How to get number of rows processed, time taken for different stages for Dataflow activity

Last updated by | Ranjith Katukojwala | Sep 18, 2022 at 3:28 PM PDT

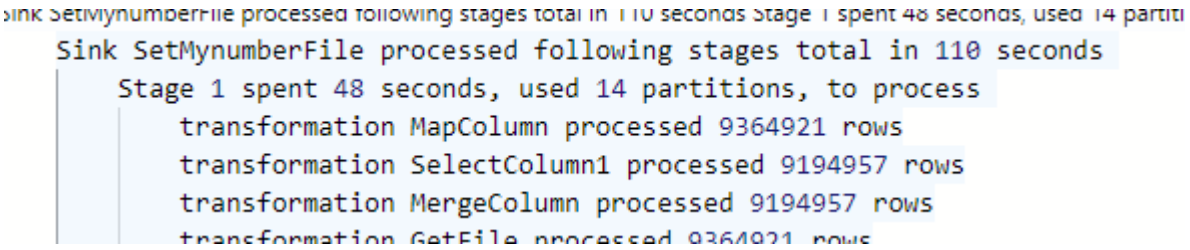
Contents

Issue

If you want to know progress on the spark side for Dataflow activity for various reasons, like how many rows are processed, how much time it took, where run failed, what is the error.

Dataflow execution uses spark behind the scene. Multiple transformations are clubbed together and then executes as a single unit called stages, following insight present similar information what user see in his monitoring view. You will be able to see the information of sink times, stages for sinks, number of partitions used, number of rows etc.
Few other points

- While writing data to any tabular sinks, first data is being written to temporary tables, then from temporary tables data is being moved to target tables.
- In case of tabular sinks, if pre-SQL, Truncate, post-SQL is being used, that is going to take time, and are reported in metrics shown.
- While reading file sources, if wild card path is specified then first what files to be read are being processed.
- For file based sources, if move files option is used that also take time.
- Partitions should be used only for SQL based sources (exceptions are always there).
- To know if customer is using partitions or not search for partitionBy in <dataflowName>.dsl file in the support files, other options can be searched as well.
- Look at this public performance guide.

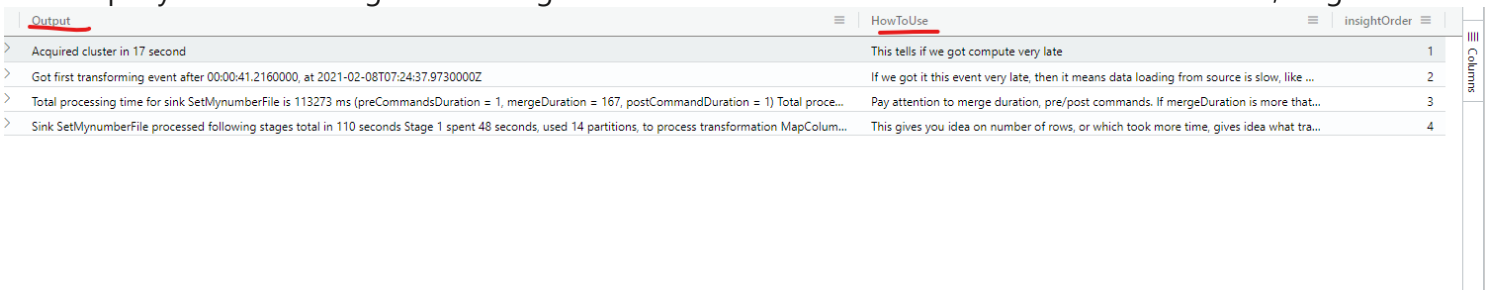


Query

To get the logs from spark execution, find activity run id (either from customer or from ActivityRuns table) and run the following query.

```
let activityRunId = "xxxxxxxxxxxx";
cluster('adfcus.kusto.windows.net').database('AzureDataFactory').GetDataflowActivityRuntimeDetails(activityRun
```

Above query will return insight on the logs. Like how much time it took to run the different sinks, stages etc.



Output	HowToUse	insightOrder
Acquired cluster in 17 second	This tells if we got compute very late	1
Got first transforming event after 00:00:41.2160000, at 2021-02-08T07:24:37.9730000Z	If we got it this event very late, then it means data loading from source is slow, like ...	2
Total processing time for sink SetMynumberFile is 113273 ms (preCommandsDuration = 1, mergeDuration = 167, postCommandDuration = 1) Total proce...	Pay attention to merge duration, pre/post commands. If mergeDuration is more that...	3
Sink SetMynumberFile processed following stages total in 110 seconds Stage 1 spent 48 seconds, used 14 partitions, to process transformation MapColum...	This gives you idea on number of rows, or which took more time, gives idea what tra...	4

How to interpret these results?

- writeStageDuration - This metrics represent the time to write the data to staging location for Synapse DW system.
- tableOperationSQLDuration - This metric represents time spent in moving data from temporary table to target table.
- preSQLDuration/postSQLDuration - This metrics represents time spent in running pre/post SQL commands.
- preCommandsDuration/postCommandsDuration - This metrics represents time spent in running any pre/post operations for file based source/sinks. For example move or delete files after processing.
- mergeDuration - This metrics represents time spent in merging the file, merge files are used for file based sinks when writing to single file or when "File name as column data" is used. If this is significant time, ask user if he can avoid these options.

Usual reason for slow performance

- SQL Source with no partitioning
- File based sources with partitioning.
- Single file output or "file name as column data" is being used.
- More rows than previous runs, contributing to more time.

To check source/sinks types

Run following query to check the logs.

```
cluster('adfcus.kusto.windows.net').database('AzureDataFactory').DataflowClusterLogs
| union cluster('adfneu.kusto.windows.net').database('AzureDataFactory').DataflowClusterLogs
| where ActivityRunId == "<activity-id>"
```

Search for text similar to following, anything above "Retrieved source details" mentioned sources, similarly sinks are logged above "Retrieved sink details".

>	2021-04-27 09:40:54.6920	Dataset Telemetry: sqlserver, AzureSqlTable, SQLAuthentication format: 'query' allowSchemaDrift validateSchema isolationLevel query
>	2021-04-27 09:40:54.6920	Dataset Telemetry: sqlserver, AzureSqlTable, SQLAuthentication format: 'query' allowSchemaDrift validateSchema isolationLevel query
>	2021-04-27 09:40:54.6930	Dataset Telemetry: sqlserver, AzureSqlTable, SQLAuthentication format: 'table' allowSchemaDrift validateSchema isolationLevel
>	2021-04-27 09:40:54.6930	Dataset Telemetry: sqlserver, AzureSqlTable, SQLAuthentication format: 'table' allowSchemaDrift validateSchema isolationLevel
>	2021-04-27 09:40:54.6940	Retrieved the source details
>	2021-04-27 09:40:54.8680	Retrieved functions and transform details
>	2021-04-27 09:40:54.8750	Transform Telemetry Transform type: source allowSchemaDrift: true validateSchema: false isolationLevel query format Transform type: source allowS
>	2021-04-27 09:40:54.8780	Dataset Telemetry: sqlserver, AzureSqlTable, SQLAuthentication format: 'table' errorHandlingOption: 'stopOnFirstError' allowSchemaDrift validateSch

Additional Information:

- Icm Reference: N/A
- Author: Anudeep Sharma
- Reviewer: Anudeep Sharma
- Keywords: Performance, Comparing 2 runs, Number of rows, number of partitions, time spend on a task, error f

How good have you found this content?