message: [FLINK-16007][table-planner][table-planner-blink][python] Add PythonCorrelateSplitRule to push down the Java Calls contained in Python Correlate node This closes #11242.

github_issues:

github_issues_comments:

github_pulls:

1. **title:** [FLINK-16007][table-planner][table-planner-blink][python] Add PythonCorrelateSplitRule to push down the Java Calls contained in Python Correlate node

body: ## What is the purpose of the change *This rule will add PythonCorrelateSplitRule to support java calls in Python UDTF* ## Brief change log - *Add PythonCorrelateSplitRule* ## Verifying this change This change added tests and can be verified as follows: - *UT in PythonCorrelateSplitRuleTest* - *IT in test_table_function_with_java_calls of test_udtf* ## Does this pull request potentially affect one of the following parts: - Dependencies (does it add or upgrade a dependency): (no) - The public API, i.e., is any changed class annotated with `@Public(Evolving)`: (no) - The serializers: (no) - The runtime per-record code paths (performance sensitive): (no) - Anything that affects deployment or recovery: JobManager (and its components), Checkpointing, Kubernetes/Yarn/Mesos, ZooKeeper: (no) - The S3 file system connector: (no) ## Documentation - Does this pull request introduce a new feature? (yes) - If yes, how is the feature documented? (not applicable)

2. **title:** [FLINK-16007][table-planner][table-planner-blink][python] Add PythonCorrelateSplitRule to push down the Java Calls contained in Python Correlate node
   body: ## What is the purpose of the change *This rule will add PythonCorrelateSplitRule to support java calls in Python UDTF* ## Brief change log - *Add PythonCorrelateSplitRule* ## Verifying this change This change added tests and can be verified as follows: - *UT in PythonCorrelateSplitRuleTest* - *IT in test_table_function_with_java_calls of test_udtf* ## Does this pull request potentially affect one of the following parts: - Dependencies (does it add or upgrade a dependency): (no) - The public API, i.e., is any changed class annotated with `@Public(Evolving)`: (no) - The serializers: (no) - The runtime per-record code paths (performance sensitive): (no) - Anything that affects deployment or recovery: JobManager (and its components), Checkpointing, Kubernetes/Yarn/Mesos, ZooKeeper: (no) - The S3 file system connector: (no) ## Documentation - Does this pull request introduce a new feature? (yes) - If yes, how is the feature documented? (not applicable)

3. **title:** [FLINK-16007][table-planner][table-planner-blink][python] Add PythonCorrelateSplitRule to push down the Java Calls contained in Python Correlate node
   body: ## What is the purpose of the change *This rule will add PythonCorrelateSplitRule to support java calls in Python UDTF* ## Brief change log - *Add PythonCorrelateSplitRule* ## Verifying this change This change added tests and can be verified as follows: - *UT in PythonCorrelateSplitRuleTest* - *IT in test_table_function_with_java_calls of test_udtf* ## Does this pull request potentially affect one of the following parts: - Dependencies (does it add or upgrade a dependency): (no) - The public API, i.e., is any changed class annotated with `@Public(Evolving)`: (no) - The serializers: (no) - The runtime per-record code paths (performance sensitive): (no) - Anything that affects deployment or recovery: JobManager (and its components), Checkpointing, Kubernetes/Yarn/Mesos, ZooKeeper: (no) - The S3 file system connector: (no) ## Documentation - Does this pull request introduce a new feature? (yes) - If yes, how is the feature documented? (not applicable)

4. **title:** [FLINK-16007][table-planner][table-planner-blink][python] Add PythonCorrelateSplitRule to push down the Java Calls contained in Python Correlate node
   body: ## What is the purpose of the change *This rule will add PythonCorrelateSplitRule to support java calls in Python UDTF* ## Brief change log - *Add PythonCorrelateSplitRule* ## Verifying this change This change added tests and can be verified as follows: - *UT in PythonCorrelateSplitRuleTest* - *IT in test_table_function_with_java_calls of test_udtf* ## Does this pull request potentially affect one of the following parts: - Dependencies (does it add or upgrade a dependency): (no) - The public API, i.e., is any changed class annotated with `@Public(Evolving)`: (no) - The serializers: (no) - The runtime per-record code paths (performance sensitive): (no) - Anything that affects deployment or recovery: JobManager (and its components), Checkpointing, Kubernetes/Yarn/Mesos, ZooKeeper: (no) - The S3 file system connector: (no) ## Documentation - Does this pull request introduce a new feature? (yes) - If yes, how is the feature documented? (not applicable)

5. **title:** [FLINK-16007][table-planner][table-planner-blink][python] Add PythonCorrelateSplitRule to push down the Java Calls contained in Python Correlate node
   body: ## What is the purpose of the change *This rule will add PythonCorrelateSplitRule to support java calls in Python UDTF* ## Brief change log - *Add PythonCorrelateSplitRule* ## Verifying this change This change added tests and can be verified as follows: - *UT in PythonCorrelateSplitRuleTest* - *IT in test_table_function_with_java_calls of test_udtf* ## Does this pull request potentially affect one of the following parts: - Dependencies (does it add or upgrade a dependency): (no) - The public API, i.e., is any changed class annotated with `@Public(Evolving)`: (no) - The serializers: (no) - The runtime per-record code paths (performance sensitive): (no) - Anything that affects deployment or recovery: JobManager (and its components), Checkpointing, Kubernetes/Yarn/Mesos, ZooKeeper: (no) - The S3 file system connector: (no) ## Documentation - Does this pull request introduce a new feature? (yes) - If yes, how is the feature documented? (not applicable)

6. **title:** [FLINK-16007][table-planner][table-planner-blink][python] Add PythonCorrelateSplitRule to push down the Java Calls contained in Python Correlate node
   body: ## What is the purpose of the change *This rule will add PythonCorrelateSplitRule to support java calls in Python UDTF* ## Brief change log - *Add PythonCorrelateSplitRule* ## Verifying this change This change added tests and can be verified as follows: - *UT in PythonCorrelateSplitRuleTest* - *IT in test_table_function_with_java_calls of test_udtf* ## Does this pull request potentially affect one of the following parts: - Dependencies (does it add or upgrade a dependency): (no) - The public API, i.e., is any changed class annotated with `@Public(Evolving)`: (no) - The serializers: (no) - The runtime per-record code paths (performance sensitive): (no) - Anything that affects deployment or recovery: JobManager (and its components), Checkpointing, Kubernetes/Yarn/Mesos, ZooKeeper: (no) - The S3 file system connector: (no) ## Documentation - Does this pull request introduce a new feature? (yes) - If yes, how is the feature documented? (not applicable)

7. **title:** [FLINK-16007][table-planner][table-planner-blink][python] Add PythonCorrelateSplitRule to push down the Java Calls contained in Python Correlate node
   body: ## What is the purpose of the change *This rule will add PythonCorrelateSplitRule to support java calls in Python UDTF* ## Brief change log - *Add PythonCorrelateSplitRule* ## Verifying this change This change added tests and can be verified as follows: - *UT in PythonCorrelateSplitRuleTest* - *IT in test_table_function_with_java_calls of test_udtf* ## Does this pull request potentially affect one of the following parts: - Dependencies (does it add or upgrade a dependency): (no) - The public API, i.e., is any changed class annotated with `@Public(Evolving)`: (no) - The serializers: (no) - The runtime per-record code paths (performance sensitive): (no) - Anything that affects deployment or recovery: JobManager (and its components), Checkpointing, Kubernetes/Yarn/Mesos, ZooKeeper: (no) - The S3 file system connector: (no) ## Documentation - Does this pull request introduce a new feature? (yes) - If yes, how is the feature documented? (not applicable)

8. **title:** [FLINK-16007][table-planner][table-planner-blink][python] Add PythonCorrelateSplitRule to push down the Java Calls contained in Python Correlate node
   body: ## What is the purpose of the change *This rule will add PythonCorrelateSplitRule to support java calls in Python UDTF* ## Brief change log - *Add PythonCorrelateSplitRule* ## Verifying this change This change added tests and can be verified as follows: - *UT in PythonCorrelateSplitRuleTest* - *IT in test_table_function_with_java_calls of test_udtf* ## Does this pull request potentially affect one of the following parts: - Dependencies (does it add or upgrade a dependency): (no) - The public API, i.e., is any changed class annotated with `@Public(Evolving)`: (no) - The serializers: (no) - The runtime per-record code paths (performance sensitive): (no) - Anything that affects deployment or recovery: JobManager (and its components), Checkpointing, Kubernetes/Yarn/Mesos, ZooKeeper: (no) - The S3 file system connector: (no) ## Documentation - Does this pull request introduce a new feature? (yes) - If yes, how is the feature documented? (not applicable)
   label: code-design

9. **title:** [FLINK-16007][table-planner][table-planner-blink][python] Add PythonCorrelateSplitRule to push down the Java Calls contained in Python Correlate node
   body: ## What is the purpose of the change *This rule will add PythonCorrelateSplitRule to support java calls in Python UDTF* ## Brief change log - *Add PythonCorrelateSplitRule* ## Verifying this change This change added tests and can be verified as follows: - *UT in PythonCorrelateSplitRuleTest* - *IT in test_table_function_with_java_calls of test_udtf* ## Does this pull request potentially affect one of the following parts: - Dependencies (does it add or upgrade a dependency): (no) - The public API, i.e., is any changed class annotated with `@Public(Evolving)`: (no) - The serializers: (no) - The runtime per-record code paths (performance sensitive): (no) - Anything that affects deployment or recovery: JobManager (and its components), Checkpointing, Kubernetes/Yarn/Mesos, ZooKeeper: (no) - The S3 file system connector: (no) ## Documentation - Does this pull request introduce a new feature? (yes) - If yes, how is the feature documented? (not applicable)

10. **title:** [FLINK-16007][table-planner][table-planner-blink][python] Add PythonCorrelateSplitRule to push down the Java Calls contained in Python Correlate node
    body: ## What is the purpose of the change *This rule will add PythonCorrelateSplitRule to support java calls in Python UDTF* ## Brief change log - *Add PythonCorrelateSplitRule* ## Verifying this change This change added tests and can be verified as follows: - *UT in PythonCorrelateSplitRuleTest* - *IT in test_table_function_with_java_calls of test_udtf* ## Does this pull request potentially affect one of the following parts: - Dependencies (does it add or upgrade a dependency): (no) - The public API, i.e., is any changed class annotated with `@Public(Evolving)`: (no) - The serializers: (no) - The runtime per-record code paths (performance sensitive): (no) - Anything that affects deployment or recovery: JobManager (and its components), Checkpointing, Kubernetes/Yarn/Mesos, ZooKeeper: (no) - The S3 file system

connector: (no) ## Documentation - Does this pull request introduce a new feature? (yes) - If yes, how is the feature documented? (not applicable)

github_pulls_comments:

1. Thanks a lot for your contribution to the Apache Flink project. I'm the @flinkbot. I help the community to review your pull request. We will use this comment to track the progress of the review. ## Automated Checks Last check on commit 2de60095d8046648dd942cf0915a18c8b4a3a854 (Thu Feb 27 13:33:56 UTC 2020) **Warnings:** * No documentation files were touched! Remember to keep the Flink docs up to date! * **This pull request references an unassigned [Jira ticket] (https://issues.apache.org/jira/browse/FLINK-16007).** According to the [code contribution guide] (https://flink.apache.org/contributing/contribute-code.html), tickets need to be assigned before starting with the implementation work. _{Mention the bot in a comment to re-run the automated checks.} ## Review Progress * ❓ 1. The [description] looks good. * ❓ 2. There is [consensus] that the contribution should go into to Flink. * ❓ 3. Needs [attention] from. * ❓ 4. The change fits into the overall [architecture]. * ❓ 5. Overall code [quality] is good. Please see the [Pull Request Review Guide] (https://flink.apache.org/contributing/reviewing-prs.html) for a full explanation of the review process. <details> The Bot is tracking the review progress through labels. Labels are applied according to the order of the review items. For consensus, approval by a Flink committer of PMC member is required <summary>Bot commands</summary> The @flinkbot bot supports the following commands: - `@flinkbot approve description` to approve one or more aspects (aspects: `description`, `consensus`, `architecture` and `quality`) - `@flinkbot approve all` to approve all aspects - `@flinkbot approve-until architecture` to approve everything until `architecture` - `@flinkbot attention @username1 [@username2 ..]` to require somebody's attention - `@flinkbot disapprove architecture` to remove an approval you gave earlier </details>

2. <!-- Meta data { "version" : 1, "metaDataEntries" : [{ "hash" : "2de60095d8046648dd942cf0915a18c8b4a3a854", "status" : "DELETED", "url" : "https://dev.azure.com/rmetzger/5bd3ef0a-4359-41af-abca-811b04098d2e/_build/results?buildId=5694", "triggerID" : "2de60095d8046648dd942cf0915a18c8b4a3a854", "triggerType" : "PUSH" }, { "hash" : "2de60095d8046648dd942cf0915a18c8b4a3a854", "status" : "DELETED", "url" : "https://travis-ci.com/flink-ci/flink/builds/150841822", "triggerID" : "2de60095d8046648dd942cf0915a18c8b4a3a854", "triggerType" : "PUSH" }, { "hash" : "ad21752d6e9ac8c6d2dc1c0d2d824f86d77d69c0", "status" : "SUCCESS", "url" : "https://travis-ci.com/flink-ci/flink/builds/151299714", "triggerID" : "ad21752d6e9ac8c6d2dc1c0d2d824f86d77d69c0", "triggerType" : "PUSH" }, { "hash" : "ad21752d6e9ac8c6d2dc1c0d2d824f86d77d69c0", "status" : "FAILURE", "url" : "https://dev.azure.com/rmetzger/5bd3ef0a-4359-41af-abca-811b04098d2e/_build/results?buildId=5796", "triggerID" : "ad21752d6e9ac8c6d2dc1c0d2d824f86d77d69c0", "triggerType" : "PUSH" }] }--> ## CI report: * ad21752d6e9ac8c6d2dc1c0d2d824f86d77d69c0 Travis: [SUCCESS](https://travis-ci.com/flink-ci/flink/builds/151299714) Azure: [FAILURE](https://dev.azure.com/rmetzger/5bd3ef0a-4359-41af-abca-811b04098d2e/_build/results?buildId=5796) <details> <summary>Bot commands</summary> The @flinkbot bot supports the following commands: - `@flinkbot run travis` re-run the last Travis build - `@flinkbot run azure` re-run the last Azure build </details>

3. Thanks a lot for @hequn8128 review, I have addressed the comments at the latest commit.

github_pulls_reviews:

1. Since we have Rule tests, we don't need the IT test.
2. We haven't supported python udtf for blink planner under batch mode.
3. Rule will split the => Rule will split the Python ..., making it more explicit.
4. Change to ``` RexNode rexNode = pythonTableFuncScan.getCall(); if (rexNode instanceof RexCall) { return PythonUtil.isPythonCall(rexNode) && PythonUtil.containsNonPythonCall(rexNode); } ```
5. Do we need to merge Calcs here? This can be done in the CalcMergeRule.
6. **body:** Could you extract some smaller methods from this method? The code will be more readable. **label:** code-design
7. Remove the blank after `new`.

jira_issues:

1. **summary:** Add rules to push down the Java Calls contained in Python Correlate node

description: The Java Calls contained in Python Correlate node should be extracted to make sure the TableFunction works well.
2. **summary:** Add rules to push down the Java Calls contained in Python Correlate node
   description: The Java Calls contained in Python Correlate node should be extracted to make sure the TableFunction works well.
3. **summary:** Add rules to push down the Java Calls contained in Python Correlate node
   description: The Java Calls contained in Python Correlate node should be extracted to make sure the TableFunction works well.

jira_issues_comments:

1. Resolved in 1.11.0 via f0bdf3179c1deff41eaaccb81544f19150519c30