Item 214
git_comments:

1. Check if the file is local, if that is the case, upload it to a bucket

git_commits:

1. **summary:** [AIRFLOW-2124] Upload Python file to a bucket for Dataproc
   message: [AIRFLOW-2124] Upload Python file to a bucket for Dataproc If the Python Dataproc file is on local storage, we want to upload this to google cloud storage before submitting it to the dataproc cluster Closes #3130 from Fokko/airflow-stash-files-on-gcs

github_issues:

github_issues_comments:

github_pulls:

1. **title:** [AIRFLOW-2124] Upload the main file to a bucket
   body: We want to supply a bucket where we can store stuff to pass it on to a dataproc job. Just tested in on our test-cluster. Make sure you have checked _all_ steps below. ### JIRA - [x] My PR addresses the following [Airflow JIRA](https://issues.apache.org/jira/browse/AIRFLOW/) issues and references them in the PR title. For example, "[AIRFLOW-XXX] My Airflow PR" - https://issues.apache.org/jira/browse/AIRFLOW-XXX ### Description - [x] Here are some details about my PR, including screenshots of any UI changes: ### Tests - [x] My PR adds the following unit tests __OR__ does not need testing for this extremely good reason: ### Commits - [x] My commits all reference JIRA issues in their subject lines, and I have squashed multiple commits if they address the same issue. In addition, my commits follow the guidelines from "[How to write a good git commit message](http://chris.beams.io/posts/git-commit/)": 1. Subject is separated from body by a blank line 2. Subject is limited to 50 characters 3. Subject does not end with a period 4. Subject uses the imperative mood ("add", not "adding") 5. Body wraps at 72 characters 6. Body explains "what" and "why", not "how" - [x] Passes `git diff upstream/master -u -- "*.py" | flake8 --diff`

github_pulls_comments:

1. @kaxil @fenglu-g Any thoughts?
2. **body:** And `Flake8` errors: ``` airflow/contrib/operators/dataproc_operator.py:902:91: E501 line too long (115 > 90 characters) "If you want Airflow to upload the local file to a temporary bucket, set the 'temp_bucket' key in " ^ airflow/www/views.py:2676:91: E501 line too long (93 > 90 characters) 'extra__google_cloud_platform__temp_bucket': StringField('Temporary Airflow bucket'), ```
   label: test
3. Thanks @kaxil for the comments. What are your thoughts on general on this? When we submit using the `gcloud` tool, I expect something similar in the background: ``` gcloud dataproc jobs submit pyspark aggregations.py ``` This is a local file and apparently also uploaded. Maybe this is using the configBucket of Google, but it isn't transparent to me: https://cloud.google.com/dataproc/docs/reference/rest/v1/projects.regions.clusters#nodeinitializationaction
4. # [Codecov](https://codecov.io/gh/apache/incubator-airflow/pull/3130?src=pr&el=h1) Report > Merging [#3130](https://codecov.io/gh/apache/incubator-airflow/pull/3130?src=pr&el=desc) into [master](https://codecov.io/gh/apache/incubator-airflow/commit/37072ab521a8ef77cd1e85b75ecf439e64b70eec?src=pr&el=desc) will **decrease** coverage by `<.01%`. > The diff coverage is `n/a`. [![Impacted file tree graph](https://codecov.io/gh/apache/incubator-airflow/pull/3130/graphs/tree.svg?width=650&token=WdLKlKHOAU&height=150&src=pr)](https://codecov.io/gh/apache/incubator-airflow/pull/3130?src=pr&el=tree) ```diff @@ Coverage Diff @@ ## master #3130 +/- ## == - Coverage 73.06% 73.05% -0.01% == Files 180 180 Lines 12654 12654 == - Hits 9246 9245 -1 - Misses 3408 3409 +1 ``` | [Impacted Files](https://codecov.io/gh/apache/incubator-airflow/pull/3130?src=pr&el=tree) | Coverage Δ | | |---|---|---| | [airflow/www/views.py](https://codecov.io/gh/apache/incubator-airflow/pull/3130/diff?src=pr&el=tree#diff-YWlyZmxvdy93d3cvdmlld3MucHk=) | `71.7% <ø> (ø)` | :arrow_up: | | [airflow/models.py](https://codecov.io/gh/apache/incubator-airflow/pull/3130/diff?src=pr&el=tree#diff-YWlyZmxvdy9tb2RlbHMucHk=) | `87.28% <0%> (-0.05%)` | :arrow_down: | ------ [Continue to review full report at Codecov](https://codecov.io/gh/apache/incubator-airflow/pull/3130?src=pr&el=continue). > **Legend** - [Click here to learn more](https://docs.codecov.io/docs/codecov-delta) > `Δ = absolute <relative> (impact)`, `ø = not affected`, `? = missing data` > Powered by [Codecov](https://codecov.io/gh/apache/incubator-airflow/pull/3130?src=pr&el=footer). Last update [37072ab...d7ba338](https://codecov.io/gh/apache/incubator-airflow/pull/3130?src=pr&el=lastupdated). Read the [comment docs](https://docs.codecov.io/docs/pull-request-comments).
5. @Fokko Yes, that is correct. When you submit a job, it uploads it to Cloud Storage in a temporary bucket it creates for that dataproc cluster. Below is the location it uploads my pyspark job file `pi.py` when I run `gcloud dataproc jobs submit pyspark pi.py --cluster=CLUSTER_NAME --region=europe-west2` ``` gs://dataproc-272e8095-45e1-UUID-d4ed588428db-europe-west2/google-cloud-dataproc-metainfo/3ce3db05-a2dd-4f47-baba-c602e8d75e01/jobs/4ffd806b-c302-4931-905a-109ccd7d9be2/staging/pi.py ``` It also outputs the details of location of this file in the logs in the key: `mainPythonFileUri`: ``` [Stage 0:> (0 + 0) / 2]18/03/16 15:05:40 WARN org.apache.spark.scheduler.TaskSetManager: Stage 0 contains a task of very large size (368 KB). The maximum recommended task size is 100 KB. Pi is roughly 3.133240 18/03/16 15:05:43 INFO org.spark_project.jetty.server.AbstractConnector: Stopped Spark@36e4ca9{HTTP/1.1,[http/1.1]}{0.0.0.0:4040} Job [4ffd806b-c302-4931-905a-109ccd7d9be2] finished successfully. placement: clusterName: cluster-RANDOM clusterUuid: ############# pysparkJob: loggingConfig: {} mainPythonFileUri: gs://dataproc-272e8095-45e1-UUID-europe-west2/google-cloud-dataproc-metainfo/3ce3db05-a2dd-4f47-baba-c602e8d75e01/jobs/4ffd806b-c302-4931-905a-109ccd7d9be2/staging/pi.py reference: jobId: 4ffd806b-c302-4931-905a-109ccd7d9be2 projectId: sb01-185511 status: state: DONE stateStartTime: '2018-03-16T15:05:45.790Z' statusHistory: - state: PENDING stateStartTime: '2018-03-16T15:05:04.203Z' - state: SETUP_DONE stateStartTime: '2018-03-16T15:05:04.280Z' - details: Agent reported job success state: RUNNING stateStartTime: '2018-03-16T15:05:04.762Z' yarnApplications: - name: PythonPi progress: 1.0 state: FINISHED trackingUrl: http://cluster-RANDOMUUID-m:8088/proxy/application_1521212161610_0001/ ```
6. You can also find the path of the temporary bucket created for dataproc cluster using `project.regions.clusters.get` on dataproc resource under `config.configBucket` Check: https://cloud.google.com/dataproc/docs/reference/rest/v1/projects.regions.clusters#ClusterConfig You can use an alternate approach to upload this file i.e. instead of uploading it to a bucket use the temporary bucket created for the DataProc cluster by first doing a get on the respective cluster and retrieving the name of the temporary bucket. However, I think the idea of having a temporary bucket for Airflow makes more sense and this bucket can be used for multiple things.
7. Thanks @kaxil, exactly the pointer that I needed. I've refactored the code to store it in the temporary bucket.

github_pulls_reviews:

1. It would be good if you can add prefix `gs://` to the path where the file would be uploaded. ```python self.log.info("Uploading {} to gs://{}/{}", local_file, bucket, temp_filename) ```
2. **body:** typo... `am` to `an`
   label: documentation
3. same thing here.. Adding `gs://` would make the log more helpful. ```python self.log.info("Deleting gs://{}/{}", bucket, object) ```

jira_issues:

1. **summary:** Allow local mainPythonFileUri
   description: For our workflow, we currently are in the transition from using BashOperator to using the DataProcPySparkOperators. While rewriting the DAG we came to the conclusion that it is not possible to submit a (local) path as our main Python file, and a Hadoop Compatible Filesystem (HCFS) is required. Our main Python drivers are located in a Git repository. Putting our main Python files in a GS bucket would require manual updating/overwriting these files. In terms of code, this works using the BashOperator: {code:java} gcloud dataproc jobs submit pyspark \ /usr/local/airflow/git/airflow-dags/jobs/main_python_driver.py \ --cluster {cluster_name}{code} But cannot be replicated using the DataProcPySparkOperator: {code:java} DataProcPySparkOperator(main="/usr/local/airflow/git/airflow-dags/jobs/main_python_driver.py", cluster_name=cluster_name) {code} Error: {code:java} =========== Cloud Dataproc Agent Error =========== java.lang.NullPointerException at sun.nio.fs.UnixPath.normalizeAndCheck(UnixPath.java:77) at

sun.nio.fs.UnixPath.<init>(UnixPath.java:71) at sun.nio.fs.UnixFileSystem.getPath(UnixFileSystem.java:281) at com.google.cloud.hadoop.services.agent.job.AbstractJobHandler.registerResourceForDownload(AbstractJobHandler.java:442) at com.google.cloud.hadoop.services.agent.job.PySparkJobHandler.buildCommand(PySparkJobHandler.java:93) at com.google.cloud.hadoop.services.agent.job.AbstractJobHandler$StartDriver.call(AbstractJobHandler.java:538) at com.google.cloud.hadoop.services.agent.job.AbstractJobHandler$StartDriver.call(AbstractJobHandler.java:532) at com.google.cloud.hadoop.services.repackaged.com.google.common.util.concurrent.TrustedListenableFutureTask$TrustedFutureInterruptibleTask.runInterruptibly(T at com.google.cloud.hadoop.services.repackaged.com.google.common.util.concurrent.InterruptibleTask.run(InterruptibleTask.java:57) at com.google.cloud.hadoop.services.repackaged.com.google.common.util.concurrent.TrustedListenableFutureTask.run(TrustedListenableFutureTask.java:80) at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511) at java.util.concurrent.FutureTask.run(FutureTask.java:266) at java.util.concurrent.ScheduledThreadPoolExecutor$ScheduledFutureTask.access$201(ScheduledThreadPoolExecutor.java:180) at java.util.concurrent.ScheduledThreadPoolExecutor$ScheduledFutureTask.run(ScheduledThreadPoolExecutor.java:293) at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142) at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617) at java.lang.Thread.run(Thread.java:748) ======== End of Cloud Dataproc Agent Error ======== {code} What would be best practice in this case? Is it possible to add the ability to submit local paths as main Python file?

jira_issues_comments:

1. **body:** [~kaxilnaik] [~fenglu] We're moving from the bash-operators to the very nice DataProc* operators, but we're running into this. What would we the best practice to solve this? Maybe upload it to a temporary bucket. Previous the gcloud this was handled for us: `gcloud dataproc jobs submit pyspark ../submit_job.py` Any thoughts on this?
   label: code-design
2. Dataproc submit job API accepts job file stored in HCFS, details [here]([https://cloud.google.com/dataproc/docs/reference/rest/v1/projects.regions.jobs).] If you could pre-stage the job files in a GCS bucket, passing the gs://path-to-the-job-file as the mainPythonFileUri would work. Feel free to let me know it that doesn't work for you.
3. Agree with [~fenglu] above
4. We would like to integrate this in the DataProcOperator. We don't want to have additional steps We'll develop something internal which will take care of this and then push it back to Airflow. Cheers
5. I will also double check on this and update you [~Fokko] once I am back from holidays.
6. Commit ec80f944183b744ff7cab7b72f5350240a8ac4ae in incubator-airflow's branch refs/heads/master from [~Fokko] [https://git-wip-us.apache.org/repos/asf?p=incubator-airflow.git;h=ec80f94] [AIRFLOW-2124] Upload Python file to a bucket for Dataproc If the Python Dataproc file is on local storage, we want to upload this to google cloud storage before submitting it to the dataproc cluster Closes #3130 from Fokko/airflow-stash-files-on-gcs
7. Issue resolved by pull request #3130 [https://github.com/apache/incubator-airflow/pull/3130]