

# Dynamics 365 Business Central & Azure data Lake

#### DISCLAIMER



This presentation is provided for informational and training purpose only. It represents Microsoft's view as of the presentation date. Microsoft cannot guarantee the accuracy of any information after the presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted as a commitment on Microsoft's part. This presentation is provided "as-is".

Sample code included in this presentation is made available AS IS.

• © Copyright Microsoft Corporation. All rights reserved

## **Today Speakers!!**



**Stefano Demiliani**MVP, D365 Business Central
<a href="https://www.linkedin.com/in/stefano-demiliani/">https://www.linkedin.com/in/stefano-demiliani/</a>



Henri Schulte
Cloud Solution Architect, Data & Al
https://www.linkedin.com/in/henrischulte/



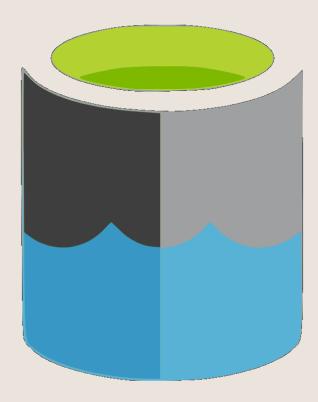
**Giuseppe Russo**Cloud Solution Architect, Business Applications <a href="https://www.linkedin.com/in/bepperusso">https://www.linkedin.com/in/bepperusso</a>



**Soumya Dutta**Cloud Solution Architect, Business Applications
<a href="https://www.linkedin.com/in/soumya-dutta-07813a5/">https://www.linkedin.com/in/soumya-dutta-07813a5/</a>



#### What is the data lake?



#### The problem

Many organizations have spent the last two decades building data warehouses and business intelligence (BI) solutions based on relational database systems.

Many BI solutions have lost out on opportunities to store unstructured data due to cost and complexity in these types of data in databases.

A data lake provides <u>file-based storage</u>, usually in a distributed file system that supports high scalability for massive volumes of data. Organizations can store structured, semi-structured, and unstructured files in the data lake and then consume them from there in big data processing technologies

Azure Data Lake Storage Gen2 provides a cloud-based solution for data lake storage in Microsoft Azure

## **Understanding Azure Data Lake Storage Gen2**

**Azure Data Lake Storage** is a comprehensive, massively scalable, secure, and cost-effective data lake solution for high performance analytics built into Azure.

Azure Data Lake Storage combines a file system with a storage platform to help you quickly identify insights into your data. Data Lake Storage builds on Azure Blob storage capabilities to optimize it specifically for analytics workloads.

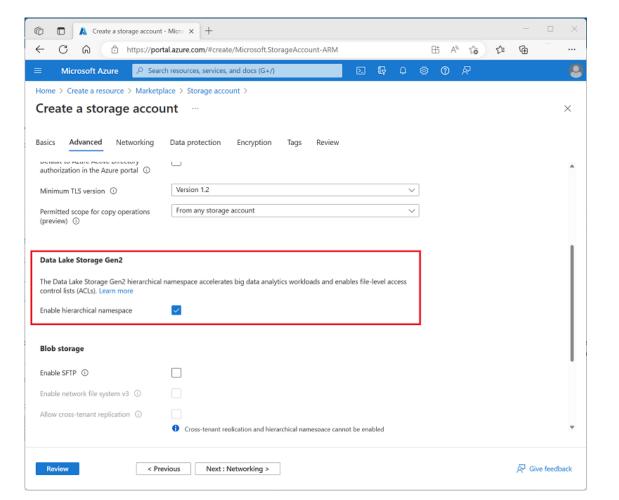
#### **Benefits:**

- Data Lake Storage is designed to deal with variety and volume of data at exabyte scale while securely handling hundreds of gigabytes of throughput.
- You can store the data in one place and access it through compute technologies including Azure Databricks, Azure
  HDInsight, and Azure Synapse Analytics without moving the data between environments.
- Ability to use storage mechanisms such as the parquet format, which is highly compressed and performs well across multiple platforms using an internal columnar storage.
- Security (directory or file level)
- Azure Data Lake Storage organizes the stored data into a hierarchy of directories and subdirectories, much like a file system, for easier navigation. As a result, data processing requires less computational resources, reducing both the time and cost.
- Data Lake Storage takes advantage of the Azure Blob replication models (LRS, GRS etc)

## Enabling Azure Data Lake Storage Gen2 in Azure Storage

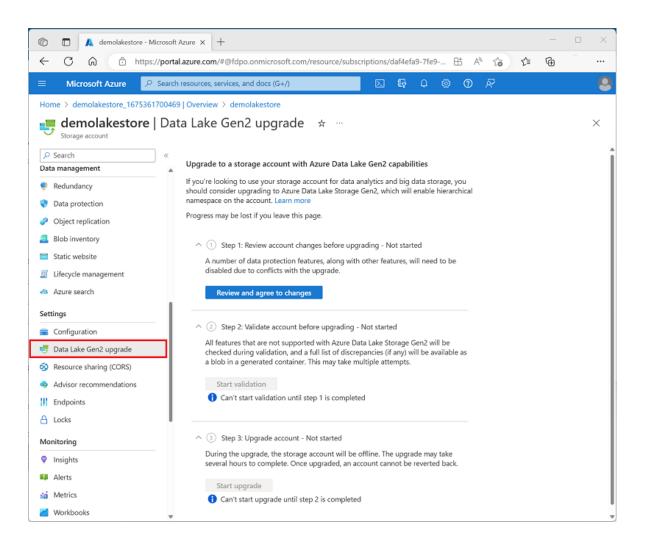
Azure Data Lake Storage Gen2 <u>IS NOT</u> a standalone Azure service, but <u>IS a configurable capability</u> of a **StorageV2** (**General Purpose V2**) Azure Storage.

To enable Azure Data Lake Storage Gen2 in an Azure Storage account, you can select the option to **Enable hierarchical namespace** in the **Advanced** page when creating the storage account in the Azure portal:



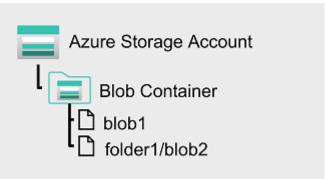
### Enabling Azure Data Lake Storage Gen2 in Azure Storage

If you already have an Azure Storage account and want to enable the Azure data Lake Storage Gen2 capability, you can use the **Data Lake Gen2 upgrade** wizard in the Azure portal page for your storage account resource:

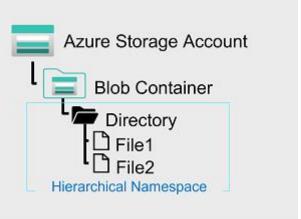


#### Azure Data Lake Storage vs Blob Storage

In **Azure Blob storage**, you can store large amounts of unstructured data in a flat namespace within a blob container.



**Azure Data Lake Storage Gen2** builds on blob storage and optimizes I/O of high-volume data by using a hierarchical namespace that organizes blob data into *directories*, and stores metadata about each directory and the files within it:



#### **Scenarios**



Big data analytical workloads that involve massive *volumes* of data in a *variety* of formats that needs to be processed at a fast *velocity* 



Data warehousing: data is extracted from operational data stores (Azure, Azure Cosmos DB etc., and transformed into structures more suitable for analytical workloads:

Data extracted from operational data sources and load it into a data lake hosted in an Azure Data Lake Storage Gen2 container. The data is then processed and loaded into a relational data warehouse in an Azure Synapse Analytics dedicated SQL pool, from where it can support data visualization and reporting using Microsoft Power BI.



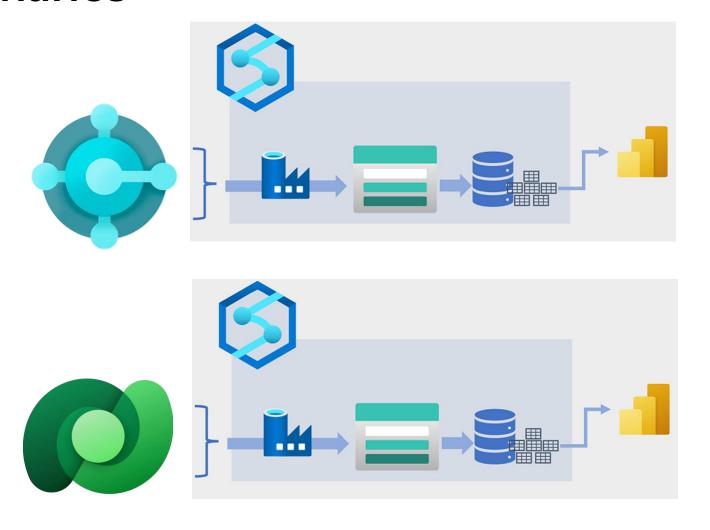
#### Real-time data analytics:

Azure Stream Analytics enables you to create *jobs* that query and aggregate event data as it arrives, and write the results in an output *sink*. One such sink is Azure Data Lake Storage Gen2; from where the captured real-time data can be analyzed and visualized.



Data science and Machine Learning

#### **Scenarios**





#### The bc2adls idea



## Challenges

#### Storage in BC could be expensive

Azure data lake	Business central storage
Price rates (https://azure.microsoft.com/en-us/pricing/) Azure data lake (Hot tier) Storage: starts at \$0.0184 per GB/month 10000 Write operations: \$0.065 10000 Read operations: \$0.0052  Azure Synapse Pipelines	Starts @ ~\$10 per GB per month
1000 Pipeline runs: \$1	
Data flows: \$0.325 per vCore-hour	

Limited ability to combine with other data sources Limited ability to do analytics and machine learning

#### Our proposed solution: bc2adls



- Easily export BC data to a Data Lake, incl. table extensions
- Little traffic: only export incremental changes
- Get unified data for analytics on Azure
- Store data in CDM format as CSV or parquet
- Visualize data in Power BI or query with Synapse Serverless SQL
- Powered by low-code, easy-to-maintain Azure components





#### How does bc2adls work?



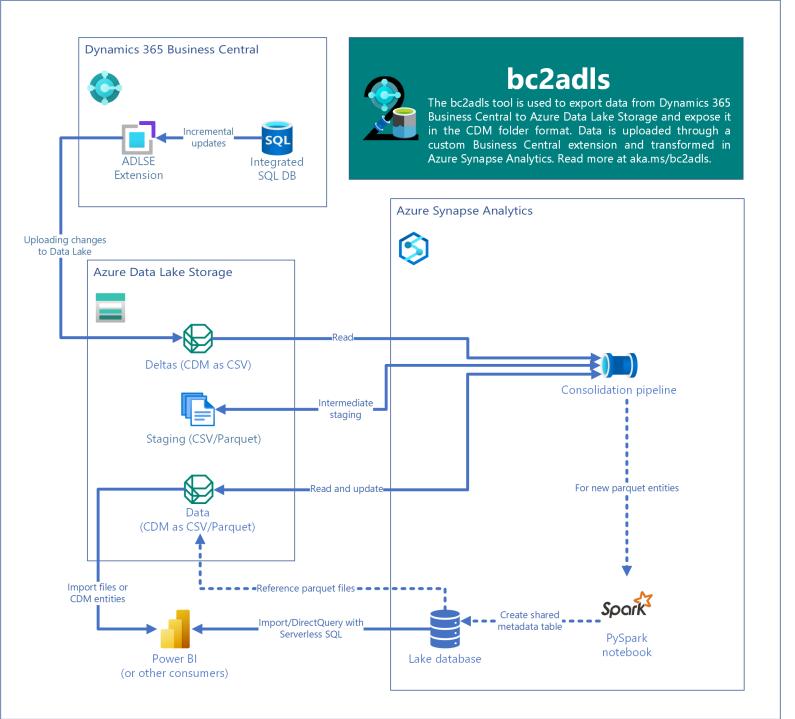




Step 1: The BC extension tracks new changes and uploads them to an Azure Data Lake.

Step 2: The Synapse Pipeline reads the incremental changes and applies them to your data.

Step 3: Your consolidated data is ready to be queried by Power BI, Serverless SQL, etc.





## Ok, but how does it actually work?

#### The Common Data Model (CDM) format

Learn more <a href="learn.microsoft.com/en-us/common-data-model/">learn.microsoft.com/en-us/common-data-model/</a>

#### Supported in,

- Power BI data flows integration
- CDM format in Azure Data Factory & Synapse Analytics
- Azure Synapse Spark CDM Connector



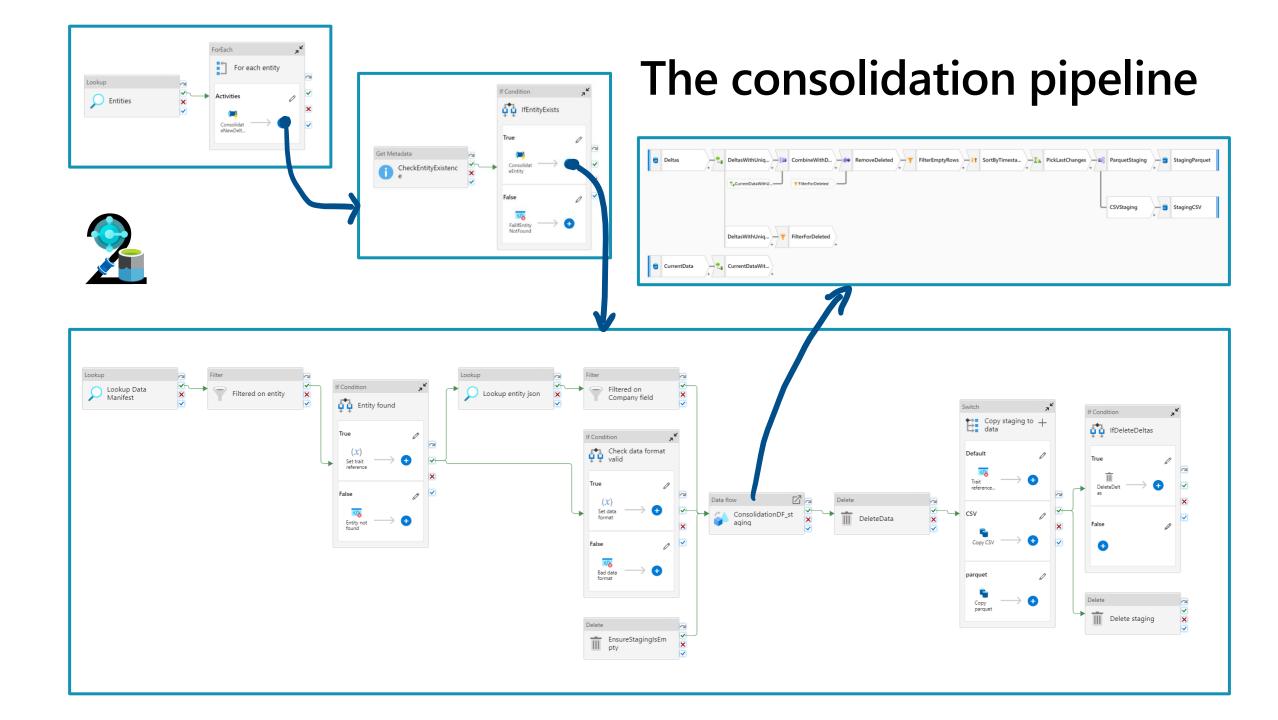


								Sandbox 🔎	Ů 🕸
Export to Azure Data Lake Storage				<b>→</b> +	Î			✓ Saved	] [] <sub>7</sub> [
business-centralz									
Dadiness certiful									
Export Stop export	Schedule export 🔀 Clear tracked delete	ed records 🔑 Clear exec	cution log						
Setup									
Account				Execution					
Container	b ionicos ilz	E rantes IlZ			ad size (MiBs)				4
Tenant ID		ah-20. cc. 11.21.11.		CDM data	format · · · · · · · · · · · · · · · · · · ·	Parquet			~
Account name · · · · · · · · · · · · · · · · · · ·	bs` " a			Multi- com	npany export · · · · · · · · · · · · · · · · · · ·				
App registration				I					
App registration				Emit telem	netry				
Client ID									
Client secret									
Client secret									· · · · · · · · · · · · · · · · · · ·
Client secret									년 E3
Client secret  Tables Manage		# Fields selected	Entity name		Last exported state	Last started at	Last error		
Tables Manage	rose fields 🔊 Reset						Last error		
Client secret  Tables Manage	oose fields 🔊 Reset					, -,	Last error		
Tables Manage	rose fields & Reset  Enabled	124 106	PurchRcptLine-121		Success	10/6/2022 11:46 AM	Last error		
Tables  Manage  *Add Delete	oose fields   Reset  Enabled	124 106 121	PurchRcptLine-121 PurchInvHeader-122		Success Success	10/6/2022 11:46 AM 10/6/2022 11:46 AM	Last error		
Tables Manage  ** Add Delete ** Choo  Table  Purch. Rcpt. Line  Purch. Inv. Header  Purch. Inv. Line	oose fields Reset  Enabled  —	124 106 121 72	PurchRcptLine-121 PurchInvHeader-122 PurchInvLine-123		Success Success	10/6/2022 11:46 AM 10/6/2022 11:46 AM 10/6/2022 11:46 AM	Last error		
Tables  Manage  Add Delete Coco  Table  Purch. Rcpt. Line  Purch. Inv. Header  Purch. Inv. Line  Job	oose fields Reset  Enabled  Compared to the co	124 106 121 72	PurchRcptLine-121 PurchInvHeader-122 PurchInvLine-123 Job-167		Success Success Success	10/6/2022 11:46 AM 10/6/2022 11:46 AM 10/6/2022 11:46 AM 10/6/2022 11:46 AM	Last error		
Tables  Manage  ** Add Delete	Prose fields Reset    Enabled	124 106 121 72 5	PurchRcptLine-121 PurchInvHeader-122 PurchInvLine-123 Job-167 NoSeries-308		Success Success Success Success Success	10/6/2022 11:46 AM 10/6/2022 11:46 AM 10/6/2022 11:46 AM 10/6/2022 11:46 AM 10/6/2022 11:46 AM	Last error		

## Step 1. Inside the Data Lake: After the export

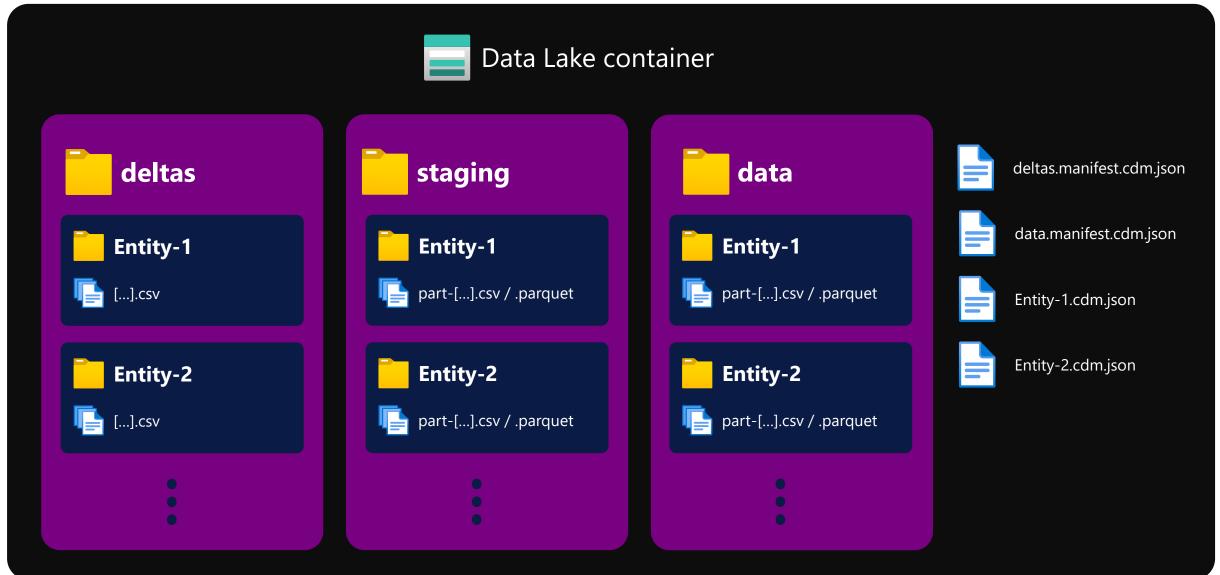






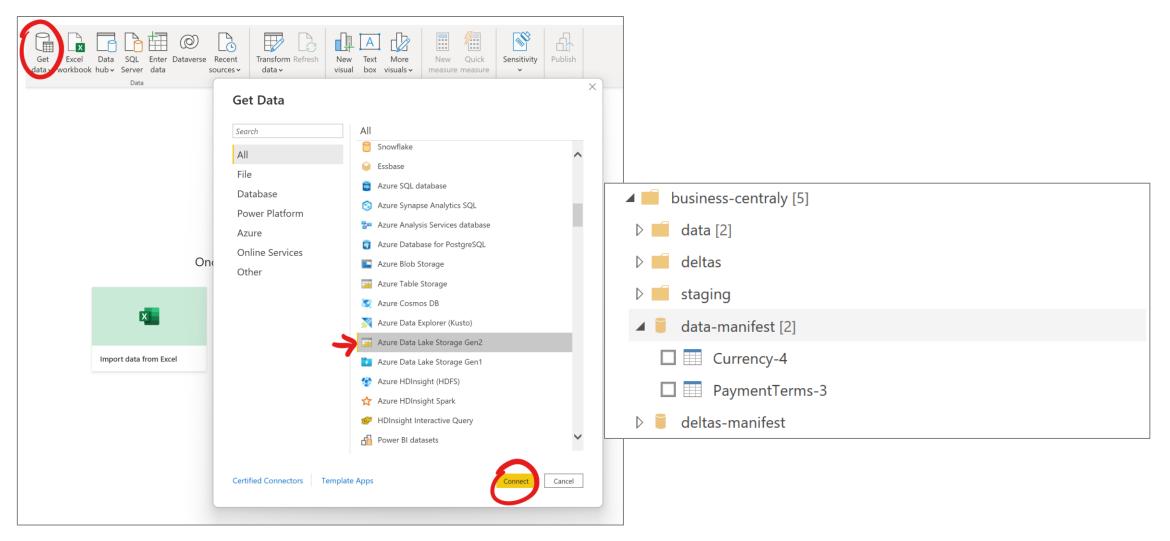
## Step 2. Inside the Data Lake: After the pipeline





## Step 3. Consume the data in Power BI





#### Where's the catch?





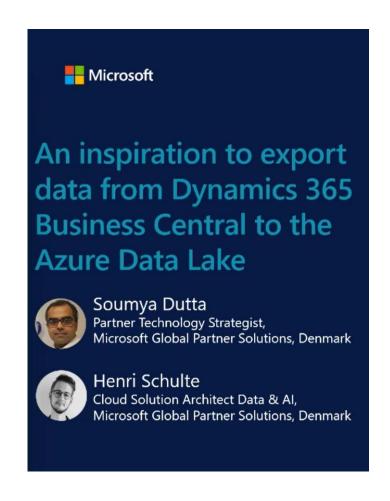
bc2adls is an open-source prototype that is not officially supported or maintained by Microsoft!

It's under active development and things may change frequently!

If Business Central receives an official export feature, it will likely be different and there may not be an upgrade path!

You can reach us at <a href="mailto:bc2adls@microsoft.com">bc2adls@microsoft.com</a>

#### For more details, watch the webinar from Jan 2022





## What's new in bc2adls?

Updates since last webinar (January 2022)



#### Exports from the read-only replica

- Exports data using the DataAccessIntent property
- Keeps your production workloads un-impacted

## **Performance improvements**



Data export from companies

Synapse consolidation

## Storing data in the Parquet format

- Columnar format
- Includes data types
- More robust data fidelity to BC
- Supports more efficient reads



### T-SQL: Querying from Synapse Serverless SQL

- Query your ADLS data through familiar SQL syntax
- Based on parquet, makes for efficient data retrieval
- How to create shared metadata tables: github.com/microsoft/bc2adls/blob/main/.assets/SharedMetadataTables.md



# Demo

## Bug fixes and changelog

#### Changelog

This page lists all pull requests that made significant changes to bc2adls.

Pull request	Changes
79	The step to clean up tracked deleted records from the export process has now been removed to make exports more efficient. This clean up step can instead be performed either by clicking on the action <b>Clear tracked deleted records</b> on the main setup page, or by invoking the new codeunit <b>ADLSE Clear Tracked Deletions</b> through a low- frequency custom job queue entry.
78	Upgrading to new versions may lead the export configuration to enter an incorrect state, say, if a field that was being exported before gets obsoleted in the new version. This fix prevents such an occurence by raising an error during the upgrade process. If corrective actions, say, disabling such fields are not taken after multiple upgrade attempts, the bc2adls extension is uninstalled and upgrade is forced. A subsequent re-install of the extension will then disable such tables from being exported, so that the user can then react to the change in schema later on.
56	The table ADLSE Run has now been added to the retention policy so that the logs for the executions can be cleared periodically, thus taking up less space in the database.
55	A much awaited request to allow the BC extension to read from the replica database saves up resources that can otherwise be dedicated to normal ERP operations, has now been implemented. This change is dependent on the version 21 of the application.
	The default rounding principles caused the consolidated data to have a maximum of two decimal places even though the data

## Looking ahead to the future

No commitments

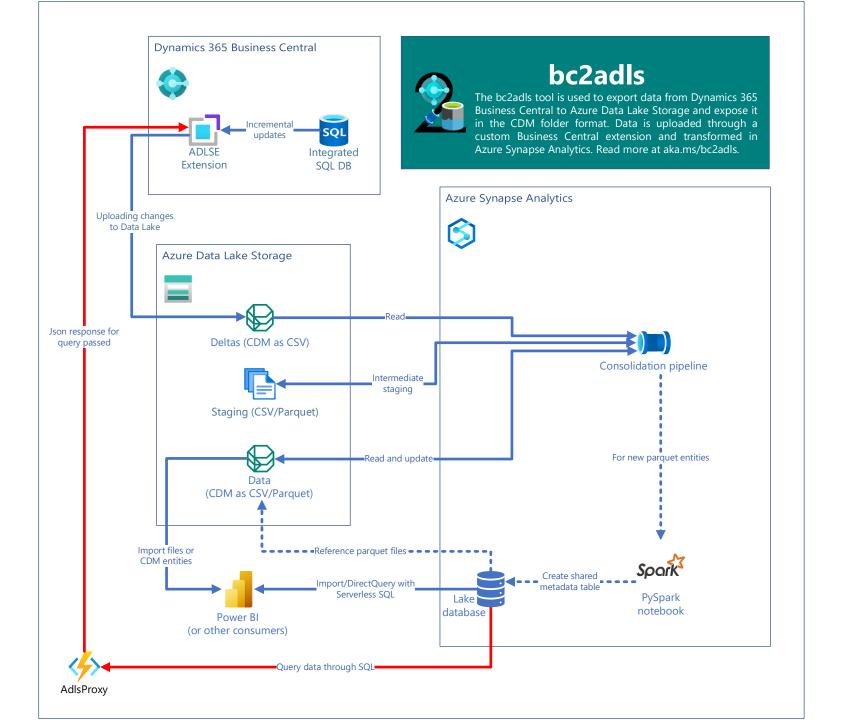
We are looking towards you to validate the features



#### 1. Querying data residing in the lake with bc2adls

#### Use cases

- Refer to archived and removed data
- External data needs to be used in Business Central calculations
- Data lake as a cheap storage for miscellaneous tabular data





#### 1. Querying data residing in the lake with bc2adls

#### BC tables

```
// first set the table to be queried
ADLSEQueryTable.Open(Database::"Cust. Ledger Entry");
// set the filters to be applied
ADLSEQueryTable.SetRange(CustLedgerEntry.FieldNo("Document Type"), "Gen. Journal Document Type"::Invoice);
ADLSEQueryTable.SetFilter(CustLedgerEntry.FieldNo("Customer No."), '>=%1', '40000');
ADLSEQueryTable.SetFilter(CustLedgerEntry.FieldNo("Posting Date"), '>%1', 20211023D);
// set the result to be sorted first by descending order of Posting Date and then ascending order of Entry No.
ADLSEQueryTable.SetOrderBy(CustLedgerEntry.FieldNo("Posting Date"), false);
ADLSEQueryTable.SetOrderBy(CustLedgerEntry.FieldNo("Entry No."));
// make the find set query
if ADLSEOueryTable.FindSet() then
    // records found
    repeat
        EntryNo := ADLSEQueryTable.Field(CustLedgerEntry.FieldNo("Entry No.")); // get an integer value
        CustomerNo := ADLSEQueryTable.Field(CustLedgerEntry.FieldNo("Customer No.")); // get a code value
        DocumentType := Enum::"Gen. Journal Document Type".FromInteger(
            ADLSEQueryTable.Field(CustLedgerEntry.FieldNo("Document Type"))); // get an enum value
        PostingDate := ADLSEQueryTable.Field(CustLedgerEntry.FieldNo("Posting Date")); // get a date value
        ModifiedDateTime := ADLSEQueryTable.Field(CustLedgerEntry.FieldNo(SystemModifiedAt)); // get a datetime value
        Timestamp := ADLSEQueryTable.Field(0); // get a big integer
        RecordsFound += 1:
    until not ADLSEQueryTable.Next(); // Next will return false when there are no more records in the result set
Message('Records found: %1.', RecordsFound);
// first set the table to be queried
ADLSEQueryTable.Open(Database::"G/L Register");
if ADLSEQueryTable.FindSet() then
    repeat
        GLRegisterNo := ADLSEQueryTable.Field(GLRegister.FieldNo("No.")); // get an integer
        CreationTime := ADLSEQueryTable.Field(GLRegister.FieldNo("Creation Time")); // get a time
    until not ADLSEQueryTable.Next(); // Next will return false when there are no more records in the result set
```

#### 1. Querying data residing in the lake with bc2adls

#### Other CDM data

```
// initialize with the table name
ADLSEQuery.Init('custledgerentry 21');
// optionally set the connection string
ADLSEQuery.SetServer('bc2adlspipelinetemp-ondemand.sql.azuresynapse.net');
ADLSEQuery.SetDatabase('bc query2');
// set filtering
ADLSEQuery.SetRange('DocumentType-5', 'Invoice');
ADLSEQuery.SetFilter('CustomerNo-3', '>=%1', '40000');
ADLSEQuery.SetFilter('PostingDate-4', '>%1', 20211023D);
ADLSEQuery.SetRange('$Company', 'CRONUS USA, Inc.');
// set sequence of results
ADLSEQuery.SetOrderBy('PostingDate-4', false);
ADLSEQuery.SetOrderBy('EntryNo-1');
// set fields to be fetched
ADLSEQuery.AddLoadField('EntryNo-1');
ADLSEQuery.AddLoadField('CustomerNo-3');
ADLSEOuerv.AddLoadField('DocumentType-5');
ADLSEQuery.AddLoadField('PostingDate-4');
ADLSEQuery.AddLoadField('SystemModifiedAt-2000000003');
ADLSEQuery.AddLoadField('timestamp-0');
// make the find set query
if ADLSEQuery.FindSet() then
    // records found
    repeat
        EntryNo := ADLSEQuery.Field('EntryNo-1').AsInteger(); // get an integer value
        CustomerNo := ADLSEQuery.Field('CustomerNo-3').AsCode(); // get a code value
        DocumentType := ADLSEQuery.Field('DocumentType-5').AsText(); // get a text value
        PostingDate := DT2Date(ADLSEQuery.Field('PostingDate-4').AsDateTime()); // get a date value
        ModifiedDateTime := ADLSEOuery.Field('SystemModifiedAt-2000000003').AsDateTime(); // get a datetime value
        Timestamp := ADLSEQuery.Field('timestamp-0').AsBigInteger(); // get a big integer value
        RecordsFound += 1:
    until not ADLSEQuery.Next(); // Next will return false when there are no more records in the result set
Message('Records found: %1.', RecordsFound);
```

### 1. Querying data residing in the lake with bc2adls

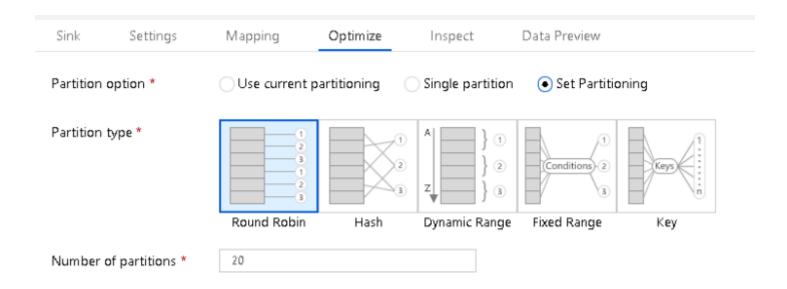
Use for non-business critical operations!

• Read more github.com/microsoft/bc2adls/blob/ReadDataFromLakeToBC/.assets/QueryLakeData.md

• Pull request github.com/microsoft/bc2adls/pull/101

### 2. New partitioning strategy

<u>Dataflow partitioning by HenriSchulte-MS · Pull Request #108 · microsoft/bc2adls (github.com)</u>



### **Frequently Asked Questions**

github.com/microsoft/bc2adls/blob/ main/.assets/FAQs.md

### Other Issues

github.com/microsoft/bc2adls/issues

Here you will find answers to the most frequently asked questions. Please also refer to the issues site to know more or to ask your own questions in the community.

#### How do I run the export to the lake in a recurring schedule?

The Job Queue feature in Business Central is used to schedule background tasks in a periodic way. You may invoke the Codeunit ADLSE Execution through the feature to export the data increments to the lake as a scheduled job. You may click Schedule export on the main setup page to create this job queue.

#### How should I distribute the BC data to my lake?

We recommend that a data lake container holds data only for **only one** Business Central environment. After copying environments, ensure that the export destination on the setup page on the new environment points to a new data lake container.

#### Can I export calculated fields into the lake?

No, only persistent fields on the BC tables can be exported. But, the issue #88 describes a way to show up those fields when consuming the lake data.

#### How can I export BLOB data to the lake?

Data from blob fields in tables are not exported today to the lake. It should be possible however to convert the (possibly, binary) data to text using the Codeunit Base64 Convert and then store it as a separate field in a new table and exporting it to the lake using the bc2adls solution.

#### How do I export some tables at a different frequency than the rest?

Normally, all the tables that are setup for export will export at the same time. However you may invoke exports of selected tables by using the API pages available. The issue #87 describes this possibility in more detail.

#### How do I track the files in the deltas folder in my data lake container?

Incremental exports create files in the deltas folder in the lake container. Each such file has a Modified field that indicates the time when it was last updated, in other words, when the export process finished with that file. Each export process for an entity and in a company logs its

### **Testimonials**

## Magnus digital



Fellowwind

twoday kapacity

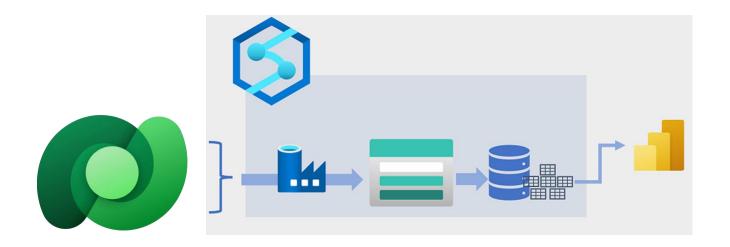
Share YOUR experience with us at <a href="mailto:bc2adls@microsoft.com">bc2adls@microsoft.com</a>



# The Azure Synapse Link for Dataverse



### Scenario



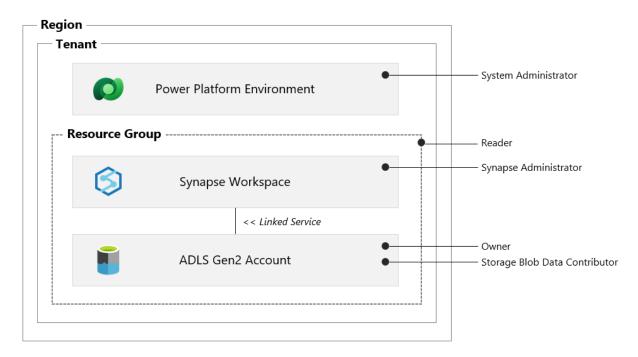
**Azure Synapse Link for Dataverse** provides a continuous pipeline of data from Dataverse to Azure Synapse Analytics or Azure Data Lake Storage Gen2. Once the link is established, the service will export all the initial data as well as sync any incremental changes, enabling customers to glean near real-time insights from data sourced from the Microsoft Dataverse service.

You can use Azure Synapse Link to continuously export data from:

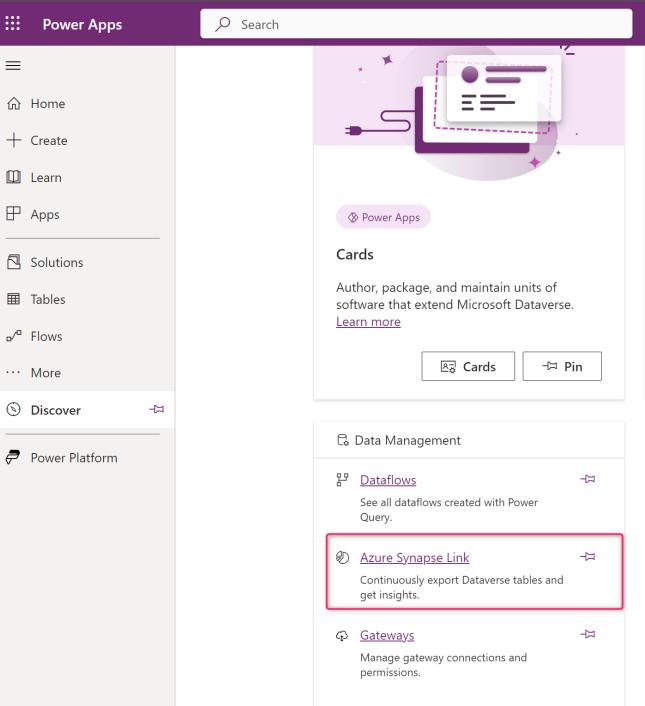
- Dataverse to Azure Synapse Analytics
- Dataverse to Azure Data Lake Storage Gen2

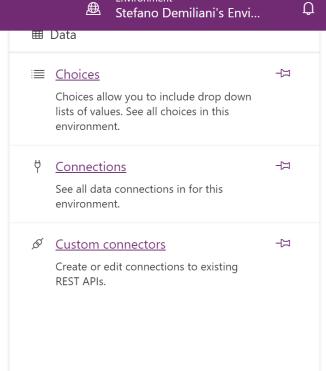
Minimum impact on source systems as there is a continuous replication of incremental changes. Reduced complexity, as there are no ETL jobs to manage, this is all handled by the service.

### Requirements



- You must have an Azure Data Lake Storage Gen2 account.
- You must have Owner and Storage Blob Data Contributor role access.
- You must have a Synapse workspace.
- You must provision the Synapse workspace in the same region as your Azure Data Lake Storage Gen2 account.
- You must provision the Synapse workspace in the same **resource group** as your Azure Data Lake Storage Gen2 account.
- The Azure Data Lake Storage Gen2 account and Synapse workspace must be created in the same Azure Active Directory (Azure AD) **tenant** as your Power Apps tenant.
- You must have the Dataverse System Administrator security role.
- Only tables that have change tracking enabled can be exported.



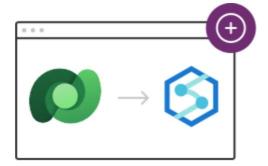


Environment



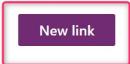
√ Filter

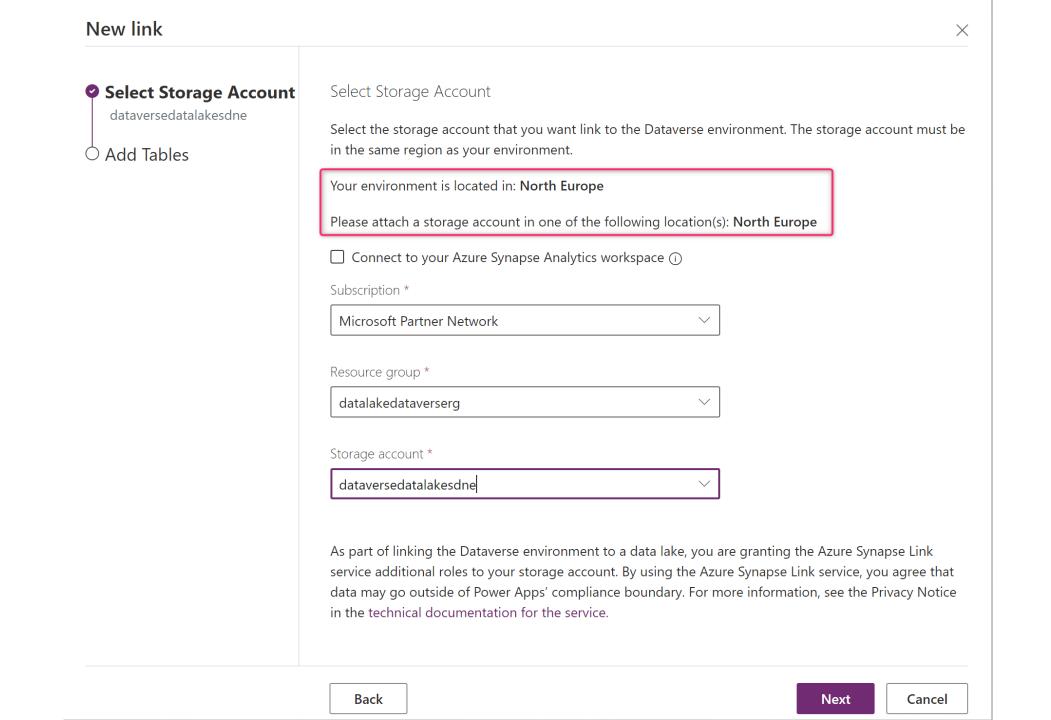
#### Azure Synapse Link for Dataverse

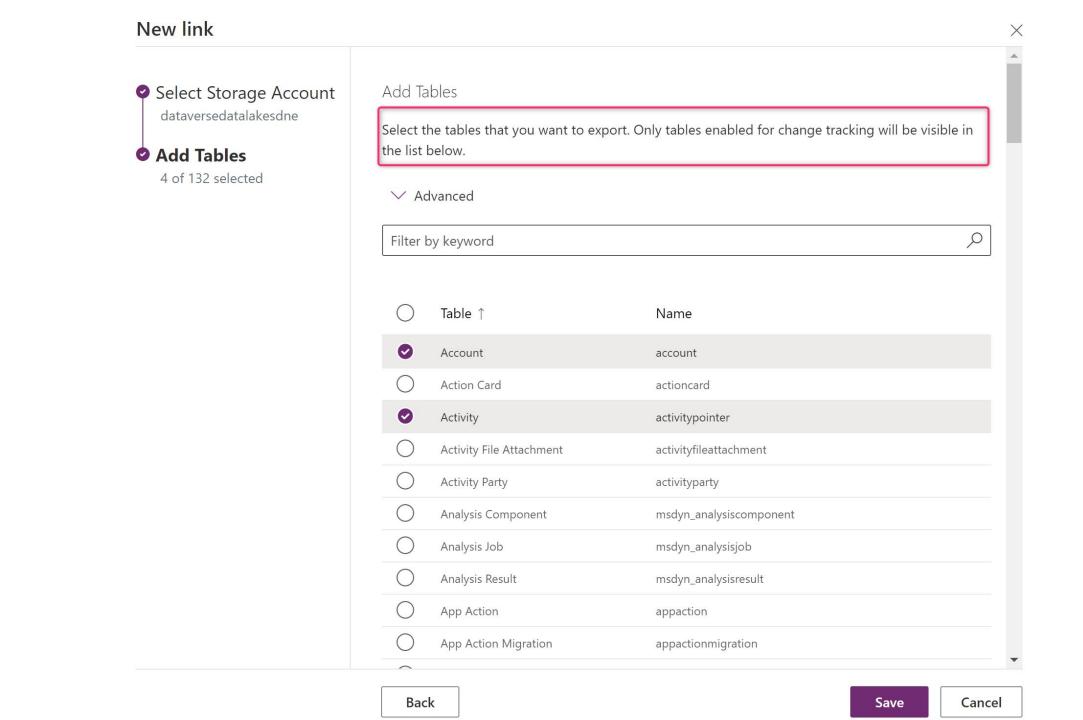


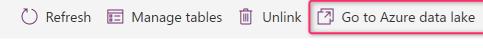
You have not linked the Dataverse environment to Azure Synapse Analytics.

Before you can export, link the Dataverse environment to an Azure Synapse Analytics workspace.









∀ Filter

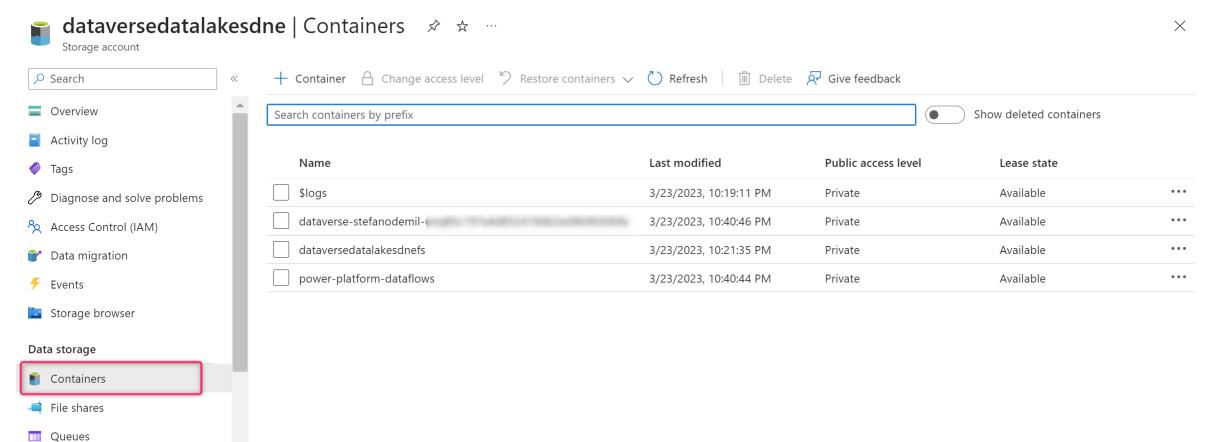
### Azure Synapse Link for Dataverse $\geq$ dataversedatalakesdne

**Tables** Details Discover hub

Table ↑	Name	Sync status	Last synchronized	Count	Append only	Partition
Account	account		03/23/2023 10:41:32 PM	4	No	Month
Activity	activitypointer		03/23/2023 10:41:37 PM	2	No	Month

#### Home > dataversedatalakesdne

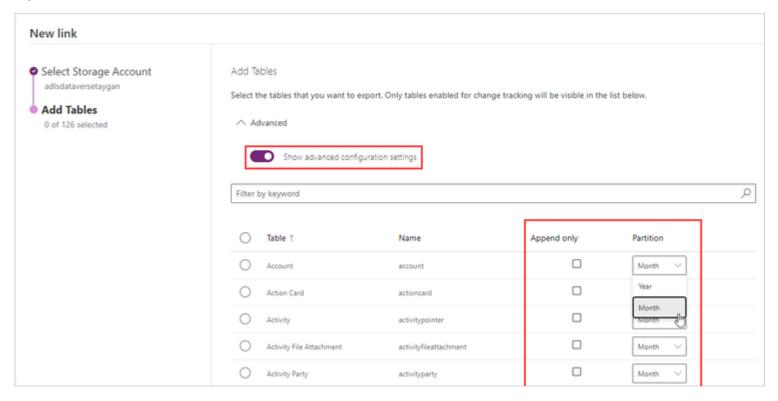
**Tables** 



#### Home > dataversedatalakesdne | Containers > dataverse-stefanodemil- $\times$ Container ↑ Upload + Add Directory ♦ Refresh | ♦ Rename 🗓 Delete ≥ Change tier 🔊 Acquire lease 🔊 Break lease 🕏 Give feedback Search Overview Authentication method: Access key (Switch to Azure AD User Account) Location: dataverse-stefanodemil-ung85c197e4d852419db2ed9b9b50bfa Diagnose and solve problems Show deleted objects Search blobs by prefix (case-sensitive) Access Control (IAM) Settings Modified Archive status Blob type Size Name Access tier Shared access tokens account Manage ACL activitypointer Access policy Microsoft. Athena. Trickle Feed Service Properties model.json 3/23/2023, 10:41:29 PM Hot (Inferred) Block blob 0 B Metadata

### **Advanced Configuration**

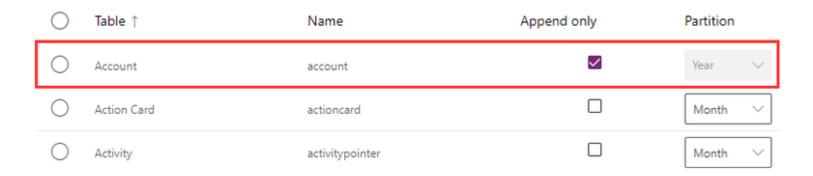
When adding tables to be in scope, you can alter the default settings that will determine how data will be exported and ultimately written to the data lake. These additional options can be revealed by toggling on the *advanced configuration* option.



### **Advanced Configuration**

### In-place updates vs append-only writes

- For tables where createdOn is available, the default setting is in-place update (AppendOnlyMode = False).
- For tables where *createdOn* is not available, the default setting is append-only writes (AppendOnlyMode = True).
- Using the advanced configuration settings, append-only mode can be enabled, this is available as a per table setting.
- If append-only is enabled, the partition strategy defaults to year.



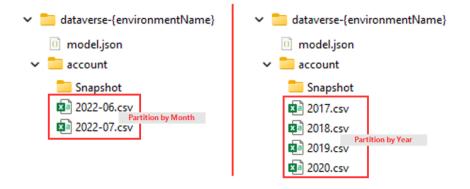
EVENT	IN-PLACE UPDATE	APPEND-ONLY		
CREATE	Row is inserted	Row is added to the end		
UPDATE	Row is replaced	Row is added to the end		
DELETE	Row is removed	Row is added to the end with isDeleted column = True		

- In-place updates: If you are planning to query the data directly from the data lake and only require the current state (as history will not be preserved).
- Append-only: If you require historical changes to be preserved. If you are not planning to query the data directly and would like to incrementally copy changes to another target.

### **Advanced Configuration**

#### **Data Partitioning**

- For tables where createdOn is available, the tables are partitioned by time (default is Month).
- For tables where *createdOn* is not available, the tables are partitioned every five million records.
- This is a per table setting.
- For high data volumes, it is recommended to partition by month as this will result in smaller files (compared to partition by year).



### Thank you for your questions!





# THANK YOU!!

© 2023 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries.

The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.