StriimWatcher: Monitoring Source OP

Many customers want an ability to retrieve monitoring data without needing to create separate apps (such as using Python or curl to run API commands). Instead, some customers want to periodically retrieve various this data in table structure. Striim Watcher is a custom Source OP, which can be used to get this data.

Installing

Current Jar File (Built on 4.2.0)

Reach out to Field Engineering for the latest source code.

Loading

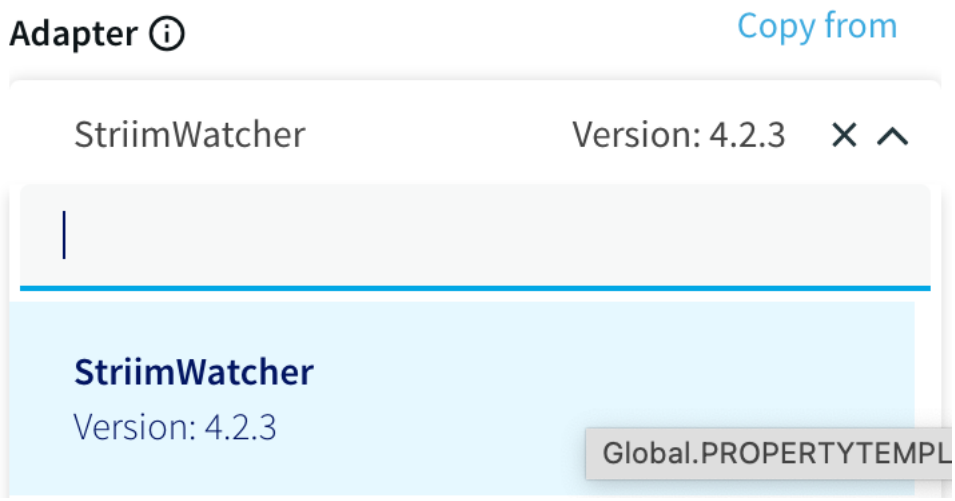
Upload into Uploaded Files and run this command:

```
LOAD OPEN PROCESSOR 'UploadedFiles/StriimWatcher-4.2.3.jar';
```

Refresh the browser UI after running this UI for it to show in the sources list.

Note about **Striim Cloud**: This takes a few minutes to appear in 'Sources'. Wait about 5 minutes, refresh the page, and it'll appear.

Drag 'Sources' (generic) over, and select 'Striim Watcher' from the Drop-Down:



Unloading

If you need to remove this, run this command:

```
UNLOAD OPEN PROCESSOR 'UploadedFiles/StriimWatcher-4.2.2.jar';
```

Performance

In testing, a simple application of StriimWatcher → Database Writer (PostgreSQL), the following was experienced, while running this with a 30 second interval (**NOTE: We do not recommend less than 120 seconds as an interval time.**):

- CPU Usage:
  - Average (while RUNNING): 0.1% / Maximum (while RUNNING): 0.127%
- Memory Usage:
  - Run ***mon <componentName> memorysize;*** on StriimWatcher component.
  - Maximum Memory Utilized: Depends on number of apps, tables, and settings.

Simple App

```
CREATE OR REPLACE APPLICATION Striim_Watcher;

CREATE OR REPLACE SOURCE myStriim_Watcher USING Global.StriimWatcher VERSION '4.2.3'
(
  StartOn: '2023-12-31T1:20:00',
  RepeatInSeconds: '300',
  IncludeNodeMonitor: true,
  IncludeNodeCluster: true,
  IncludeNodeES: true,
  IncludeAppDetail: true,
  IncludeAppDescribeDetail: true,
  IncludeAppStatusDetail: true,
  IncludeLee: true,
  IncludeTableComparisonDetail: true,
  IncludeTableComparisonDetail_SinceLastInterval: true,
  IncludeCreatedApplicationDetail: false,
  IncludeDeployedApplicationDetail: false,
  EndOn: '',
  adapterName: 'StriimWatcher',
  IncludeTypeDetailsAsOutput: false,
  IncludeComponentDetailsAsOutput: false,
  AdditionalCommandList: '',
  PreserveStriimWatcherPosition: true,
  IncludeDataWarehouseDetails: true,
  IncludeLogWatcher: true)
OUTPUT TO Striim_WatcherStream;

CREATE OR REPLACE TARGET PG_Target_Watcher USING Global.DatabaseWriter (
  ConnectionRetryPolicy: 'retryInterval=30, maxRetries=3',
  ConnectionURL: 'jdbc:postgresql://localhost:5432/customers',
  BatchPolicy: 'EventCount:10,Interval:10',
  CheckPointTable: 'CHKPOINT',
  Password_encrypted: 'false',
  CDDLAction: 'Process',
  Password: '',
  Username: 'striim',
  CommitPolicy: 'EventCount:10,Interval:10',
  StatementCacheSize: '50',
  DatabaseProviderType: 'Postgres',
  Tables: 'mon.%,public.%',
  PreserveSourceTransactionBoundary: 'false',
  adapterName: 'DatabaseWriter' )
INPUT FROM Striim_WatcherStream;

END APPLICATION Striim_Watcher;
```

Key

- **MetaData-KeyName:** Information about the MetaData.

⇒ **Parameter**: Information about the parameter

∅ **Table that gets created:** Output name of the table that gets created, starting with mon schema and table name.

METADATA

ALL STRIIM MONITORING EVENTS WILL HAVE CERTAIN METADATA. THE FOLLOWING WILL BE ON EACH EVENT:

- **TimeStamp:** The current DateTime of when the event is being created.
- **NextRun:** The next DateTime of when StriimWatcher is scheduled to be run.
- **TotalRuns:** The total number of runs StriimWatcher has executed.
- **TableName:** The StriimWatcher table name (mon.<name>).
- **OperationName:** Always INSERT for StriimWatcher.
- **ColumnCount:** The number of columns in this StriimWatcher table.

OPERATION_TS: The current DateTime of when the event is being created, in long.

EVENTS WITH APPNAME

If a Striim Monitoring table contains “App Name” as a column, then the following additional Metadata fields will be included in the event:

- **RelatedAppName:** The related Striim App name that is being monitored.

LEE EVENTS

For LEE monitoring events, the following additional differences are included in the Metadata:

- **RelatedAppName:** The related Striim App name that is being monitored.
- **RelatedSourceAppName:** The Lee Sourece related Striim App name that is being monitored.
- **RelatedTargetAppName:** The Lee Target related Striim App name that is being monitored.

PARAMETERS

Run-time Parameters

⇒ **Repeat in Seconds**: The number of seconds between each run (default 300 seconds → 5 minutes).

Data Output Parameters

The following tables will get created, depending on if a parameter is chosen.

Run History Table:

⇒ All runs will include a run history table. This table cannot be disabled.

∅ **Source Table Mapping:** `mon.striim_mon_table_runhistory`

Field Name	Description
runid	A unique bigint value for PK of the row.
batchdate	A FK reference, datetime of when this was batch was run.
runtimeEnd	The time when the StriimWatcher app finished all its internal calls.
runtimeDurationMS	The length of time, in milliseconds, for how long it took StriimWatcher from start to finish.
clustername	The 'clustername' from the startUp.properties file.
companyname	The 'companyname' from the startUp.properties file.
lastrun	Either a default start value, or the actual last run of the StriimWatcher app.
nextrun	The next planned run, based on configured value 'RepeatInSeconds'.

Include Node Monitor:

⇒ **Include Node Monitor**: Chooses whether to include a full output app list under 'Striim Applications'. (Runs the `mon;` command)

∅ **Source Table Mapping:** mon.striim_mon_node_applications

Field Name	Description
monappid	A unique bigint value for PK of the row.
batchdate	A FK reference to runtime in striim_mon_table_runhistory table, datetime of when this was batch was run.
command	The equivalent console command to gather the necessary data provided here.
montype	What type of monitoring command was run.
appname	The Striim namespace.appName of the app on the node.
status	The application state (such as: CREATED, DEPLOYED, RUNNING)
rate	The rate provided from the mon command for each app.
sourcerate	The source rate provided from the mon command for each app.
cpurate	The cpu rate provided from the mon command for each app.
numservers	List the number of servers this application is running on. Normally, this should indicate 1 or 0.
latestactivity	The latest activity seen by the application overall.

INCLUDE NODE CLUSTER:

⇒ **Include Node Cluster**: Chooses whether to include Cluster info under 'Striim Cluster Nodes'

∅ **Source Table Mapping:** `mon.striim_mon_node_cluster`

Field Name	Description
monnodeclusterid	A unique bigint value for PK of the row.
batchdate	A FK reference to runtime in striim_mon_table_runhistory table, datetime of when this was batch was run.
command	The equivalent console command to gather the necessary data provided here.
montype	What type of monitoring command was run.
nodename	The name of the node.
striimversion	The Striim version number of the node.
freemem	The amount of free memory on the server.
cpurate	The current CPU rate. NOTE: it is usually a 100%/core based; this means that an 8 core system can utilize 800% cpurate.
uptime	Indicates how long the Striim Server has been up for.

INCLUDE NODE ES:

⇒ **Include Node ES**: Chooses whether to include Elastic Search info under 'ElasticSearch'

∅ **Source Table Mapping:** `mon.striim_mon_node_elasticsearch`

Field Name	Description
monnodeesid	A unique bigint value for PK of the row.
batchdate	A FK reference to runtime in striim_mon_table_runhistory table, datetime of when this was batch was run.
command	The equivalent console command to gather the necessary data provided here.
montype	What type of monitoring command was run.
elasticsearchReceiveThroughput	Current ES receive throughput.
elasticsearchTransmitThroughput	Current ES transmit throughput
elasticsearchClusterStorageFree	ES storage free availability.
elasticsearchClusterStorageTotal	ES storage total.

INCLUDE APP DETAIL:

⇒ **Include App Detail**: Chooses whether to include detailed app information. Some columns are not included unless enabled (see below). Runs the `mon <app>;` command.

∅ **Source Table Mapping:** `mon.striim_mon_appdetail`

Field Name	Description
monid	A unique bigint value for PK of the row.
batchdate	A FK reference to runtime in striim_mon_table_runhistory table, datetime of when this was batch was run.
command	The equivalent console command to gather the necessary data provided here.
appName	The app name.
appStatus	Indicates the app status. Will not include CREATED or DEPLOYED app details unless Include Created Application Detail or Include Deployed Application Detail are enabled.
totalInput	Lists the total input count from the app (what is displayed in the UI).
totalOutput	Lists the total output count from the app (what is displayed in the UI).
isBackpressured	Boolean: indicates if the app is backpressured.
isRecoveryEnabled	Boolean: indicates if recovery is enabled.
recoverySetting	Requires **Include App Describe Detail** to be enabled. If Recovery is enabled, the recovery setting (such as 1 MINUTE INTERVAL) will be listed.
checkpointStatus	Indicates if the checkpoint is progressing, lagging, etc.
checkpointDetail	Requires **Include App Describe Detail** to be enabled. Displays the checkpoint detailed information as a nested JSONArray (Converted to String) `(i.e. [{"Source Current Position":{"CheckpointText":"{OpenSCN[5126439]-CommitSCN[5126440]-SeqNum[2]}"},"Source Name":"ADMIN:SOURCE:ORACLECDCREAD:4","Source Restart Position":{"CheckpointText":"{OpenSCN[5126439]-CommitSCN[5126440]-SeqNum[2]}"},"Application Name":"admin.PGCDCTest"}])`
isEncryptionEnabled	Requires **Include App Describe Detail** to be enabled. Boolean: indicates if encryption is enabled.
deploymentOn	Requires **Include App Status Detail** to be enabled. Displays the server(s) the app is deployed in. (i.e. S192_168_1_30)
deploymentIn	Requires **Include App Status Detail** to be enabled. Displays the deployment group the app is deployed in. (i.e. default)
appCreatedDate	Requires **Include App Describe Detail** to be enabled. The datetime the app was created within Striim.
latestActivity	The datetime of the latest activity the app has seen.

Include App Describe Detail: Include data in these columns (otherwise, these columns will be set as NULL):

⇒ **Include App Describe Detail**: Gathers `describe <app>;` information, to provide detailed information for `mon.striim_mon_appdetail`

∅ **Source Table Mapping (additional columns):** Enabling this provides additional information to these columns: `recoverySetting`, `checkpointDetail`, `isEncryptionEnabled`, and `appCreatedDate`.

Include App Status Detail: Include data in these columns (otherwise, these columns will be set as NULL):

⇒ **Include App Status Detail**: Include App Status Detail: Gathers `status <app>;` information.

∅ **Source Table Mapping (additional columns):** Enabling this provides additional information to these columns: `deploymentOn`, `deploymentIn`.

INCLUDE TABLE COMPARISON DETAIL:

The following is meant to provide a way to see the differences between tables on a source and target, along with either count only (for Initial Load), or specific CDC details (insert, update, delete, pkupdate, ddl). It should successfully do this, even for workloads that are split to multiple targets using routers or CQ logic.

⇒ **Include Table Comparison Detail**: Chooses whether to include detailed table-level information (counts for source/target insert/update/delete/pkupdate/ddl).

∅ **Source Table Mapping:** `mon.striim_mon_table_comparison`

Field Name	Description
tblcompareid	A unique bigint value for PK of the row.
batchdate	A FK reference to runtime in striim_mon_table_runhistory table, datetime of when this was batch was run.
appname	The Striim namespace.appName of the app on the node.
sourceName	The table name of the source.
targetName	The table name of the target.
srcNumOfDeletes, srcNumOfDDLs, srcNumOfPkUpdates, srcNumOfUpdates, srcNumOfInserts	The total number of Delete, DDL, PKUpdate, Update, and Insert operations seen by the source reader for sourcename (source table name).
tgtNumOfDeletes, tgtNumOfDDLs, tgtNumOfPkUpdates, tgtNumOfUpdates, tgtNumOfInserts	The total number of Delete, DDL, PKUpdate, Update, and Insert operations seen by the target reader for targetname (target table name).
diffNumOfDeletes, diffNumOfDDLs, diffNumOfPkUpdates, diffNumOfUpdates, diffNumOfInserts	The difference of the total number of Delete, DDL, PKUpdate, Update, and Insert operations for this source/target table combination. This subtracts source from target (NumOfDeletes/DDLs/PkUpdates/Updates/Inserts) to get the difference.

INCLUDE TABLE COMPARISON DETAIL _ SINCE LAST INTERVAL:

The following is meant to provide a way to see the interval differences between runs. This table will return a comparison of inserts/updates/deletes/etc as compared to the last interval. Note: It will have nothing to compare against during the first run and will not insert an initial record.

⇒ **Include Table Comparison Detail _ Since Last Interval**: Chooses whether to include the difference between this and last run's detailed table-level information (counts for source/target insert/update/delete/pkupdate/ddl).

∅ **Source Table Mapping:** `mon.striim_mon_table_comparison_sli`

Field Name	Description
tblcompareslid	A unique bigint value for PK of the row.
batchdate	A FK reference to runtime in striim_mon_table_runhistory table, datetime of when this was batch was run.
timeSinceLastBatch	How many seconds have passed since the last run.
appname	The app which has this source/target.
sourcename	The table name of the source.
targetname	The table name of the target.
srcNumOfDeletes_sli, srcNumOfDDLs_sli, srcNumOfPkUpdates_sli, srcNumOfUpdates_sli, srcNumOfInserts_sli	The total number of Delete, DDL, PKUpdate, Update, and Insert operations seen by the source reader for sourcename (source table name), since the last time it was checked (timeSinceLastBatch). If this is the first run of the application, it will not return a result (since there will be nothing to compare against).
tgtNumOfDeletes_sli, tgtNumOfDDLs_sli, tgtNumOfPkUpdates_sli, tgtNumOfUpdates_sli, tgtNumOfInserts_sli	The total number of Delete, DDL, PKUpdate, Update, and Insert operations seen by the target reader for targetname (target table name), since the last time it was checked (timeSinceLastBatch). If this is the first run of the application, it will not return a result (since there will be nothing to compare against).
diffNumOfDeletes_sli, diffNumOfDDLs_sli, diffNumOfPkUpdates_sli, diffNumOfUpdates_sli, diffNumOfInserts_sli	The difference of the total number of Delete, DDL, PKUpdate, Update, and Insert operations for this source/target table combination. This subtracts source from target (NumOfDeletes / DDLs / PkUpdates / Updates / Inserts) to get the difference. If this is the first run of the application, it will not return a result (since there will be nothing to compare against).

INCLUDE LEE:

⇒ **Include Lee**: Gathers `report lee;` and `report lee+;` information.

∅ **Source Table Mapping:** `mon.striim_mon_lee`

Field Name	Description
monleeid	A unique bigint value for PK of the row.
batchdate	A FK reference to runtime in striim_mon_table_runhistory table, datetime of when this was batch was run.
sourceApp	(If detectable) The App name related to the Type. App name is not provided in the Lee report; however, by using the sourceName, we can attempt to find the App.
sourceName	The Source Name as reported by the Lee report.
sourceType	The Source Type as reported by the Lee report.
targetApp	(If detectable) The App name related to the Type. App name is not provided in the Lee report; however, by using the targetName, we can attempt to find the App.
targetName	The Target Name as reported by the Lee report.
targetType	The Target Type as reported by the Lee report.
lagendtoend	The currently measured LAG returned from lee;
measuredAt	The time the lag was measured at.
sourcetime	The source time returned by report lee;
minlee	The min lee reported by lee. This represents the fastest end-to-end record creation and delivery, from the samplesize provided.
maxlee	The max lee reported by lee. This represents the slowest end-to-end record creation and delivery, from the samplesize provided.
avglee	The average lee reported by lee. Based on the sample size, this represents the average time for the end-to-end record creation and delivery, from the samplesize.
samplesize	The sample size used to calculate the lee statistics.

OTHER NOT REQUIRED PARAMETERS

⇒ **Start On**: A datetime that reflects when the app can start. Using a historical date will mean it runs.

- Example: `2023-12-31T1:20:00`

⇒ **End On**: A datetime that reflects when the app should stop. It can be left blank to run indefinitely.

⇒ **Include Created Application Detail:** Chooses whether to include detailed app information for CREATED status apps.

⇒ **Include Deployed Application Detail:** Chooses whether to include detailed app information for DEPLOYED status apps.

INCLUDE COMPONENT DETAILS AS OUTPUT (NOT REQUIRED):

⇒ **Include Component Details As Output**: When including this output, a raw output of all components will be included.

This includes:

- mon of APP, SOURCE, STREAM, and TARGETs.
- describe of SOURCE, TARGET and [*Optional: Include Type Details As Output as **True***] TYPE

∅ **Source Table Mapping:** `mon.striim_mon_component_output`

Field Name	Description
moncomoutid	A unique bigint value for PK of the row.
batchdate	A FK reference to runtime in striim_mon_table_runhistory table, datetime of when this was batch was run.
appName	The related application name for the json monitor data output.
componentName	The name of the component monitored.
command	Type type of command (mon or describe).
type	The type of component (APP, SOURCE, STREAM, TARGET, TYPE).
jsondata	The raw JSON output of the API response. This can be useful in instances where we need to track properties of sources or targets, by capturing the DESCRIBE data.

INCLUDE TYPE DETAILS AS OUTPUT (NOT REQUIRED):

⇒ **Include Type Details As Output**: Note: This may be more memory intensive, as it looks through all types in your system, in order to create a data map. If "Include App Detail" is enabled, it will also attempt to provide the related "App Name". Otherwise that field will be blank.

- Types of type "System" or "Global", and the types created through StriimWatcher will not be included in the output.

∅ **Source Table Mapping:** `mon.striim_mon_table_column_detail`

Field Name	Description
montblcoldtlid	A unique bigint value for PK of the row.
batchdate	A FK reference to runtime in striim_mon_table_runhistory table, datetime of when this was batch was run.
typeName	The Striim type name that was produced related to the table.
appName	The predicted app name that utilizes this type. It attempts to match by utilizing the app name + table name → type matching logic.
tableName	The predicted table name that utilizes this type. It attempts to match by utilizing the app name + table name → type matching logic.
createdDate	When the type was created. By default, StriimWatcher will only produce output entries on first run, and when a type's createdDate has changed.
columnName	The column name from the table.
columnType	The column type from the table.
isPK	Whether this column is a primary key column.

INCLUDE DATA WAREHOUSE DETAILS (NOT REQUIRED):

⇒ **Include Data Warehouse Details**: For each target that is a DataWarehouse and includes any of the following fields, it will be captured and reported as an event.

∅ **Source Table Mapping:** `mon.striim_mon_datawarehouse_detail`

Field Name	Description
dwdid	A unique bigint value for PK of the row.
batchdate	A FK reference to runtime in striim_mon_table_runhistory table, datetime of when this was batch was run.
appName	The associated app name (or N/A)
sourceName	The associated source name.
targetName	The associated target name.
target_adaptername	The target adapter name.
projectId	The projectId (if applicable).
Mode	The mode (APPENDONLY or MERGE).
streamingUpload	Boolean, if streamingUpload is enabled.
StreamingConfiguration	StreamingConfiguration details, if enabled.
optimizedMerge	Boolean, if optimizedMerge is enabled.
batch_event_count	The target adapter's eventcount batch setting.
batch_interval	The target adapter's interval batch setting.
(other columns in Description)	Every column from mon <target> for the data warehouse targets with data is included: total_batches_created INT, partition_pruned_batches INT, last_successful_merge_time TIMESTAMP, total_batches_ignored INT, max_integration_time_ms INT, avg_in_mem_compaction_time_ms NUMERIC, avg_batch_size_bytes INT64, avg_event_count_per_batch NUMERIC, min_integration_time_ms INT, total_batches_queued INT, avg_compaction_time_ms NUMERIC, avg_waiting_time_in_queue_ms NUMERIC, avg_integration_time_ms NUMERIC, total_batches_uploaded INT, avg_merge_time_ms NUMERIC, avg_stage_resources_mgmt_time_ms NUMERIC, avg_upload_time_ms NUMERIC, -- Fields from "Last batch info" last_batch_updates INT, last_batch_event_count INT, last_batch_inserts INT, last_batch_max_record_size_bytes INT64, last_batch_total_events_merged INT, last_batch_ddls INT, last_batch_sequence_number INT, last_batch_size_bytes INT64, last_batch_deletes INT, last_batch_pk_updates INT, last_batch_accumulation_time_ms INT, -- Fields from "Integration Task Time" (within "Last batch info") last_batch_compaction_time_ms INT, last_batch_stage_resources_mgmt_time_ms INT, last_batch_upload_time_ms INT, last_batch_merge_time_ms INT,

	last_batch_in_mem_compaction_time_ms INT, last_batch_pk_update_time_ms INT, last_batch_ddl_execution_time_ms INT, last_batch_total_integration_time_ms INT

INCLUDE LOG WATCHER (NOT REQUIRED):

⇒ **Include Type Details As Output**: This attempts to look at the striim.server.log, look for errors, and report those as an output event. The position is saved while the app is running (if Preserve Striim Watcher Position flag is utilized, it will save across restarts of the app). It will report the 20 lines before and the 20 lines after an ERROR is detected in the striim.server.log.

∅ **Source Table Mapping:** `mon.striim_mon_log_watcher`

Field Name	Description
errorid	A unique bigint value for PK of the row.
batchdate	A FK reference to runtime in striim_mon_table_runhistory table, datetime of when this was batch was run.
log_date	The parsed log entry date.
server	The deployed server, if applicable.
appName	If an app was related and detected to this error, it will report it here.
log_level	This is the ERROR log level.
message	This includes the reported error event.
contextbuffertext	This includes the 20 lines before the error, the actual error, and the 20 lines after the error for context.

ADDITIONAL COMMAND LIST (NOT REQUIRED):

⇒ **Additional Command List (String)**: This is an optional semi-colon separated list of extra commands, with an optional interval in seconds provided. The syntax is:

```
<firstCommand>{optionalWaitIntervalInSeconds};<secondCommand>{optionalWaitIntervalInSeconds}
```

- Values:
  - **commandList**: You can provide a semi-colon separated list of commands to run.
    - **Note**: mon, describe, status, report, list, meter, show, and usage are the only allowed commands.
  - **optionalWaitIntervalInSeconds**: This interval (`optionalWaitIntervalInSeconds`) should be a multiple of the app's "Repeat in Seconds" interval. The app will only execute every "Repeat in Seconds" time, so if you choose "Repeat in Seconds" of "60" and provide a `optionalWaitIntervalInSeconds` of 90, it will only run every 2 minutes. If left blank, it will utilize the "Repeat in Seconds" option from StriimWatcher parameters.
- Example: `mon admin.StriimWatcher memorysize{600};usage;`
  - In this example, it will run the first command mon admin.StriimWatcher memorysize every 600 seconds. It will run usage every "Repeat in Seconds" provided by the app (every instance of StriimWatcher).
- Extended Functionality: It also supports a range of find->replace options, to utilize in place of source, target, or app name:
  - `%source-all%`

        - This will replace any instance of this with every source component name in Striim, regardless of app state.
    - `%source-running%`
        - This will replace any instance of this with every source component name in Striim within an app that is running.
    - `%target-all%`
        - This will replace any instance of this with every target component name in Striim, regardless of app state.
    - `%target-running%`
        - This will replace any instance of this with every target component name in Striim within an app that is running.
    - `%app-all%`
        - This will replace any instance of this with every app name in Striim, regardless of app state.
    - `%app-running%`
        - This will replace any instance of this with every app name in Striim within an app that is running.
    - Example to monitor all source component memory size of all running apps every 15 minutes (900 seconds):
        - `mon %source-all% memorysize{900}`

∅ **Source Table Mapping:** `mon.striim_mon_component_output` with the following details into this table:

- **appName** → blank
- **componentname** → This will be the full command that is provided.
- **command** → the type (mon, describe, status, report, list, meter, show, and usage are the only allowed commands)
- **type** → CUSTOM
- **jsondata** → the raw response from the provided command

PRESERVE STRIIM WATCHER POSITION (NOT REQUIRED):

⇒ **Preserve Striim Watcher Position**: If true, StriimWatcher will attempt to save the historical table row counts, as well as the last run time. Upon restart of StriimWatcher where this is true, it will re-load the data, and continue from that point.

- Note that this attempts to save it as a file under the Striim installation directory as 'StriimWatcher.hist'.