[SAP ODP] SAP incremental load doesn't work as expect

Last updated by | Zhangyi Yu | Feb 28, 2023 at 7:06 PM PST

Contents

- Issue
- Root Cause
- Resolution
- Additional Information:

Issue

Today ADF dataflow support help customers extract delta data/logs from different data sources (e.g., SAP/Cosmos/Snowflake/...) while leverage dataflow its check-pointer/state management (invisible to customers) to store previous activity execution status (succeed/failed/cancelled/...), then serve next activity execution.

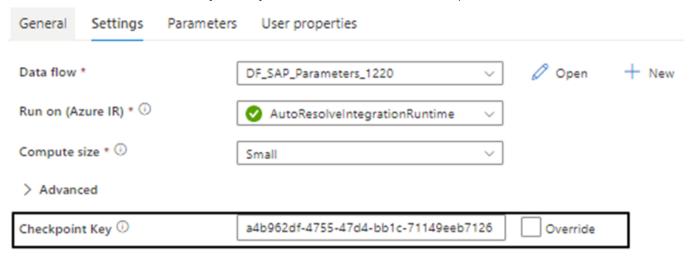
For customers who are using ADF dataflow with CDC connectors, sometime they may find connector incremental-load (delta-load) does not work as expected, for example:

- Case 1: The customer manually builds pipeline with pattern "lookup/web + foreach/IfCondition + dataflow" to replicate multiple tables from SAP. Sometimes incremental-load works fine, sometimes not. But if just run single SAP activity with same setting, incremental-load works fine.
- Case 2: The customer manually builds single pipeline to extract sap incremental data, it worked fine for a long time. But suddenly incremental-load doesn't work.

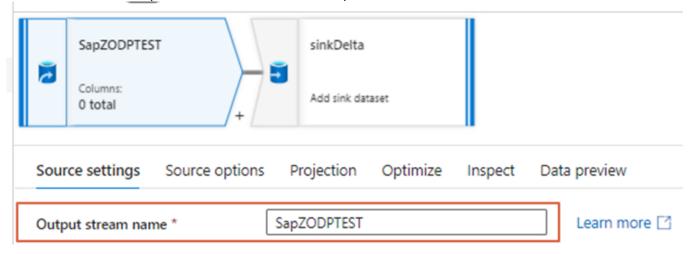
Root Cause

Today ADF dataflow automatically manages incremental-load (delta-load) status based on checkpoint while it has two parts:

- Checkpoint-Key:
 - Activity level setting



- ADF will automatically generate and maintain this value (datafactoryld + pipelineName + activityName) by default. At the same time, ADF also allows customer to control it by click the "override" button to providing customized value.
- Checkpoint-Value: it is dictionary with key-value pairs:
 - Key is source-stream-name (dataflow source setting, provide by customer, one dataflow reference can have one or multiple sources, one source is map/bind with one SAP table)



• Value is the status of read SAP source data + write sink. Based on this value, we can help customer auto recovery in next time activity execution from failure no matter it is read-source-failed or writesink-failed.

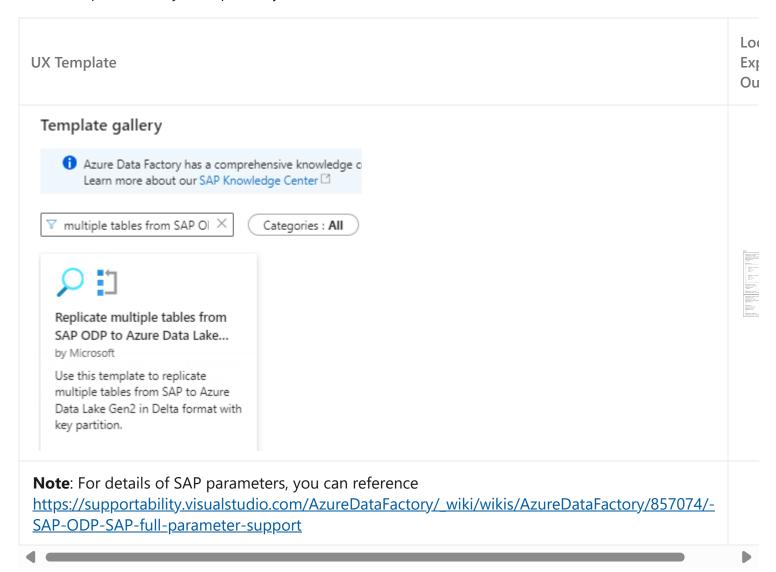
Note: If customer wrongly change the value of checkpoint-key or source-stream-name, it will lead to incremental-load (delta-load) status lose and restart from initial load again.

Case 1: If customer builds pipeline with pattern "lookup/web + foreach/IfCondition+ dataflow", then several dataflow activity executions (same activity name, one execution one SAP table) are triggered. The incremental status of table-1 will be override by table-2 because checkpoint-key is the same + sourcestream-name is the same, then it always has initial data load, for example:

Dataflow Activity (checkpoint key1, source stream1) => Execution-1 => CDC status 1 (checkpoint key1, source st Dataflow Activity (checkpoint key1, source stream1) => Execution-2 => CDC status 2 (checkpoint key1, source st • Case 2: Customer wrongly updates/modify value of checkpoint-key or source-stream-name in existed successful dataflow. For new source-stream-name or checkpoint-key, there is no incremental-load status stored in ADF micro-service, so it will run initial load to start new round of incremental-load again.

Resolution

- For case-1, we strongly suggest customers to use ADF UX template https://learn.microsoft.com/en- us/azure/data-factory/solution-template-replicate-multiple-objects-sap-cdc 2, that is because it can simplify user experience while all parameters (checkpointkey/sapContext/sapObject/sapRunMode/sapKeyColumns/sapPartitions/...) are already configured for customer.
 - If customer wants to keep delta as sink, he just need to provide stage LS, sap LS + one file (contains each SAP table settings + delta sink settings) used by lookup activity.
 - If customer wants to use other store as sink, he can feel free to choose other sink store LS and update up file used by lookup activity.



• For case-2: Don't change the source-stream-name or checkpoint-key on existed/succeed pipeline unless you want to forcedly have initial load to start new round of incremental-load again.

Additional Information:

• Icm Reference: N/A • Author: Zhangyi Yu

• Reviewer: Zhuoyang Zhang