

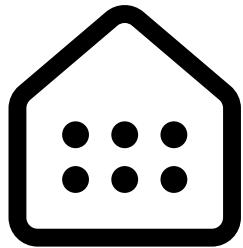
Master Microsoft Fabric: A Complete End-to-End Project- CICD



Author: Shanmukh Sattiraju
<https://in.linkedin.com/in/shanmukh-sattiraju>



Fabric Lakehouse



Fabric Warehouse



OneLake



Data Factory

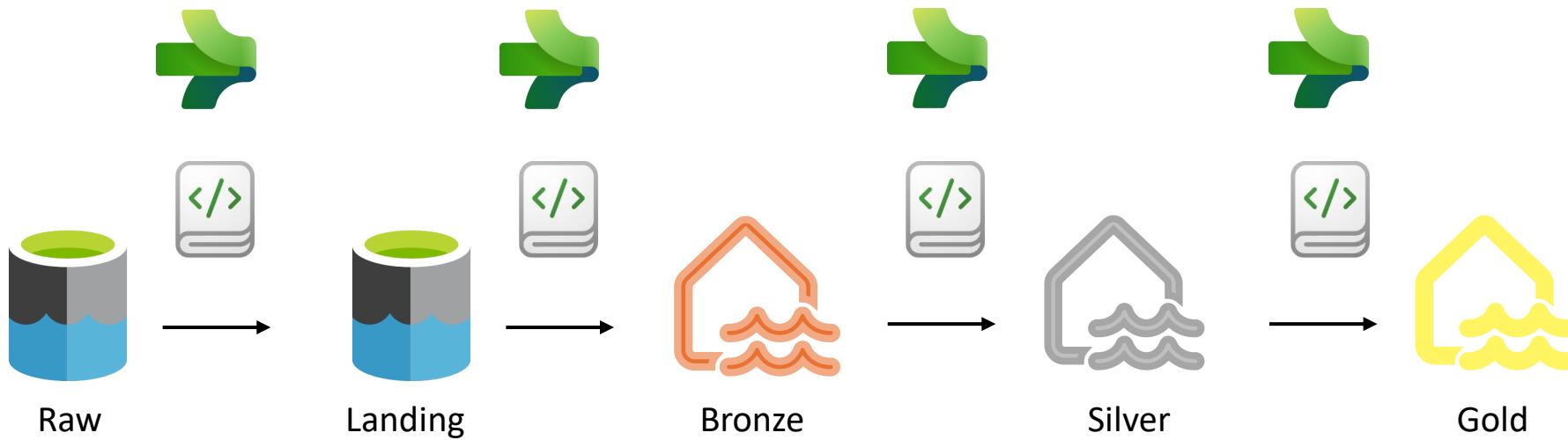


End to end project with Continuous Integration and Continuous Deployment (CI/CD)

Author: Shanmukh Sattiraju

<https://in.linkedin.com/in/shanmukh-sattiraju>

End to End flow



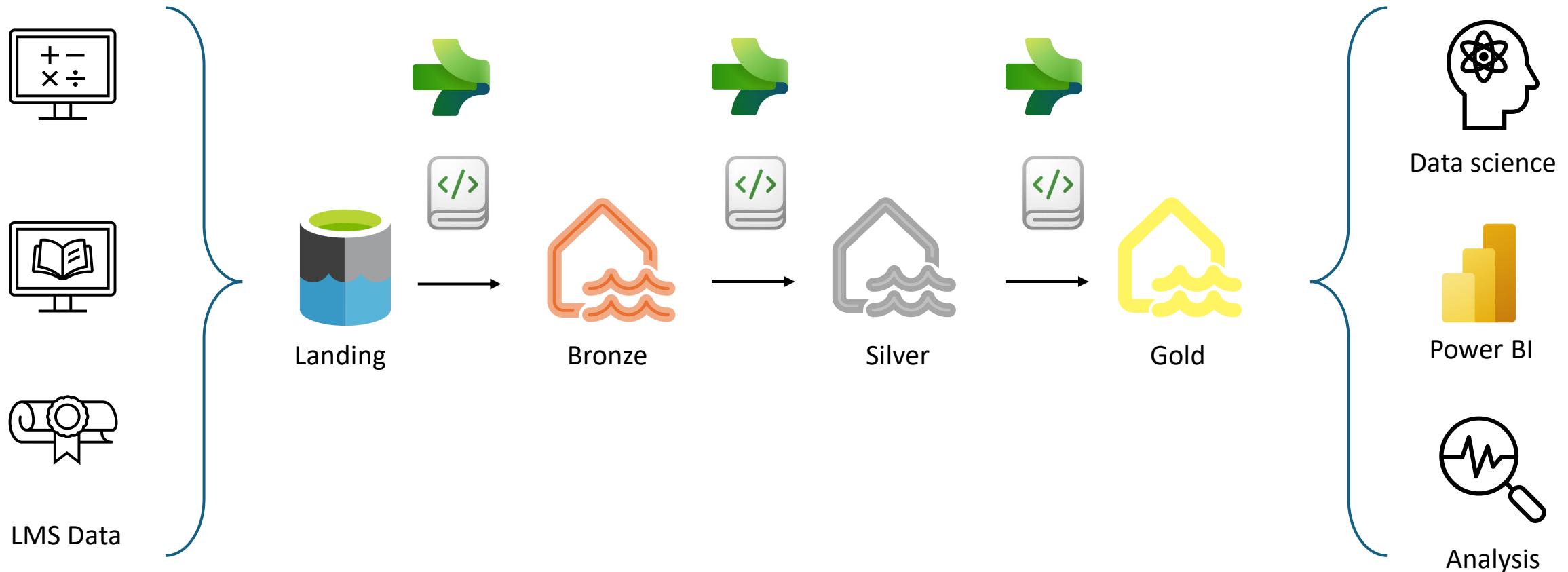
Prerequisites

- No experience is needed in Microsoft Fabric
- An Azure account for hands-on practical
- Basic knowledge on Python/PySpark and SQL/SparkSQL
- Basic knowledge on Azure Cloud Environment

What you'll get from this course?

- Nearly 21+ hours of updated learning content
- Deep dive into components of Microsoft Fabric
- Hands-on end to end project
- Implementing CICD in Microsoft Fabric
- Lifetime access to this Course
- Certificate of completion at end of the course

Fabric Project Architecture



Analyze LMS data with Power BI

45

Total Students

12

Total Courses

43.28

Average Score

98.11

Average Completion Days

86.53

Average Quiz Scores

Select all

A

A-

A+

B

B-

B+

Select all

Completed

In Progress

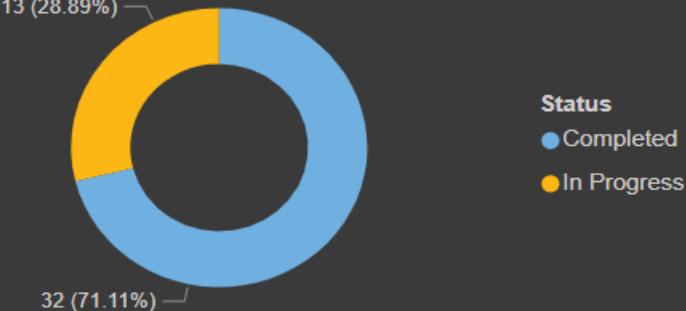
Select all

Advanced

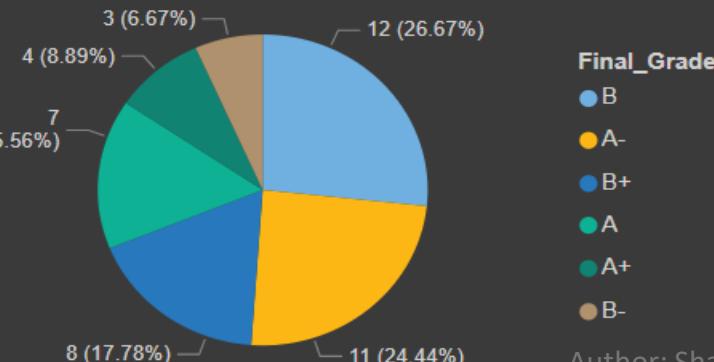
Beginner

Intermediate

Completion status by Students



Final Grades by Students



Preferred_Lea... ×

Auditory

Parent_Involv... ×

High

Average Completion...

98.11

Auditory

107.47

High

123.57

Visual

93.28

Medium

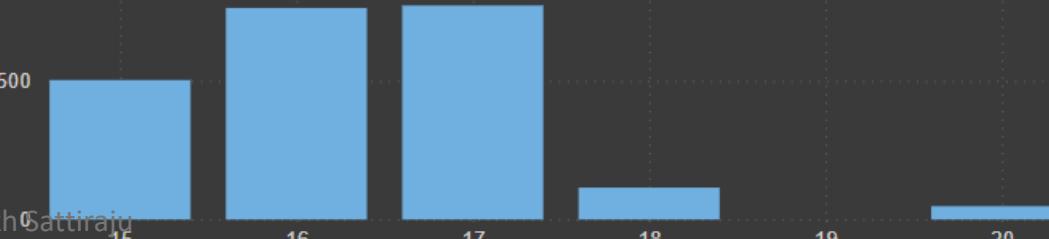
96.20

Kinesthetic

85.33

This is a test

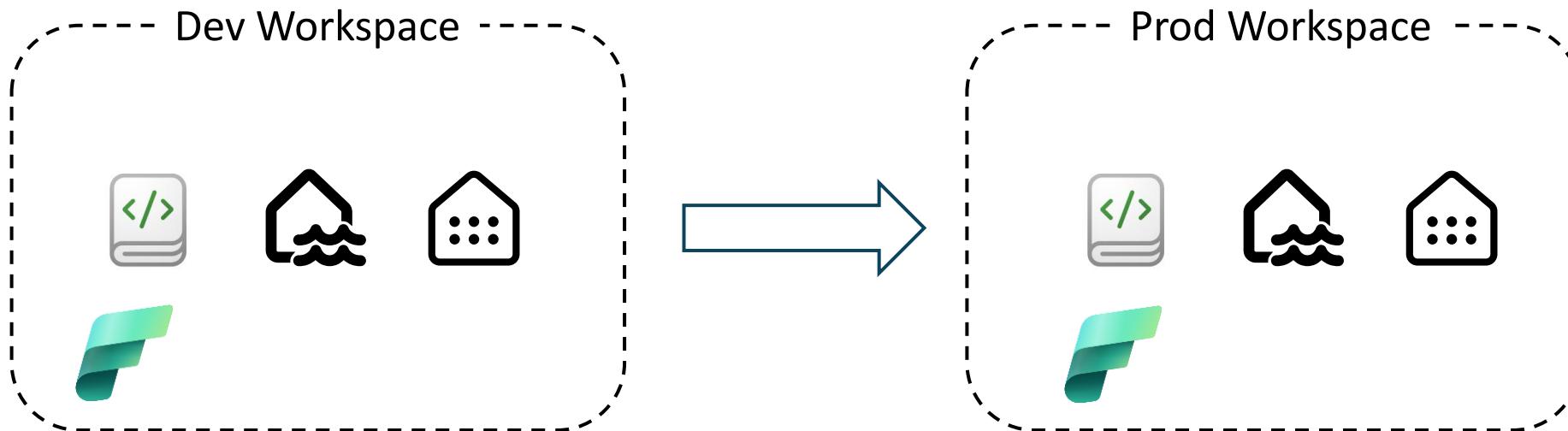
Total Time Spent on Course by Age



Author: Shanmukh Sattiraju

<https://in.linkedin.com/in/shanmukh-sattiraju>

Continuous Integration and Continuous Deployment (CICD) in Microsoft Fabric



Learning Structure

Understanding Microsoft Fabric

Lakehouse

Data Factory

OneLake

Synapse Data Engineering

Synapse migration to Fabric

Capacity Metrics App

Synapse Data Warehouse

Access Control and Permissions

Power BI in Fabric

End to End project

Git Integration

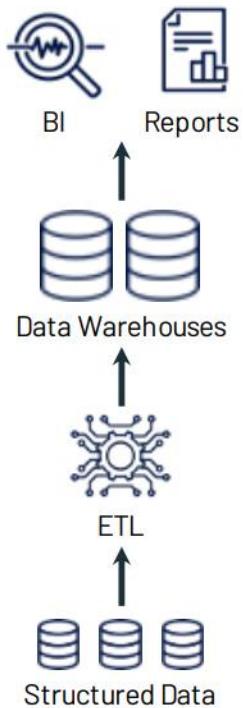
Environment Setup

Microsoft Fabric

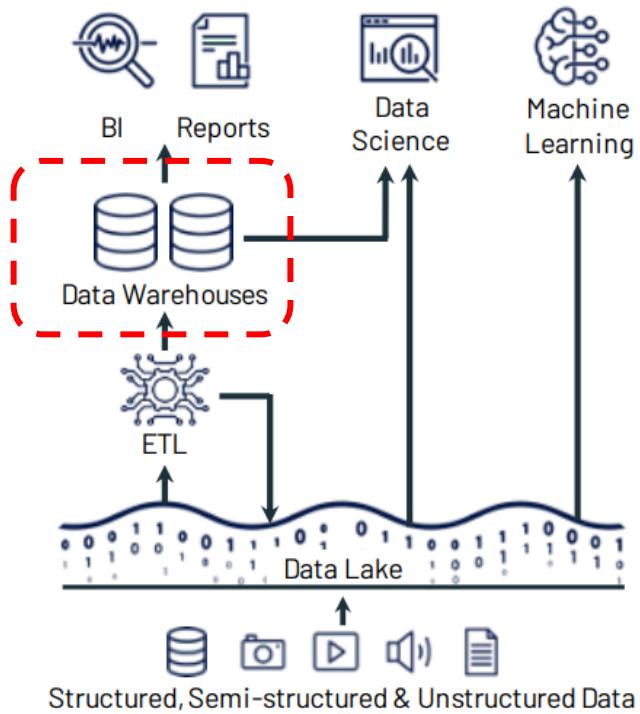
A unified analytics solution
for the era of AI



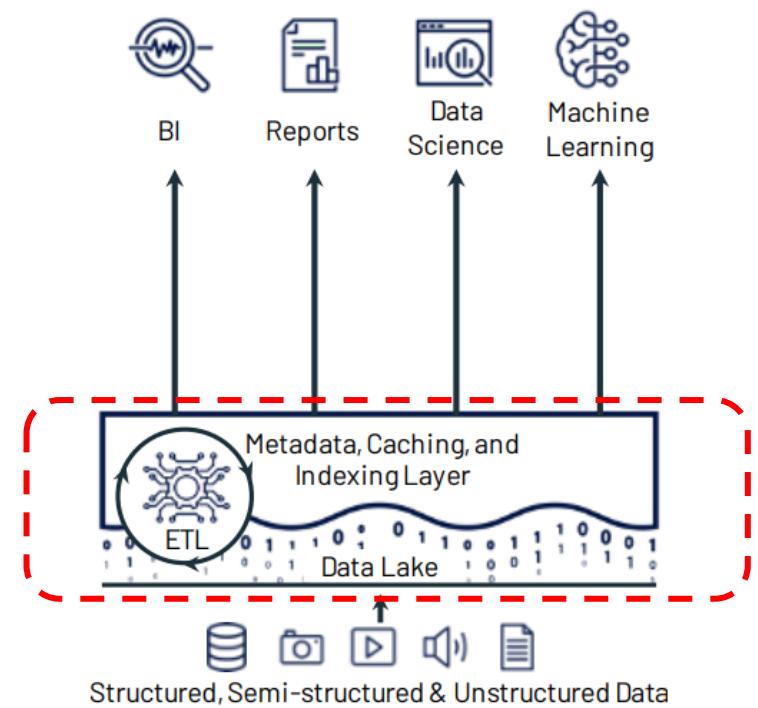
Evolution of Data Architectures



Data Warehouse



Modern Data Warehouse
(Data lake)



Lakehouse Architecture
(Delta lake)

Lakehouse Architecture

Best elements of
Data lake

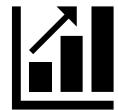
Best elements of
Data warehouse

Lakehouse

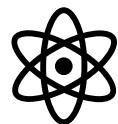
Lakehouse Architecture



BI



Reports



Data-Science



ML



Structured, Semi- Structured & Unstructured Data

Author: Shanmukh Sattiraju

<https://in.linkedin.com/in/shanmukh-sattiraju>

How to create delta lake?

Instead of parquet..

```
dataframe.  
    write\  
        .format("parquet")\  
        .save("/data/")
```

Replace with delta..

```
dataframe.  
    write\  
        .format("delta")\  
        .save("/data/")
```



Delta format

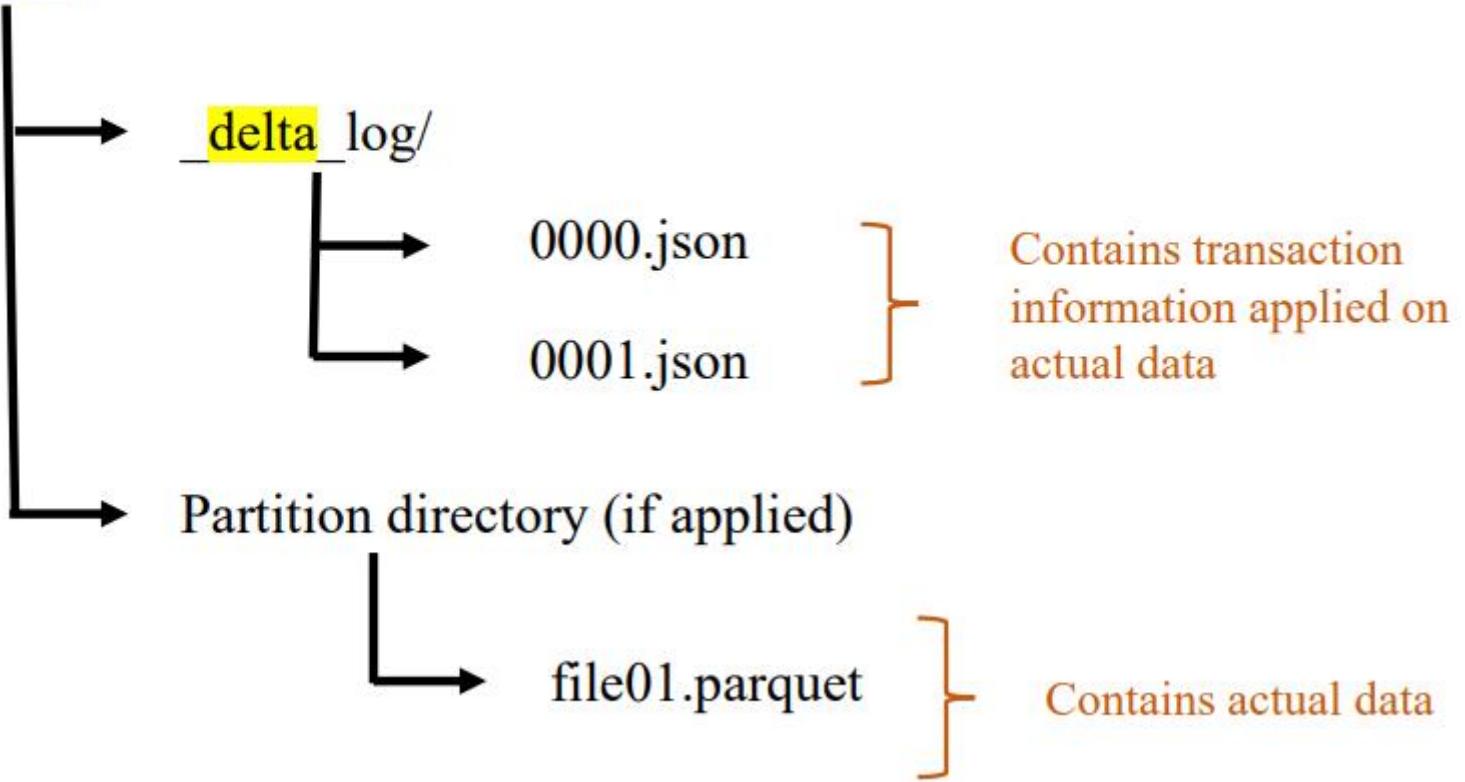


Azure Data Lake
Storage



Parquet + Transaction Log

delta/

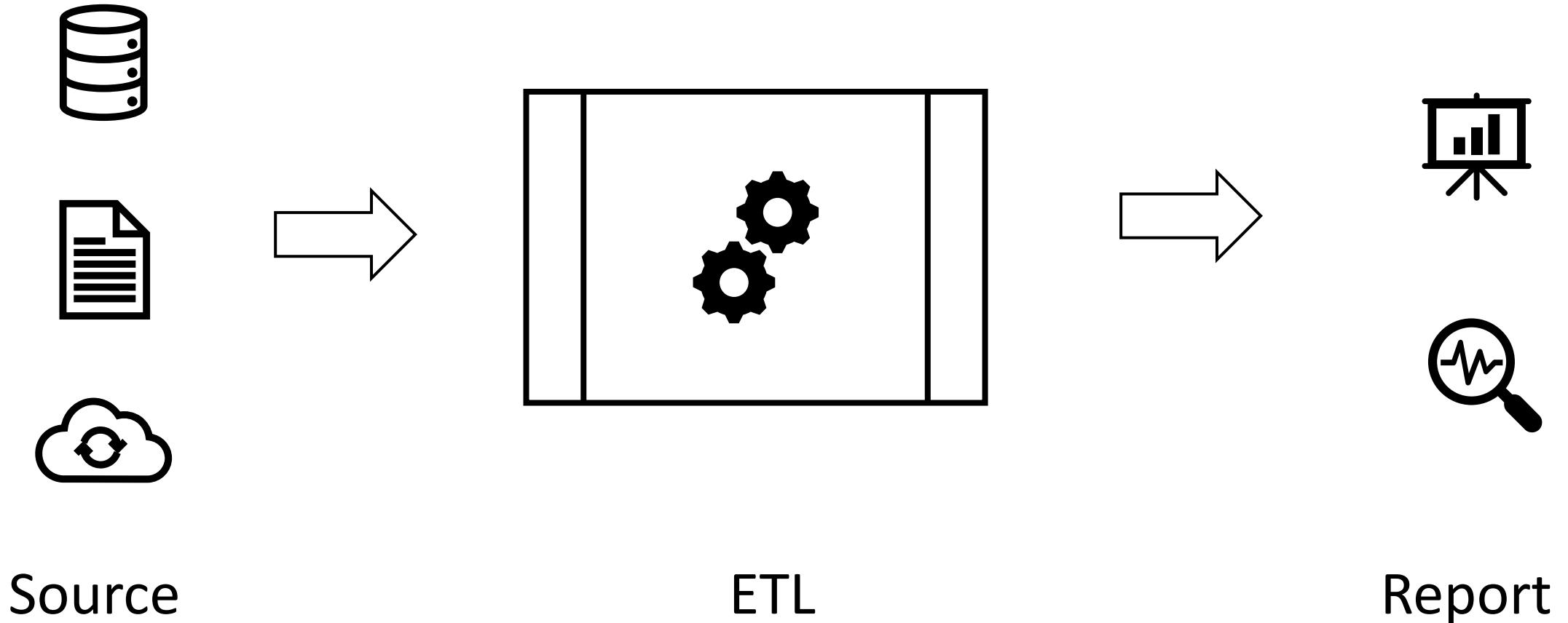


Author: Shanmukh Sattiraju

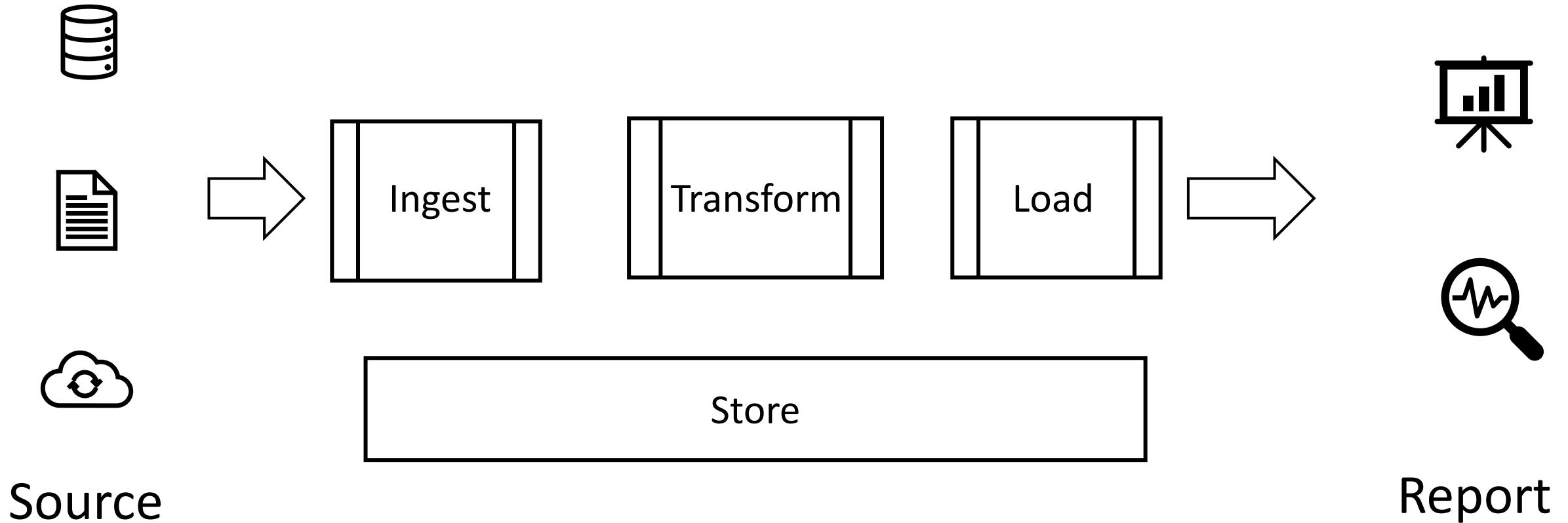
Author: Shanmukh Sattiraju
<https://in.linkedin.com/in/shanmukh-sattiraju>

Why Microsoft Fabric?

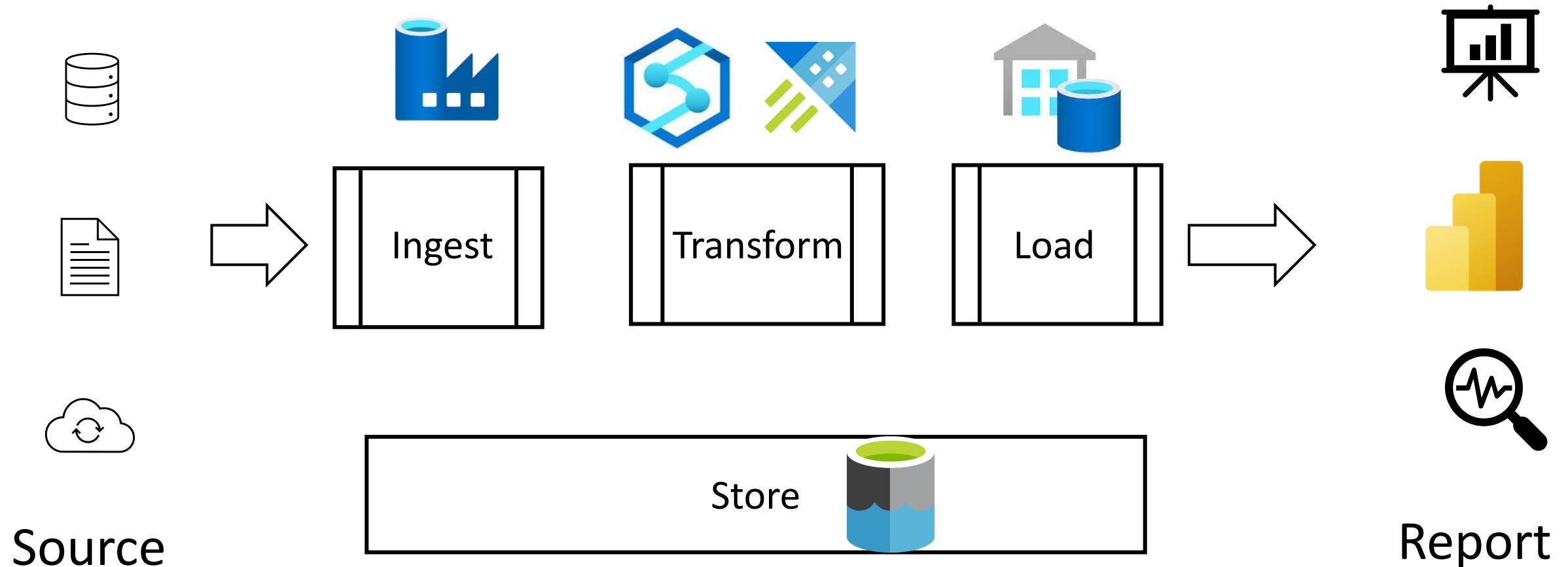
Typical Data workflow



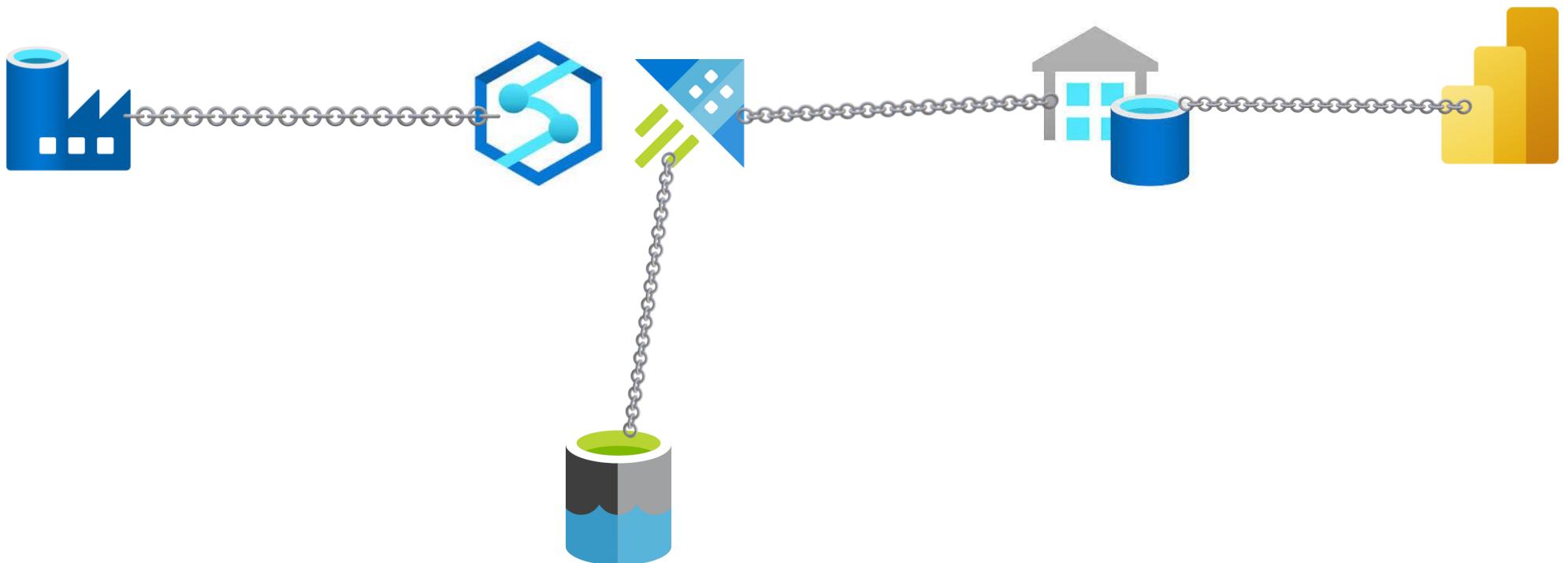
Typical Data workflow



Azure Data Services



Azure Data Services



Azure Data Services



Azure Data
Factory



Synapse
Analytics



Data
Explorer



SQL DW



Power BI



Data lake

Computes



Spark pool



Serverless
SQL Pool

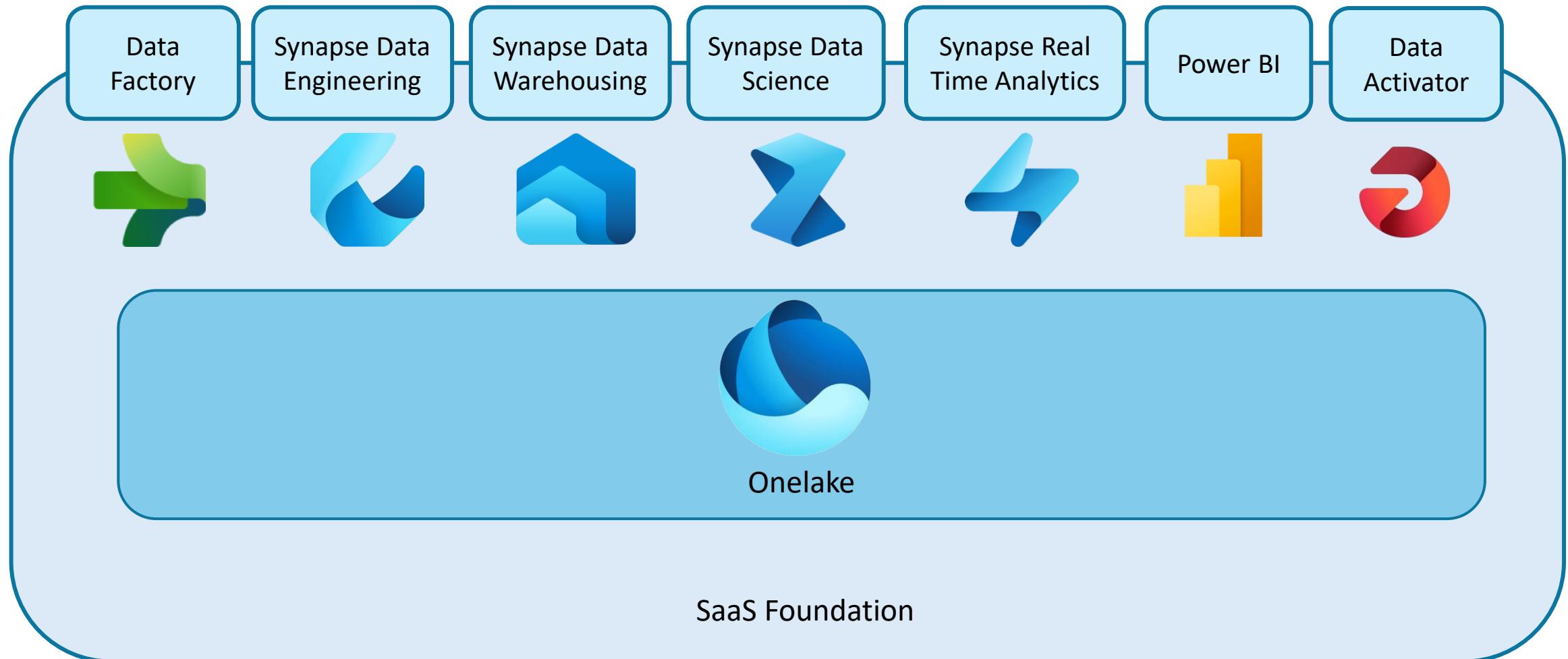


Dedicated
SQL pool



Data
Explorer pool

Microsoft Fabric



Why Microsoft Fabric?

Microsoft's definition:

“Microsoft Fabric is an all-in-one analytics solution for enterprises that covers everything from data movement to data science, Real-Time Analytics, and business intelligence. It offers a comprehensive suite of services, including data lake, data engineering, and data integration, all in one place.”

How to enable / access Fabric?

Enable / access Microsoft Fabric

The screenshot shows the Microsoft Fabric Admin portal interface. The left sidebar contains navigation links: Home, Create, Browse, OneLake data hub, Apps, Metrics, and Monitoring hub. The main content area has a header "Microsoft Fabric" and a search bar. Below the header, the title "Admin portal" is displayed. On the left, a sidebar lists "Tenant settings" (marked as New), "Usage metrics", "Users", "Premium Per User", "Audit logs", "Domains" (marked as New), "Capacity settings", "Refresh summary", "Embed Codes", "Organizational visuals", and "Azure connections". A message bubble indicates "There are new or updated tenant settings. Expand to review the changes." The "Microsoft Fabric" section displays two items: "Data Activator (preview)" (Enabled for the entire organization) and "Users can create Fabric items" (Enabled for the entire organization). Both items have a descriptive text and a "Learn More" link. A toggle switch at the bottom is set to "Enabled".

Microsoft Fabric

- ▷ Data Activator (preview)
Enabled for the entire organization
- ▷ Users can create Fabric items
Enabled for the entire organization

Users can use production-ready features to create Fabric items. Turning off this setting doesn't impact users' ability to create Power BI items. This setting can be managed at both the tenant and the capacity levels. [Learn More](#)

Enabled

Fabric Capacity License

SKU *	Capacity Units (CU)	Power BI SKU	Power BI v-cores
F2	2	-	0.25
F4	4	-	0.5
F8	8	EM/A1	1
F16	16	EM2/A2	2
F32	32	EM3/A3	4
F64	64	P1/A4	8
Trial	64	-	8
F128	128	P2/A5	16
F256	256	P3/A6	32
F512	512	P4/A7	64
F1024	1024	P5/A8	128
F2048	2048	-	256

Fabric Capacity License

SKU *	Capacity Units (CU)	Power BI SKU	Power BI v-cores
F2	2	-	0.25
F4	4	-	0.5
F8	8	EM/A1	1
F16	16	EM2/A2	2
F32	32	EM3/A3	4
F64	64	P1/A4	8
Trial	64	-	8
F128	128	P2/A5	16
F256	256	P3/A6	32
F512	512	P4/A7	64
F1024	1024	P5/A8	128
F2048	2048	-	256

Power BI Pro

Spark nodes per capacity

SKU name	Capacity units	Spark VCores	Node size	Default max nodes	Max number of nodes
F2	2	4	Medium	1	1
F4	4	8	Medium	1	1
F8	8	16	Medium	2	2
F16	16	32	Medium	3	4
F32	32	64	Medium	8	8
F64	64	128	Medium	10	16
(Trial Capacity)	64	128	Medium	10	16
F128	128	256	Medium	10	32
F256	256	512	Medium	10	64
F512	512	1024	Medium	10	128
F1024	1024	2048	Medium	10	200
F2048	2048	4096	Medium	10	200

Reference: [Configure and manage starter pools in Fabric Spark](#). - Microsoft Fabric | Microsoft Learn
Author: Shanmukh Sattiraju
<https://in.linkedin.com/in/shanmukh-sattiraju>

Cost of using Fabric?

Cost of using Fabric?

Compute cost :

SKU	Capacity unit (CU)	Pay-as-you-go
F 2	2	\$0.36/hour
F 4	4	\$0.72/hour
F 8	8	\$1.44/hour
F 16	16	\$2.88/hour
F 32	32	\$5.76/hour
F 64	64	\$11.52/hour
F 128	128	\$23.04/hour
F 256	256	\$46.08/hour
F 512	512	\$92.16/hour
F 1024	1024	\$184.32/hour
F 2048	2048	\$368.64/hour

Cost of using Fabric?

Storage cost :

Storage	Price
OneLake storage/month **	\$0.023 per GB
OneLake BCDR storage/month	\$0.0414 per GB
OneLake cache/month *	\$0.246 per GB

Resources: [Resources: Microsoft Fabric - Pricing | Microsoft Azure](#)

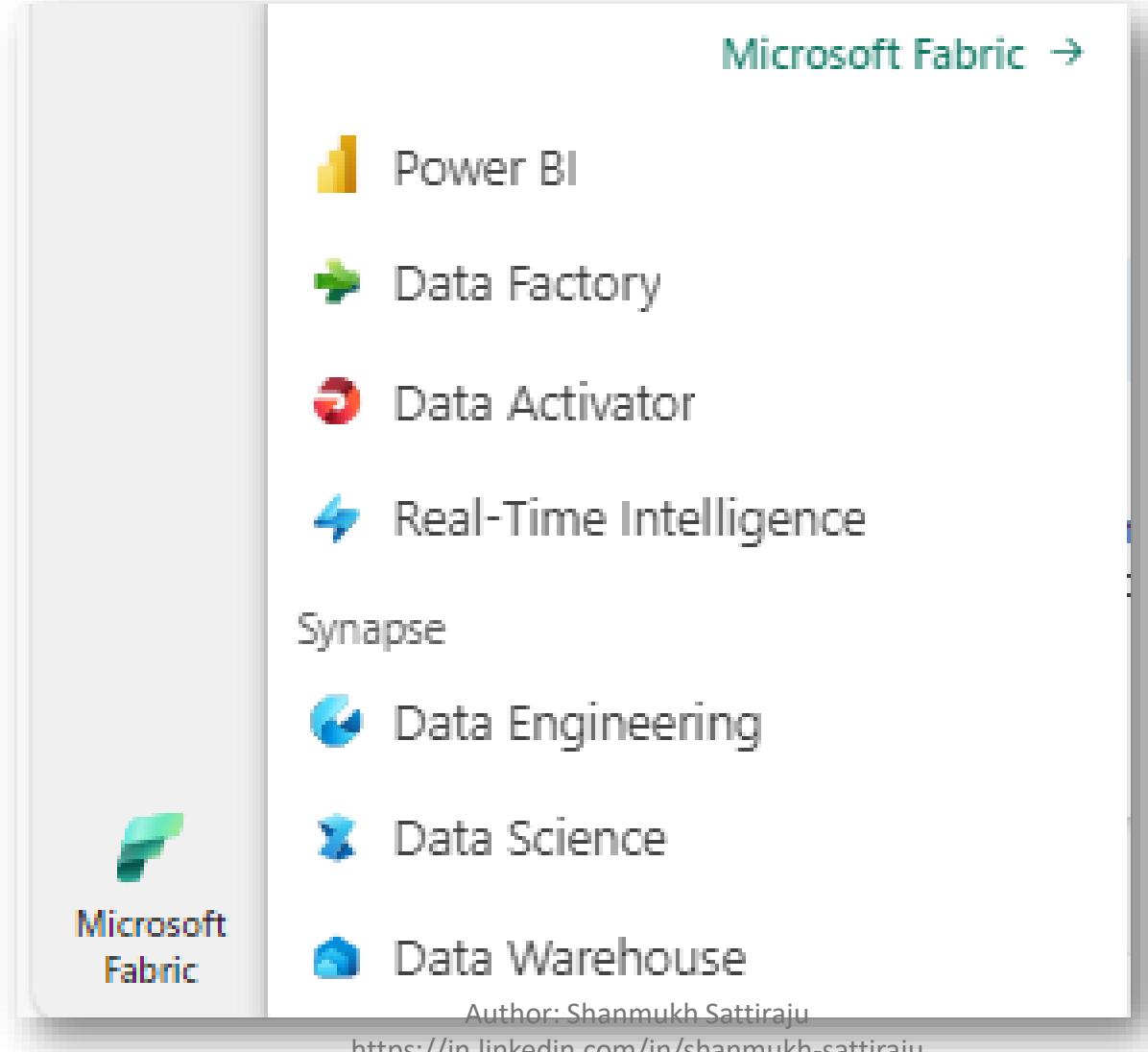
[Pricing Calculator | Microsoft Azure](#)

Benefits of Fabric

- One environment for everyone
- SaaS platform makes development easier
- Universal compute
- All data is open-source delta lake
- OneLake = one copy of data
- Sharing data between workspaces
- OneSecurity

Components / Features of Fabric

Experiences of Microsoft Fabric



The image shows a screenshot of the Microsoft Fabric interface. On the left, there's a vertical sidebar with the Microsoft Fabric logo at the bottom. The main area is titled "Microsoft Fabric" with a right-pointing arrow. Below this, there's a list of eight items, each with an icon and a title:

- Power BI (yellow bar icon)
- Data Factory (green plus icon)
- Data Activator (red circular icon)
- Real-Time Intelligence (blue lightning bolt icon)
- Synapse (grey square icon)
- Data Engineering (blue circular icon)
- Data Science (blue brain-like icon)
- Data Warehouse (blue warehouse icon)

At the bottom of the main area, it says "Author: Shanmukh Sattiraju" and provides a LinkedIn link: <https://in.linkedin.com/in/shanmukh-sattiraju>.

Fabric Home

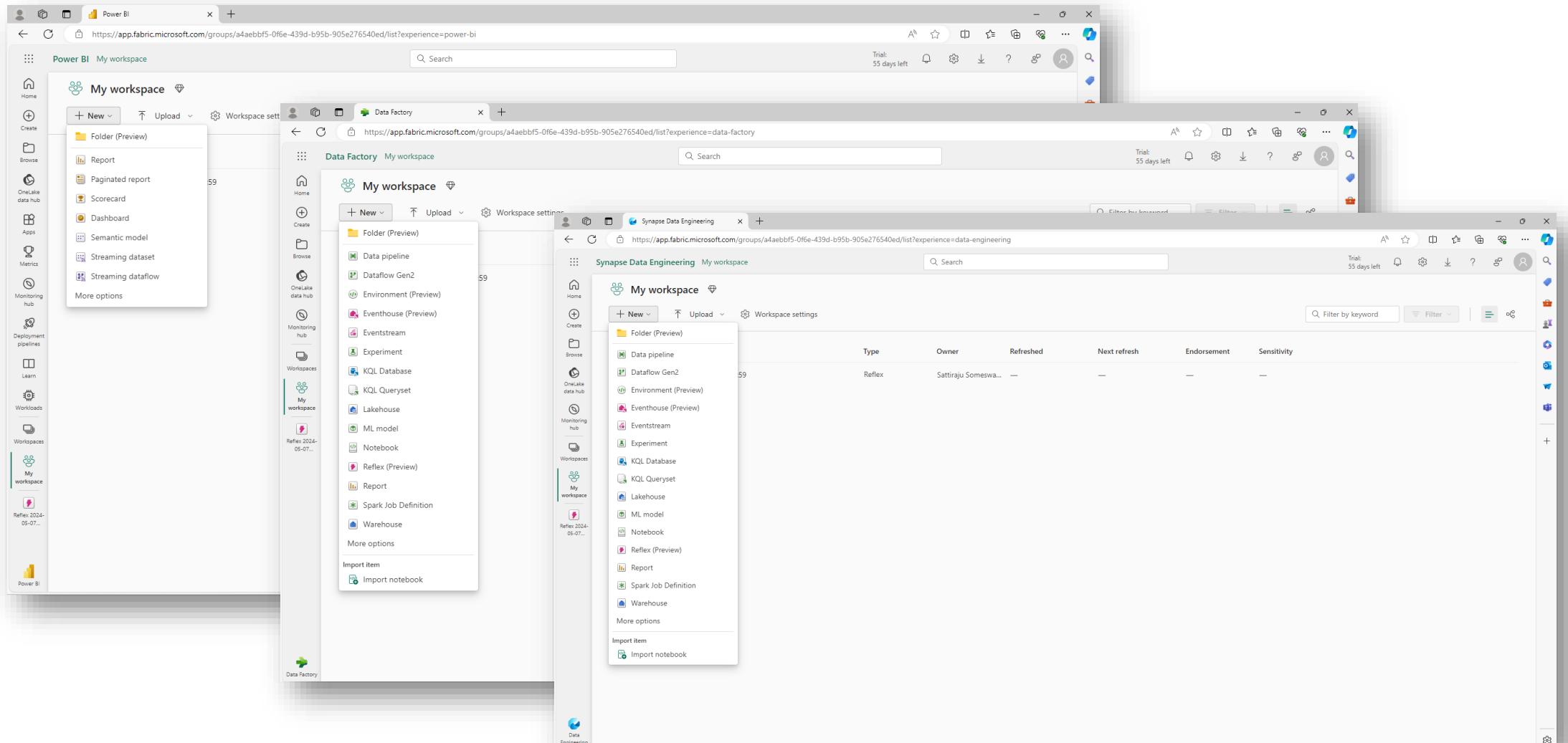
The screenshot shows the Microsoft Fabric Home page. At the top center, it says "Microsoft Fabric" and "All your data. In one location. Organize. Collaborate. Create." Below this, there's a section titled "Explore the experience" with eight cards:

- Power BI**: Find insights, track progress, and make decisions faster using rich visualizations.
- Data Factory**: Solve the most complex data integration and ETL scenarios with cloud-scale data movement and data transformation services.
- Data Activator**: Monitor data to trigger alerts and automated actions so your organization adapts to changing conditions in real time.
- Industry Solutions**: Get a head start with industry-relevant connectors, transformations, and scenario-specific tools.
- Synapse Data Engineering**: Create a lakehouse, and use Apache Spark to transform and prepare organizational data to share with the business.
- Synapse Data Science**: Explore your data, and build machine learning models to infuse predictive insights into your analytics solutions and applications.
- Synapse Data Warehouse**: Scale up your insights by storing and analyzing data in a secure, open-data-format SQL warehouse with top performance at PB scale.
- Synapse Real-Time Analytics**: Rapidly ingest, transform, and query any data source and format, from 1 GB to 1 PB, and then visualize and share the insights.

At the bottom, there are two buttons: "Read documentation" and "Explore community".

Author: Shanmukh Sattiraju
<https://in.linkedin.com/in/shanmukh-sattiraju>

Experiences



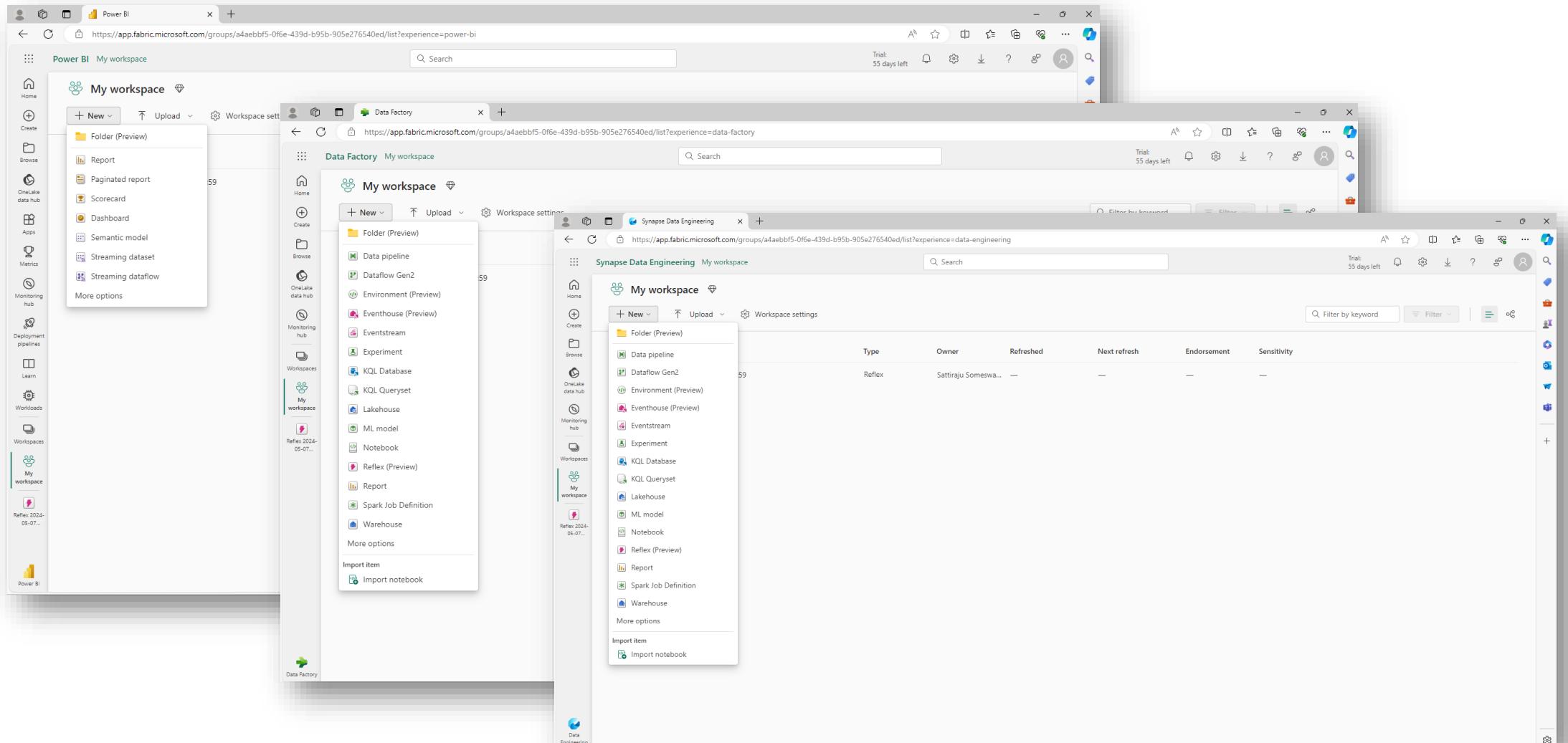
Power BI

The screenshot shows the Microsoft Power BI workspace interface. The left sidebar contains navigation links such as Home, Create, Browse, OneLake data hub, Apps, Metrics, Monitoring hub, Deployment pipelines, Learn, Workloads, Workspaces, and a selected item, My workspace. A context menu is open over a 'Report' item, listing options like New, Upload, and Workspace settings. The main area displays a table of workspace items with columns for Type, Owner, Refreshed, Next refresh, Endorsement, and Sensitivity. One item is listed: '59' (Type: Reflex, Owner: Sattiraju Someswara..., Refreshed: —, Next refresh: —, Endorsement: —, Sensitivity: —). The top right corner shows a trial status of '55 days left'.

	Type	Owner	Refreshed	Next refresh	Endorsement	Sensitivity
59	Reflex	Sattiraju Someswara...	—	—	—	—

Author: Shanmukh Sattiraju
<https://in.linkedin.com/in/shanmukh-sattiraju>

Experiences



Data Engineering

The screenshot shows the Microsoft Synapse Data Engineering workspace interface. The left sidebar includes links for Home, Create (with options like Folder (Preview), Data pipeline, Dataflow Gen2, Environment (Preview), Eventhouse (Preview), Eventstream, Experiment, KQL Database, KQL Queryset, Lakehouse, ML model, Notebook, Reflex (Preview), Report, Spark Job Definition, and Warehouse), My workspace (selected), and Import item (Import notebook). The main area displays a table with one item:

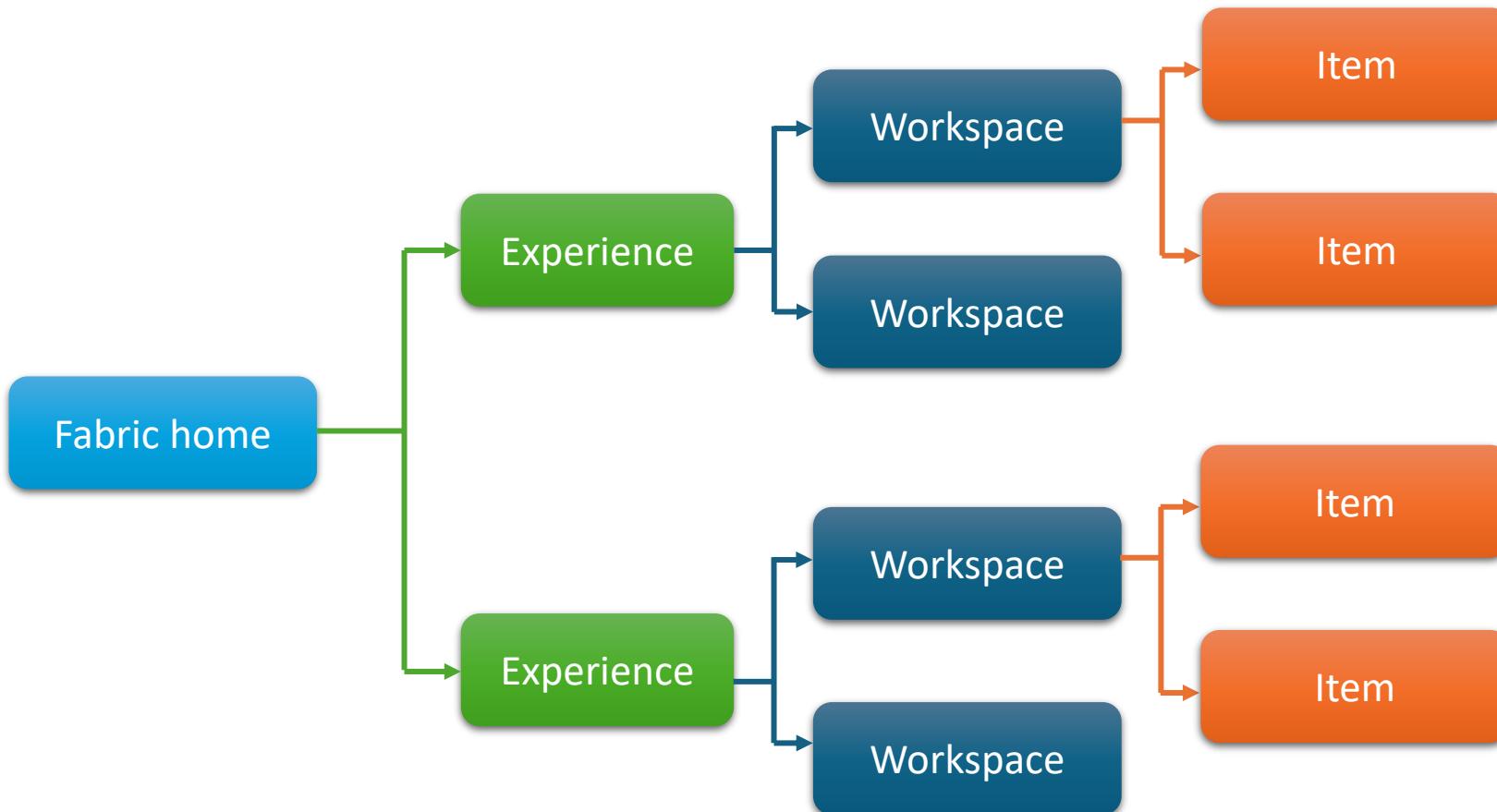
Type	Owner	Refreshed	Next refresh	Endorsement	Sensitivity
Reflex	Sattiraju Someswara...	—	—	—	—

The top navigation bar shows the title "Synapse Data Engineering" and the URL "https://app.fabric.microsoft.com/groups/a4aebbf5-0f6e-439d-b95b-905e276540ed/list?experience=data-engineering". A search bar and various toolbar icons are also present.

Author: Shanmukh Sattiraju

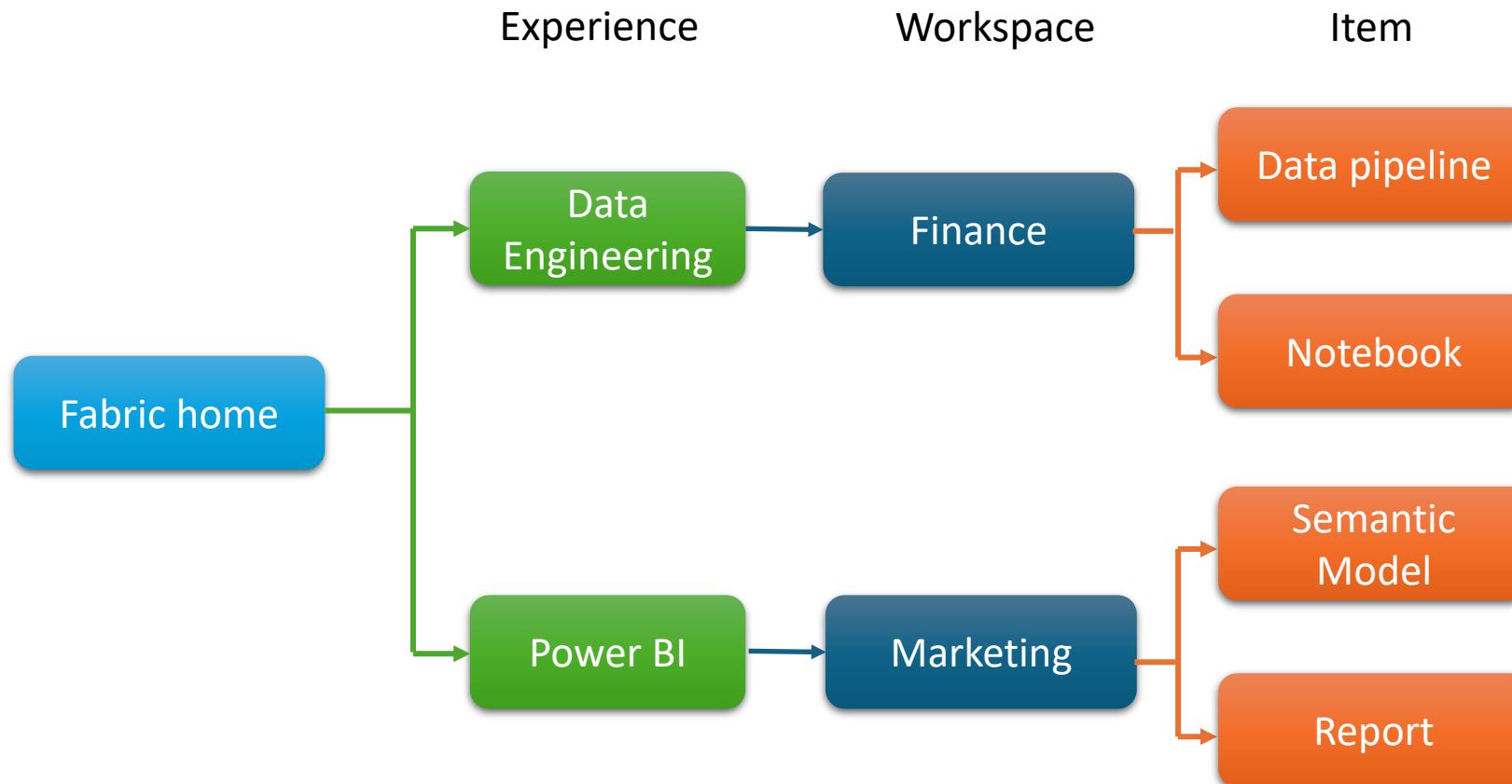
<https://in.linkedin.com/in/shanmukh-sattiraju>

Fabric Terminology

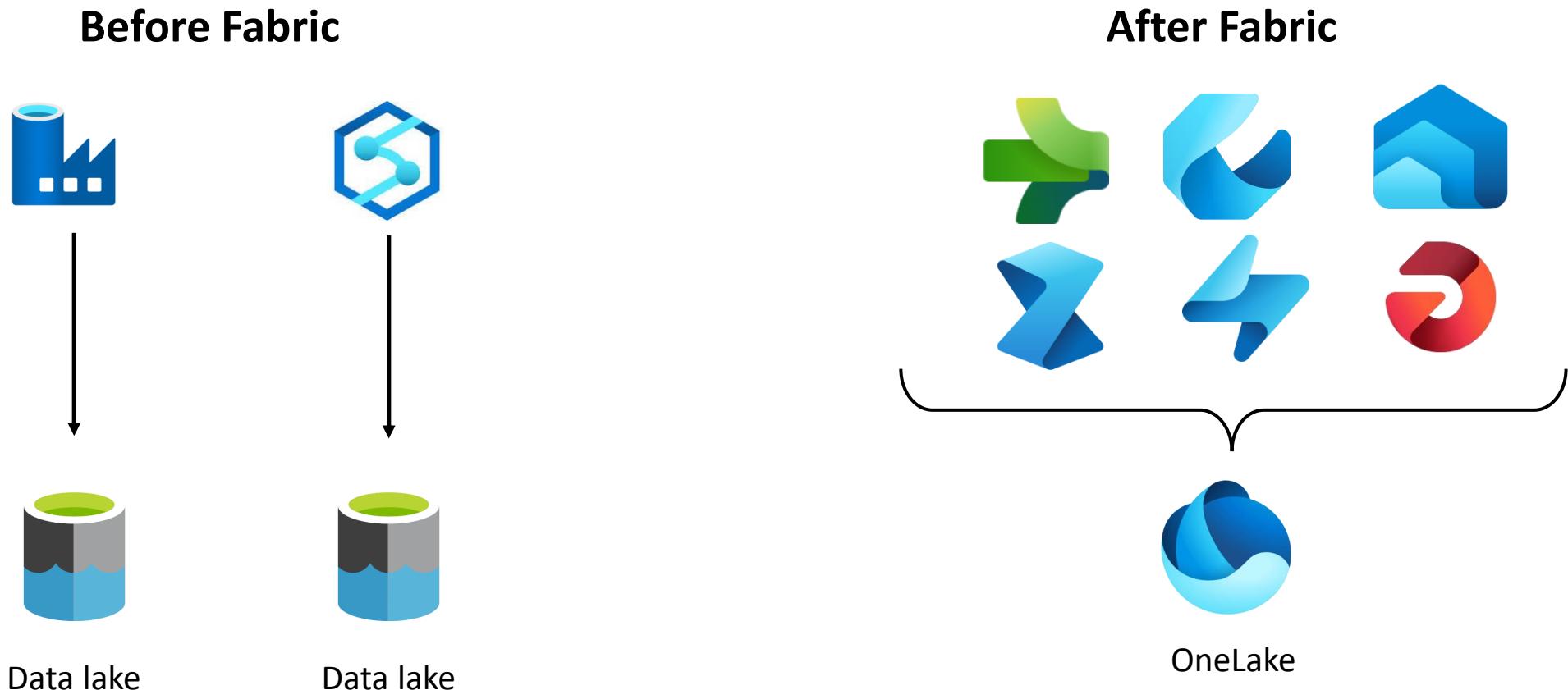


Fabric Terminology

Example:



OneLake



OneLake



OneLake



Author: Shanmukh Sattiraju
<https://in.linkedin.com/in/shanmukh-sattiraju>

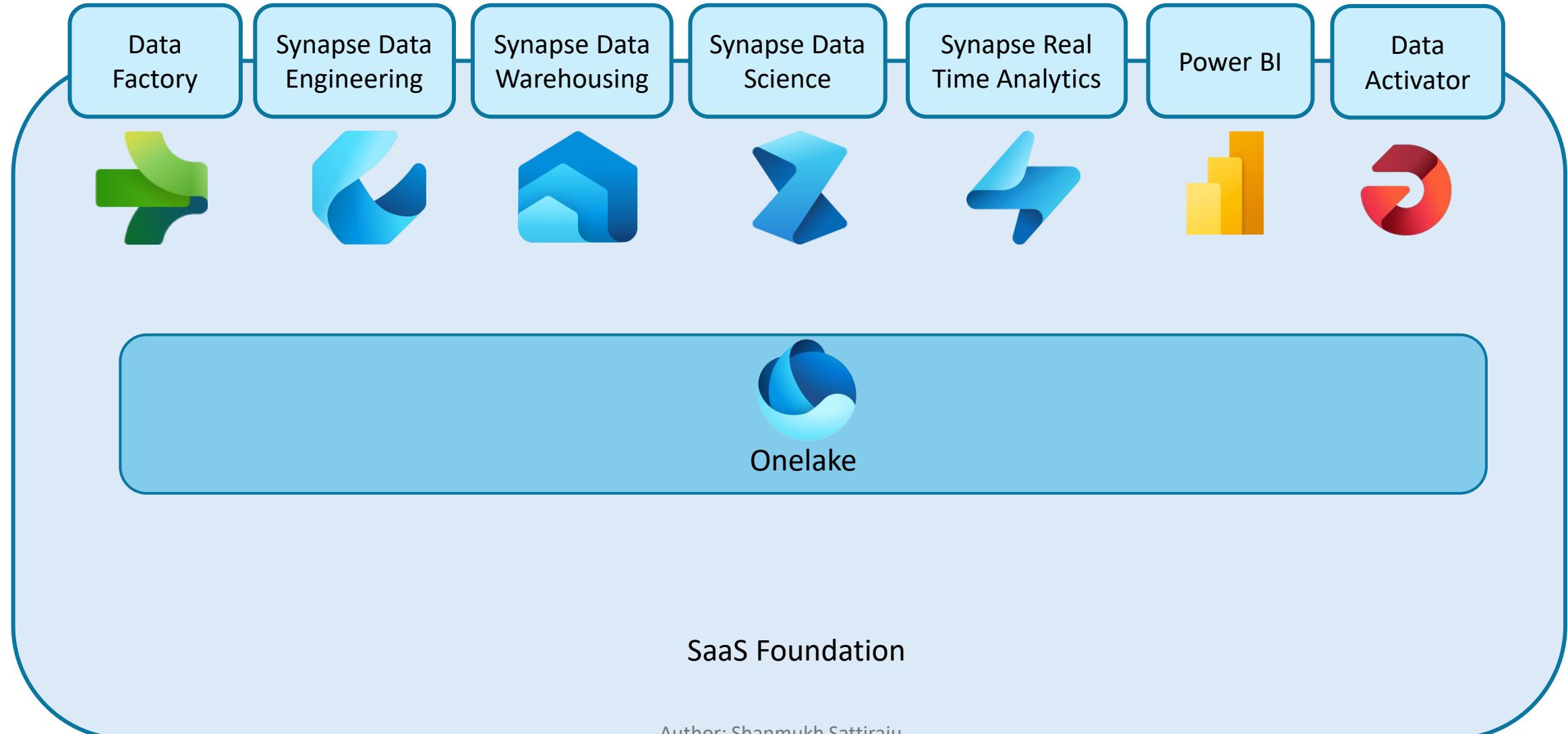
OneLake



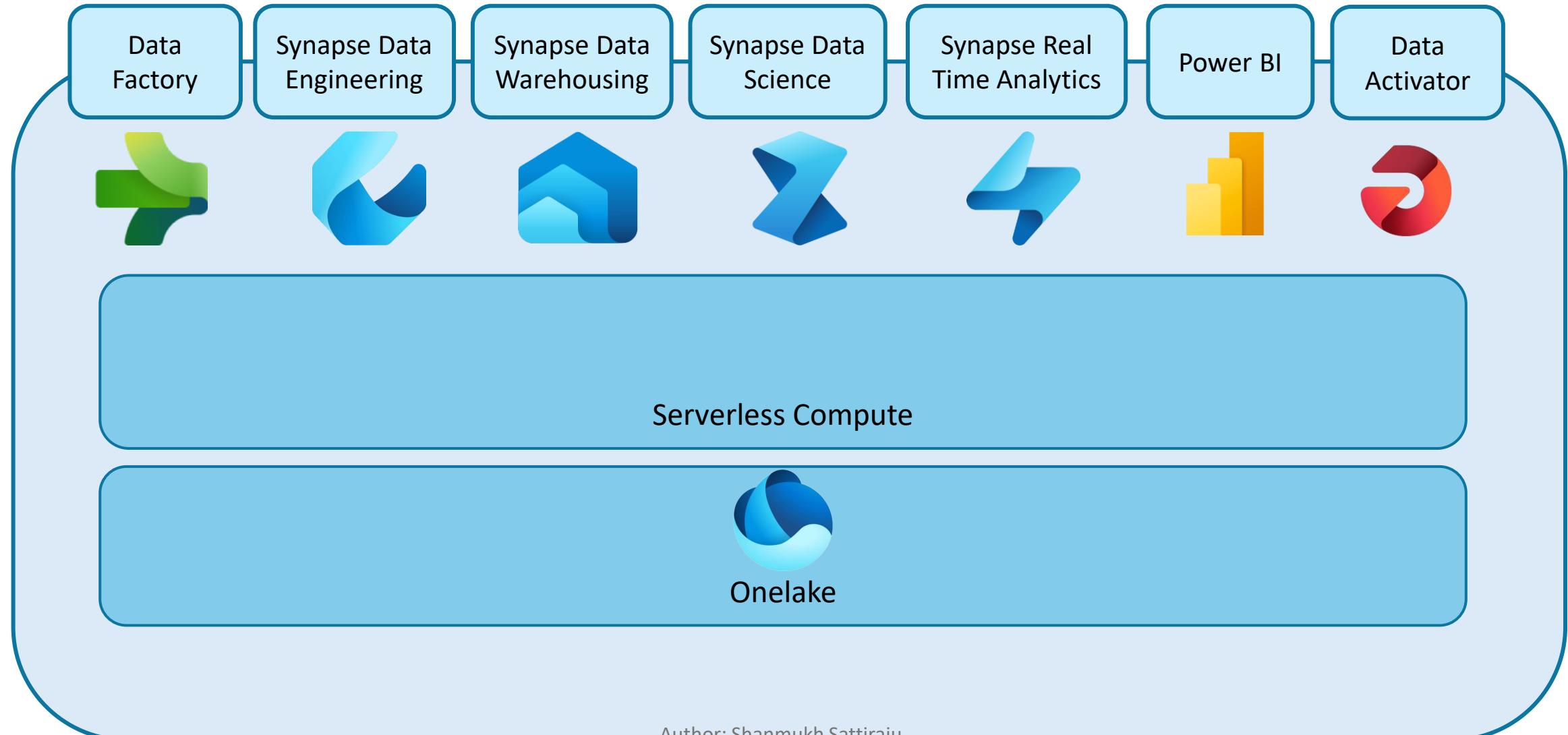
One Copy of the Data



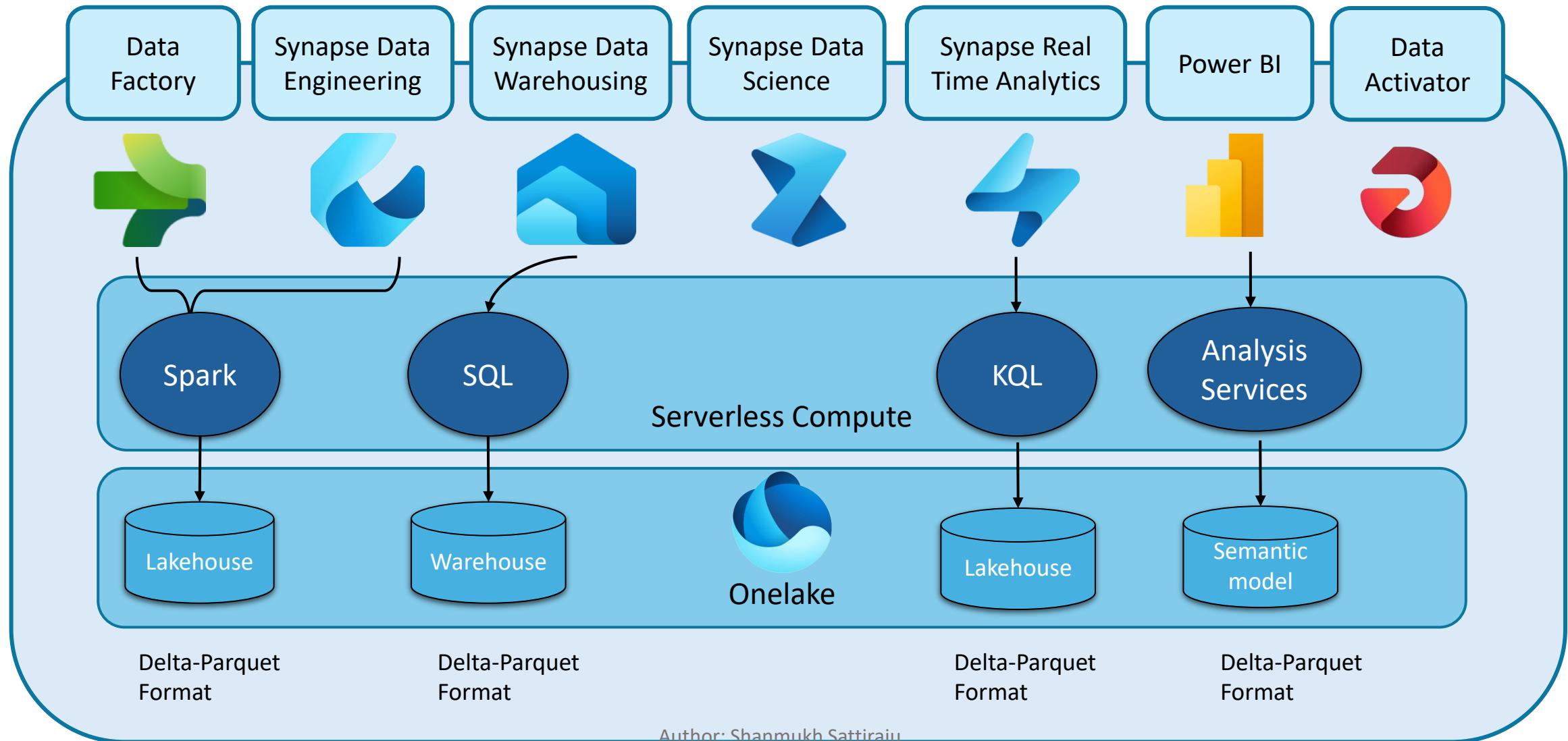
One Copy for all computers



One Copy for all computers

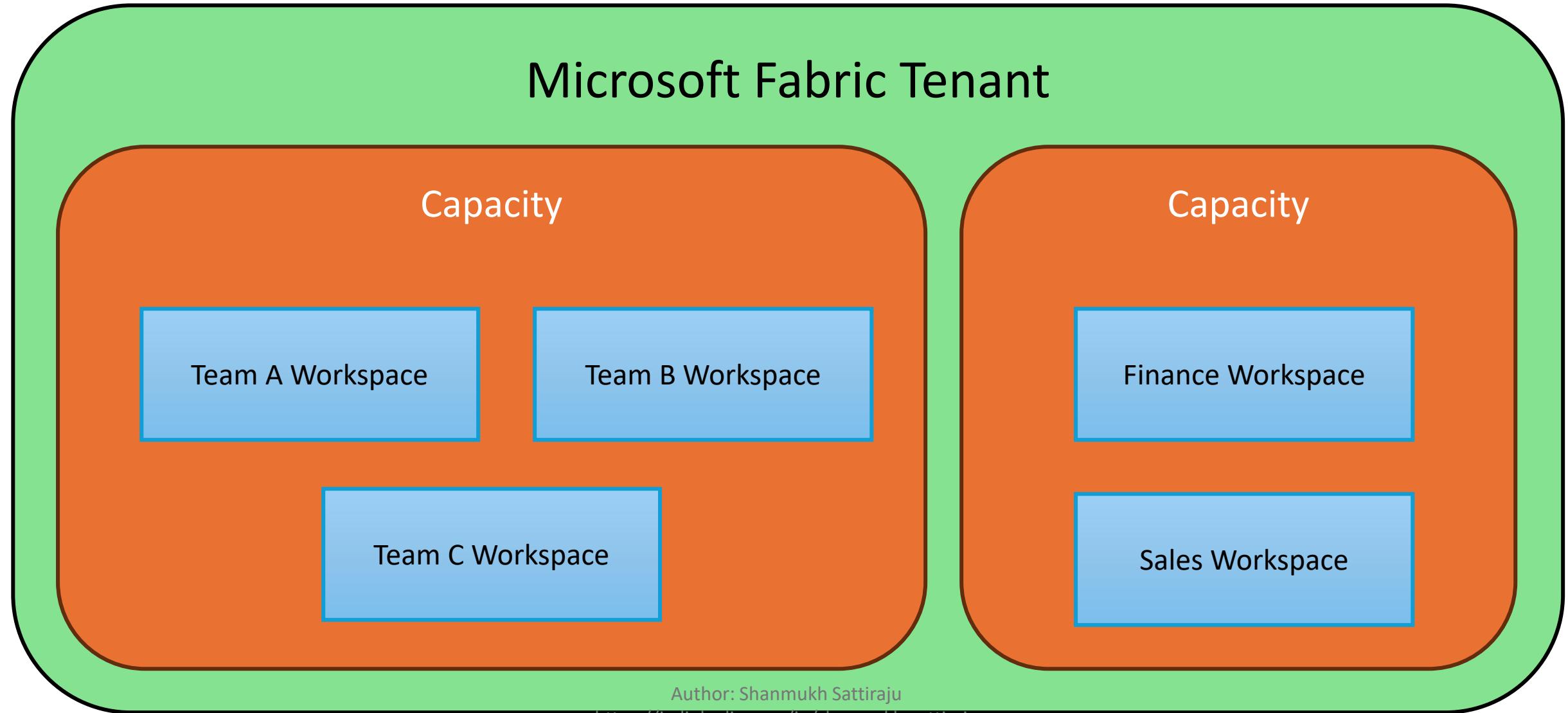


One Copy for all computers



Lakehouse

Access Microsoft Fabric



Workspace roles

Each workspace role contains permissions that allow users to perform certain actions

Role	Can add admins?	Can add members?	Can write data and create items?	Can read data?
Admin	Yes	Yes	Yes	Yes
Member	No	Yes	Yes	Yes
Contributor	No	No	Yes	Yes
Viewer	No	No	No	Yes

Creating Lakehouse

- Lakehouse
- Semantic Model (default)
- SQL analytics endpoint

	Name	Type	Task
	LH_Fabric	Lakehouse	
	LH_Fabric	Semantic model (...)	
	LH_Fabric	SQL analytics end...	

Inside Lakehouse (Lakehouse Explorer)

Managed	Tables	Contains only tables, whether they were automatically generated or explicitly created and registered in the metastore.
	Unidentified	<p>It displays any folders or files present in the managed area that lack the associated tables.</p> <p>If the table created is not a delta table, it will automatically get saved in the Unidentified folder.</p>
Unmanaged	Files	"landing zone" for raw data ingested from various sources metastore.

Upload data to Lakehouse

- To see some features of Lakehouse we need some sample data
- So we are uploading a file to get some sample data
 1. Upload CSV to Tables section (will show as unidentified)
 2. Upload CSV to files section

OneLake Explorer

- Application that integrates OneLake with Windows File Explorer
- You can install it and use it like Onedrive
- Installation URL : [Access Fabric data locally with OneLake file explorer - Microsoft Fabric | Microsoft Learn](#)

SQL analytics endpoint of Lakehouse

You can perform the following actions in the SQL analytics endpoint:

- Query the tables that reference data in your Delta Lake folders in the lake.
- Create views, inline TVFs, and procedures to encapsulate your semantics and business logic in T-SQL.
- Manage permissions on the objects.

Access SQL Analytics Endpoint using SSMS

1. Copy SQL connection string
2. Paste in SSMS
3. Mention database name as Lakehouse
4. Authenticate with same credentials of Fabric

SQL Endpoint - Visual query

1. GUI interface to query tables in your lakehouse.
2. Drag one or more tables onto the canvas, you can use the visual experience to design your queries.
3. Save the visual queries as SQL View

Semantic model

- Source of data ready for reporting, visualization.
- Created by default when creating a Lakehouse
- All tables and Views created in lakehouse will be synced to default semantic model
- You can create a Power BI report using this or you can also create a new semantic model and create report using that



Data Factory

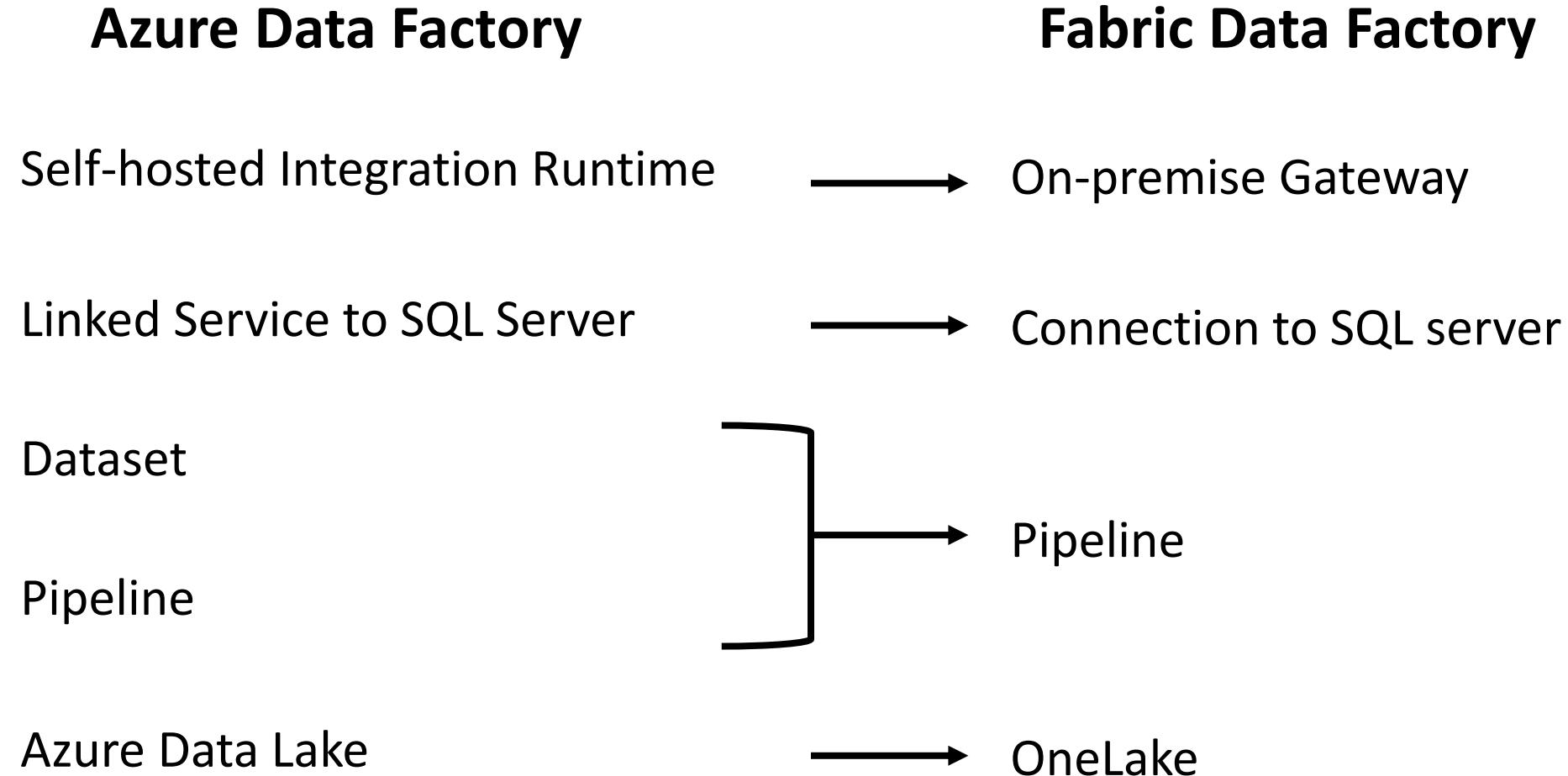
Ways to load data into Lakehouse

1. Local file/folder upload ✓
2. Copy tool in pipelines -----> Data Factory
3. Dataflow Gen2 -----> Data Factory
4. Notebook code
5. Shortcut

Consideration when loading data

Use case	Recommendation
Small file upload from local machine	Use Local file upload
Small data or specific connector	Use Dataflows
Large data source	Use Copy tool in pipelines
Complex data transformations	Use Notebook code

Ingest On-premise SQL server to OneLake



Data Gateway Types

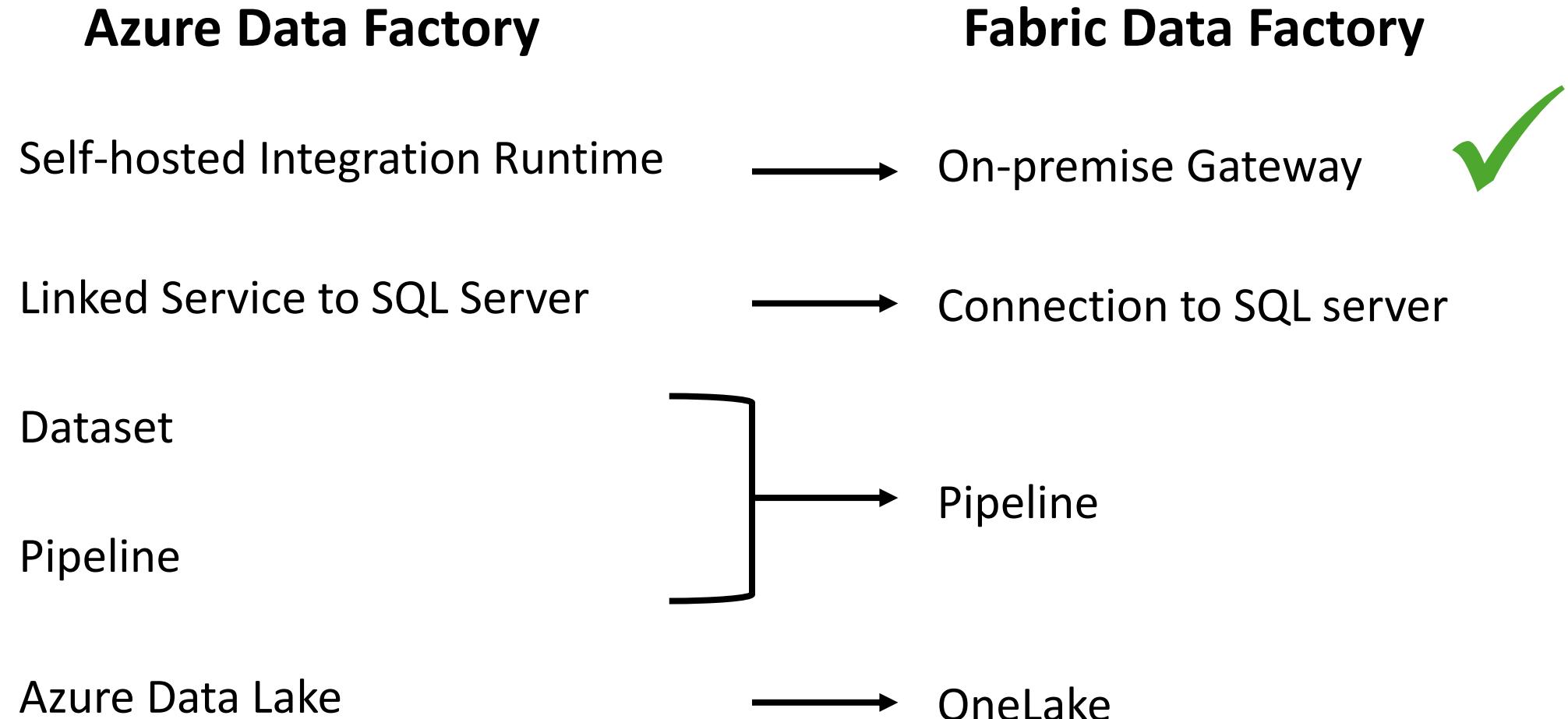
- On-premise Data Gateway
 - The on-premises data gateway provides quick and secure data transfer between on-premises data and several Microsoft cloud services, such as Power BI, Power Apps
- V-Net Data Gateway
 - A virtual network (VNet) data gateway helps you to connect from Microsoft Cloud services to your Azure data services within a virtual network

Installing On-premise Gateway

- Install a Standard Gateway using

[Install an on-premises data gateway | Microsoft Learn](#)

Ingest On-premise SQL server to OneLake

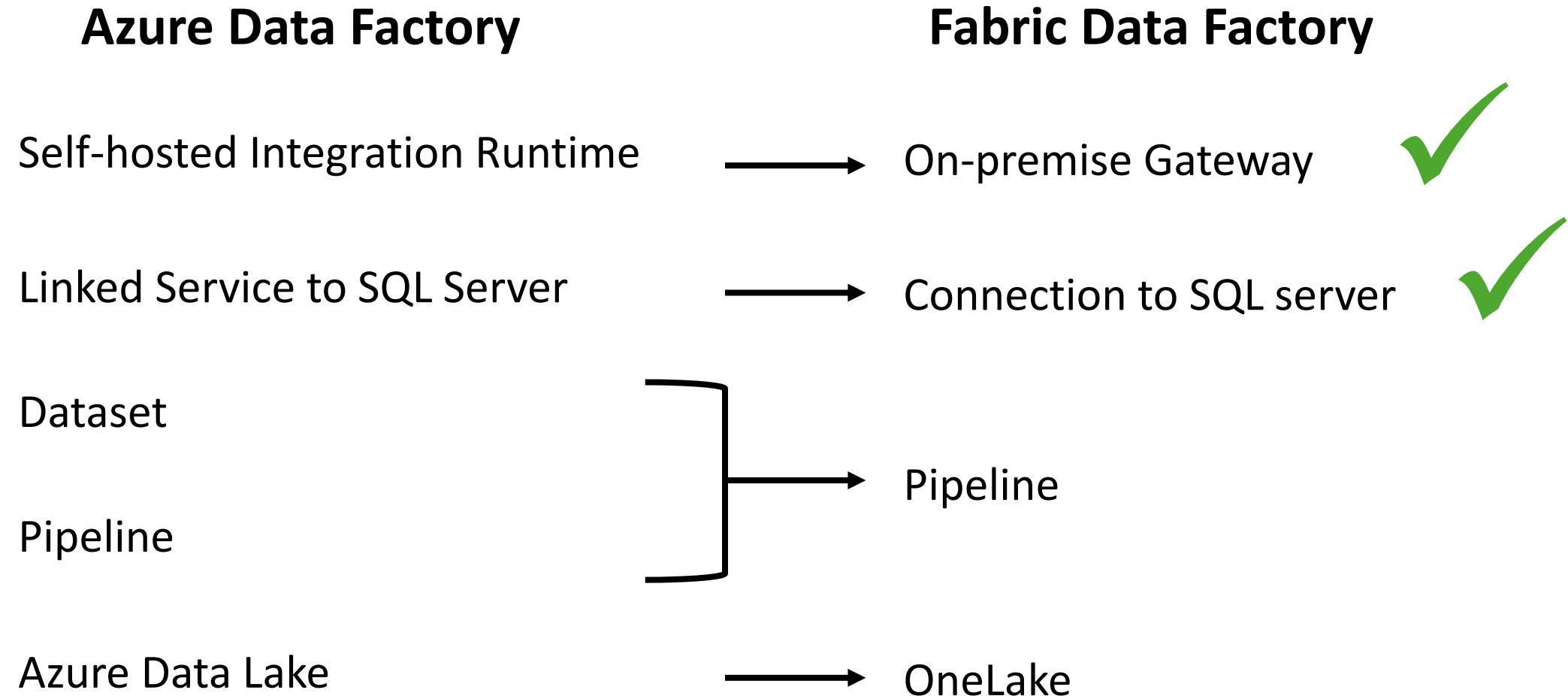


Connections

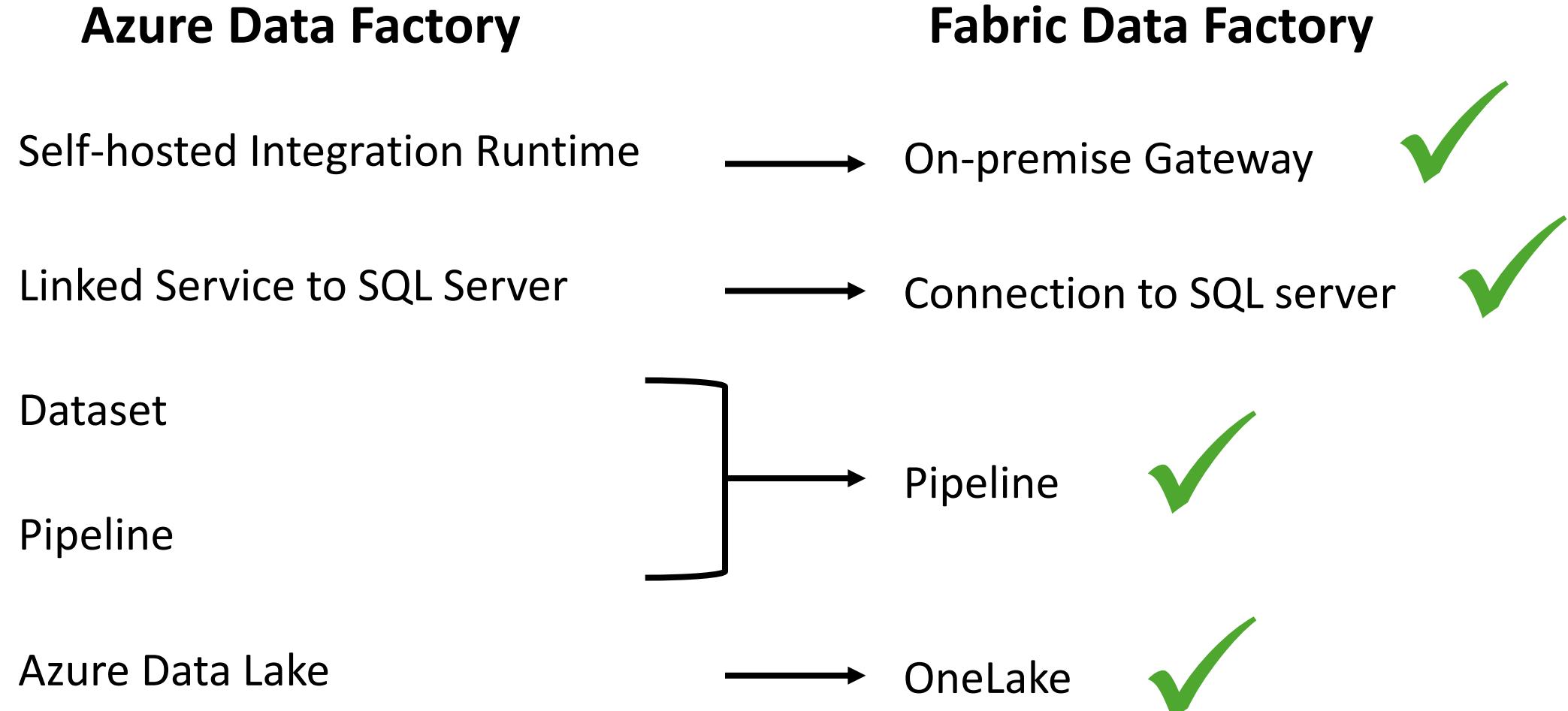
Connections establish authentication to Data Sources like linked services in Azure Data Factory / Synapse Analytics



Ingest On-premise SQL server to OneLake



Ingest On-premise SQL server to OneLake



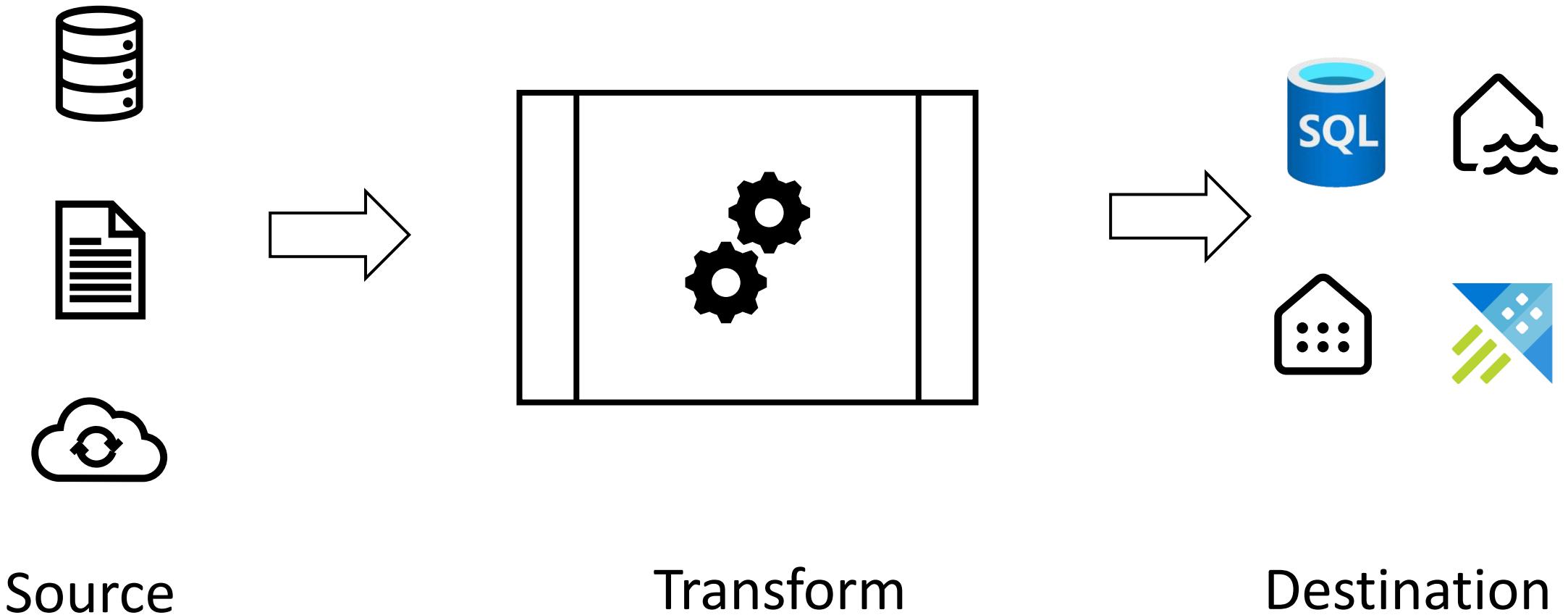
Pipeline Triggers in Fabric

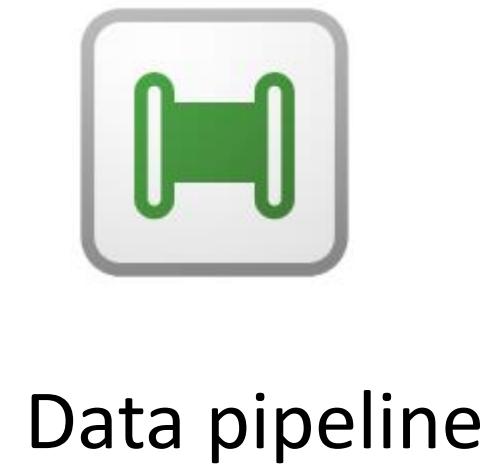
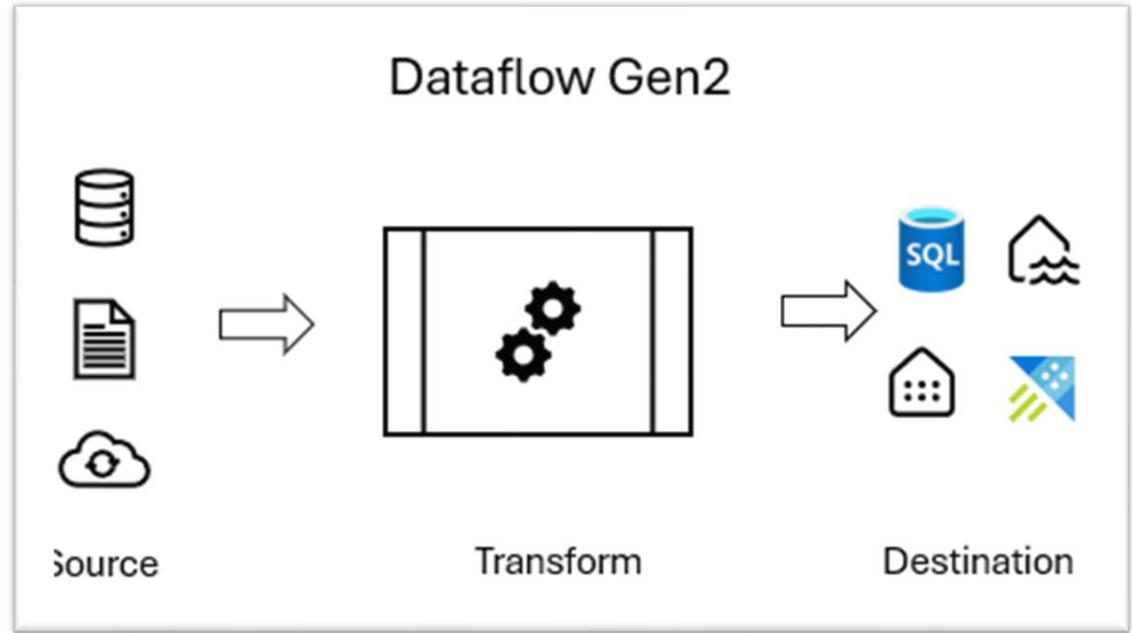
Triggers invoke pipelines in Data Factory

Trigger types are:

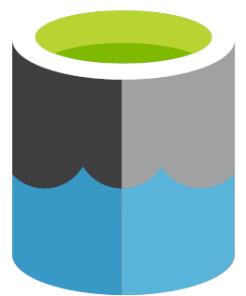
1. Schedule trigger
2. Storage event trigger (preview) -- Only for Azure Blob storage

Dataflow Gen2

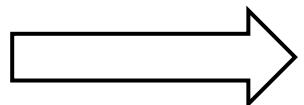




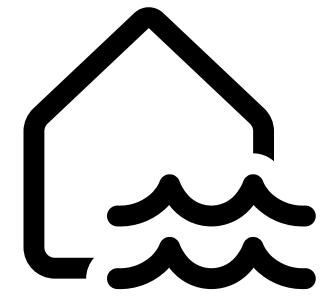
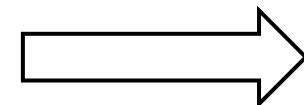
Dataflow Gen2



Datalake Gen2



Dataflow Gen2

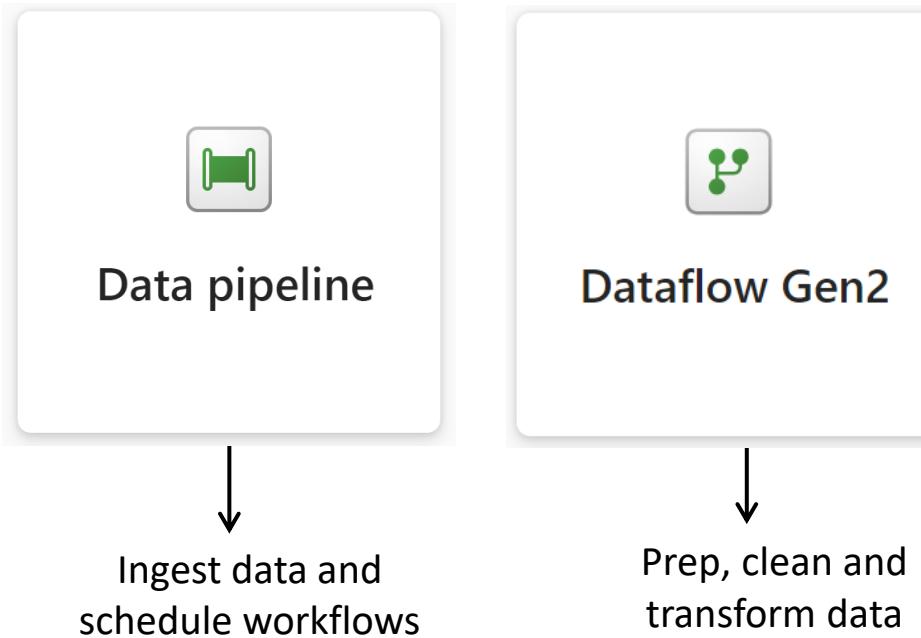


Lakehouse

Role: Storage Data
Blob Contributor



Data Factory summary





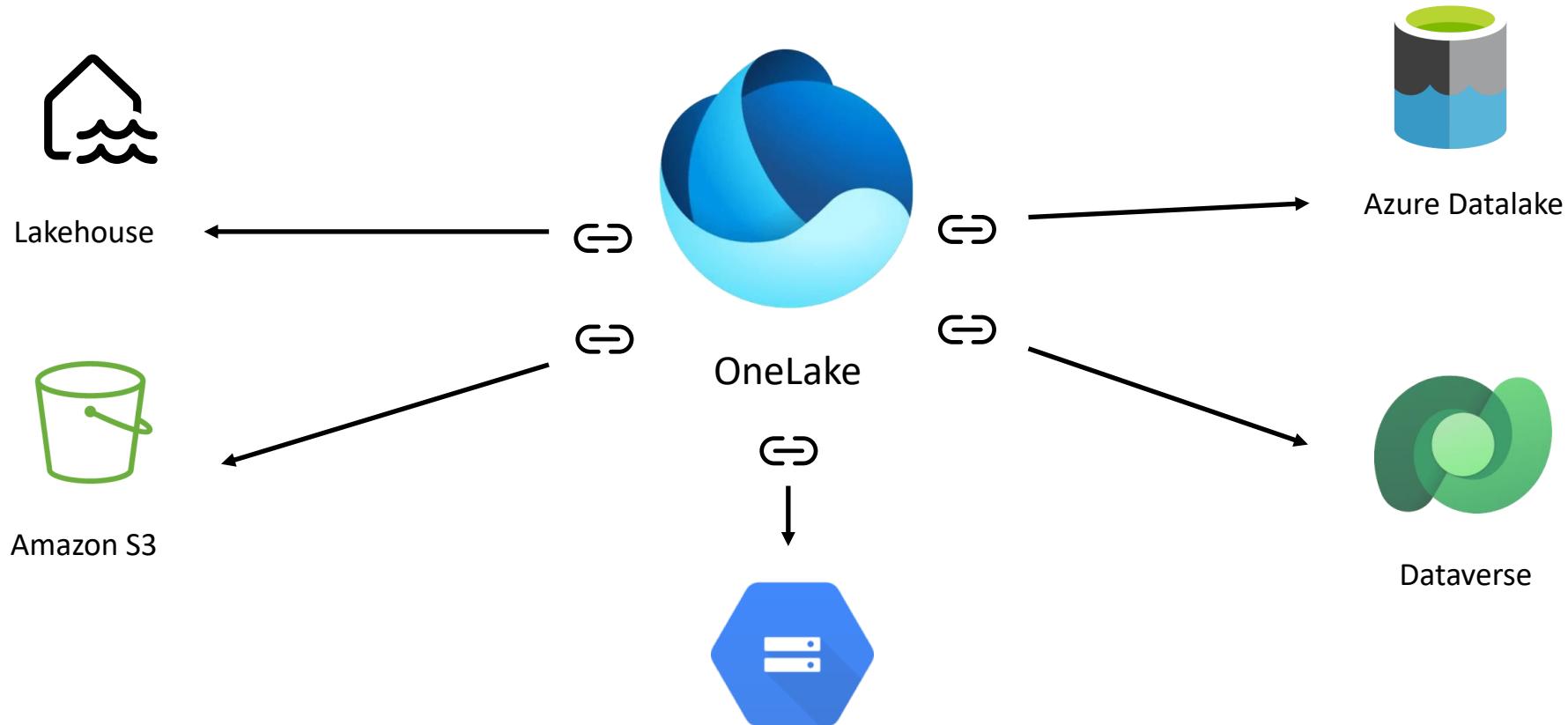
OneLake

Author: Shanmukh Sattiraju
<https://in.linkedin.com/in/shanmukh-sattiraju>

Ways to load data into Lakehouse

1. Local file/folder upload ✓
2. Copy tool in pipelines -----> Data Factory ✓
3. Dataflow Gen2 -----> Data Factory ✓
4. Shortcut -----> OneLake
5. Notebook code -----> Data Engineering

🔗 Shortcut



Google Cloud Storage
(preview)

Author: Shanmukh Sattiraju

<https://in.linkedin.com/in/shanmukh-sattiraju>

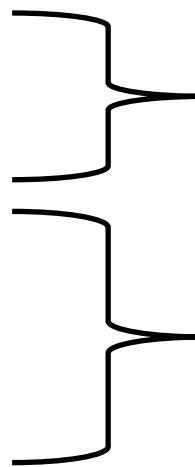
Shortcuts in Fabric



Creating a shortcut

1. Source location

- A. Lakehouse
- B. Warehouse
- C. Azure Datalake
- D. Amazon S3
- E. Dataverse



Internal to Fabric

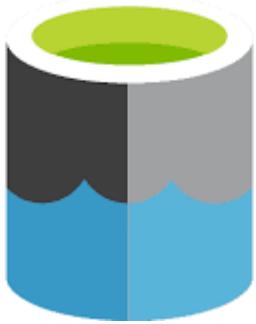
External to Fabric

2. Authentication to Source data

3. Destination

- A. Lakehouse
- B. KQL Database

Creating a shortcut in Files

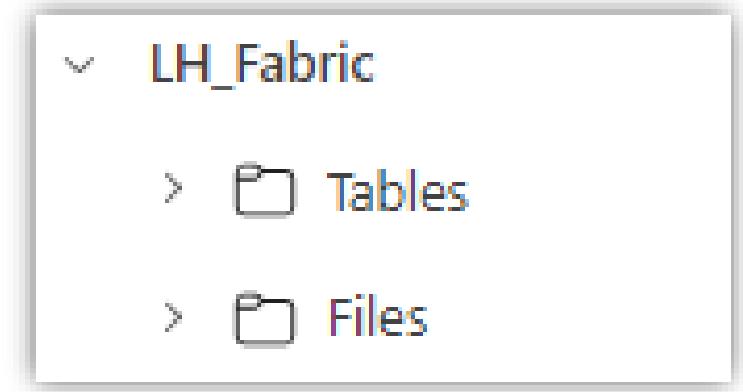
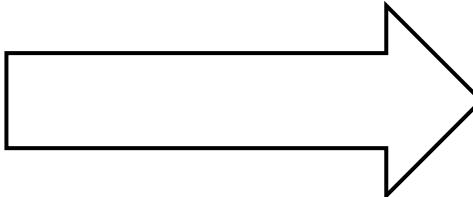


Azure Datalake Gen2

Container: shortcutfile

SubFolder: Emp

Files: Emp1.csv,Emp2.csv,etc



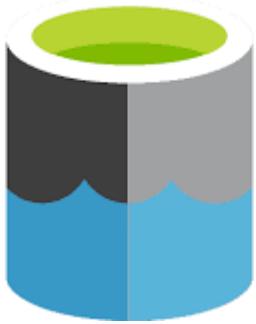
Lakehouse

Section: Files

Folder: Emp

Role: Storage Blob Data Contributor

Creating a shortcut in Table

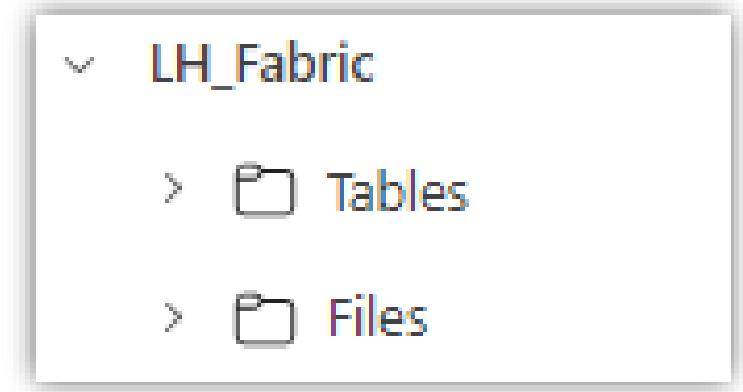
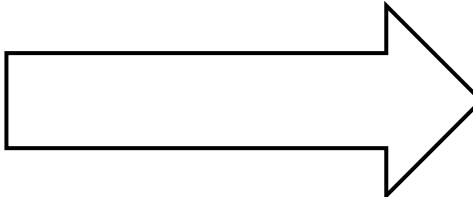


Azure Datalake Gen2

Container: shortcutfile

SubFolder: Emp

Files: Emp1.csv,Emp2.csv,etc

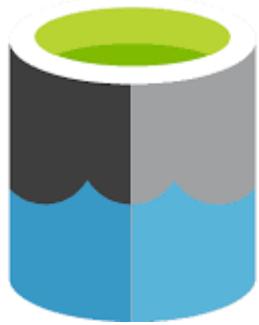


Lakehouse

Section: Table

Role: Storage Blob Data Contributor

Creating a shortcut in Table

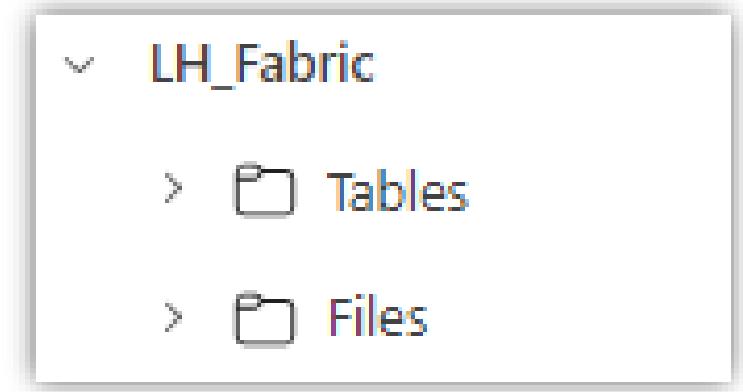
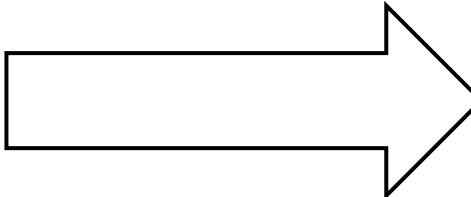


Azure Datalake Gen2

Container: shortcutfile

SubFolder: Emp

Files: Emp1.csv,Emp2.csv,etc



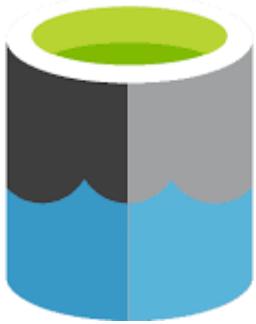
Lakehouse



Section: Table

Role: Storage Blob Data Contributor

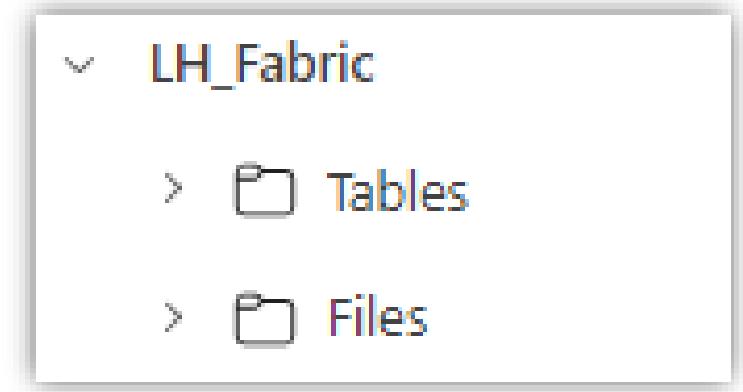
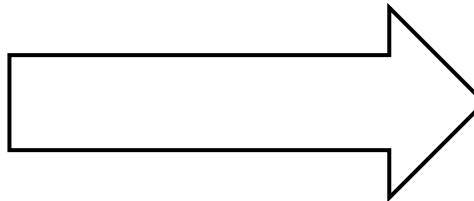
Creating a shortcut in Table



Azure Datalake Gen2

Container: shortcutdeltaroot

Files: Delta format

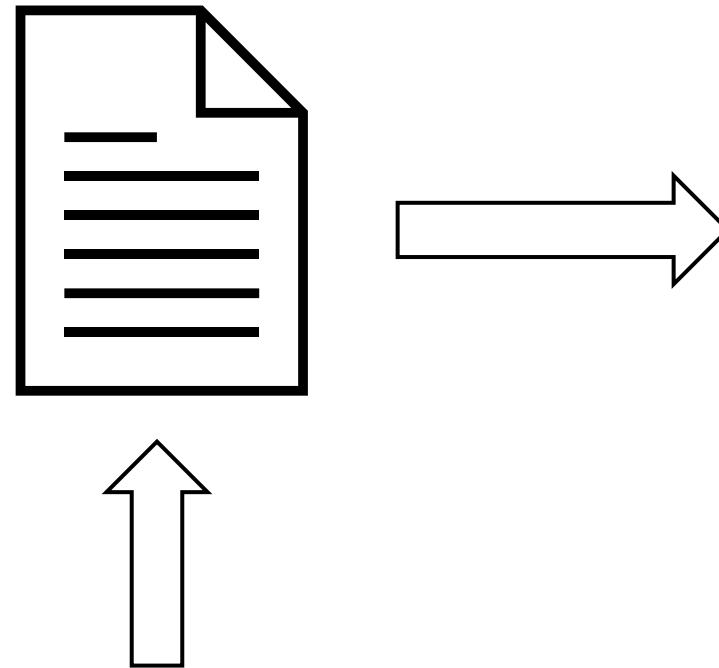


Lakehouse

Section: Table

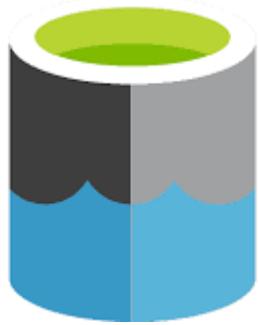
Role: Storage Blob Data Contributor

Creating a delta file



Parquet File

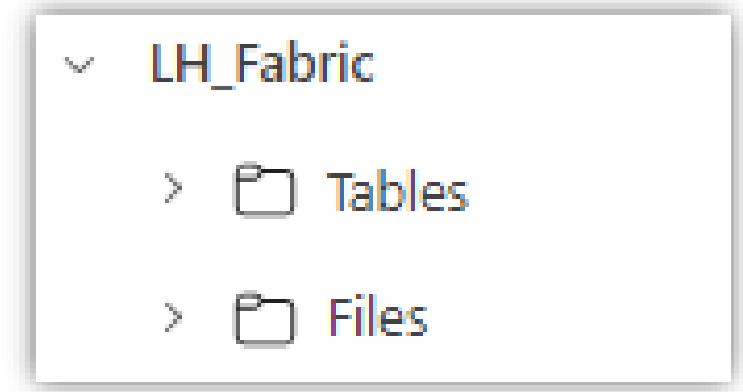
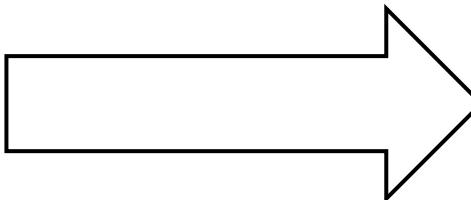
Creating a shortcut in Table



Azure Datalake Gen2

Container: shortcutdeltaroot

Files: <delta format>

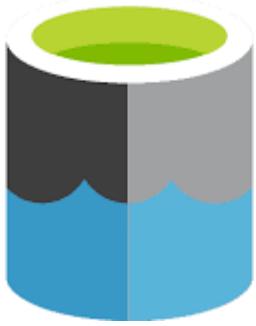


Lakehouse

Section: Table

Role: Storage Blob Data Contributor

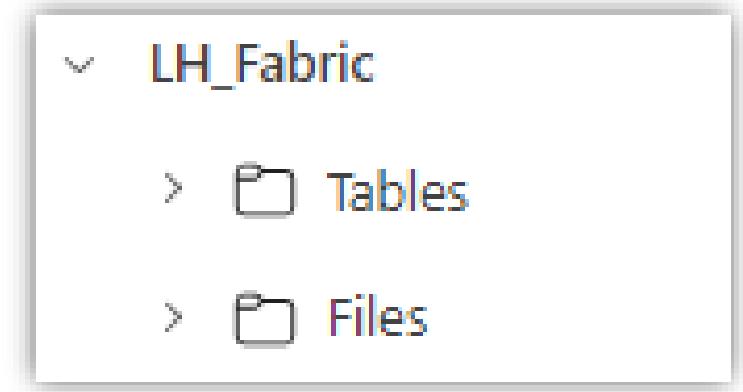
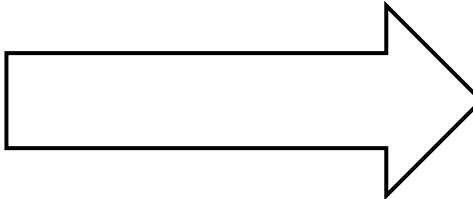
Creating a shortcut in Table



Azure Datalake Gen2

Container: shortcutdeltasub

Files: data/<delta format>

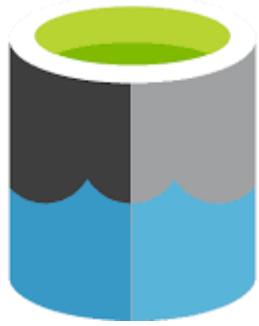


Lakehouse

Section: Table

Role: Storage Blob Data Contributor

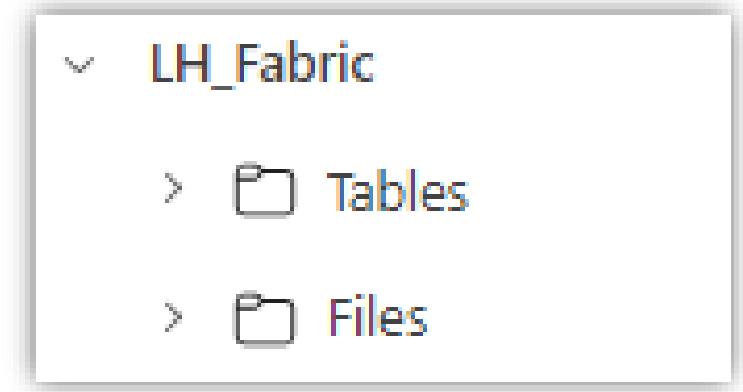
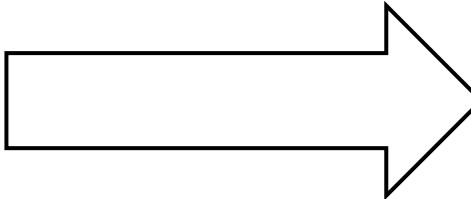
Creating a shortcut in Table



Azure Datalake Gen2

Container: shortcutparquet

Files: <parquet format>



Lakehouse

Section: Table

Role: Storage Blob Data Contributor

Shortcuts in Fabric

Shortcuts in Tables

- In the **Tables** folder, you can only create shortcuts at the top level. Shortcuts aren't supported in other subdirectories of the **Tables** folder
- Data to be in Delta/parquet format so that lakehouse automatically synchronizes the metadata and recognizes the folder as a table

Shortcuts in Fabric

Shortcuts in Files

- If your shortcut location data is in form of sub-directories go with storing them in files.
- If they are not in delta-parquet format , store them in files

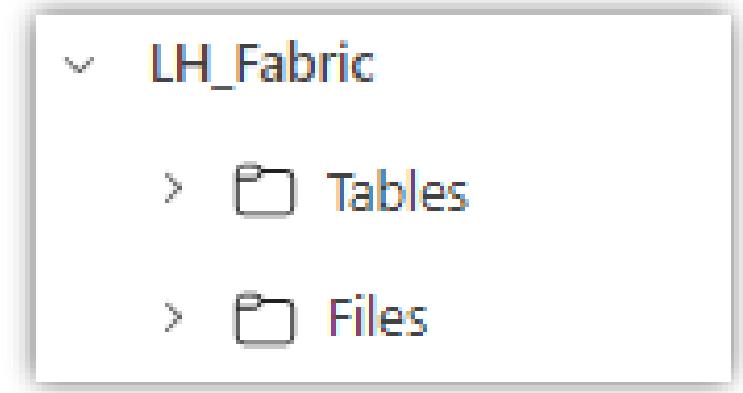
Shortcut updating scenario



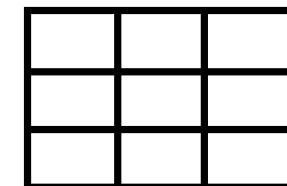
Azure Datalake Gen2

Container: shortcutdeltaroot

Files: <delta format>



Lakehouse



Updating data in Shortcut Table

Azure Datalake Gen2

Lakehouse



Read access



Update data



Write access



Update data

Updating data in Shortcut Table

Azure Datalake Gen2

Lakehouse



Read access



Update data

Gets updated



Write access



Update data

Updating data in Data lake

Azure Datalake Gen2

Lakehouse

Update data



?

Updating data in Data lake

Azure Datalake Gen2

Lakehouse

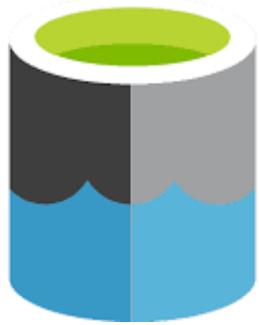
Update data



Gets updated



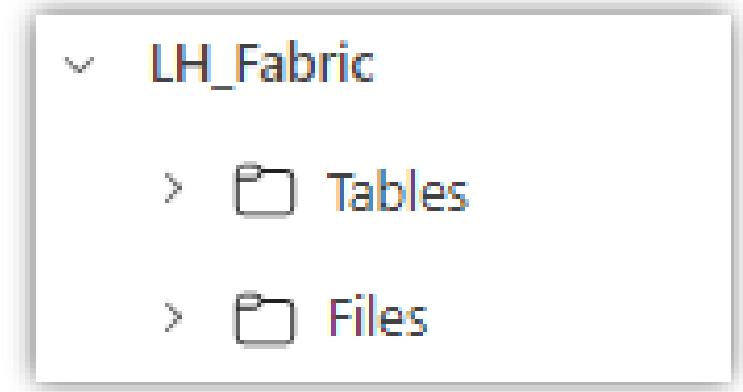
Shortcut deletion scenarios



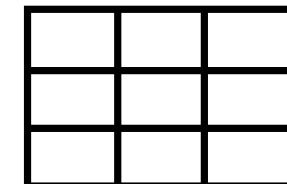
Azure Datalake Gen2

Container: shortcutdeltaroot

Files: <delta format>



Lakehouse



Scenario 1: Delete content in Shortcut of Files section

Azure Datalake Gen2

Lakehouse Files

?

A horizontal sequence of twelve black rectangular bars of varying widths, followed by a large black arrow pointing left.

Delete data

Scenario 1: Delete content in Shortcut of Files section

Azure Datalake Gen2

Lakehouse Files

Gets deleted



Delete data

Scenario 2: Delete a specific content in ADLS

Azure Datalake Gen2

Lakehouse
Files

Delete data



?

Scenario 2: Delete a specific content in ADLS

Azure Datalake Gen2

Lakehouse
Files

Delete data



Gets deleted



Scenario 3: Delete content in Shortcut of Tables section

Azure Datalake Gen2

Lakehouse Tables

?

A horizontal sequence of twelve black bars of varying widths, followed by a large black arrow pointing left.

Delete data

Scenario 3: Delete content in Shortcut of Tables section

Azure Datalake Gen2

Lakehouse Tables

Gets deleted



Delete data

Scenario 4: Delete a specific content of delta table in ADLS

Azure Datalake Gen2

Lakehouse
Tables

Delete data



?

Scenario 4: Delete a specific content of delta table in ADLS

Azure Datalake Gen2

Lakehouse
Tables

Delete data



Gets deleted



Scenario 5: Deleting shortcut completely in Lakehouse

Azure Datalake Gen2

Lakehouse Tables

?

A horizontal sequence of eleven black rectangular bars of varying widths, followed by a large black arrow pointing left.

Delete shortcut

Scenario 5: Deleting shortcut completely in Lakehouse

Azure Datalake Gen2

Lakehouse Tables

Will not be deleted ← - - - - - - - - - - - - - - Delete shortcut



Shortcut deletion scenarios

Scenario 1	File	Delete content in Shortcut of Files section	Deletes in Datalake	✓
Scenario 2	File	Delete a specific content in ADLS	Deletes in Lakehouse	✓
Scenario 3	Table	Delete content in Shortcut of Tables section	Deletes in Datalake	✓
Scenario 4	Table	Delete a specific content of delta table in ADLS	Deletes in Lakehouse	✓
Scenario 5	File & Table	Deleting shortcut completely in Lakehouse	ADLS data will not be deleted	✗



Synapse Data Engineering

Ways to load data into Lakehouse

1. Local file/folder upload ✓
2. Copy tool in pipelines -----> Data Factory ✓
3. Dataflow Gen2 -----> Data Factory ✓
4. Shortcut -----> OneLake ✓
5. Notebook code -----> Data Engineering

Spark in Microsoft Fabric

Workspace settings



General



License info



Azure connections



System storage



Git integration



OneLake



Workspace identity



Network security



Power BI



Data
Engineering/Science



Spark settings

Spark settings

Configure and manage settings for Spark workloads and the default environment for the workspace.

Pool

Environment

Jobs

High concurrency

Automatic log

Default pool for workspace

Use the automatically created starter pool or create custom pools for workspaces and items in the capacity. If the setting Customize compute configurations for items is turned off, this pool will be used for all environments in this workspace.

StarterPool

Pool details



Node family

Memory optimized

Node size

Medium

Number of nodes

1 - 10

Customize compute configurations for items

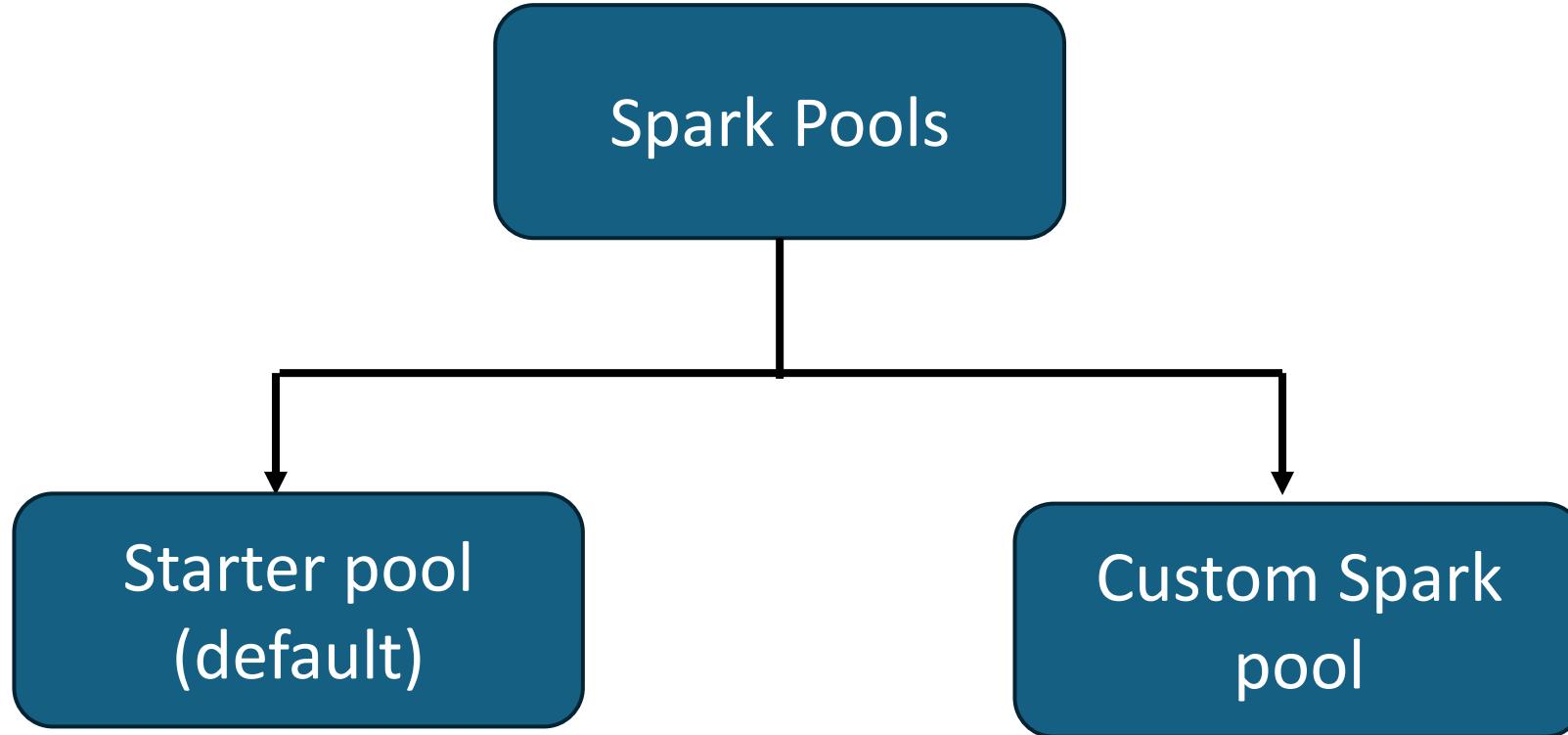


On

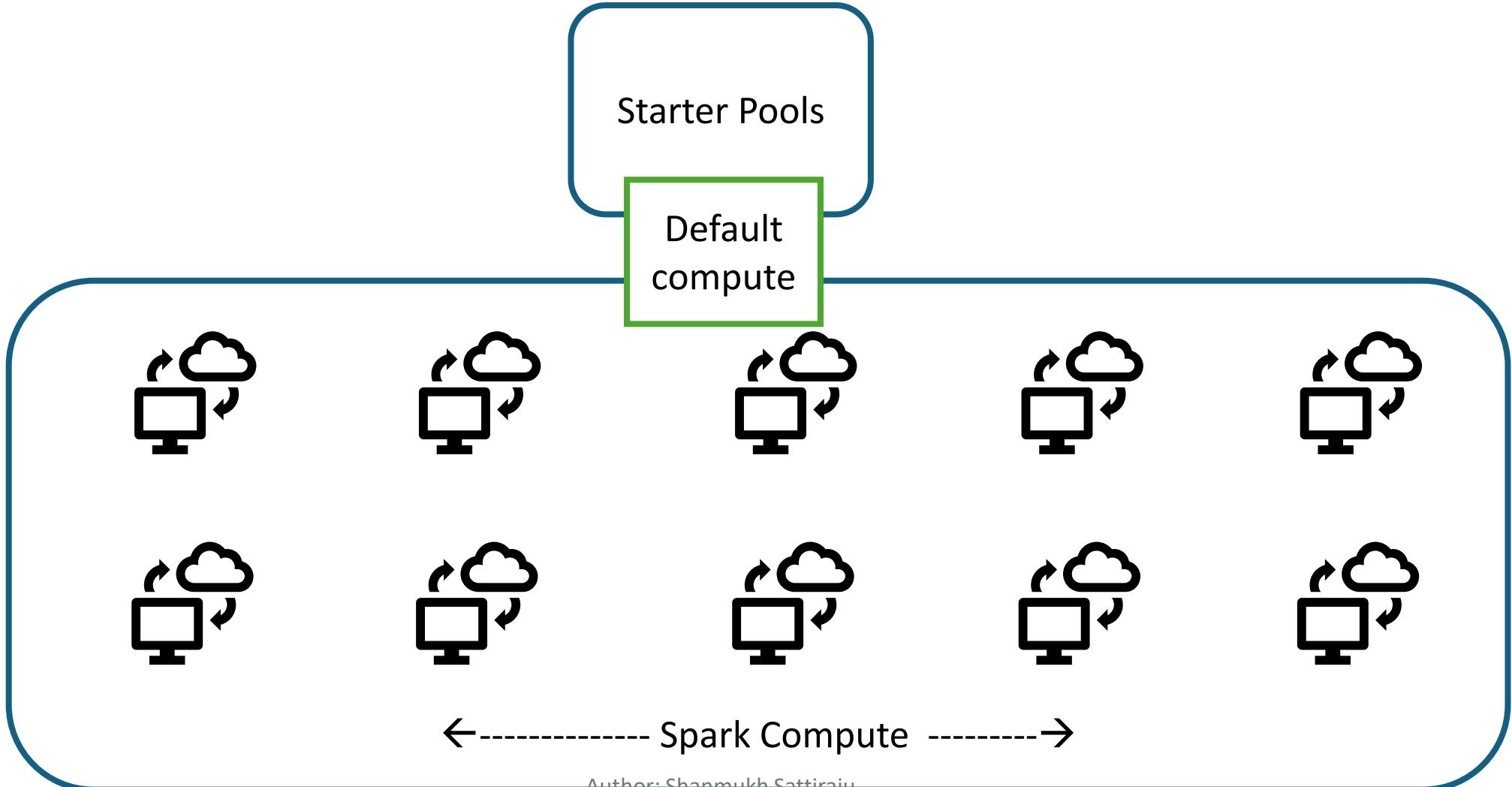
When turned on, users can adjust compute configuration for individual items

Author: Shanmukh Sattiraju

Spark pools in Microsoft Fabric



Starter pools



Node Size

Size	vCore	Memory
Small	4	32 GB
Medium	8	64 GB
Large	16	128 GB
X-Large	32	256 GB
XX-Large	64	512 GB

Starter pools

Starter pool configuration

Node Family	Memory Optimized
Node Size	Medium
Min and Max Nodes	1 to 10
Auto scale	On
Dynamic Allocation	On

Customize Starter pool

Network security

Power BI

Data

Engineering/Science

Spark settings

Data Factory

Autoscale

If enabled, your Apache Spark pool will automatically scale up and down based on the amount of activity.

Enable autoscale

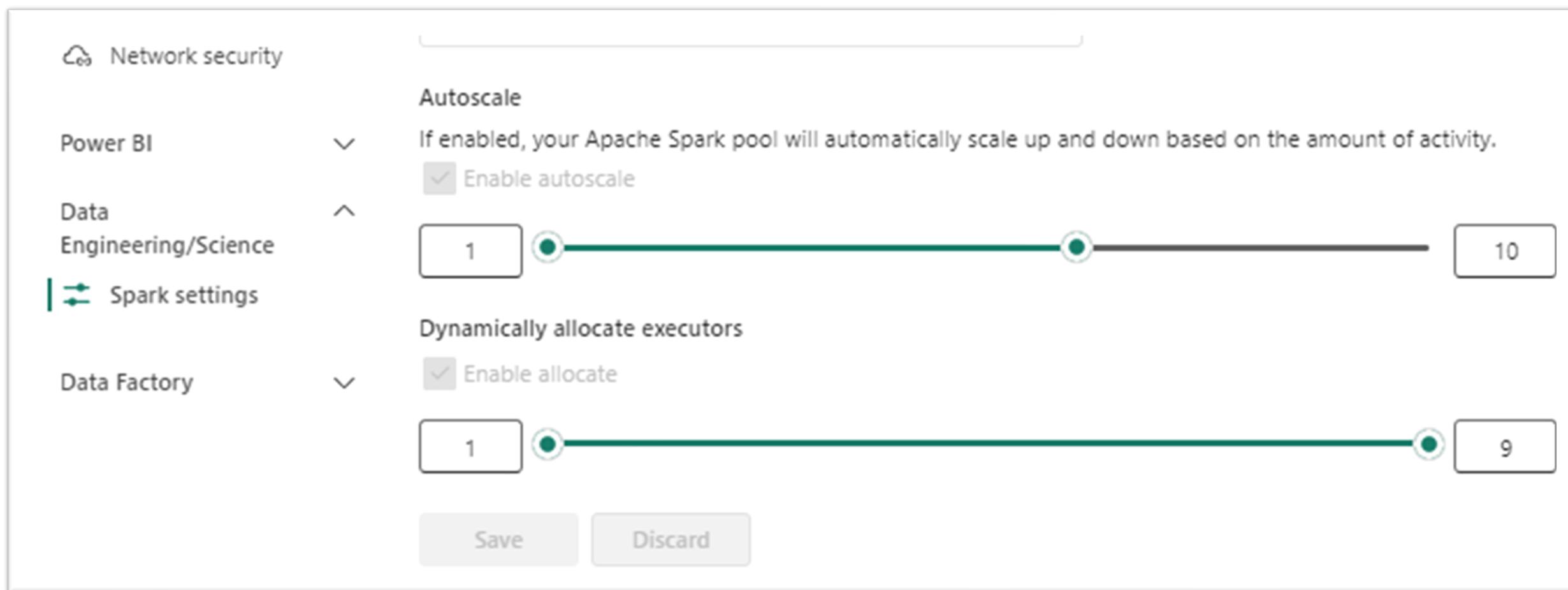
1  10

Dynamically allocate executors

Enable allocate

1  9

Save Discard



Custom pools

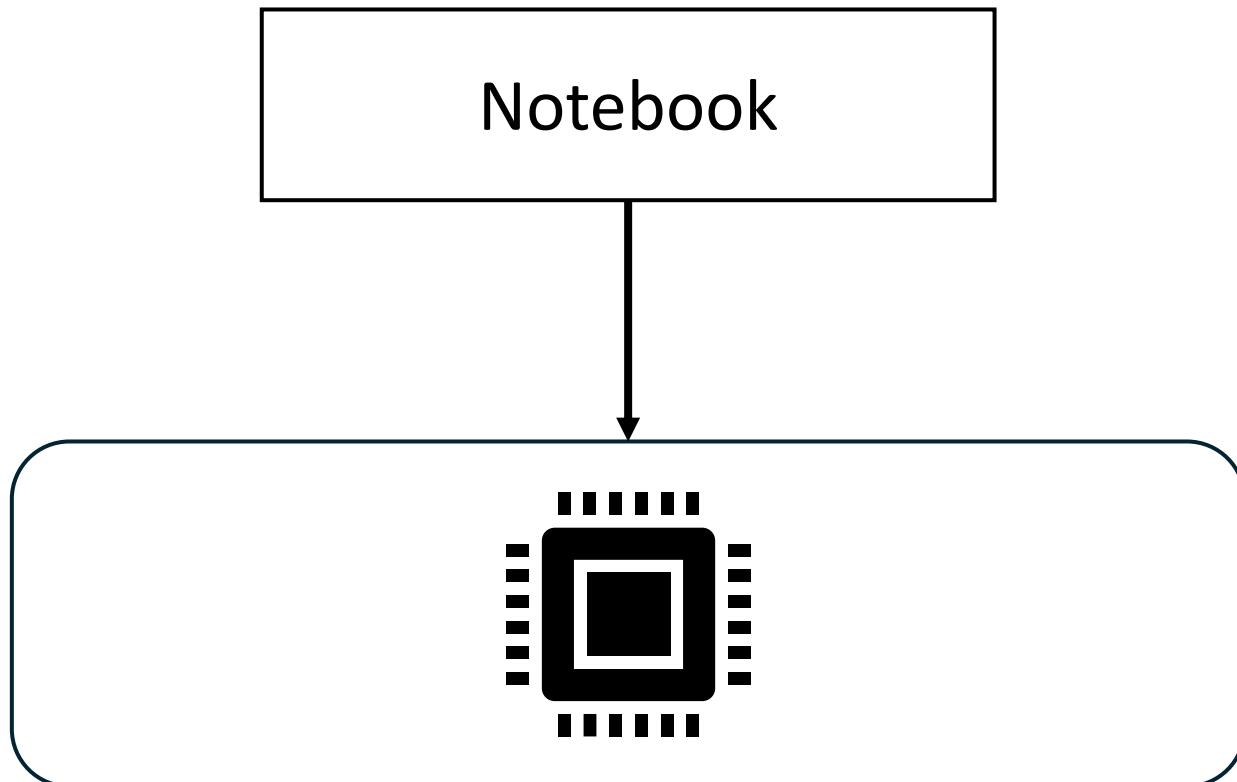
Custom Pools



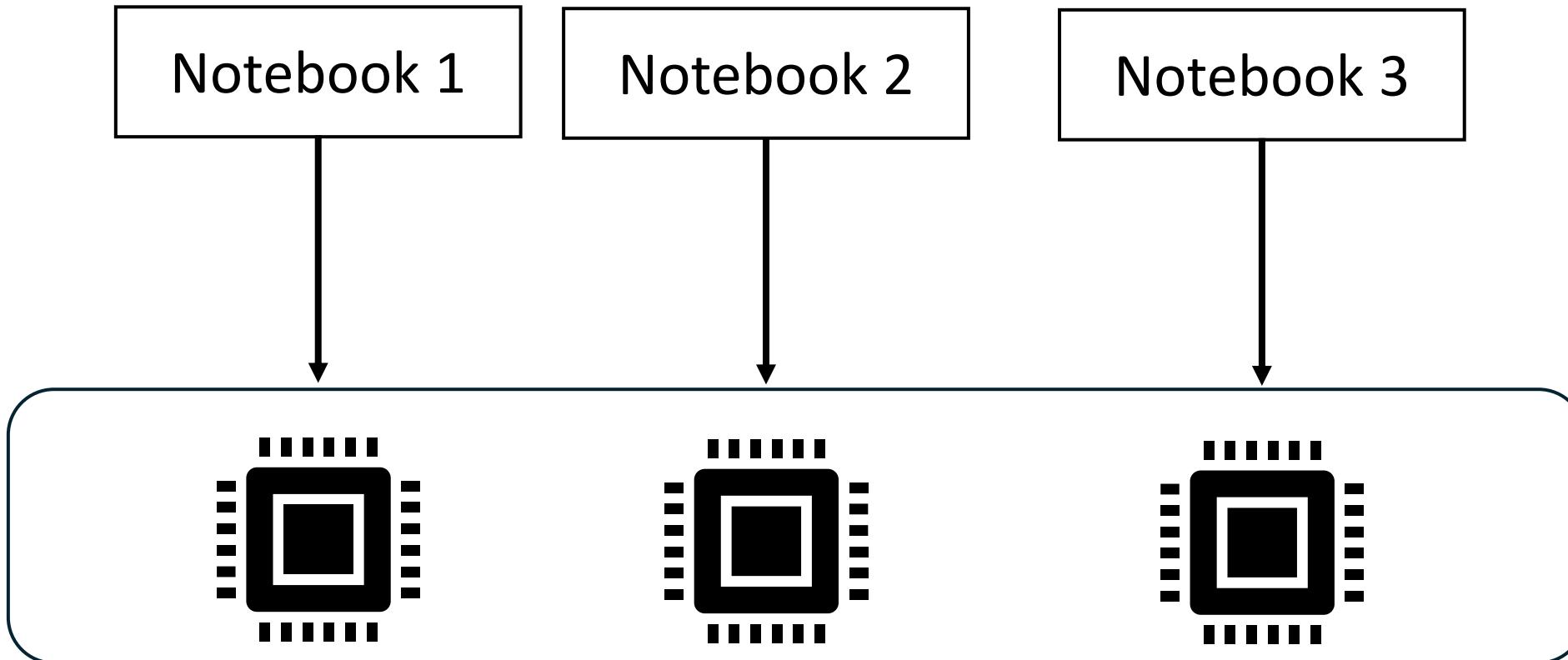
Cluster Sized based on customer
specifications

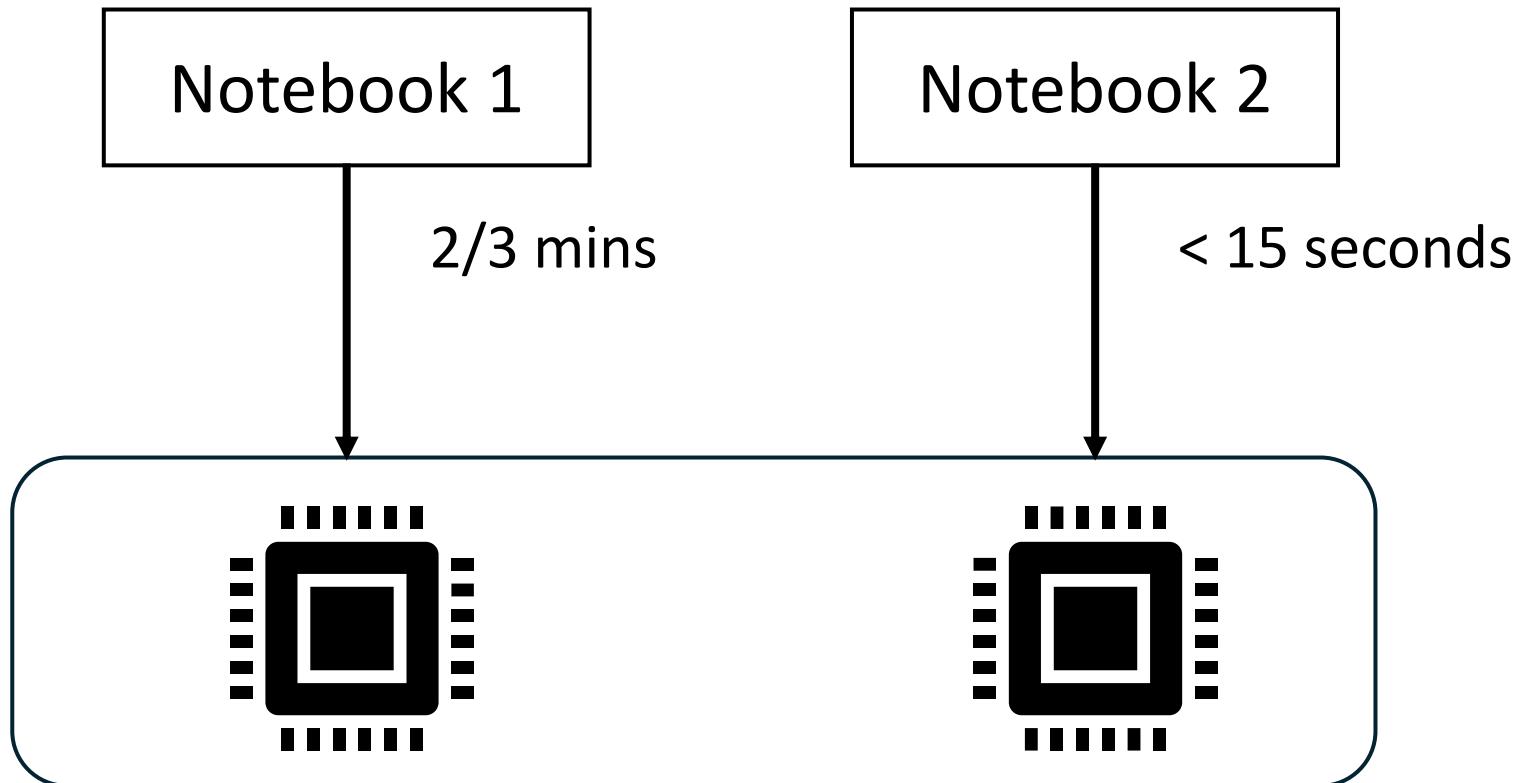


Standard session (default)



High Concurrency





High concurrency Session

Benefits of High Concurrency

- **Multi-task:**
 - One user can use multiple notebooks with one session
 - Prevents delays due to session creation
- **Security:**
 - Session sharing is always within a single user boundary
- **Cost-effective:**
 - Better resource utilization and cost-saving

Session sharing conditions

- Sessions should be within a single user boundary.
- Sessions should have the same default Lakehouse configuration.
- Sessions should have the same Spark compute properties.

Notebook basics

MSSparkUtils

Author: Shanmukh Sattiraju
<https://in.linkedin.com/in/shanmukh-sattiraju>

MSSparkUtils

- Microsoft Spark Utilities (MSSparkUtils) is a built-in package to help you easily perform common tasks
- You can use MSSparkUtils to work with file systems, to get environment variables, to chain notebooks together, and to work with secrets

Mssparkutls.help()

- **Notebook**: Utility for notebook operations (e.g, chaining Fabric notebooks together)
- **Lakehouse**: [Preview] Utility for Lakehouse operations (e.g, create, delete, update, list lakehouse)
- **fs**: Utility for filesystem operations in Fabric
- **Credentials**: Utility for obtaining credentials (tokens and keys) for Fabric resources
- **Runtime**: Utility for getting context information of runtime that matters to current session

Mssparkutils.fs.help()

Mounting:

- Mount an Azure Datalake
- Mount a lakehouse

FS utils - fastcp

- Performant copy file
- Faster way to copy files , especially large volume
- Makes use of Azcopy

Notebook utils

- Exit
- Run

Notebook – runMultiple

- Execute multiple notebooks simultaneously without waiting for each one to finish
- Specific dependencies and order of execution using JSON
- You can optimize the Spark compute resources using this
- You can view the snapshot of each notebook run record
- You can use exit value of one notebook and use them in downstream tasks

Ways to load data into Lakehouse

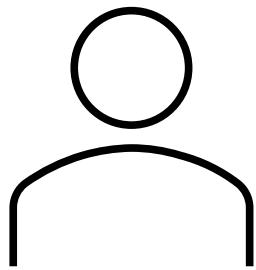
1. Local file/folder upload ✓
2. Copy tool in pipelines -----> Data Factory ✓
3. Dataflow Gen2 -----> Data Factory ✓
4. Shortcut -----> OneLake ✓
5. Notebook code -----> Data Engineering

Ingest data from ADLS to Lakehouse

Authentication methods from Notebook:

- Using Microsoft Entra ID of user
- Using Service principal
- Using Service principal with key vault

Using Microsoft Entra ID of user



Storage blob data contributor

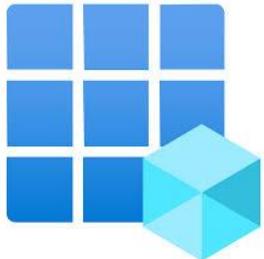


Role based access



Azure Datalake Gen2

Using Service principal



**App ID
Tenant ID
Secret Key**

Storage blob data contributor

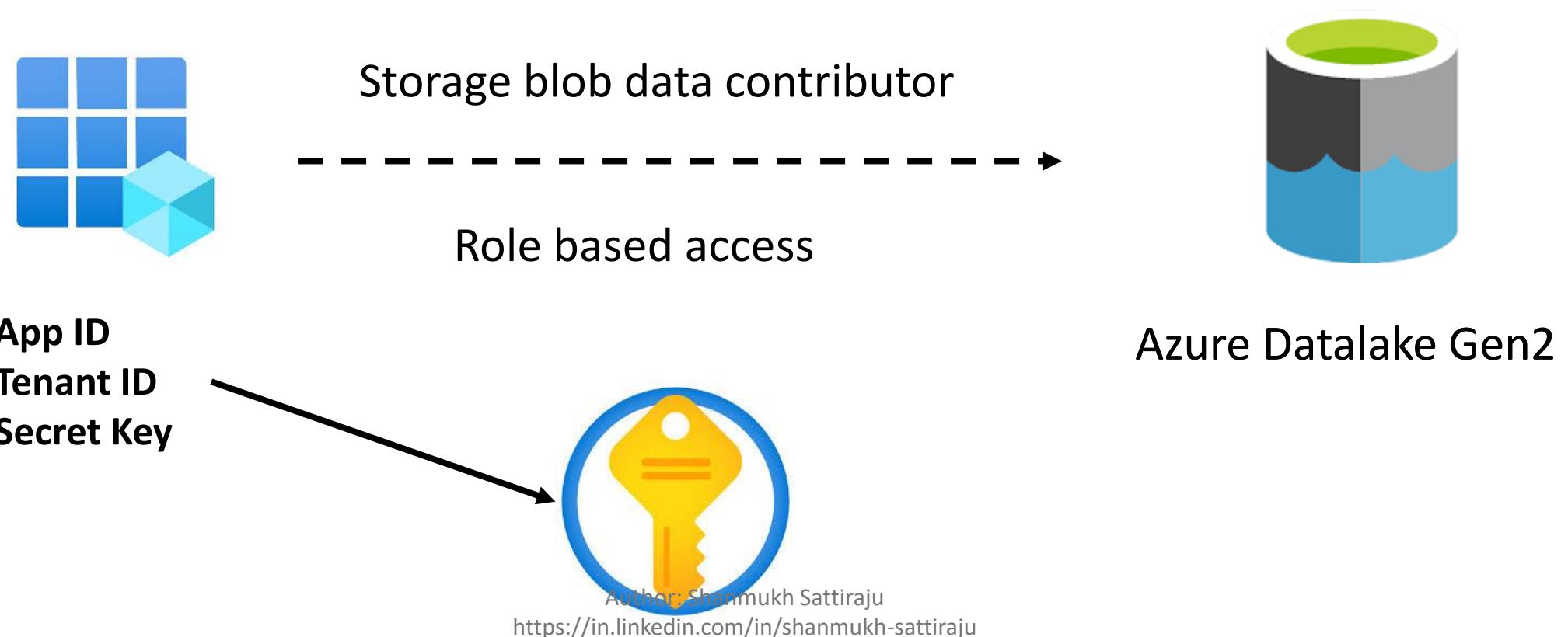


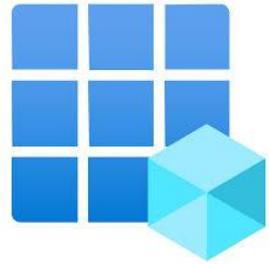
Role based access



Azure Datalake Gen2

Using Service principal using Key vault





Storage blob data contributor



**App ID
Tenant ID
Secret Key**

Role based access

Azure Datalake Gen2



Fabric Workspace
Identity

Ways to load data into Lakehouse

1. Local file/folder upload ✓
2. Copy tool in pipelines -----> Data Factory ✓
3. Dataflow Gen2 -----> Data Factory ✓
4. Shortcut -----> OneLake ✓
5. Notebook code -----> Data Engineering ✓

Managed vs External table

- Managed: Data and table definition is managed by engine of Microsoft Fabric
- External: Table definition is managed by Fabric engine but data is stored somewhere

Managed table

- Handles both data and metadata
- Data stored in Lakehouse's Table folder
- Metadata includes info about Lakehouse , Tables, schema, etc.
- Dropping table removes ALL data and metadata

Ways to create Managed Table

1. Using .saveAsTable() from dataframe
2. Using SQL CREATE TABLE syntax
3. Using DeltaTableBuilder API

External Table

- Handles metadata only
- Specify LOCATION to store table data
- Dropping table removed metadata but data persists in the specified LOCATION

Ways to create External Table

1. Using .saveAsTable() from dataframe
2. Using SQL CREATE TABLE syntax

Data wrangler

- A GUI based tool in notebooks to perform most common operations on the dataframes.
- No-code based approach
- Provided quick summary view of data

Environments in notebooks

- You can use your custom libraries, configure runtime and upload resources
- Flexible way to customize compute configurations for running your Spark jobs
- Configure session level properties to customize memory and core of executors

V-Order optimization

- Write time optimization to the parquet file format
- Enables lightning-fast reads under the fabric compute engines
- It is 100% open-source parquet format compliant
- This is enabled by default
- Works by applying special sorting, row group distribution and compression on parquet files

Spark job definition

- Allows you to submit batch/streaming jobs to Spark clusters
- To run these, you must have at least one lakehouse to serve as default file system context
- You can schedule this job definition

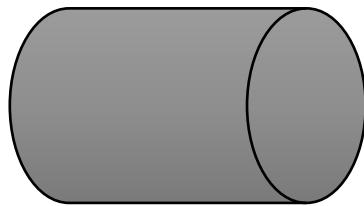
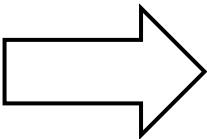
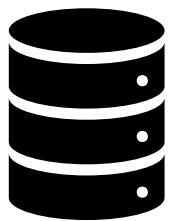
[Synapse - Choosing Between Spark Notebook vs Spark Job Definition - Microsoft Community Hub](#)



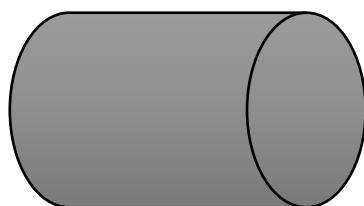
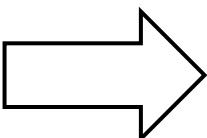
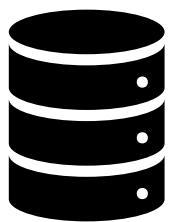
Source

Central Storage

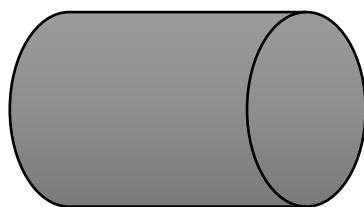
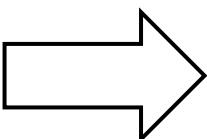
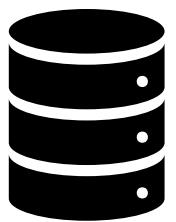
Consumers



Finance



HR



IT

Domains

- Microsoft Fabric's data mesh architecture supports organizing data into ***domains***
- Enable data consumers to be able to filter and discover content by domain
- Ensure that the data in your organization is well structured and effectively governed, and that data consumers can easily find the content they need.
- Domains is a key enabler for data mesh architecture, providing the infrastructure for decentralized architecture

Migration to Microsoft Fabric

Migrate Notebooks from Azure Synapse to Fabric

- [Migrate notebooks - Microsoft Fabric | Microsoft Learn](#)

Migrate Synapse notebooks to Fabric

Option 1 : Export from Synapse and Import to Fabric

Option 2 : Script to export notebooks from Synapse and import them to fabric using API

Migrate Synapse notebooks to Fabric using API

- Fabric workspace and Lakehouse
- Synapse workspace
- Service principal
- Access to service principal on Synapse workspace

Migrate Synapse/Data Factory pipelines to Fabric

Migrate ADLS data to Fabric Onelake

- **Option 1:** ADLS Gen2 as storage (shortcuts)
- **Option 2:** OneLake as storage
 - **mssparkutils fastcp**
 - **AzCopy**
 - **Azure Data Factory, Azure Synapse, and Data Factory in Fabric**
 - **Azure Storage Explorer**

Capacity Metrics App

Capacity Metrics App

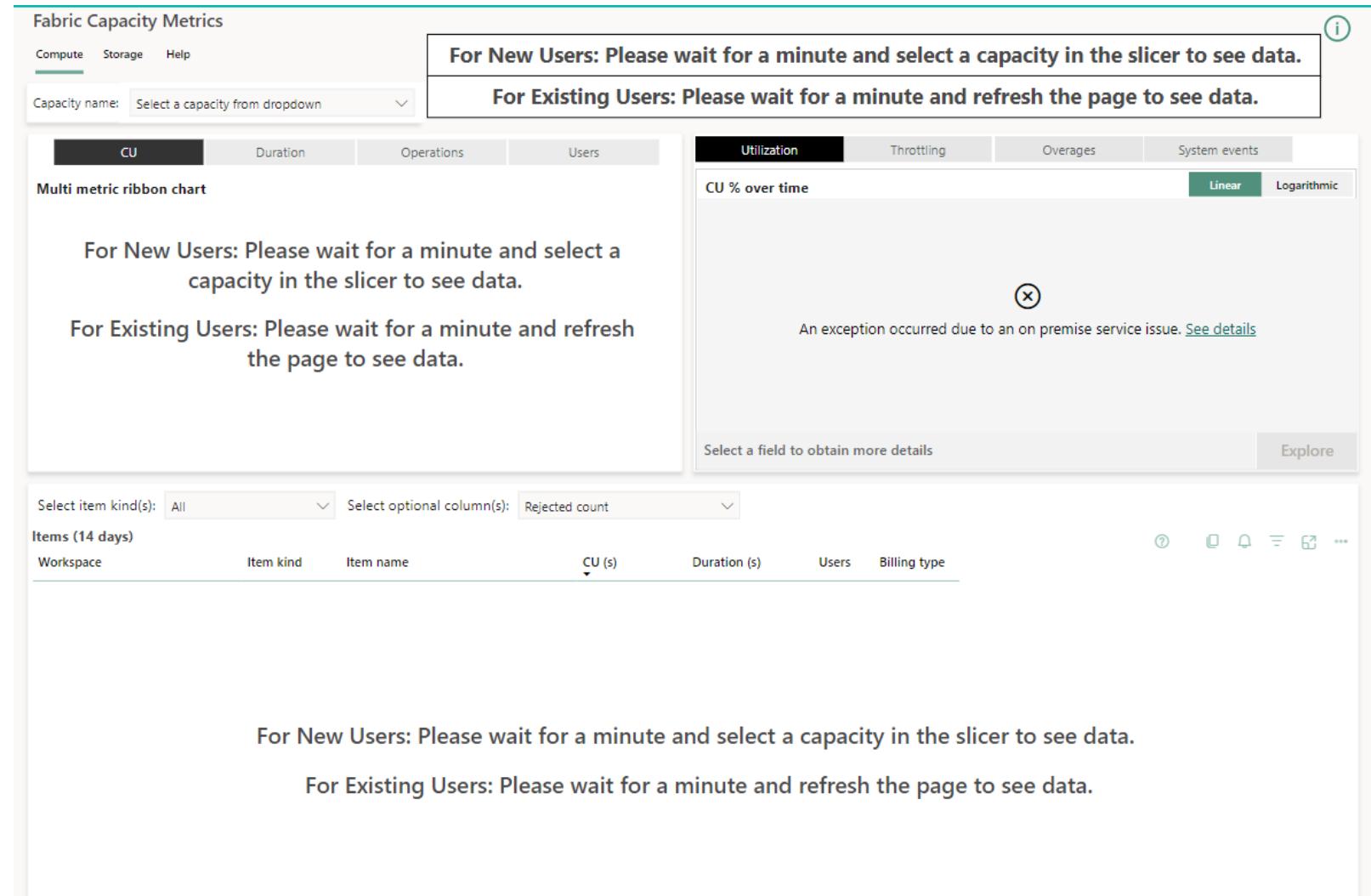
- The Microsoft Fabric Capacity Metrics app is designed to provide monitoring capabilities for Microsoft Fabric capacities
- Each capacity has its own number of Capacity Units (CU).
- CUs are used to measure the compute power available for your capacity.

Install Capacity Metrics App

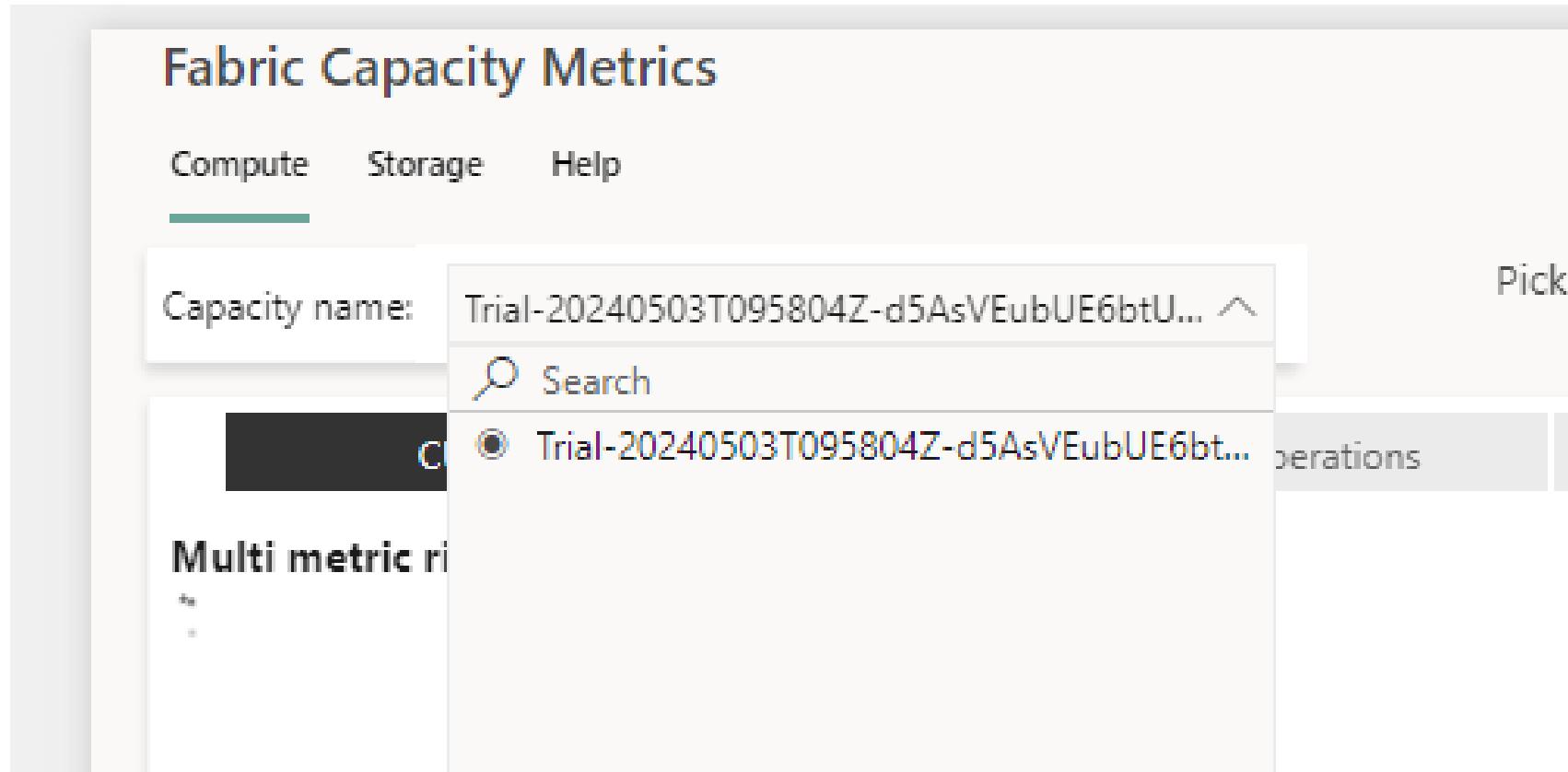
- You must be a capacity admin to install and view the Microsoft Fabric Capacity Metrics app.
- Install Capacity metrics app [Install the Microsoft Fabric capacity metrics app - Microsoft Fabric | Microsoft Learn](#)

Capacity Metrics App

- It's divided into the three visuals listed in the following sections.
- The top two visuals include
 - a ribbon chart
 - a line and stacked column chart, and
 - the bottom visual is a matrix table.

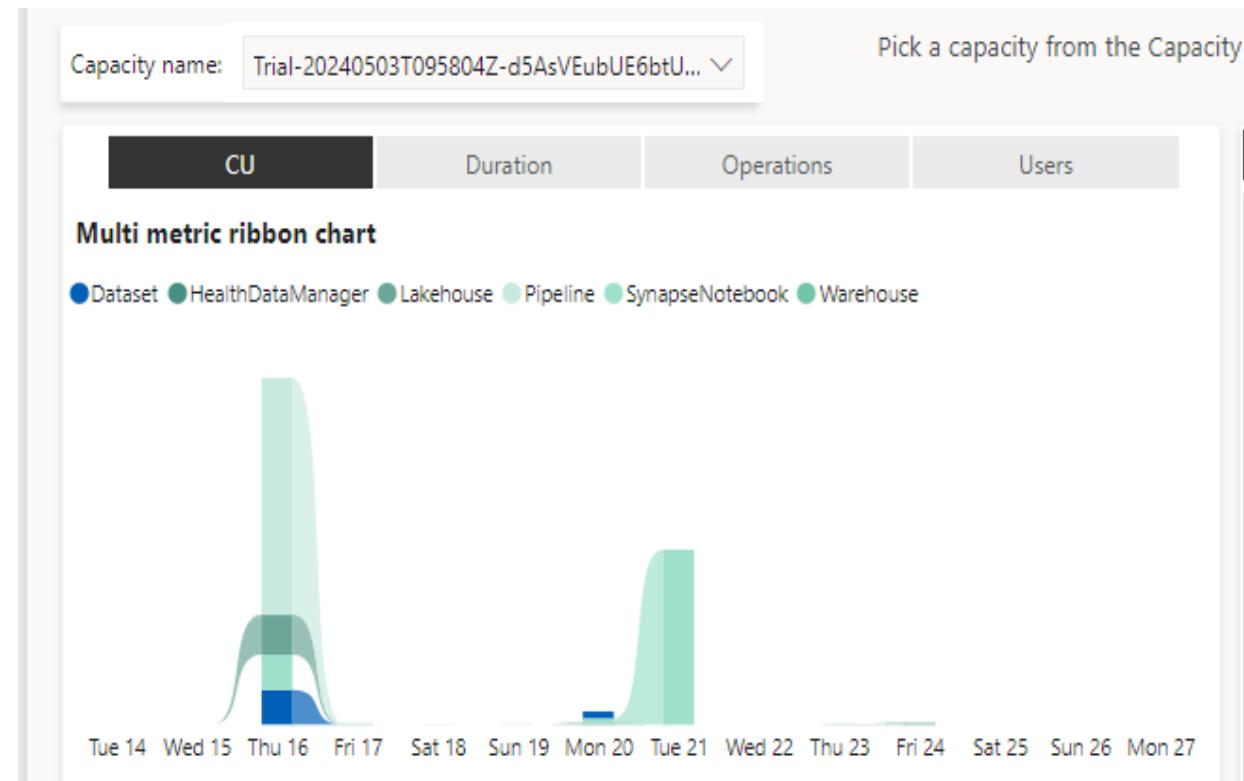


- To view metrics select the capacity name



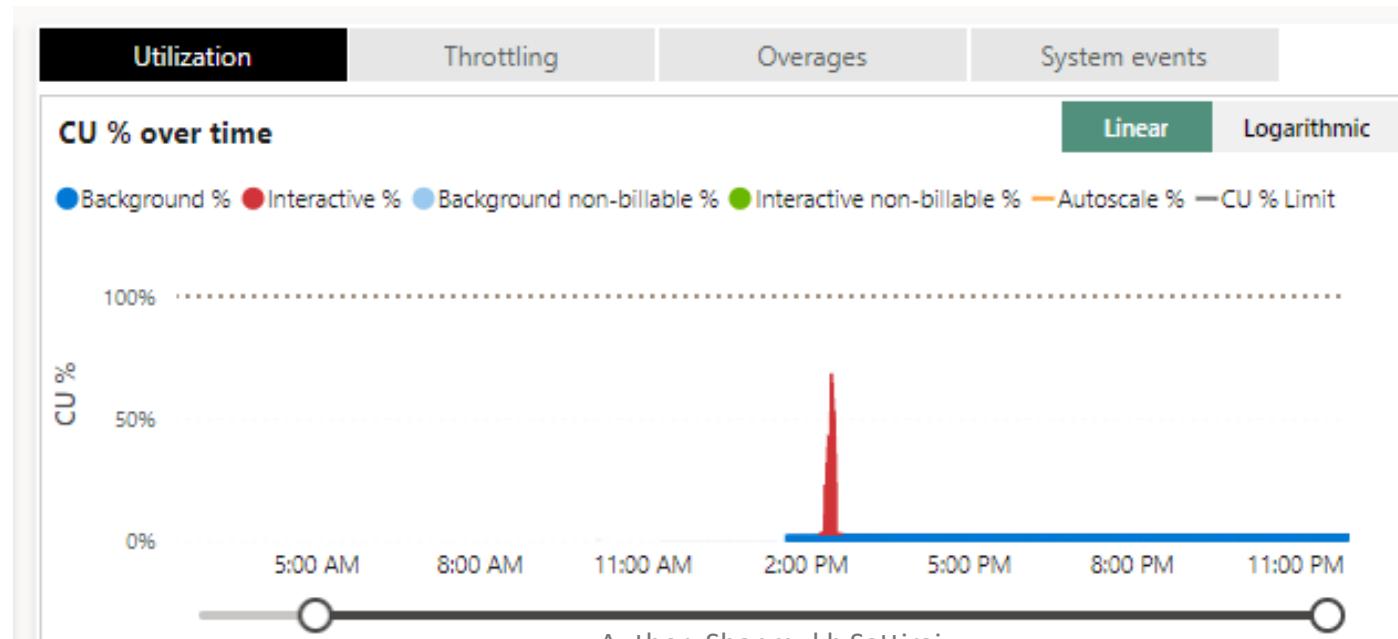
Multi metric ribbon chart

- The multi metric ribbon chart provides an hourly view of your capacity's usage. To identify daily patterns, drill down to a specific day
- The multi metric column ribbon displays the following four values. You'll see the top results for these values per item during the past two weeks.
- **CU** - Capacity Units (CU) processing time in seconds.
- **Duration** - Processing time in seconds.
- **Operations** - The number of operations that took place.
- **Users** - The number of users that performed operations.

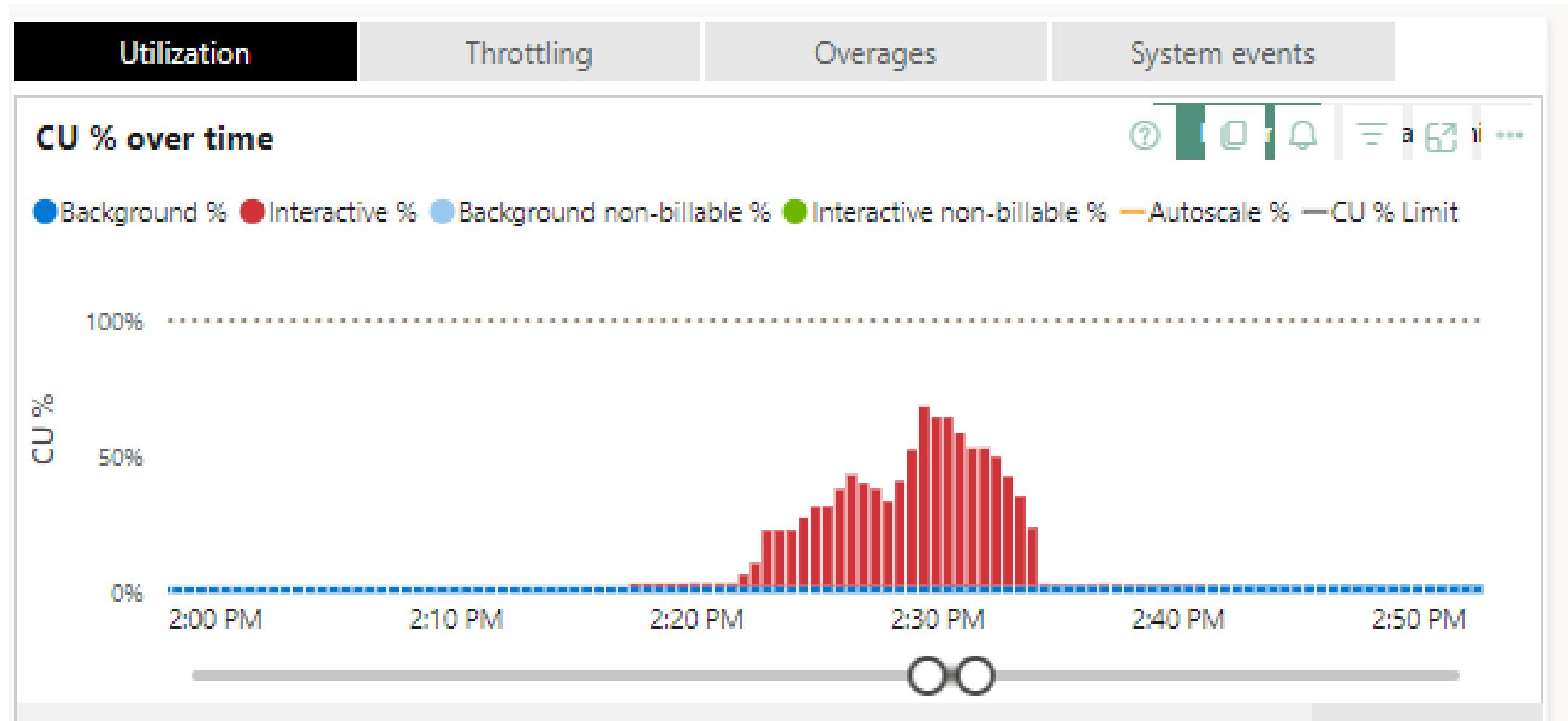


Multi metric ribbon chart

- Use the tabs at the top right corner of the visual to toggle how the visual is displayed.
- **Linear** - Display the information using a linear scale that starts at 0 percent.
- **Logarithmic** - Display the information using a logarithmic scale that depends on your CU consumption.

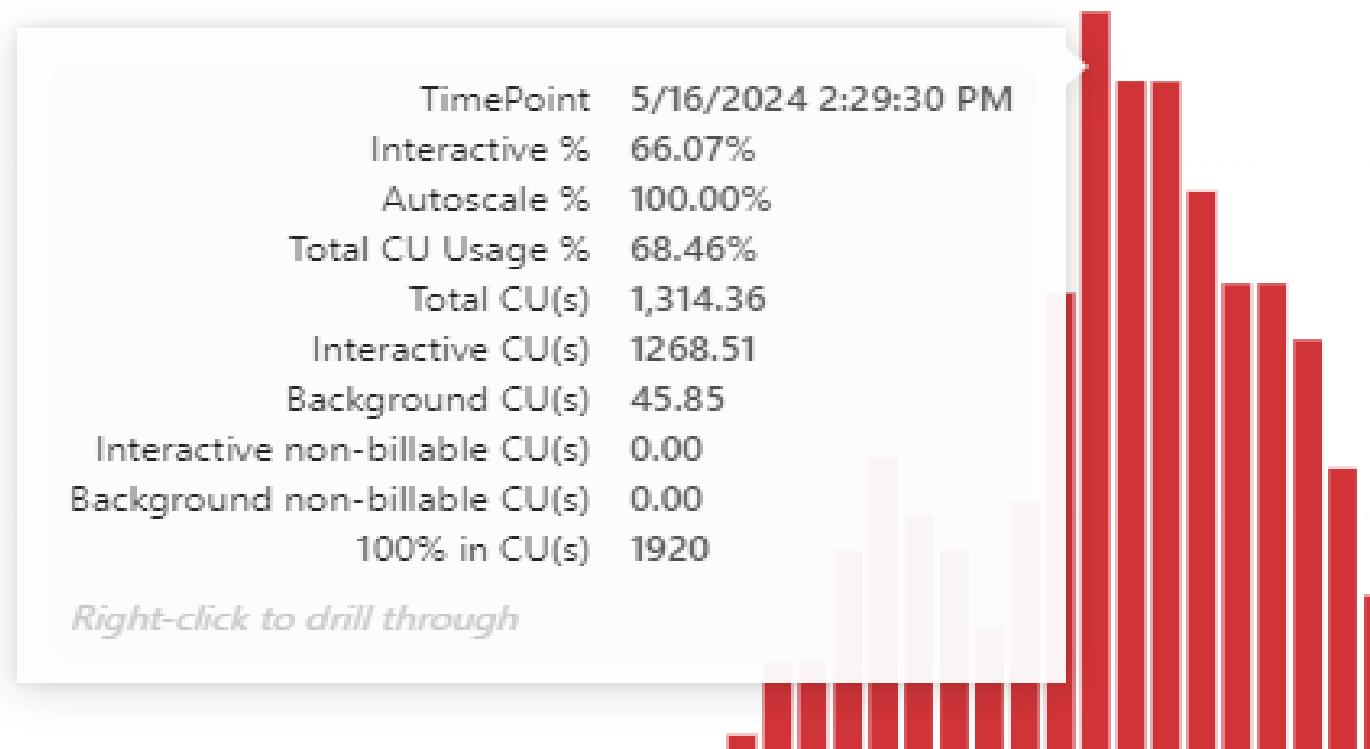


Line and stack column chart

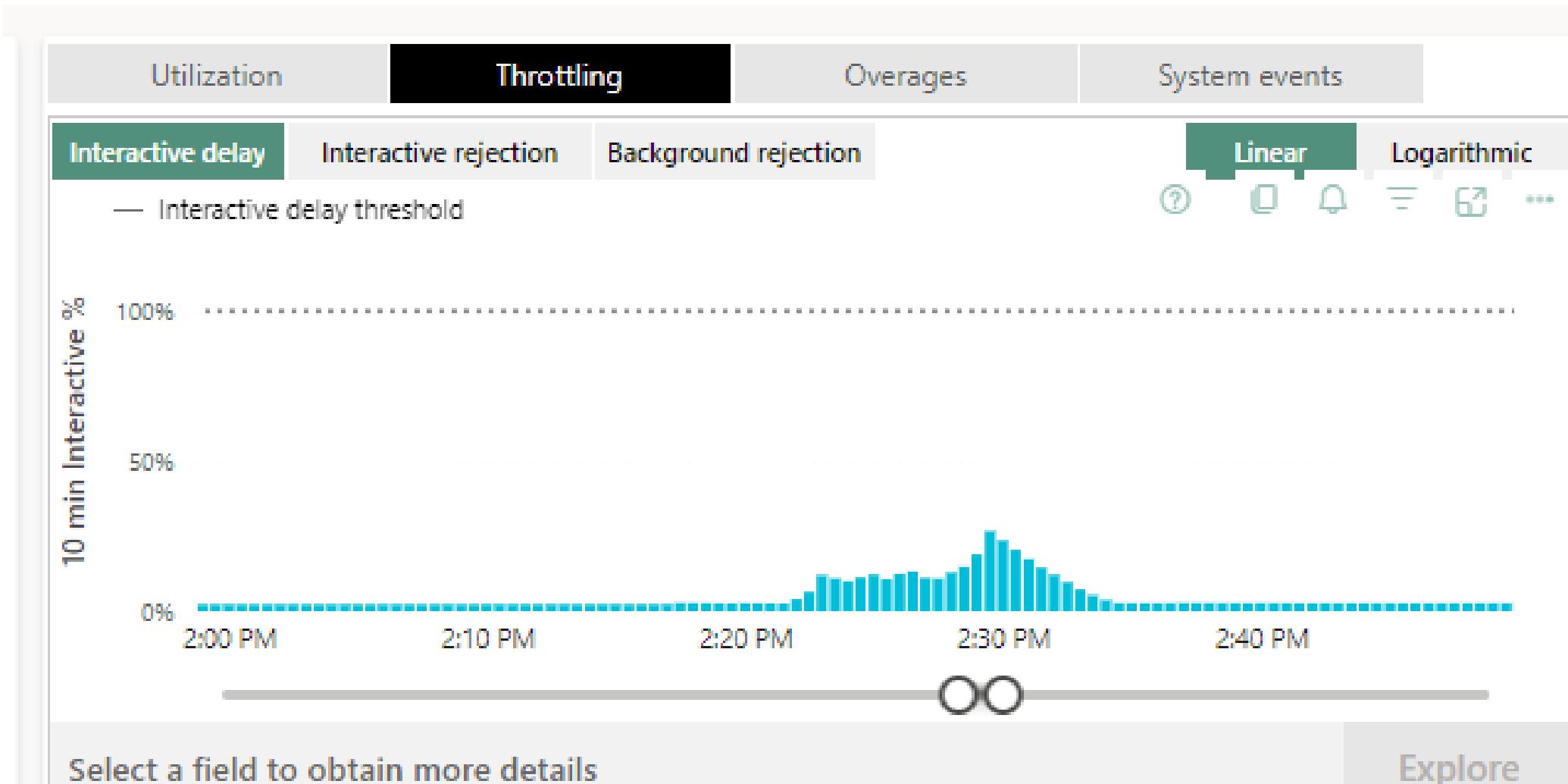


Line and stack column chart

- Understanding CU (s) Capacity Units in Seconds

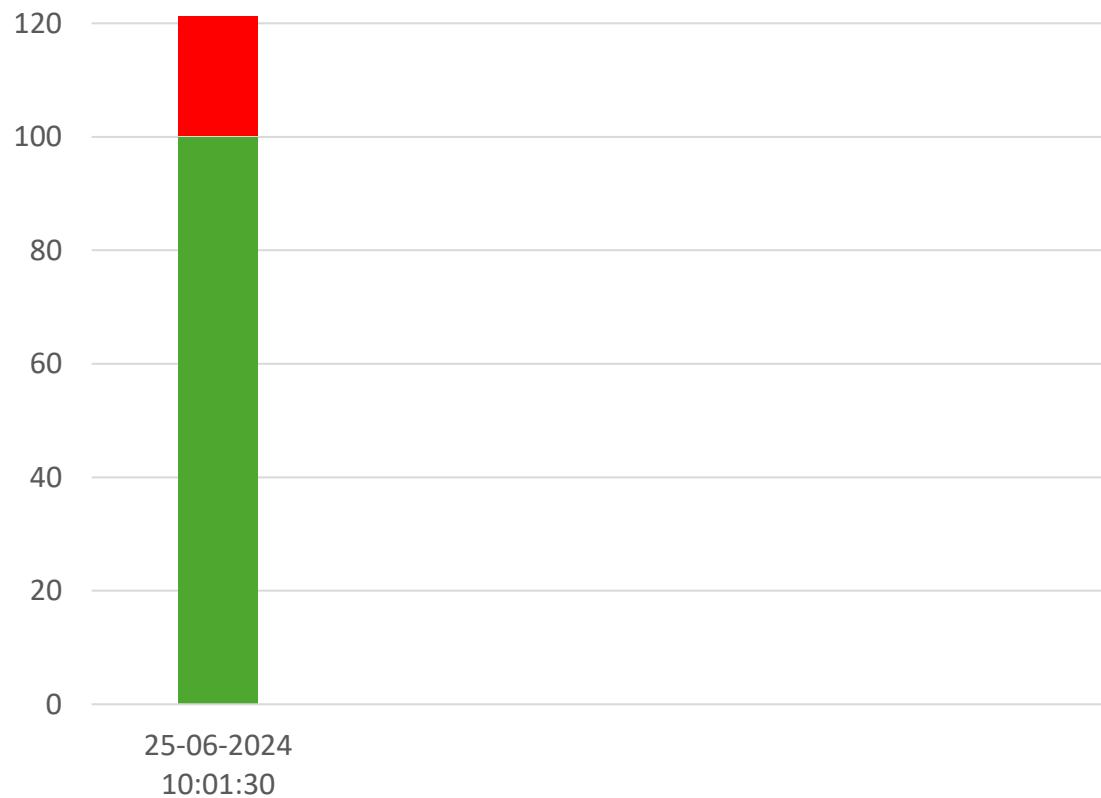


Throttling

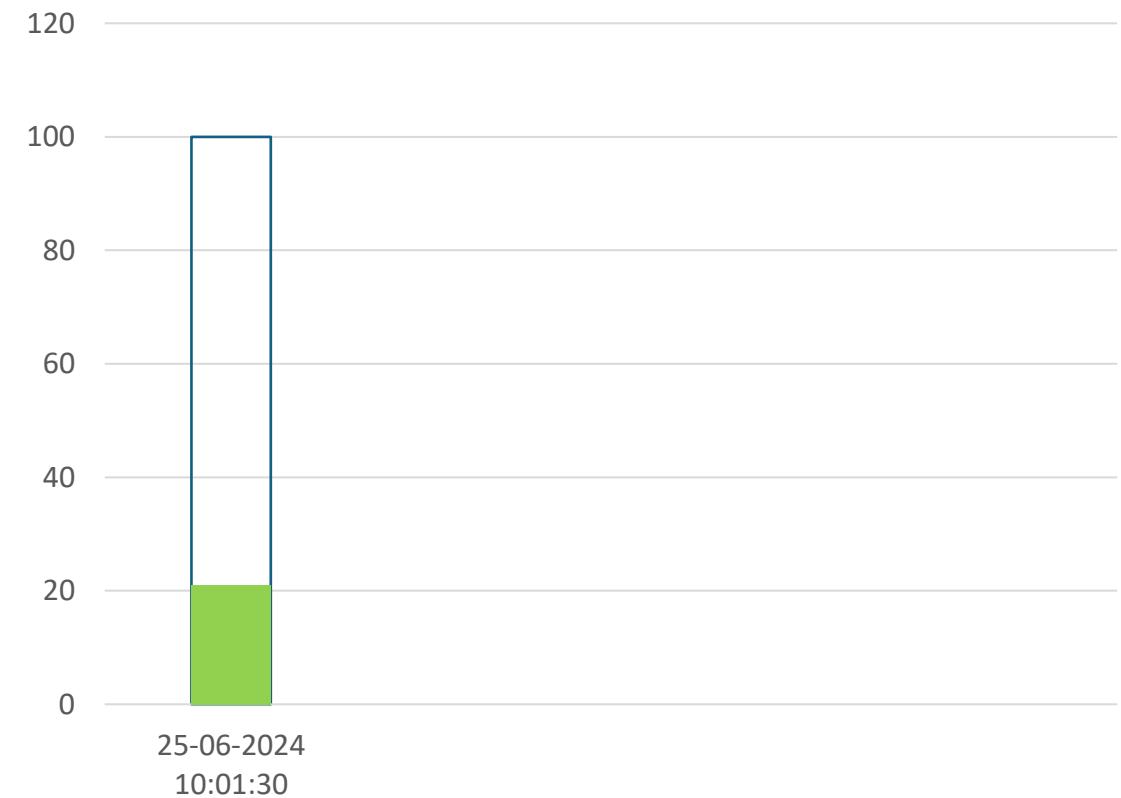


Smoothing

Over Utilized



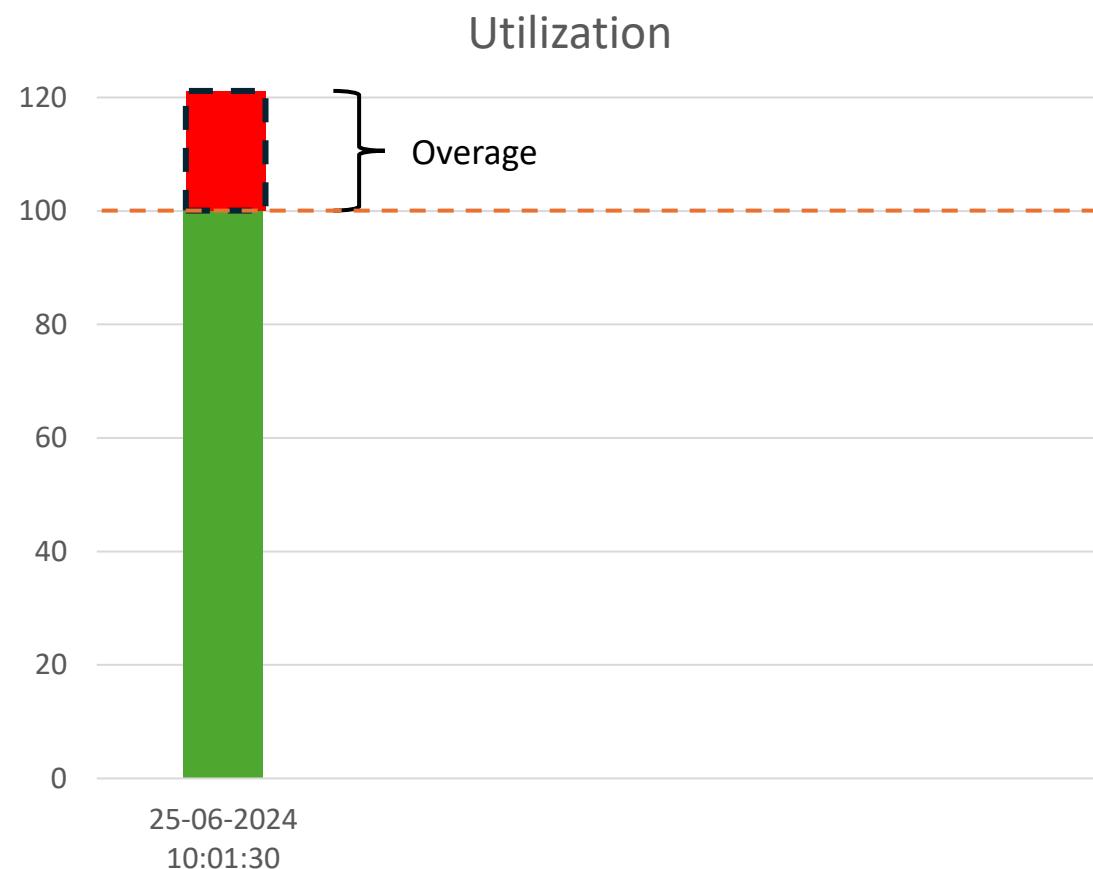
Under Utilized



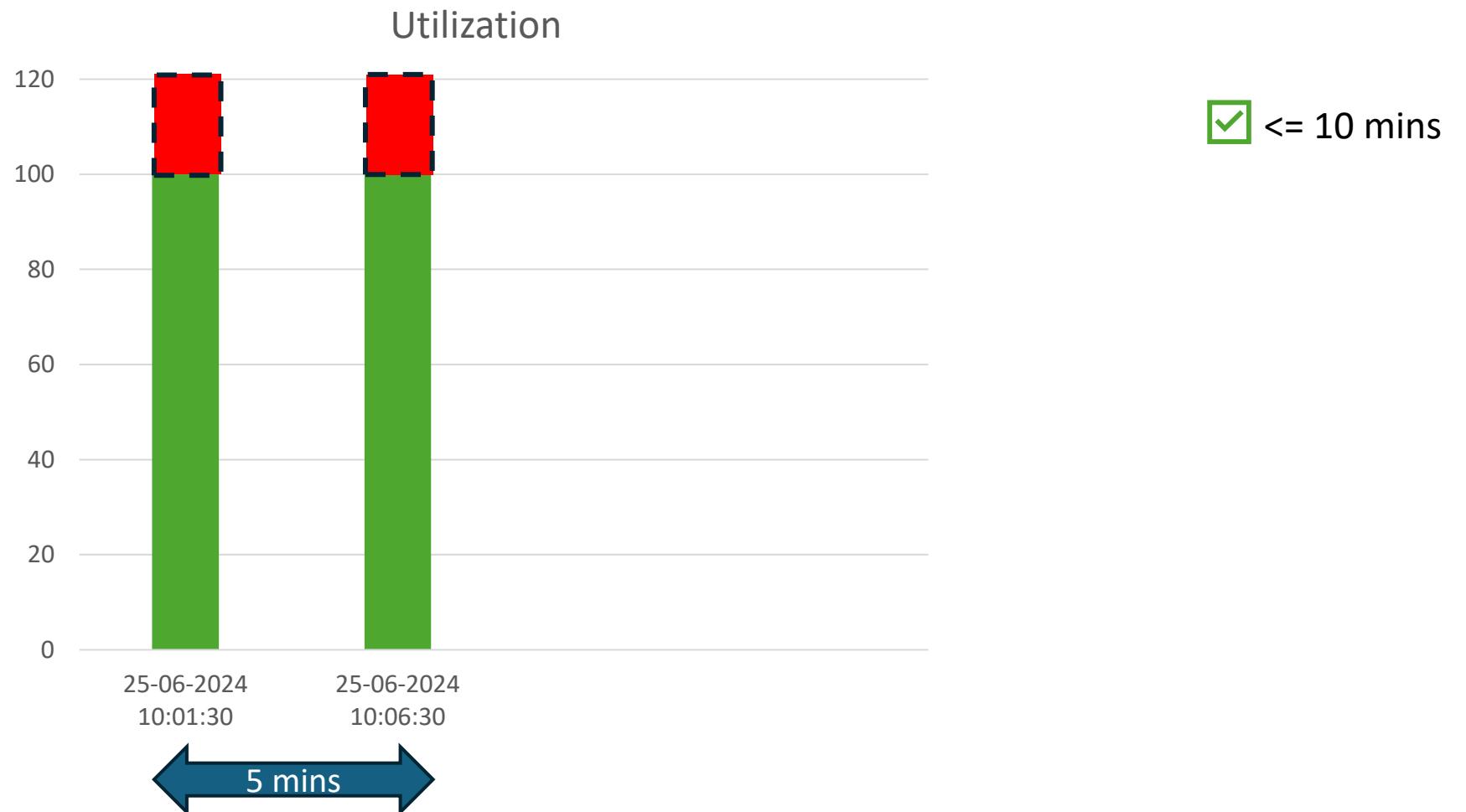
Throttling stages

Policy	Consumption	Impact
Overage protection	Usage \leq 10 minutes	Jobs can consume 10 minutes of future capacity use without throttling.
Interactive delay	10 minutes $<$ usage \leq 60 minutes	User-requested interactive jobs are delayed 20 seconds at submission.
Interactive rejection	60 minutes $<$ usage \leq 24 hours	User requested interactive jobs are rejected.
Background rejection	Usage $>$ 24 hours	User scheduled background jobs are rejected and not executed.

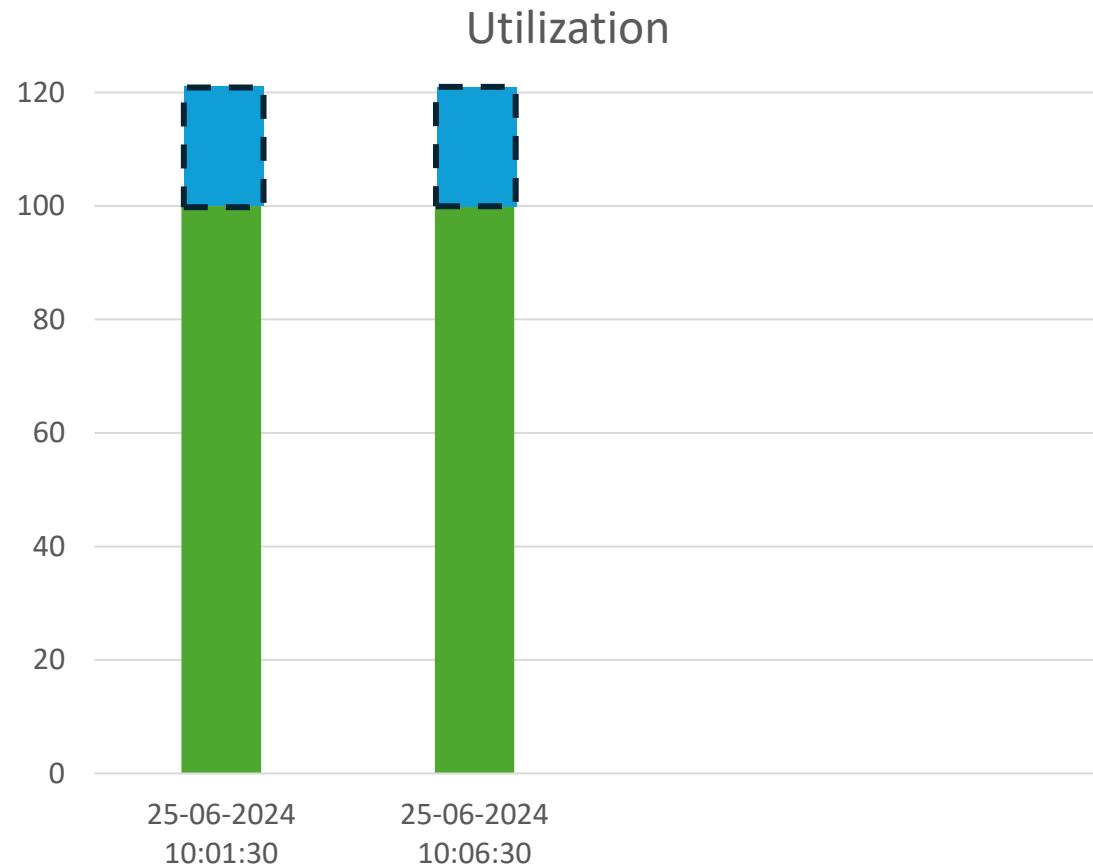
Overage protection



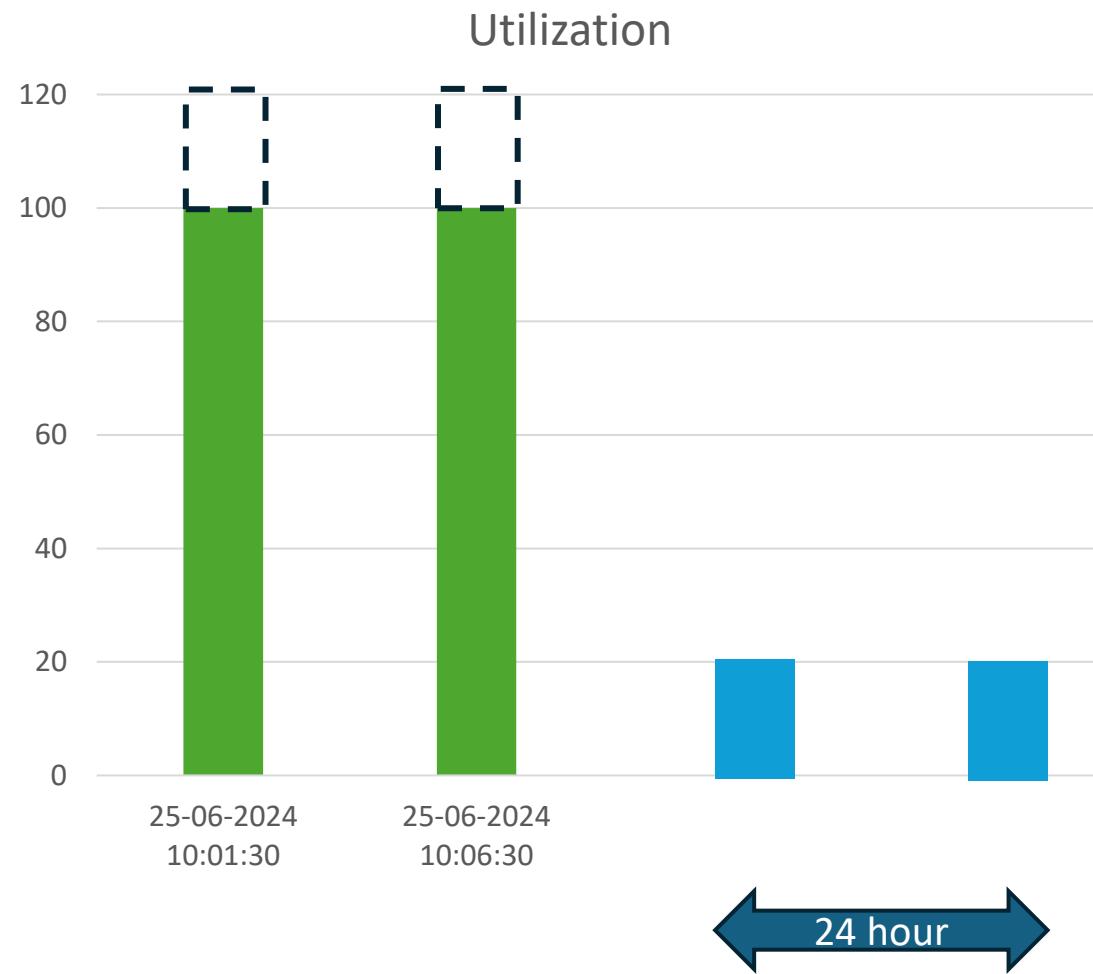
Overage protection



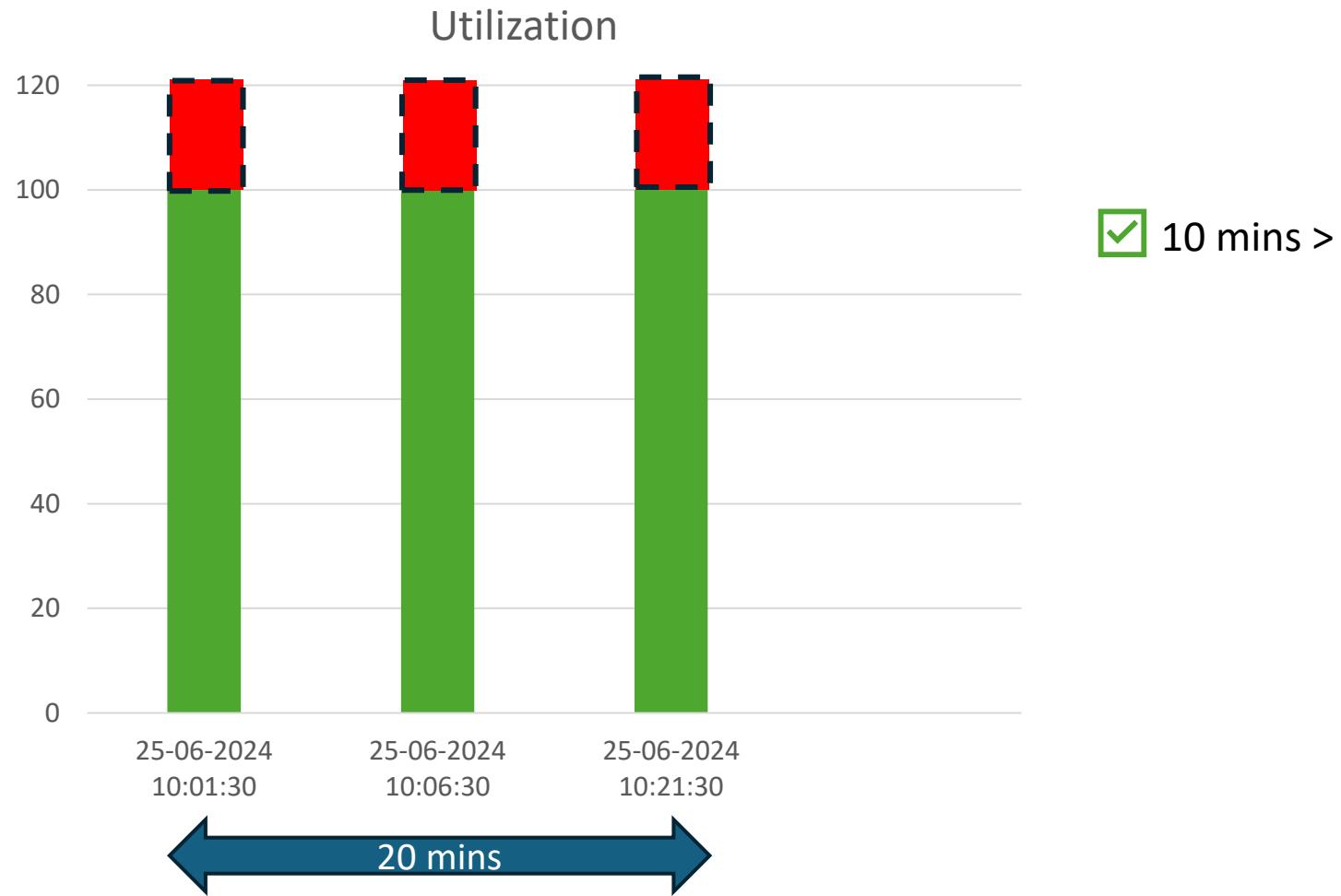
Overage protection



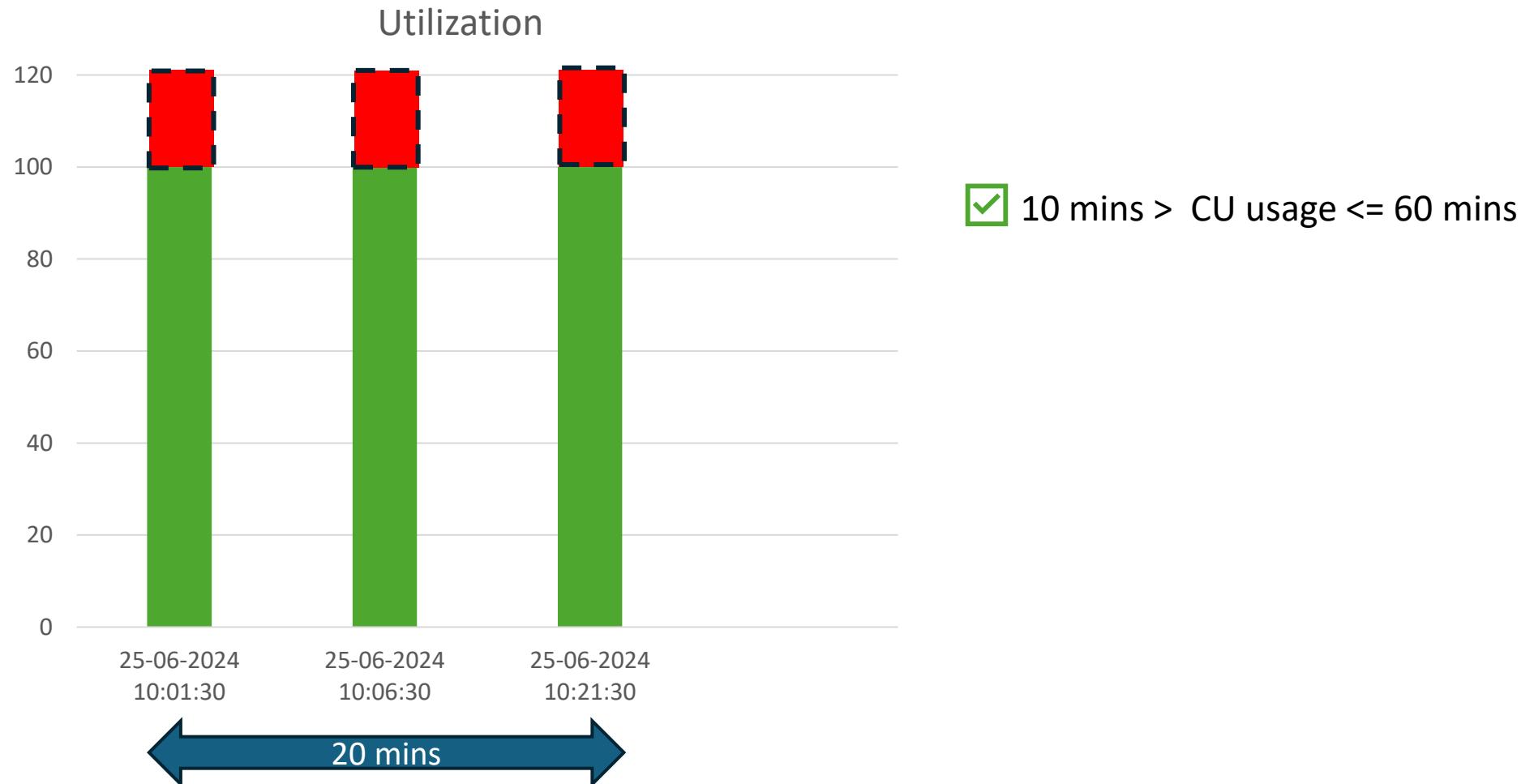
Overage protection



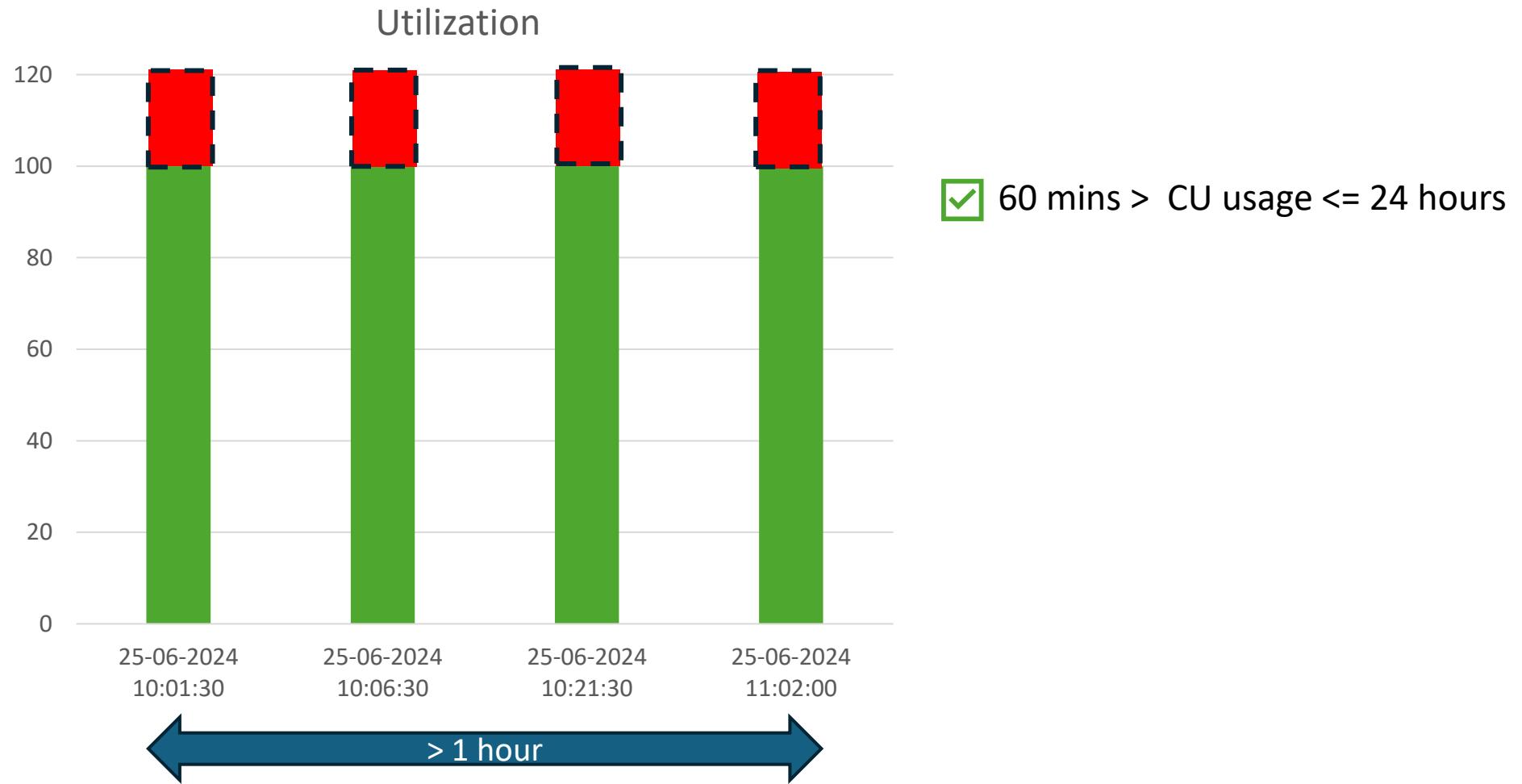
Interactive delay



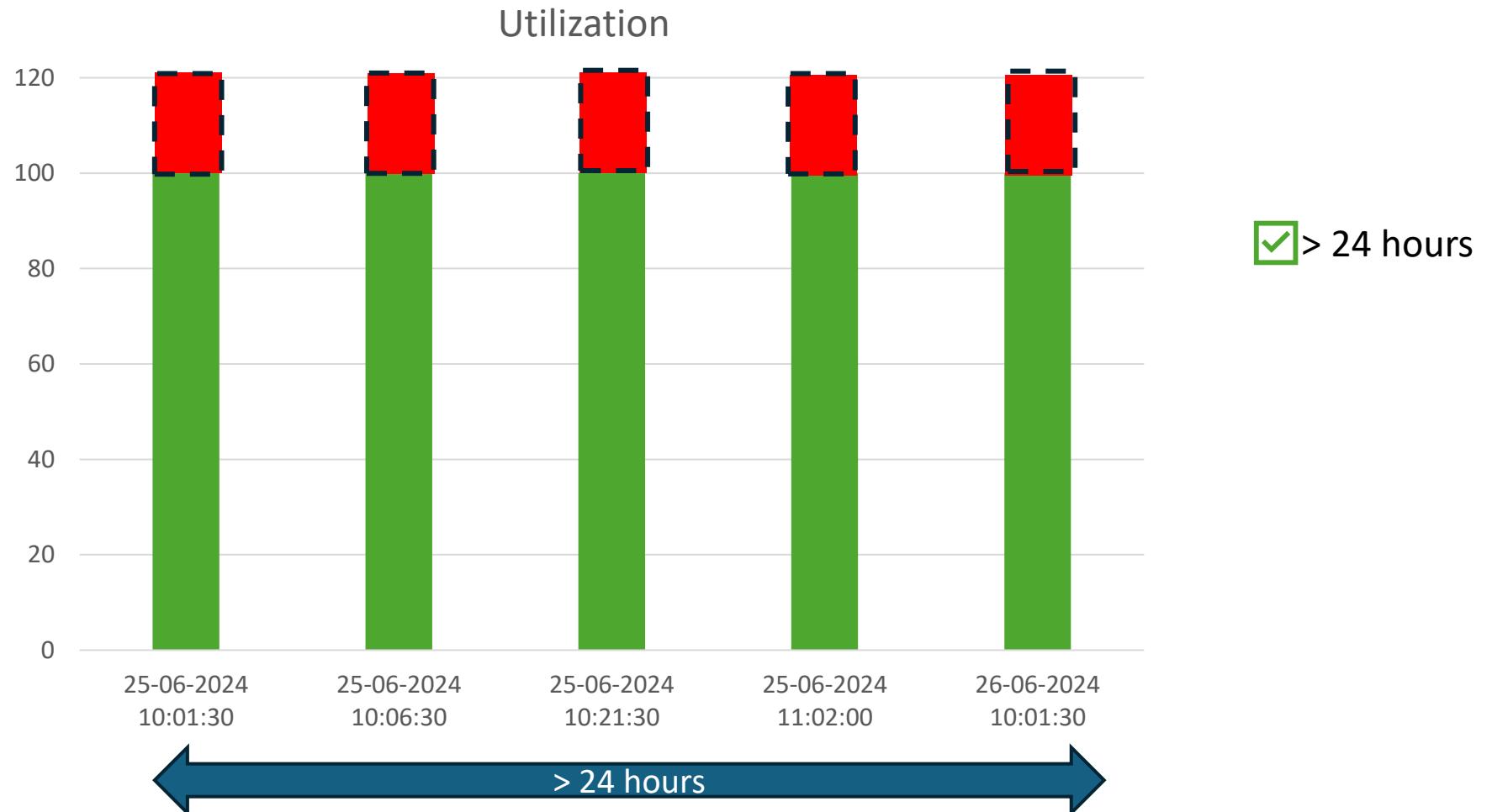
Interactive delay



Interactive rejection

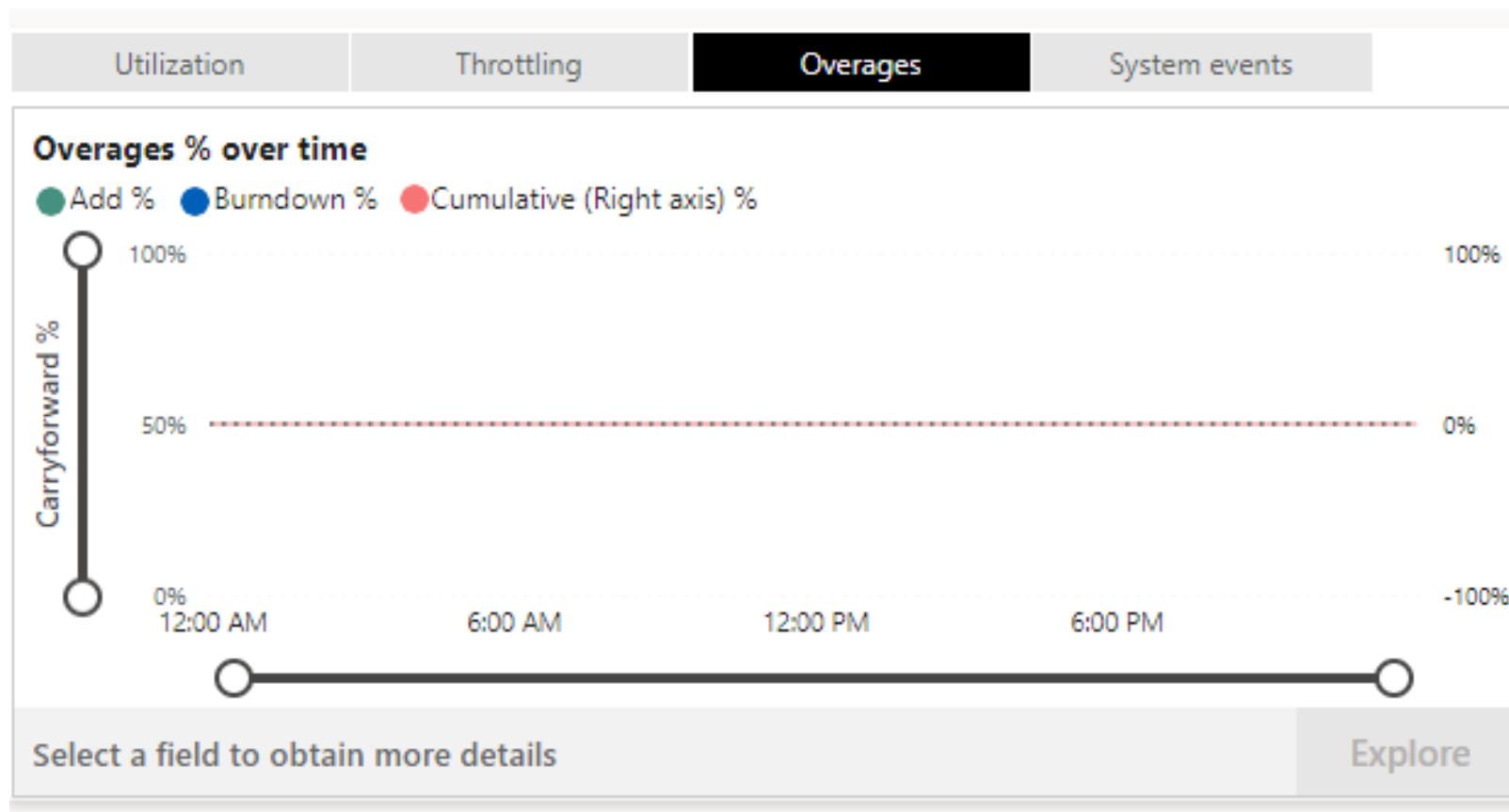


background rejection

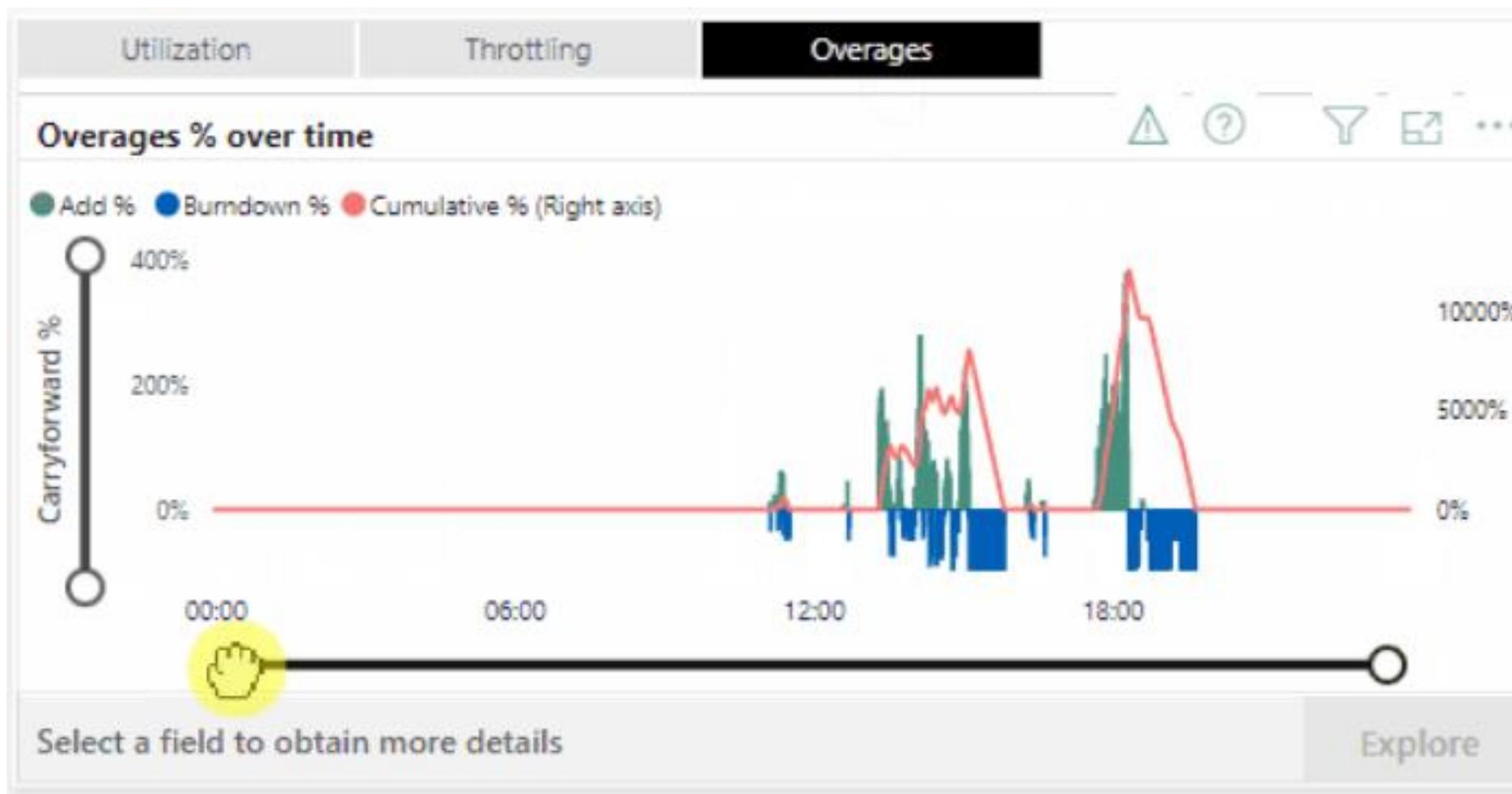


Policy	Consumption	Impact
Overage protection	Usage \leq 10 minutes	Jobs can consume 10 minutes of future capacity use without throttling.
Interactive delay	10 minutes < usage \leq 60 minutes	User-requested interactive jobs are delayed 20 seconds at submission.
Interactive rejection	60 minutes < usage \leq 24 hours	User requested interactive jobs are rejected.
Background rejection	Usage $>$ 24 hours	User scheduled background jobs are rejected and not executed.

Overages



Overages



System events

Utilization	Throttling	Overages	System events
System events			
State transition time	Capacity state	Capacity state change reason	
The state of capacity has remained unchanged in last 14 days.			
Select a field to obtain more details			Explore

Matrix

Select item kind(s): All Select optional column(s): Multiple selections

Items (14 days)

Workspace	Item kind	Item name	CU (s)	Duration (s)	Users	Successful count	Performance delta	Overloaded minutes	Billing type
My workspace	Pipeline	pipeline1	123,120.0000	2,323.5030	1	1		0.0000	Billable
My workspace	SynapseNote...	Notebook 1	43,400.2590	7,393.3250	1	0		0.0000	Billable
My workspace	SynapseNote...	ConvertToDelta	43,263.3110	6,886.8810	1	0		0.0000	Billable
My workspace	Dataset	LH_Fabric	21,689.8560	769.8280	2	387		0.0000	Billable
My workspace	Lakehouse	LH_practise	20,542.8777	4,904.4630	2	3527		0.0000	Billable
My workspace	SynapseNote...	test 2	19,600.5525	3,158.2830	1	0		0.0000	Billable
My workspace	SynapseNote...	Notebook 2	5,086.0105	1,271.4940	1	0		0.0000	Billable
Total			212,210.4	12,406.0	4	12,406		0.0000	Billable

Fabric documentation : [Understand the metrics app compute page - Microsoft Fabric | Microsoft Learn](#)



Synapse Data Warehouse

Warehouse

- Stores Structured data
- Contains Databases, Schemas, Tables
- Interact with SQL
- Supports transactions, DDL, and DML queries

Create a warehouse

When a warehouse is created:

- Warehouse
- Semantic Model (default)

Name	Location	Type
WH_Fabric	Fabric_trail	Warehouse
WH_Fabric	Fabric_trail	Semantic model (default)

Warehouse

Home Share

Get data | New SQL query | New visual query | New report | New measure | Download SQL database project

This warehouse automatically adds new objects to its default Power BI semantic model. Don't want to sync? Go to settings to turn this off. [Learn more](#)

Explorer

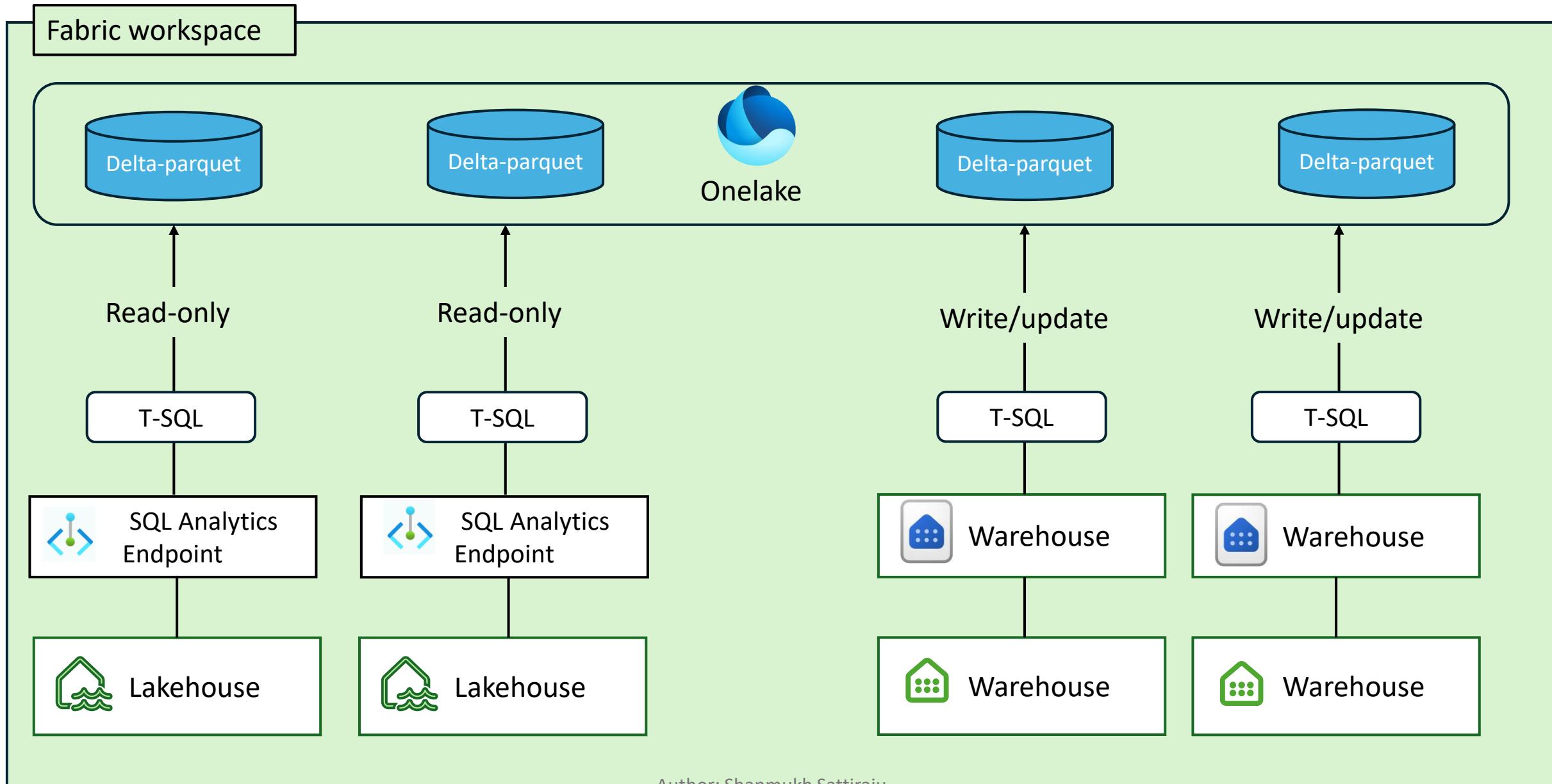
- + Warehouses
- WH_practise
 - Schemas
 - Security
- Queries
 - My queries
 - Shared queries

Build a warehouse

- T-SQL**
Create tables with T-SQL code templates
- New Dataflow Gen2**
Ingest and transform data with a low-code interface
- New data pipeline**
Ingest data at scale and schedule data workflows
- SQL database project**
Publish a database schema from an external project
- Sample data**
Automatically load sample data in your warehouse

Data Query Model

Author: Shanmukh Sattiraju
<https://in.linkedin.com/in/shanmukh-sattiraju>

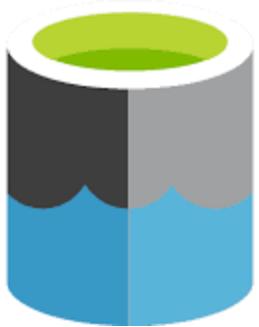


Loading data into warehouse

A Warehouse is populated by one of the supported data ingestion methods such as

1. COPY INTO statement
2. Pipelines
3. Dataflow Gen2
4. Cross database ingestion options such as CREATE TABLE AS SELECT (CTAS), INSERT..SELECT, or SELECT INTO.

Using COPY INTO command



Azure Datalake Gen2



Warehouse

Using COPY INTO command

- COPY performs high-throughput data ingestion from an external Azure storage account
- In Microsoft Fabric, currently supports the PARQUET and CSV file formats.
- For data sources, only Azure Data Lake Storage Gen2 accounts are supported
- User's Entra ID account must have access to the underlying files through [Azure role-based access control \(RBAC\)](#) or [data lake ACLs](#).
- Trusted workspace access using Fabric is in preview supported for F64 or above capacities (not trail)

Using Pipelines

Ingest data from On-premise SQL to Warehouse in Fabric

1. You need to have a On-prem gateway
2. Create a connection to SQL server
3. Pipeline
4. Staging area
 - Should be Blob or ADLS storage
 - Authenticated only via SAS or Account key (Oauth not supported)
 - Check “allow this connection to be used with On-prem data gateways”

Using DataFlow Gen2

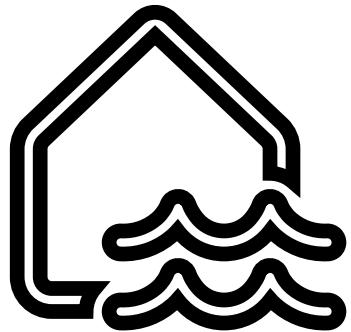


On-prem SQL Server

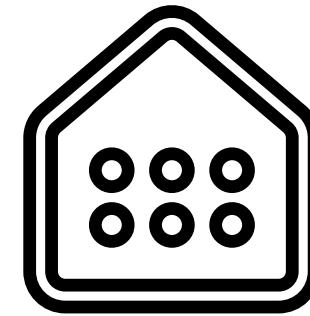


Warehouse

Data sharing - Lakehouse & Warehouse



Lakehouse



Warehouse

Cross database ingestion options

Get data using Lakehouse tables

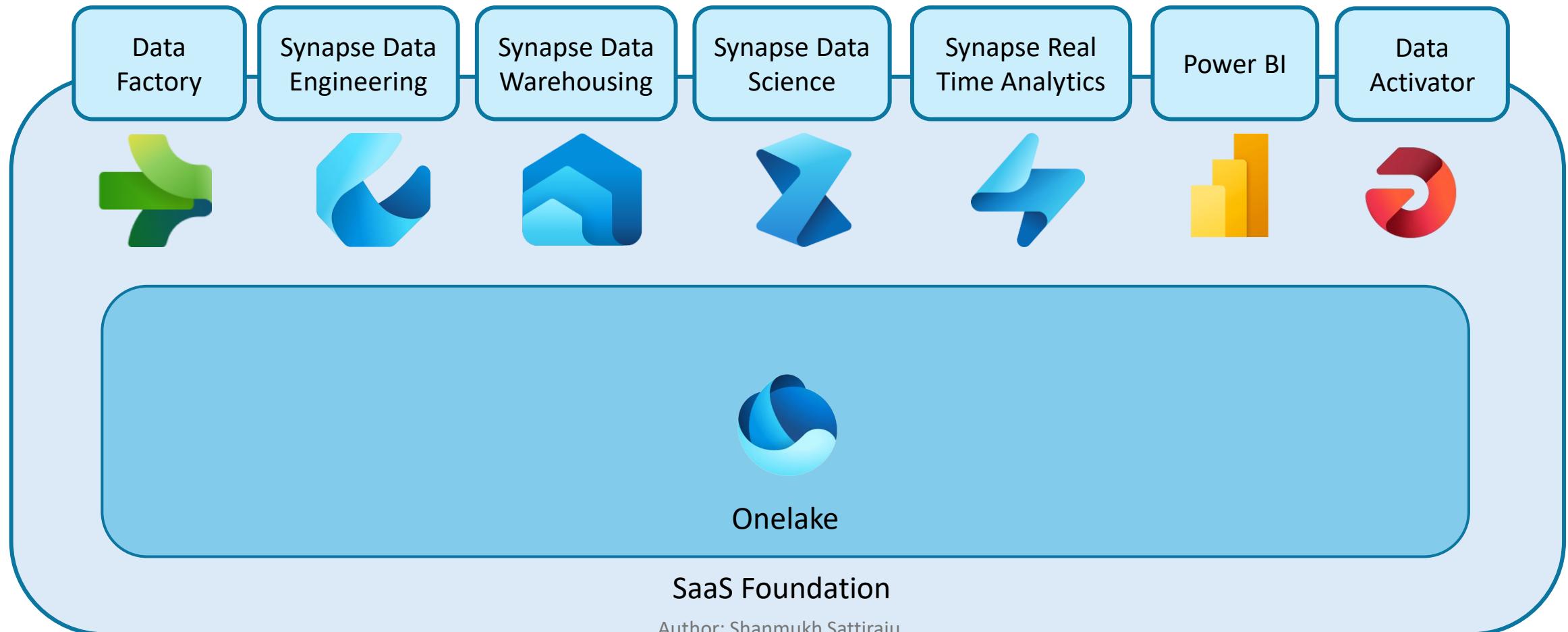
1. Using CTAS (Create Table As Select)
2. Using INSERT INTO ... SELECT *
 - Need to have a table created
3. Using SELECT INTO

Loading data into warehouse

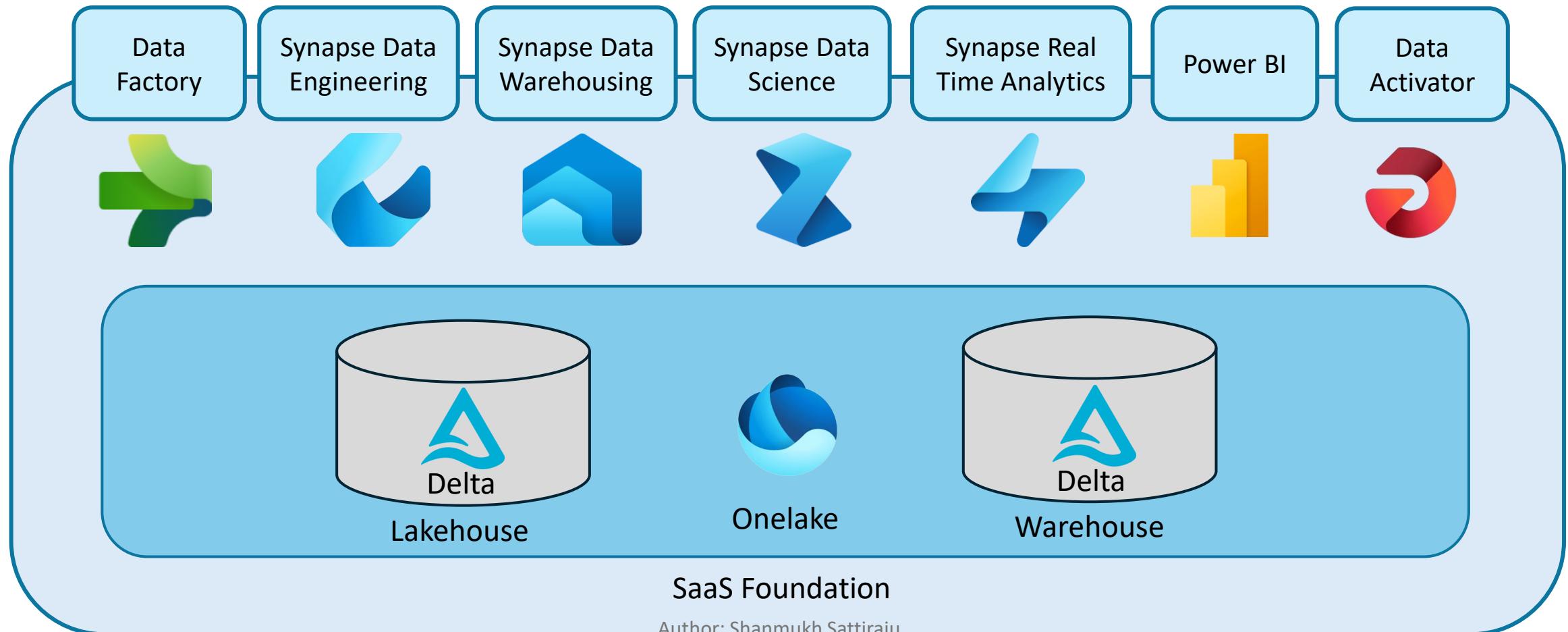
A Warehouse is populated by one of the supported data ingestion methods such as

1. COPY INTO statement ✓
2. Pipelines ✓
3. Dataflow Gen2 ✓
4. Cross database ingestion options such as CREATE TABLE AS SELECT (CTAS), INSERT..SELECT, or SELECT INTO. ✓

Microsoft Fabric Architecture



Lakehouse vs Warehouse



Author: Shanmukh Sattiraju

<https://in.linkedin.com/in/shanmukh-sattiraju>



Lakehouse

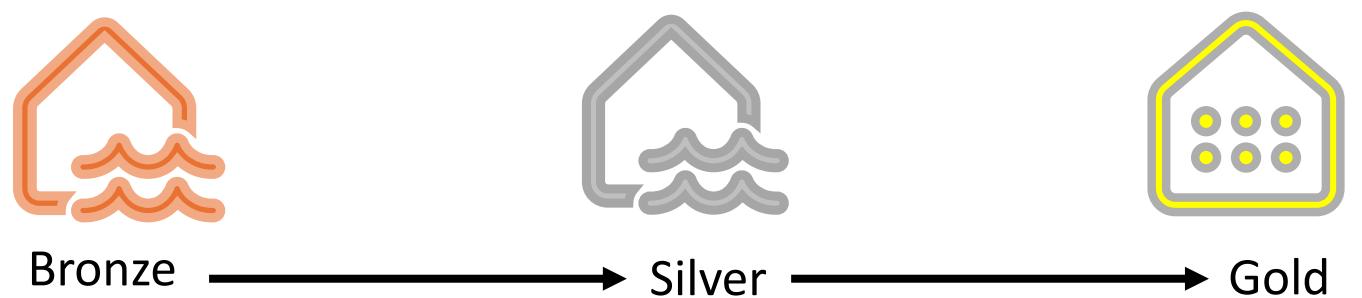
- Structured, semi-structured and unstructured
- Spark Notebooks
- PySpark, Scala, Spark SQL, Spark R
- Write transformed data into Lakehouse tables or files
- Data Engineers, Data scientists



Warehouse

- Structured
- SQL Queries, Stored Procs
- Full T-SQL support
- Write transformed data into Warehouse tables
- SQL Developer, Data Analysts

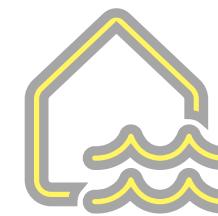
Medallion architecture patterns



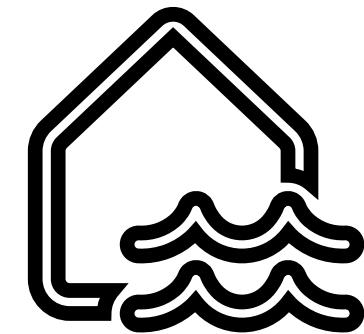
Lakehouse



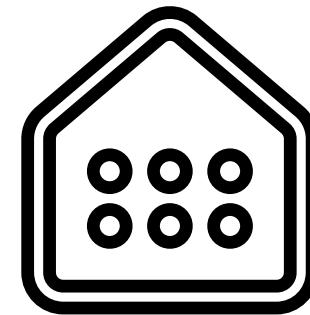
Warehouse



Update data from Lakehouse and Warehouse



Lakehouse

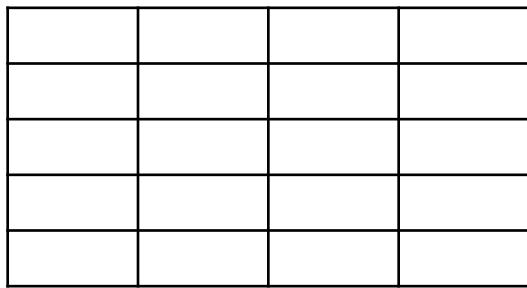


Warehouse

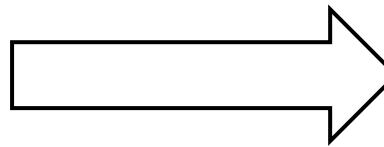
SQL query as session

- In fabric SQL editor , each statement acts as individual session
- But if you are using the same thing in SSMS it is different

Zero-Copy clone feature



Data + metadata



Metadata

Zero-Copy clone feature

- A zero-copy clone creates a replica of the table by copying the metadata , while referencing the same data files in OneLake
- The metadata is copied while the underlying data of the table stored as parquet files is not copied.
- A clone of a table can be created within or across schemas in a warehouse.
- A table clone is an independent and separate copy of the data from its source.

Creating a table clone

A Clone of a table can be created based on either:

- **Current point-in-time:**
 - The clone is based on the present state of the table.
- **Previous point-in-time**
 - The clone is based on a point-in-time up to seven days in the past.
 - This feature is known as "time travel"
 - The new table is created with a timestamp based on UTC

Syntax:

CREATE TABLE

```
{ database_name.schema_name.table_name  
(or)  
schema_name.table_name (or) table_name }
```

AS CLONE OF

```
{ database_name.schema_name.table_name |  
(or)  
schema_name.table_name (or) table_name }  
[AT {point_in_time}] -- 'YYYY-MM-DDThh:mm:ss'
```

Time travel in Warehouse

- The ability to query a data from a specific timestamp of past
- Low-cost comparisons between previous versions of data.
- Audit data changes over time
- Default retention period of seven calendar days.
- Any modifications made to the schema of a table, including but not limited to adding or removing columns from the table, cannot be queried before the schema change
- Time travel is not supported for the SQL analytics endpoint of the Lakehouse.

Retention of data history

- Warehouse automatically preserves and maintains the data history for seven calendar days, allowing for clones to be made at a point in time.
- All inserts, updates, and deletes made to the data warehouse are retained for seven calendar days.
- There is no limit on the number of clones created both within and across schemas.

Benefits of Zero-Copy clone

- Development and testing
- Low-cost, near-instantaneous recovery
- Data archiving

Limitations

- Table clones across warehouses in a workspace are not currently supported.
- Table clones across workspaces are not currently supported.
- Clone table is not supported on the SQL analytics endpoint of the Lakehouse.
- Clone of a warehouse or schema is currently not supported.
- Table clones submitted before the retention period of seven days cannot be created.
- Changes to the table schema prevent a clone from being created before to the table schema change.

Visual query editor

- Applies to SQL analytics endpoint , warehouse and mirrored database in Fabric
- You can use the visual query editor for a no-code experience to create your queries.

Limitations of Visual query editor

- In the visual query editor, you can only run DQL (Data Query Language) or read-only SELECT statements. DDL or DML statements are not supported.
- Only a subset of Power Query operations that support Query folding are currently supported.

Query Insights

- The query insights feature provides a central location for historic query data and actionable insights for 30 days, helping you to make informed decisions to enhance the performance of your Warehouse or SQL analytics endpoint.
- Available in SQL analytics endpoint and Warehouse in Microsoft Fabric
- Provides information on queries run in a user's context only, system queries aren't considered.

Type of Query insights:

- **Historical Query Data:** The query insights feature stores historical data about query executions, enabling you to track performance changes over time. System queries aren't stored in query insights.
- **Aggregated Insights:** The query insights feature aggregates query execution data into insights that are more actionable, such as identifying long-running queries or most active users.

exec_requests_history

Provides information about each complete SQL request.

Column name	Data type	Description
distributed_statement_id	uniqueid	Unique ID for each query.
start_time	datetime2	Time when the query started running.
command	varchar(8000)	Complete text of the executed query.
login_name	varchar(128)	Name of the user or system that sent the query.
row_count	bigint	Number of rows retrieved by the query.
total_elapsed_time_ms	int	Total time (ms) taken by the query to finish.
status	varchar(30)	Query status (Succeeded, Failed, or Canceled).
session_id	smallint	ID linking the query to a specific user session.
connection_id	uniqueid (nullable)	Identification number for the query's connection.
batch_id	uniqueid (nullable)	ID for grouped queries (if applicable).
root_batch_id	uniqueid (nullable)	Author: Sharmukh Sattiraju https://in.linkedin.com/in/sharmukhsattiraju ID for the main group of queries (if nested).

frequently_run_queries

Provides information about frequently run queries in Fabric Data Warehousing.

Column name	Data type	Description
last_run_start_time	datetime2	Time of the most recent query execution.
last_run_command	varchar(8000)	Text of the last query execution.
number_of_runs	int	Total number of times the query was executed.
avg_total_elapsed_time_ms	int	Average query execution time (ms) across all runs.
last_run_total_elapsed_time_ms	int	Time taken by the last execution (ms).
last_dist_statement_id	uniqueidentifier	ID linking the query to queryinsights.exec_requests_history.
last_run_session_id	smallint	User session ID for the last execution.
min_run_total_elapsed_time_ms	int	Shortest query execution time (ms).
max_run_total_elapsed_time_ms	int	Longest query execution time (ms).
number_of_successful_runs	int	Number of successful query executions.
number_of_failed_runs	int	Number of failed query executions.
number_of_cancelled_runs	int	Number of canceled query executions.

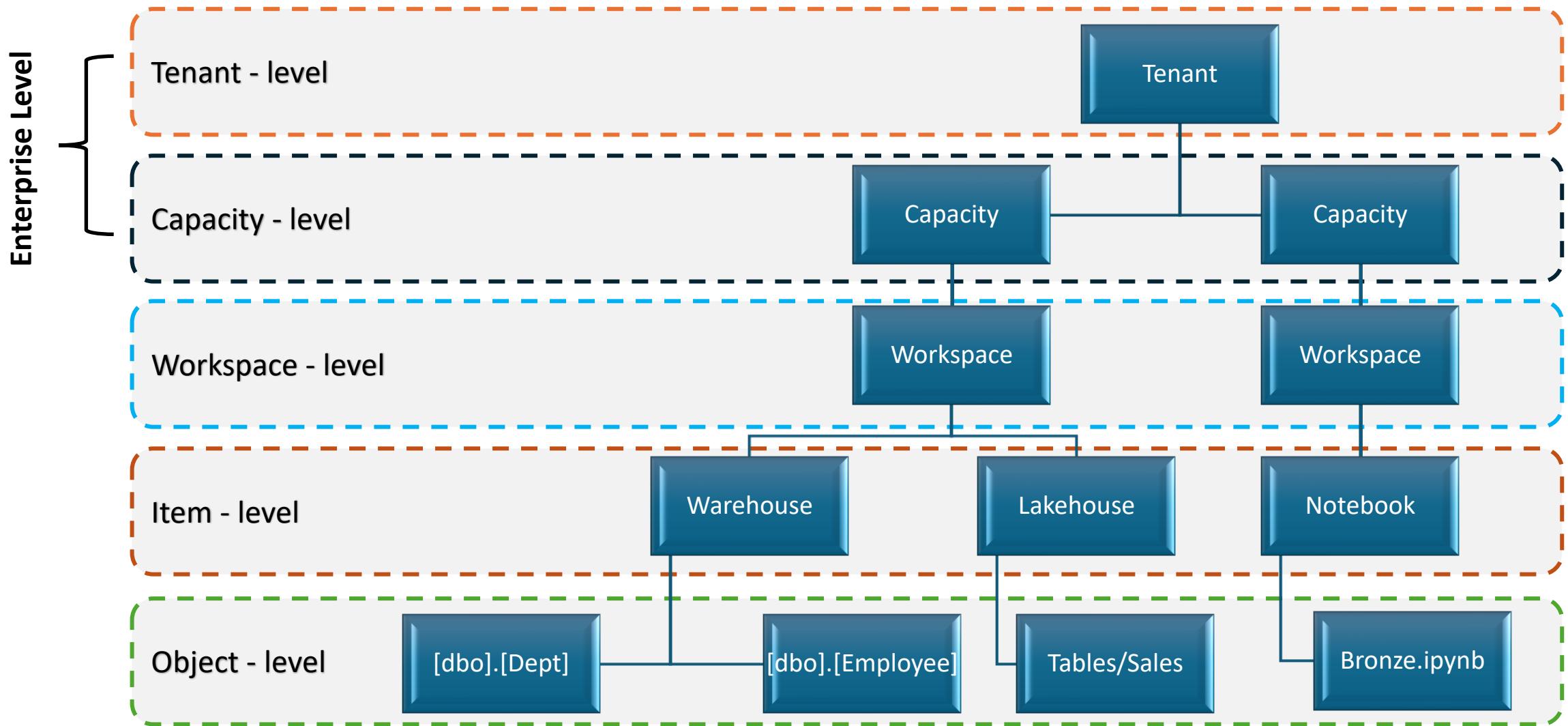
long_running_queries

provides information about SQL query execution times.

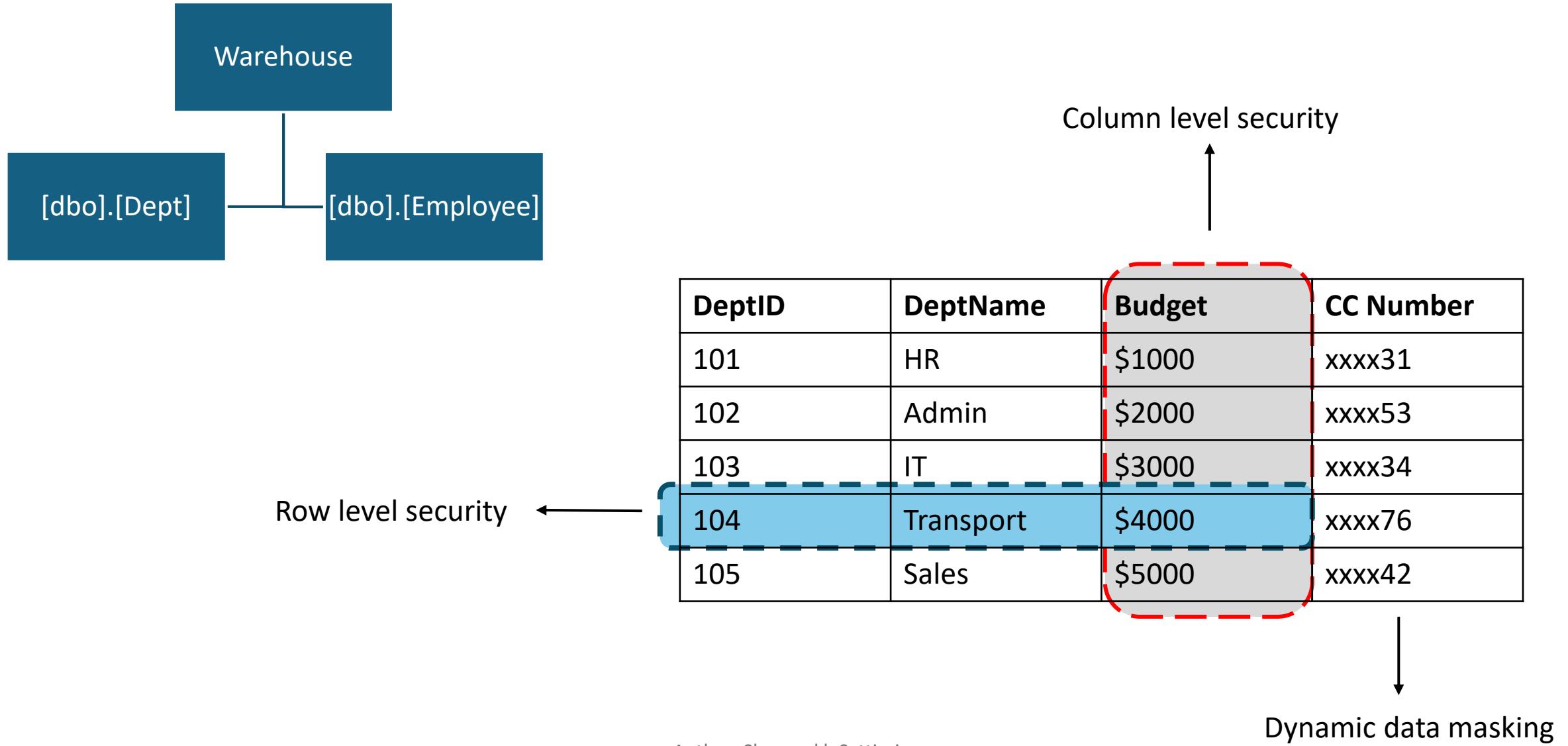
Column name	Data type	Description
last_run_start_time	datetime2	Time of the most recent query execution.
last_run_command	varchar(8000)	Text of the last query execution.
median_total_elapsed_time_ms	int	Median query execution time (ms) across runs.
number_of_runs	int	Total number of times the query was executed.
last_run_total_elapsed_time_ms	int	Time taken by the last execution (ms).
last_dist_statement_id	uniqueidentifier	ID linking the query to queryinsights.exec_requests_history.
last_run_session_id	smallint	User session ID for the last execution.

Access control and permissions in Fabric

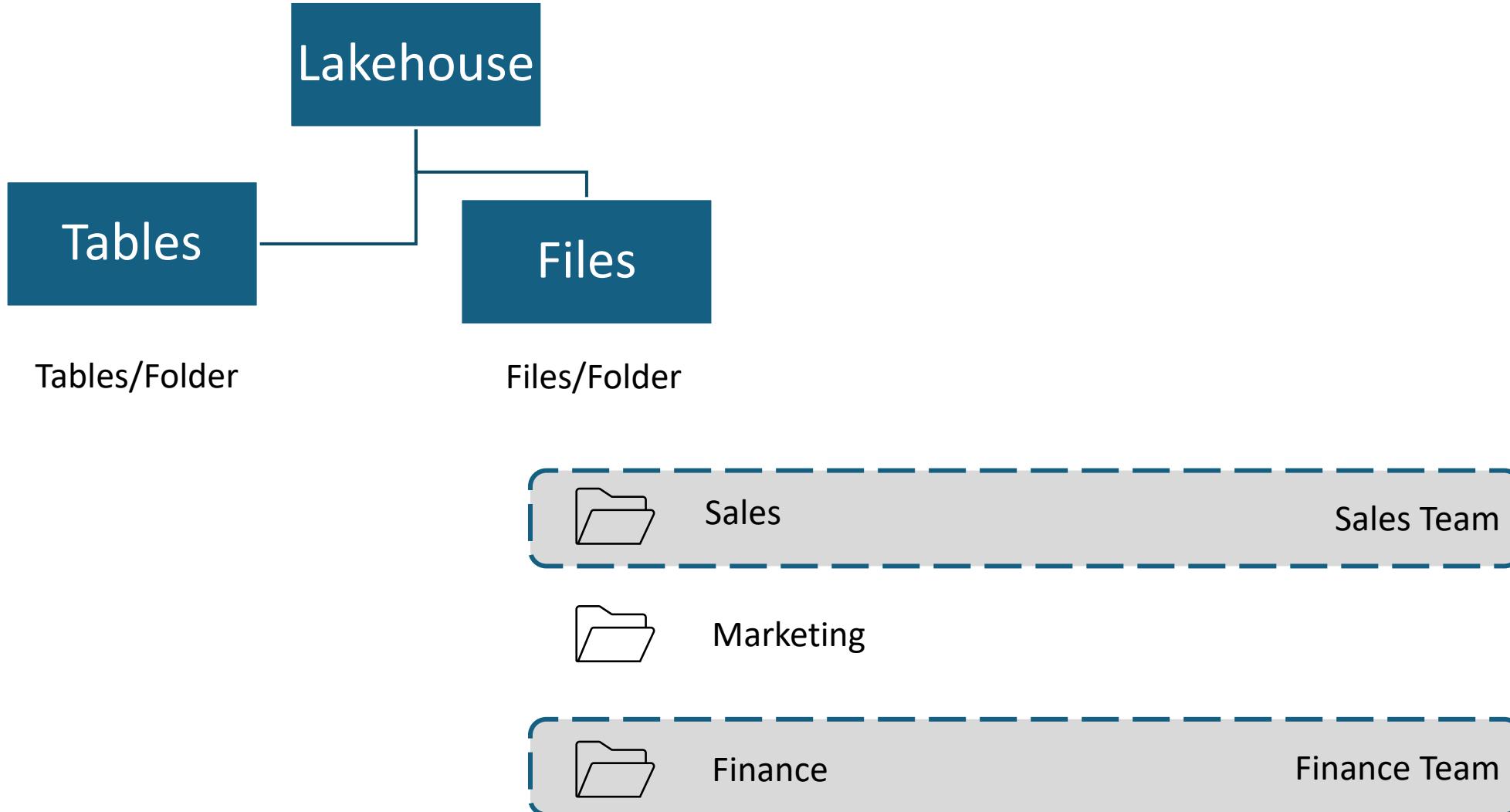
Microsoft Fabric structure



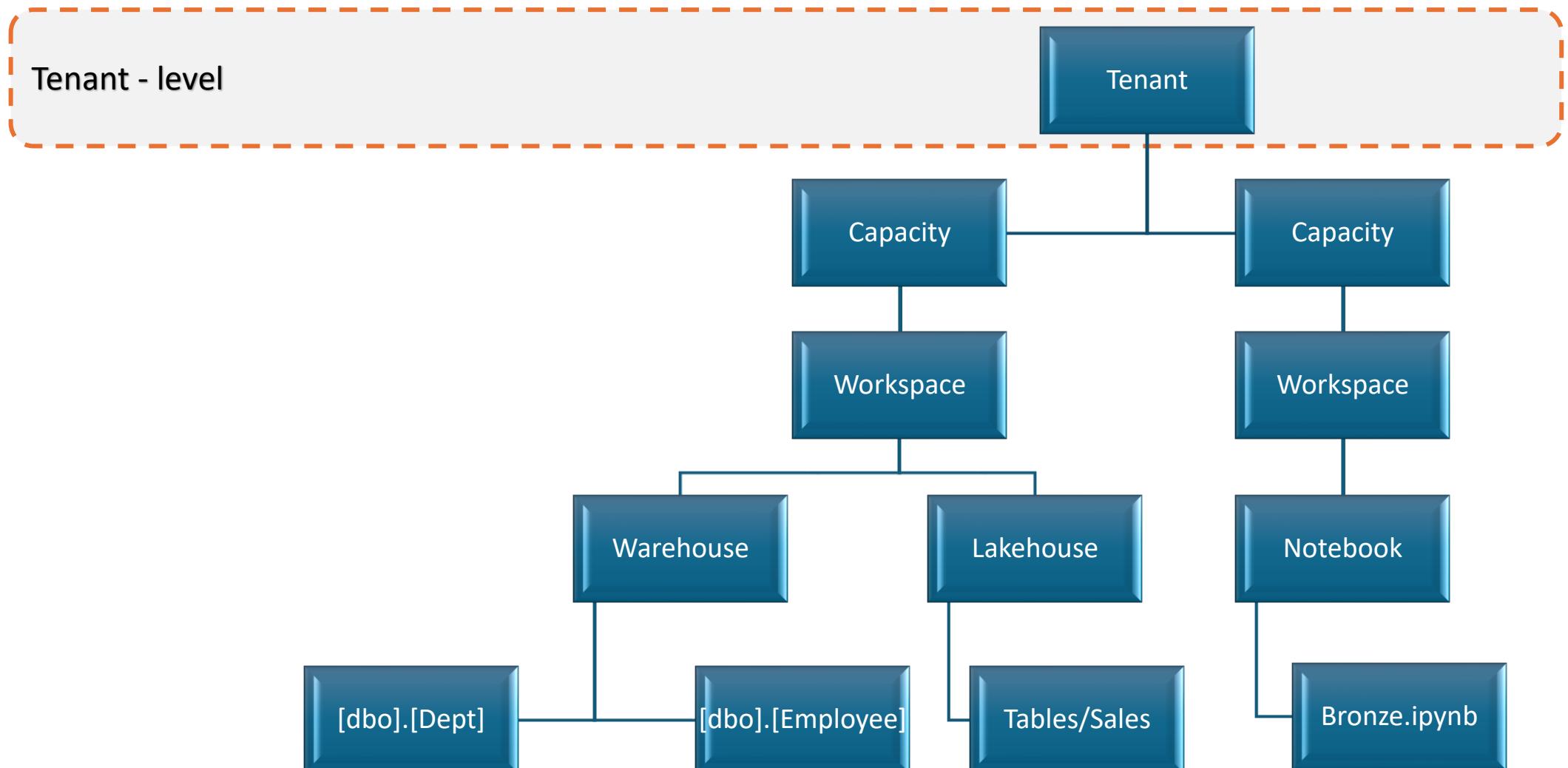
Warehouse security



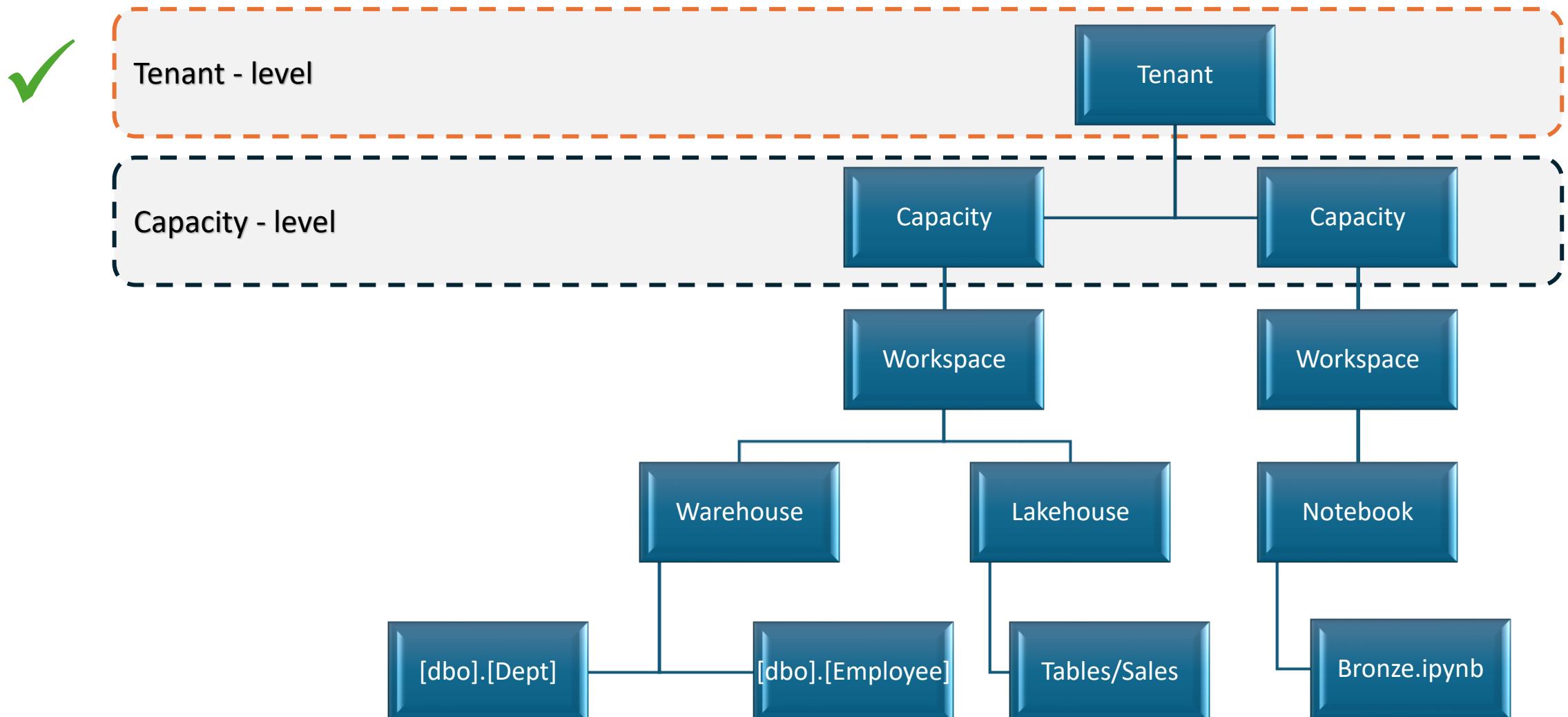
Lakehouse security



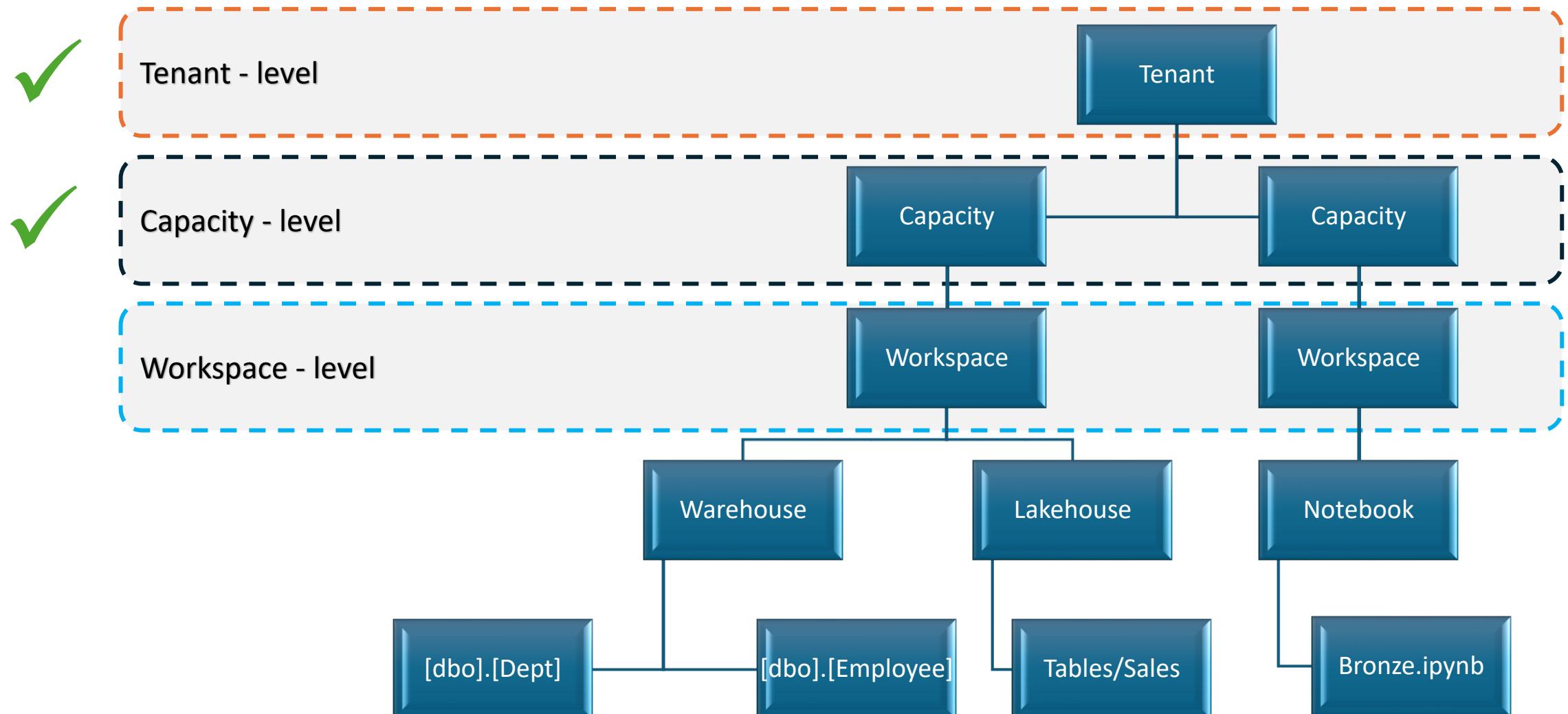
Tenant – Level permissions



Capacity level permissions



Workspace level security



Workspace roles

Each workspace role contains permissions that allow users to perform certain actions

Role	Can add admins?	Can add members?	Can write data and create items?	Can read data?
Admin	Yes	Yes	Yes	Yes
Member	No	Yes	Yes	Yes
Contributor	No	No	Yes	Yes
Viewer	No	No	No	Yes

Workspace Administration

Workspace action	Admin	Member	Contributor	Viewer
Update/ Delete workspace	✓	✗	✗	✗
Add/ Remove users	✓	✗	✗	✗
Add/ Remove members (and below)	✓	✓	✗	✗
Allow others to re-share items	✓	✓	✗	✗
Schedule refresh (via on-prem data gateway)	✓	✓	✓	✗
Modify gateway connection strings	✓	✓	✓	✗

Keep in mind that you also need permissions on the gateway. Those permissions are managed elsewhere, independent of workspace roles and permissions.

Data pipeline permissions

Data Pipeline actions	Admin	Member	Contributor	Viewer
View content and output	✓	✓	✓	✓
Execute and cancel pipeline execution	✓	✓	✓	✗
Schedule pipeline refreshes	✓	✓	✓	✗
Create , modify and delete pipelines	✓	✓	✓	✗

Notebook, Spark Jobs, Experiments, ML Models, Event streams permissions

Item actions	Admin	Member	Contributor	Viewer
View content and output	✓	✓	✓	✓
Execute and cancel item/pipeline execution	✓	✓	✓	✗
Create , modify and delete items	✓	✓	✓	✗

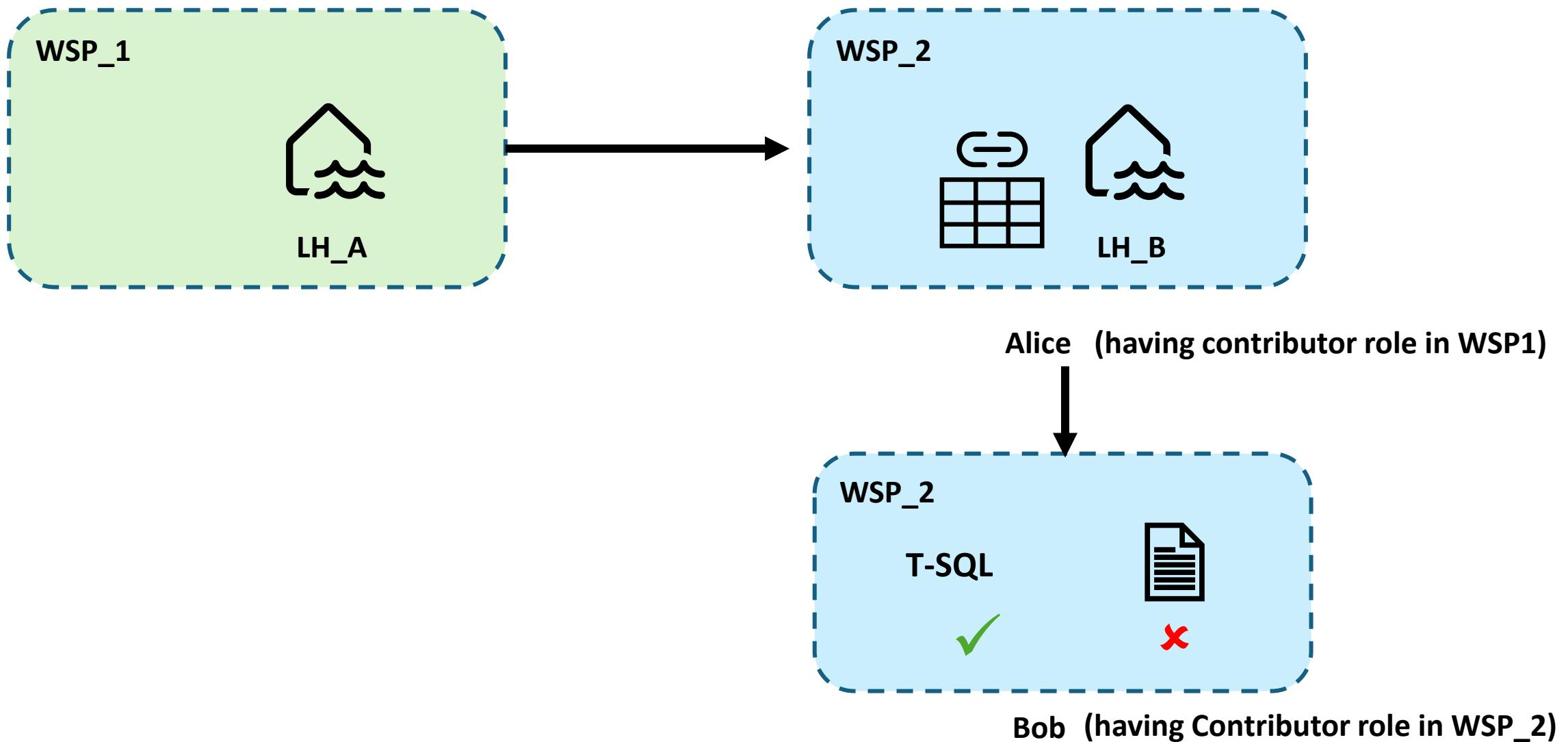
Data warehouse permissions

Data warehouse actions	Admin	Member	Contributor	Viewer
Connect to SQL Analytics endpoint	✓	✓	✓	✓
Read data and shortcuts through SQL endpoint	✓	✓	✓	✓
Read through OneLake API	✓	✓	✓	✗
Read through Spark (Shortcut)	✓	✓	✓	✗
Create , Modify tables / views, etc.	✓	✓	✓	✗

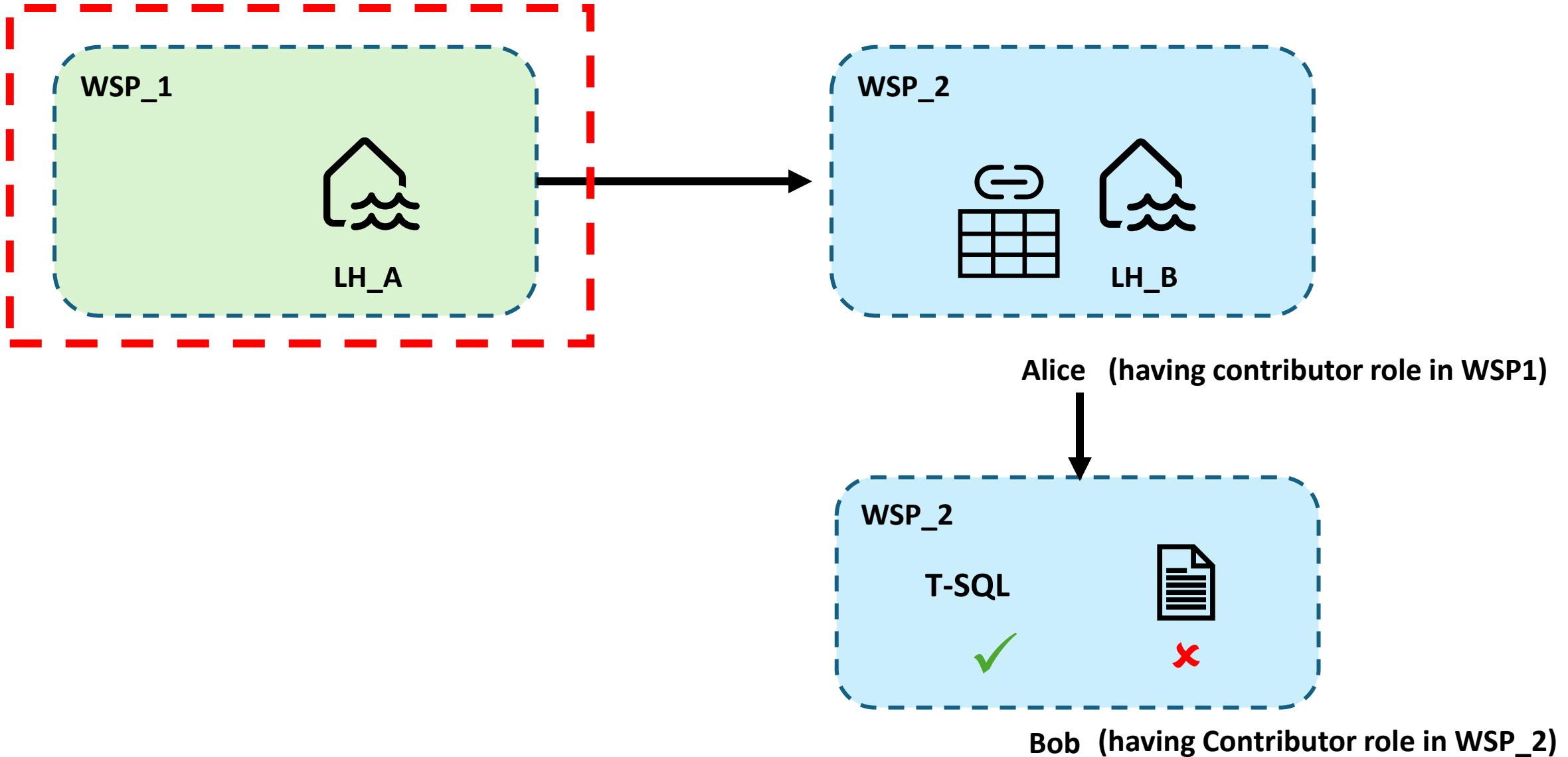
Shortcuts :

1. Reading through shortcuts need additional permission from shortcut destination for objects internal to Fabric
2. ADLS shortcuts use delegated authorization model

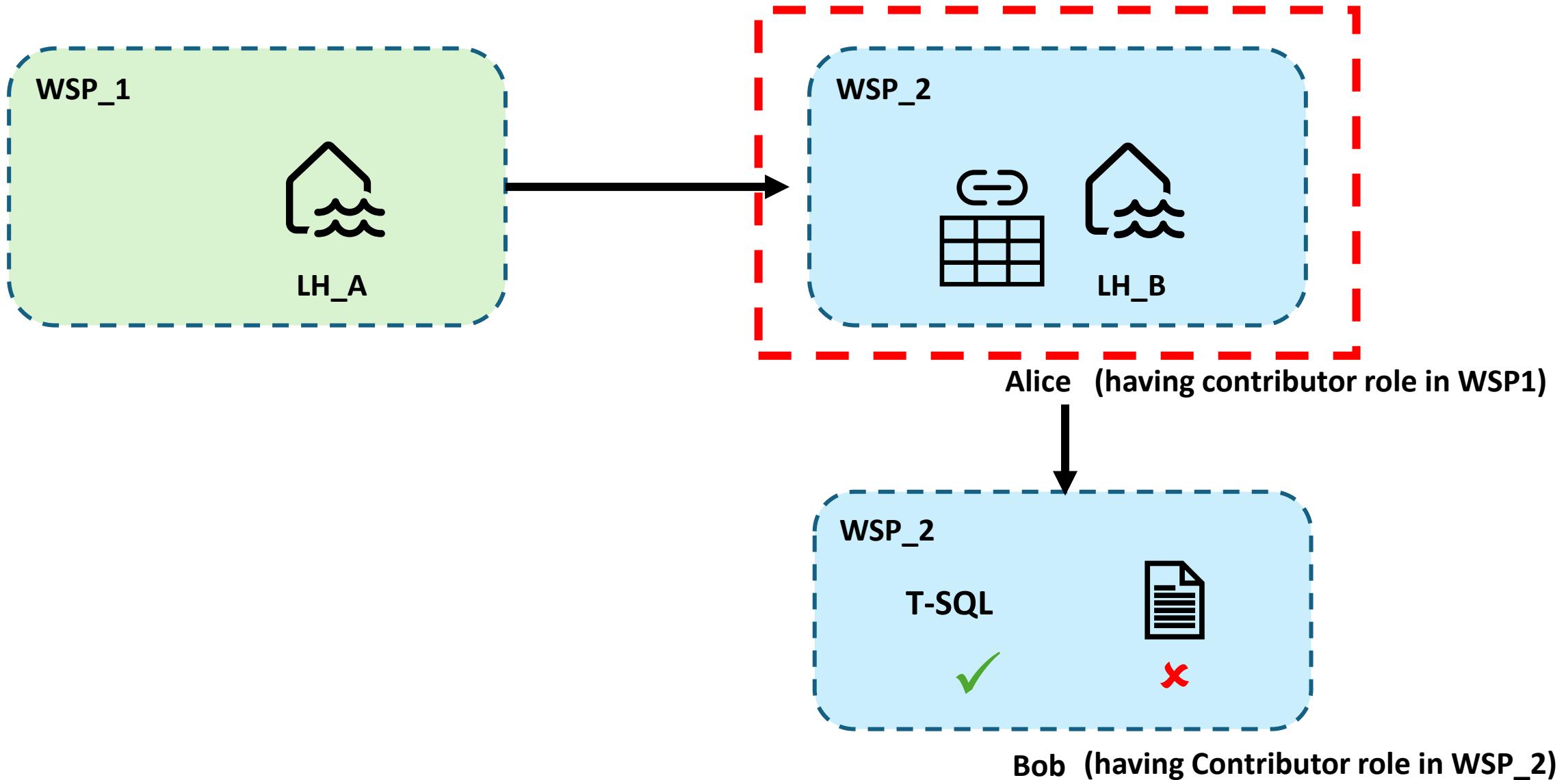
Accessing shortcuts internal to fabric



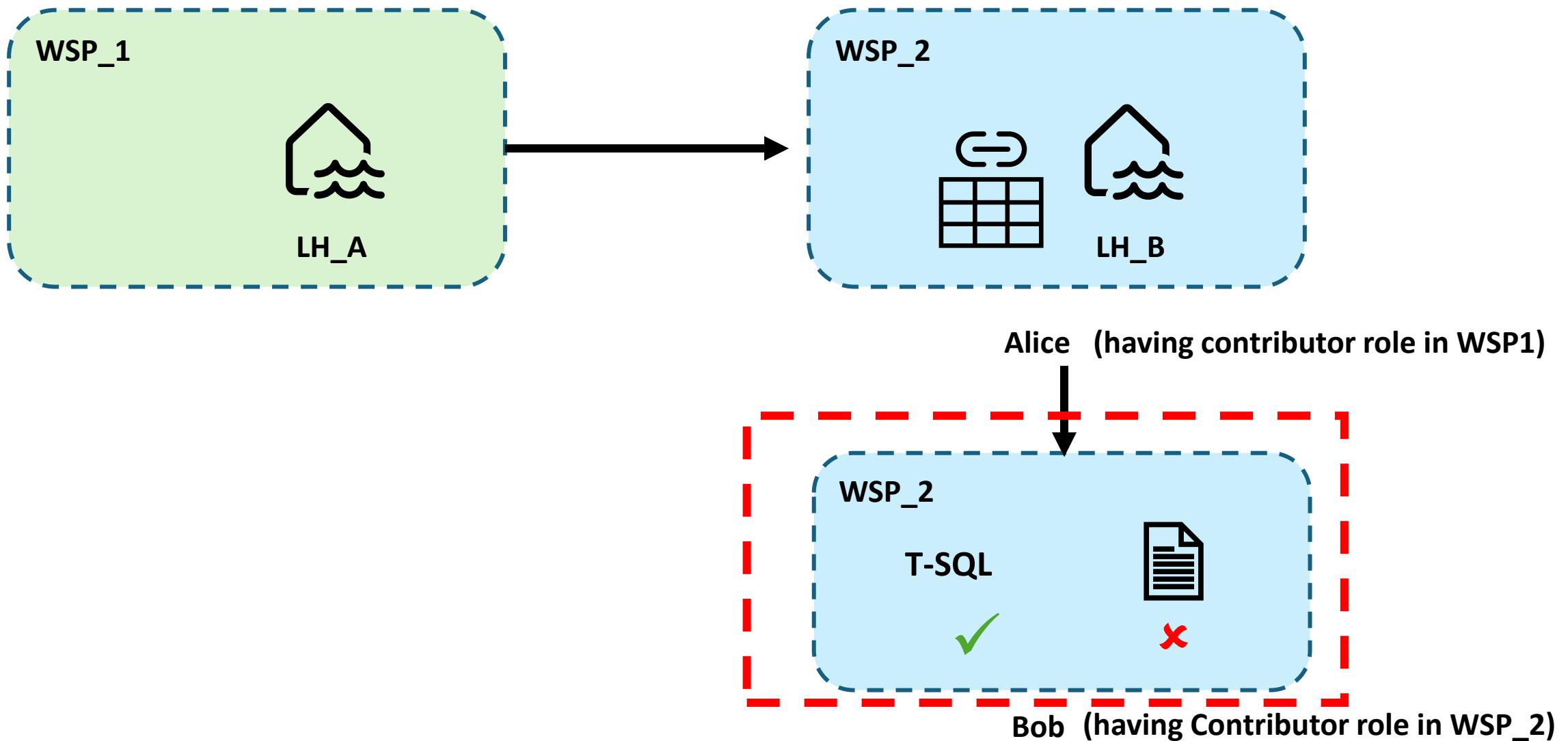
Accessing shortcuts internal to fabric



Accessing shortcuts internal to fabric



Accessing shortcuts internal to fabric



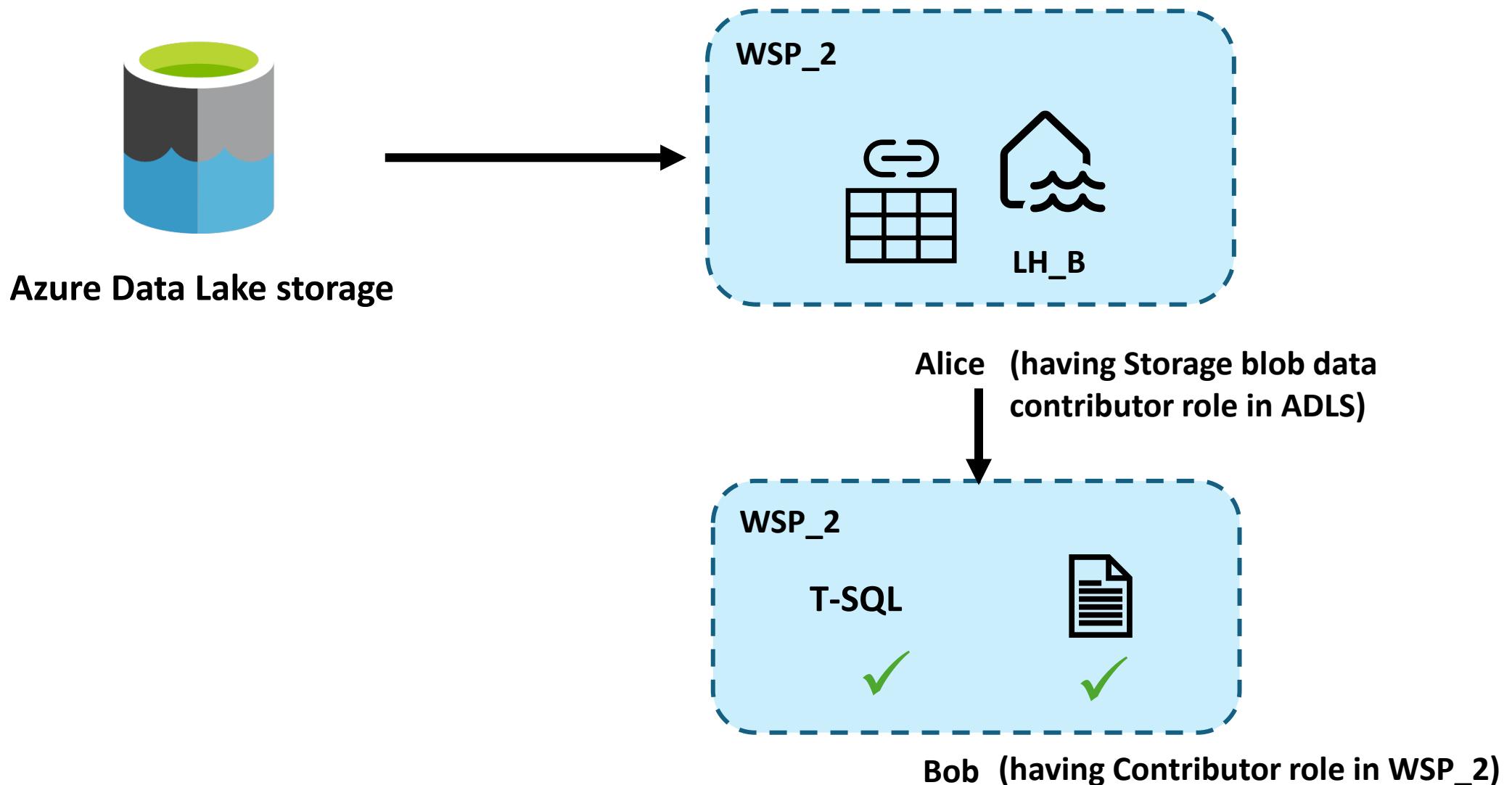
Accessing shortcuts internal to fabric

When accessing shortcuts through Power BI semantic models or T-SQL, **the calling user's identity** (who is currently using the session) **is not passed through to the shortcut target**. The calling item owner's identity (who created that shortcut item) is passed instead, delegating access to the calling user.

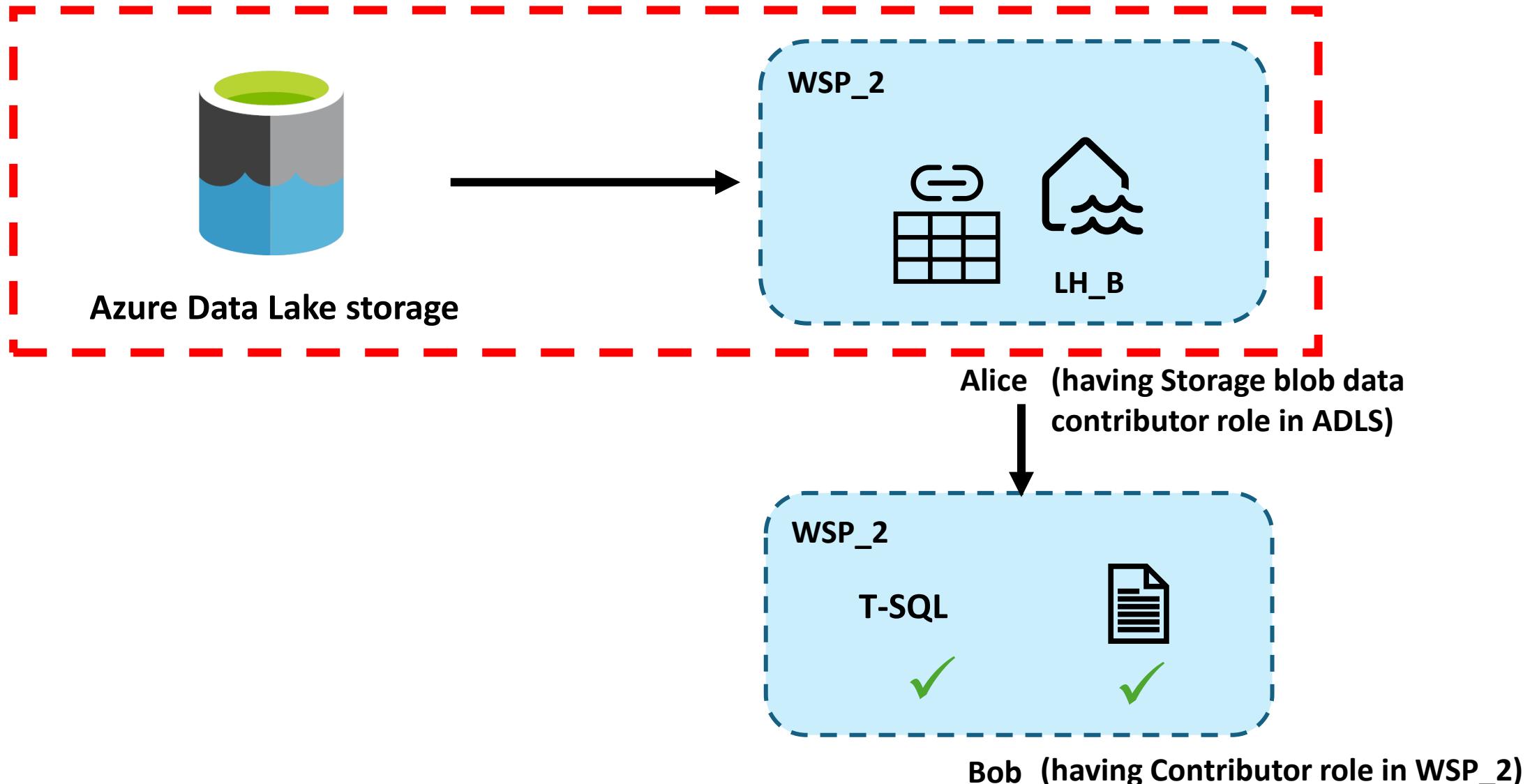
But this behavior will be different when accessing from notebook, the caller need to have access at shortcut destination

Reference: [OneLake shortcuts - Microsoft Fabric | Microsoft Learn](#)

Accessing ADLS shortcuts



Accessing ADLS shortcuts



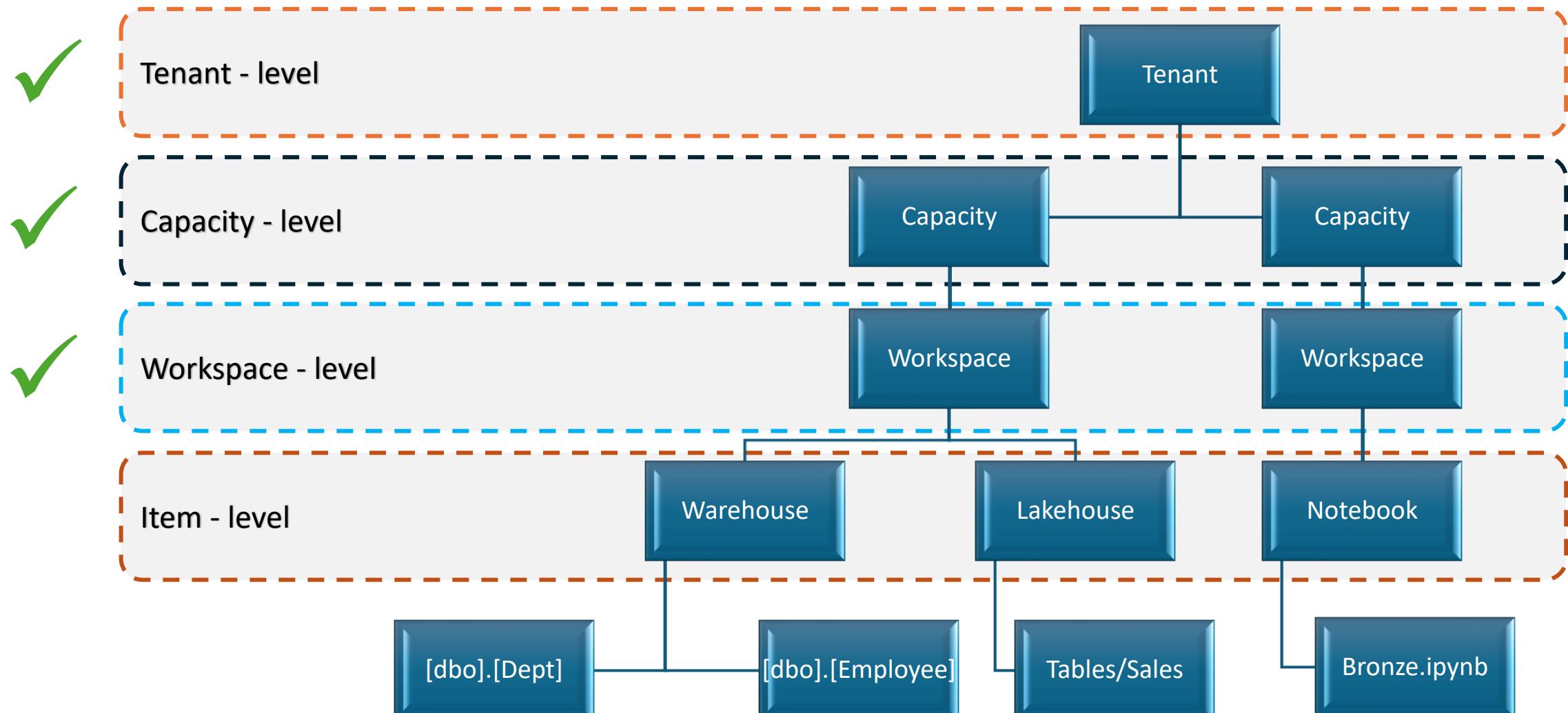
Lakehouse permissions

Lakehouse actions	Admin	Member	Contributor	Viewer
Connect to SQL Analytics endpoint	✓	✓	✓	✓
Read data and shortcuts through SQL endpoint	✓	✓	✓	✓
Read data and shortcuts through Lakehouse Explorer	✓	✓	✓	✗
Read through OneLake API	✓	✓	✓	✗
Read through Spark	✓	✓	✓	✗
Reading Files	✓	✓	✓	✗
Create / modify / delete tables / files	✓	✓	✓	✗

Shortcuts :

1. Reading through shortcuts need additional permission from shortcut destination for objects internal to Fabric
2. ADLS shortcuts use delegated authorization model

Item- level permissions



Item-Level permissions

- You can share individual items by clicking on 3 dots beside that item

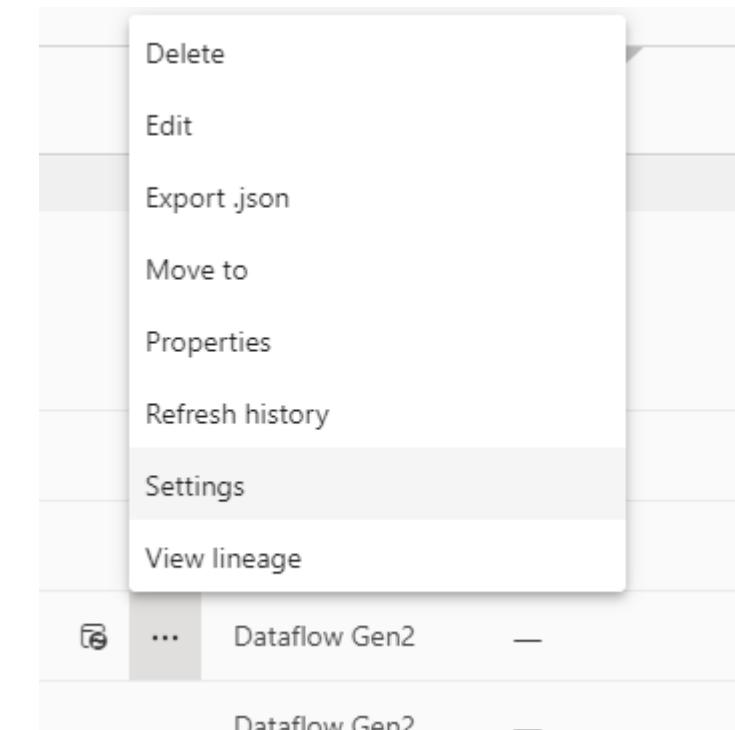
Item level sharing is useful when:

- You want to collaborate with colleagues who don't have a role in workspace
- You want to grant additional item level permissions for colleagues who already have a role in the workspace

Official documentation: [Share items in Microsoft Fabric - Microsoft Fabric | Microsoft Learn](https://learn.microsoft.com/en-us/fabric/tutorials/share-items)

Item Level permissions

Items	Sharing
Data pipeline	✗
Data flow Gen2	✗
Event Stream	✗



Data warehouse sharing

Grant people access

WH_practise

People you share this warehouse with can connect to it and use any selected additional permissions (permission to build reports is selected by default). Select additional permissions to give access to data.

Enter a name or email address

Additional permissions

- Read all data using SQL (ReadData) ⓘ
- Read all OneLake data (ReadAll) ⓘ
- Build reports on the default semantic model (Build) ⓘ

Notification Options

- Notify recipients by email

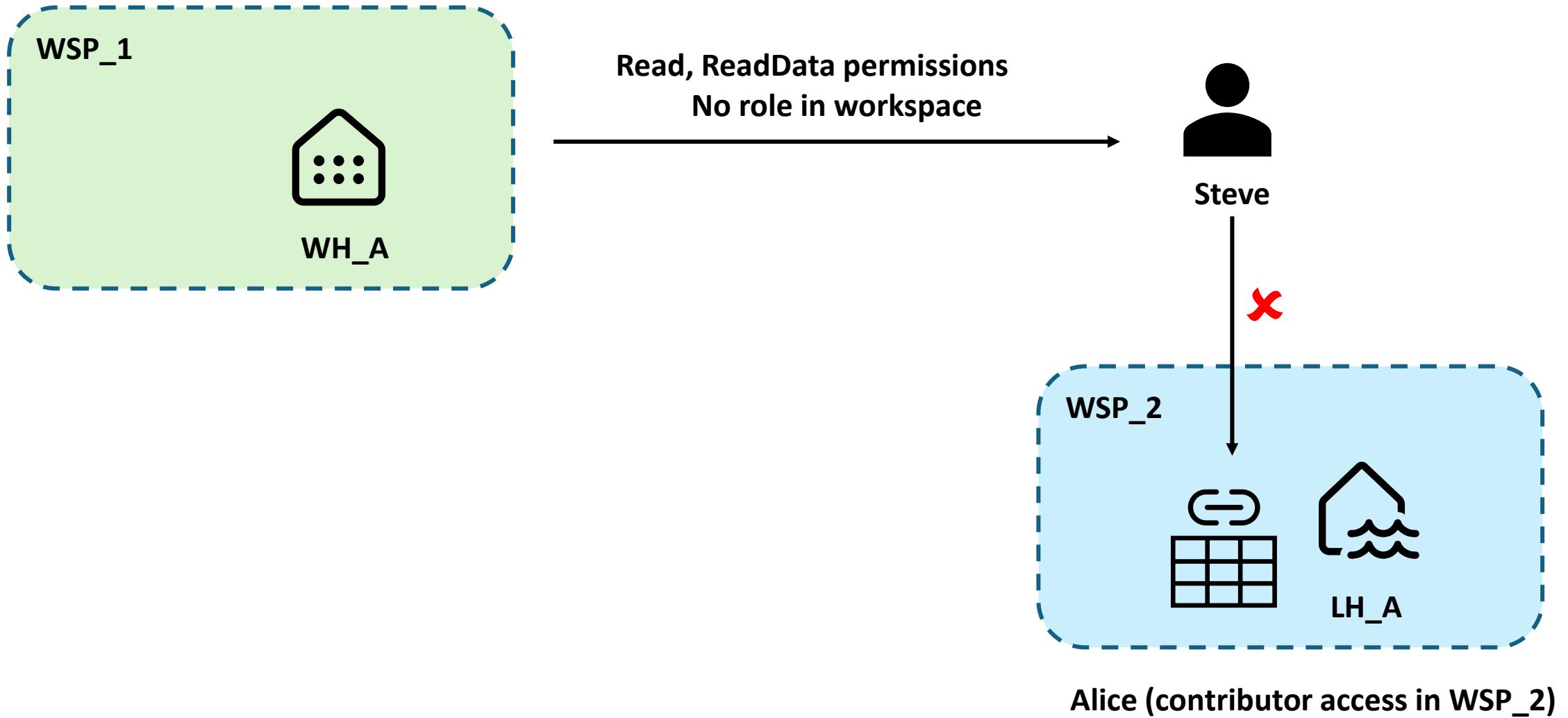
Add a message (optional)

ⓘ To define granular object-level security (OLS) for specific objects in the warehouse, use GRANT and DENY statements in T-SQL.

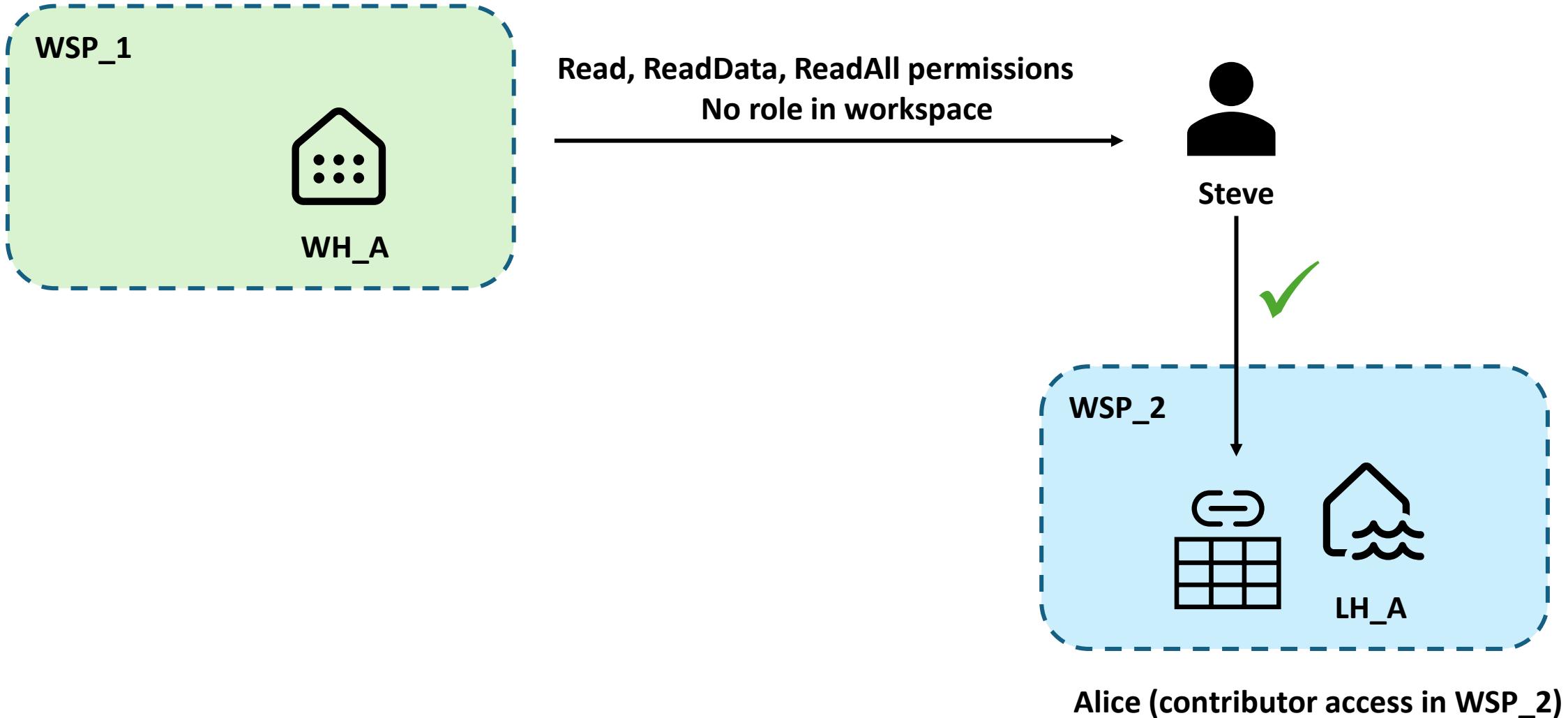
Warehouse item sharing permissions

Permission granted while sharing	Effect
If no additional permissions are selected	<p>The shared recipient by default receives "Read" permission, which only allows recipient to</p> <ul style="list-style-type: none">• Connect to SQL analytics endpoint• Recipient cannot query any table or view• Recipient cannot execute any function or Stored proc
Read all data using SQL (ReadData)	<p>Read all objects within the warehouse using T-SQL</p> <p>The shared recipient can read all database objects within warehouse</p> <ul style="list-style-type: none">• ReadData is equivalent to db_datareader role in SQL server• Further restriction can be done by using GRANT/REVOKE/DENY statements
Read all OneLake data (ReadAll)	<p>Read the warehouse's underlying OneLake files using</p> <ul style="list-style-type: none">• Apache Spark• Pipelines, or Shortcut• Other apps that access the OneLake data directly
Build reports on the default semantic Model (Build)	<ul style="list-style-type: none">• Build reports on top of the default semantic model connected to the warehouse

Warehouse – Read all data using SQL (ReadData)



Warehouse - Read all OneLake (ReadAll)



Lakehouse sharing

Grant people access



LH_practise

People you share this Lakehouse with can open it and its SQL endpoint and read the default dataset. To allow them to read directly in the Lakehouse, grant additional permissions.

Enter a name or email address

Additional permissions

- Read all SQL endpoint data ⓘ
- Read all Apache Spark ⓘ
- Build reports on the default semantic model

Notification Options

- Notify recipients by email

Add a message (optional)

Lakehouse item sharing permissions

Permission granted while sharing	Effect
If no additional permissions are selected	<p>The shared recipient by default receives "Read" permission, which only allows recipient to</p> <ul style="list-style-type: none">• Connect to SQL analytics endpoint• Recipient cannot query any table or view• Recipient cannot see anything in Lakehouse Explorer
Read all SQL Endpoint data	<ul style="list-style-type: none">• Read data from the SQL analytics endpoint of the Lakehouse• User cannot create or modify tables• Need GRANT / MODIFY from admins to make changes
Read all Apache Spark (ReadAll)	<ul style="list-style-type: none">• Read Lakehouse data through OneLake APIs and Spark.• Read Lakehouse data through Lakehouse explorer.
Build reports on the default semantic Model (Build)	<ul style="list-style-type: none">• Build reports on top of the default semantic model connected to the warehouse

Notebook - sharing

Select permissions

Notebook 1



People who can view this Notebook



People in your organization



People with existing access



Specific people

Additional permissions

Authorized users can view this Notebook by default.
Select additional permissions.

Share ⓘ

Edit ⓘ

Run ⓘ

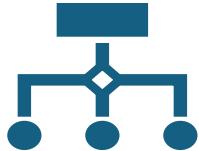


You must also grant run permission to any user who gets edit permission.

Apply

Back

Item-level permissions



People in your organization:

This type of link allows people in your organization to access this item. It doesn't work for external users or guest users. Use this link type when:

- You want to share with someone in your organization.
- You're comfortable with the link being shared with other people in your organization.
- You want to ensure that the link doesn't work for external or guest users.



People with existing access:

This type of link generates a URL to the item, but it doesn't grant any access to the item. Use this link type if you just want to send a link to somebody who already has access.

Select permissions X

Sparknew

People who can view this Environment

People in your organization

People with existing access

Specific people
Author: Shanmukh Sattiraju
<https://in.linkedin.com/in/shanmukh-sattiraju>



Specific people:

This type of link allows specific people or groups to access the report.

This link type also lets you share to guest users in your organization's Microsoft Entra ID

You can't share to external users who aren't guests in your organization.

Notebook – Additional permissions

Additional permission	Effect
If no additional permissions are selected	The shared recipient will only see the notebook cells but cannot execute it
Share	Share the notebook with other people. This permission is also known as Reshare.
Edit	Edit all notebook cells. This permission is also known as Write.
Run	Run all notebook cells. This permission is also known as Execute.

You must also grant run permission to any user who gets edit permission.

OneLake data access

- This is a new feature that enables you to apply role-based access control (RBAC) to your data stored in OneLake.
- You can define security roles that grant read access to specific folders within a Fabric item, and assign them to users or group
- By default , everyone have DefaultReader role to read all folders

Manage OneLake data access (preview) X

LH_practise

Create data access roles to manage OneLake access for folders within the Lakehouse. Assign users to roles to grant them access to the folders covered by that role. [Learn more](#)

+ New role Assign Delete Search

Role name

 DefaultReader

Author: Shanmukh Sattiraju
<https://in.linkedin.com/in/shanmukh-sattiraju>

OneLake data access

- You can provide access to tables or files section in Lakehouse
- All will be shown in the format of folders
- For OneLake shortcuts, the permissions must be defined in the destination table. Defining permissions on the shortcut itself is not allowed.

Assign role

Role name *

Role name cannot contain whitespaces.

Included folders *

All folders

Selected folders

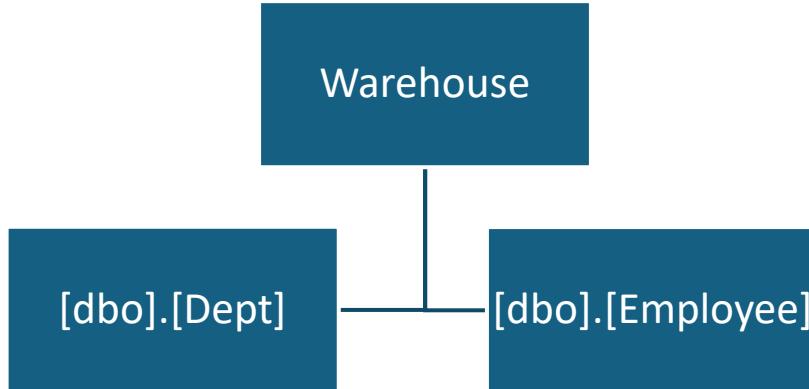
Select one or more folders or files.

> \Tables Folder

> \Files Folder

Author: Shanmukh Sattiraju
<https://in.linkedin.com/in/shanmukh-sattiraju>

Row-level security



A table showing departmental data with a highlighted row. The table has columns: DeptID, DeptName, Budget, and CC Number. The highlighted row (row 4) is shaded blue and has a dashed border. An arrow points from the text 'Row level security' to this highlighted row.

DeptID	DeptName	Budget	CC Number
101	HR	\$1000	xxxx31
102	Admin	\$2000	xxxx53
103	IT	\$3000	xxxx34
104	Transport	\$4000	xxxx76
105	Sales	\$5000	xxxx42

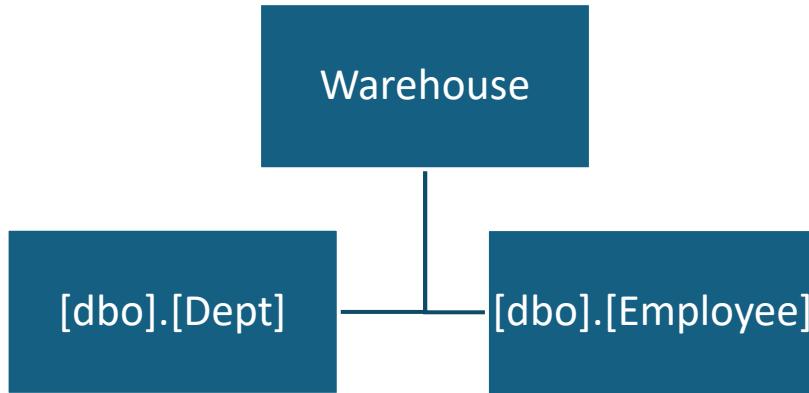
Available in :

- Synapse Data Warehousing
- SQL analytics endpoint in lakehouse

Row-level security

- The access restriction logic is in the database tier
- The database applies the access restrictions every time data access is attempted
- Access to row-level data in a table is restricted by a security predicate defined as an inline table-valued function.
- The function is then invoked and enforced by a security policy.

Dynamic data masking



DeptID	DeptName	Budget	CC Number
101	HR	\$1000	xxxx31
102	Admin	\$2000	xxxx53
103	IT	\$3000	xxxx34
104	Transport	\$4000	xxxx76
105	Sales	\$5000	xxxx42



Row level security

Available in :

- Synapse Data Warehousing
- SQL analytics endpoint in lakehouse



Dynamic data masking

Dynamic Data Masking

- Dynamic data masking limits sensitive data exposure by masking it to nonprivileged users.
- Prevent unauthorized viewing of sensitive data by enabling administrators to specify how much sensitive data to reveal
- Can be configured on designated database fields to hide sensitive data in the result sets of queries
- Users without the Administrator, Member, or Contributor rights on the workspace, and without elevated permissions on the Warehouse, will see masked data.

Function	Description
Default	<p>.</p> <p>For string data types, use XXXX (or fewer) if the size of the field is fewer than 4 characters (char, nchar, varchar, nvarchar, text, ntext).</p> <p>For numeric data types use a zero value (bigint, bit, decimal, int, money, numeric, smallint, smallmoney, tinyint, float, real).</p> <p>For date and time data types, use 1900-01-01 00:00:00.0000000 (date, datetime2, datetime, datetimeoffset, smalldatetime, time).</p> <p>For binary data types use a single byte of ASCII value 0 (binary, varbinary, image).</p>
Email	
Random	A random masking function for use on any numeric type to mask the original value with a random value within a specified range.
Custom String	<p>Masking method that exposes the first and last letters and adds a custom padding string in the middle. prefix,[padding],suffix</p> <p>If the original value is too short to complete the entire mask, part of the prefix or suffix isn't exposed.</p>

default() function

Full masking according to the data types of the designated fields

ABC	SalesRep	ABC
	xxxx	Sm
	xxxx	Lap
	xxxx	He
	xxxx	Tab
		-

Default() function

Column type	DataTypes	Masked value
Strings	char, nchar, varchar, nvarchar, text, ntext	XXXX
Numeric	bigint, bit, decimal, int, money, numeric, smallint, smallmoney, tinyint, float, real	0
Date and DateTime	date, datetime2, datetime, datetimeoffset, smalldatetime, time	1900-01-01 00:00:00.0000000
Binary	binary, varbinary, image	ASCII value of 0

Email() function

- Masking method that exposes the first letter of an email address and the constant suffix ".com", in the form of an email address. aXXX@XXXX.com.



Bypassing masking using inference or brute-force techniques

- Dynamic Data Masking can be useful to prevent accidental exposure of sensitive data when accessing data directly
- It's important to note that unprivileged users with query permissions can apply techniques to gain access to the actual data.
- dynamic data masking shouldn't be used alone to fully secure sensitive data from users with query access to the Warehouse or SQL analytics endpoint.
- It's appropriate for preventing sensitive data exposure, but doesn't protect against malicious intent to infer the underlying data.
- Follow principle of Least privilege and use Object-level security with SQL GRANT/REVOKE/ statements

Column level security



DeptID	DeptName	Budget	CC Number
101	HR	\$1000	xxxx31
102	Admin	\$2000	xxxx53
103	IT	\$3000	xxxx34
104	Transport	\$4000	xxxx76
105	Sales	\$5000	xxxx42

Available in :

- Synapse Data Warehousing
- SQL analytics endpoint in lakehouse

✓ Dynamic data masking

Column-level security

- Allows you to restrict column access to certain users
- The access restriction logic is located in the database tier,
- The database applies the access restrictions every time data access is attempted
- If user is having "Read all with SQL or ReadData" permission , Column level security will not apply

Best practices

Follow principal of least privilege

- If users only need access to a single lakehouse or data item, use the share feature to grant them access to only that item
- Assigning a user to a workspace role should only be used if that user needs to see ALL items in that workspace
- Use [OneLake data access roles \(preview\)](#) to restrict access to folders and tables within a lakehouse for access through OneLake APIs or Apache Spark notebooks.



Power BI

Semantic Model

Semantic model

- **Semantic models** represent a source of data ready for reporting, visualization
- A default semantic model is created automatically when you create a Lakehouse or warehouse
- This default semantic model will contain all the tables that we create in lakehouse or warehouse

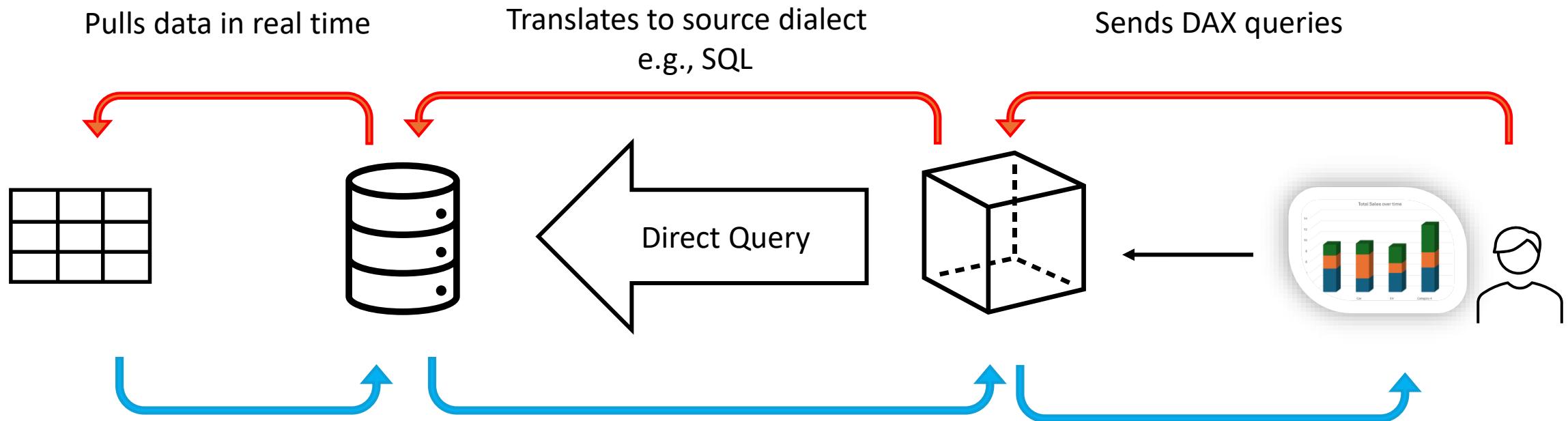
	Name	Type	Task
	LH_practise	Lakehouse	—
	LH_practise	Semantic model (...)	—
	LH_practise	SQL analytics end...	—

Author: Shanmukh Sattiraju
<https://in.linkedin.com/in/shanmukh-sattiraju>

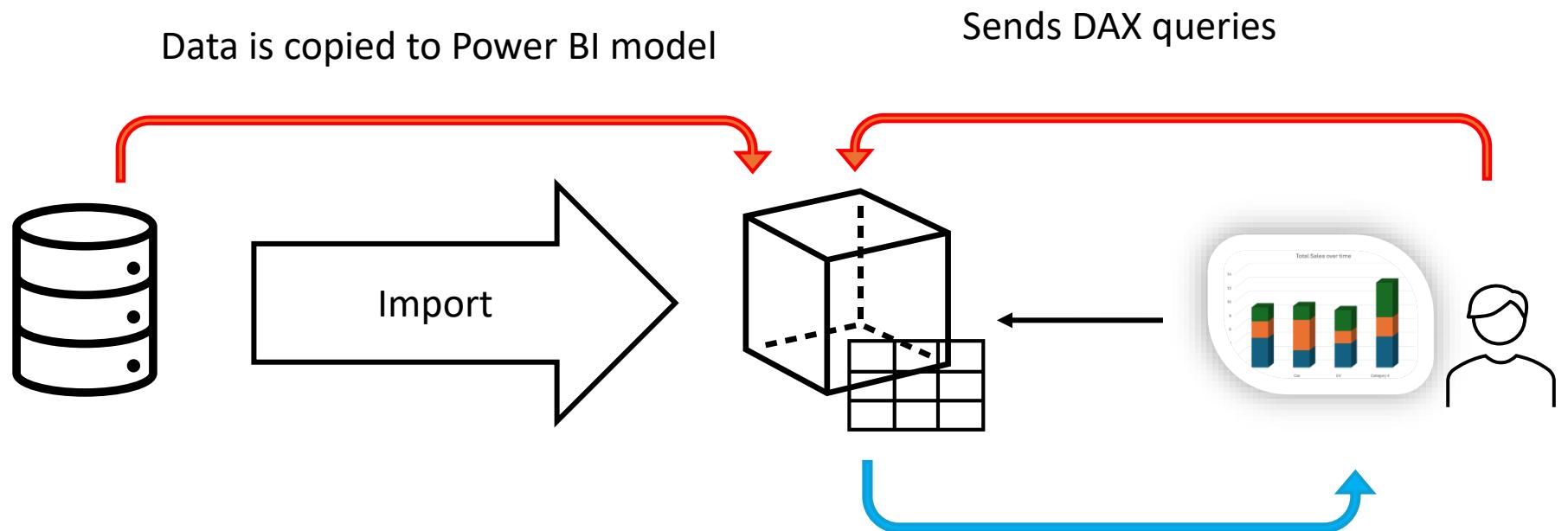
Semantic model with XMLA endpoint

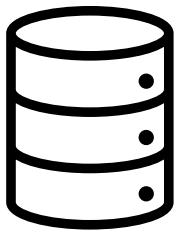
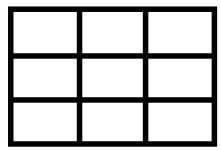
- XMLA (XML for analysis) allows client tools to communicate with semantic models in fabric
- Perform operations like read- or write- . Read is enabled by default , if you want to some write- operations like modifying the semantic model from external client tools, you need to enable the write- from Fabric admin portal
- Client tools are SSMS, DAX studio, Visual Studio, Powershell cmdlets and Tabular Editor 2
- Premium capacity feature : Premium per capacity or premium per user license types

Direct Query mode

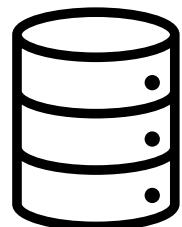
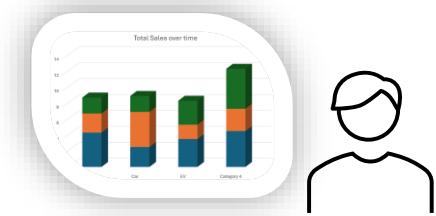
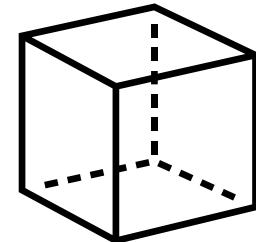
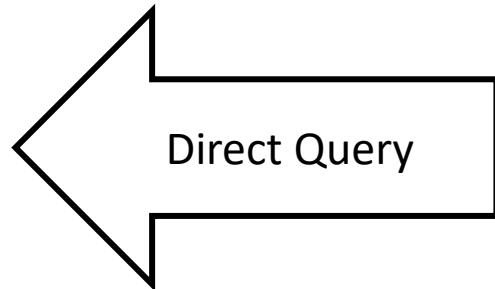


Import mode

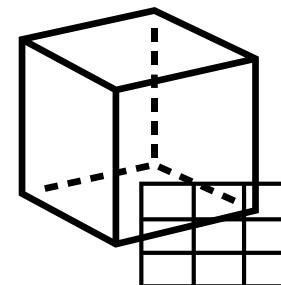
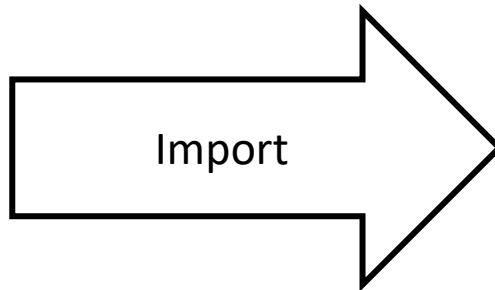




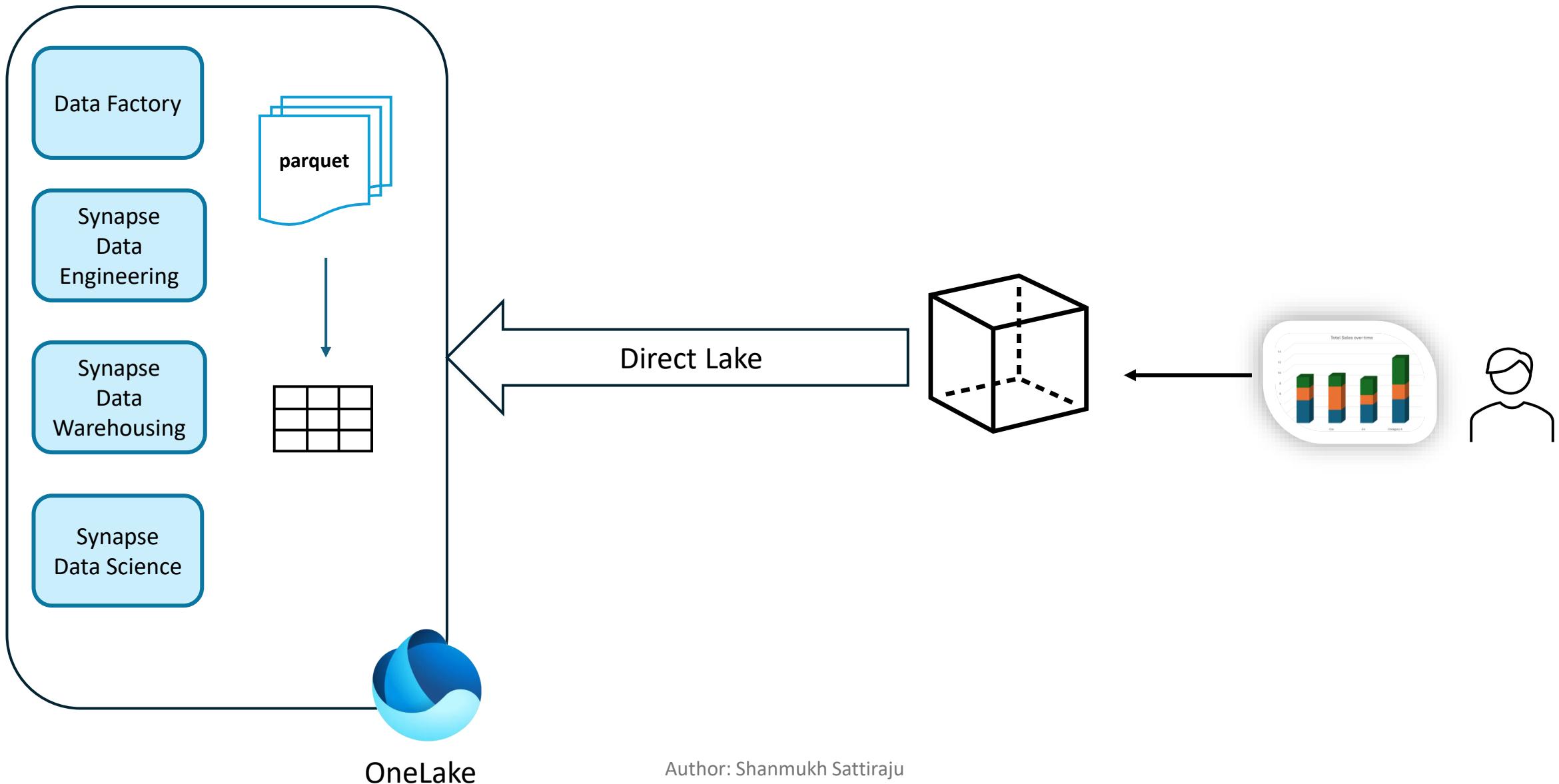
Slow , but real-time



Fast, but latent and duplicative



Direct Lake



Refresh Semantic Model

- Refresh Semantic Model – Manual
- Refresh Semantic Model – Notebook
- Refresh Semantic Model – XMLA endpoint
- Refresh Semantic Model – Data pipeline

Fallback to direct query from Direct lake

DirectLake = Direct Lake mode read Delta tables directly from OneLake

Direct query = Query use SQL to retrieve the results from the SQL endpoint of the lakehouse or warehouse, which can impact query performance.

Fallback to direct query from Direct lake

Scenario 1:
Exceeding SKU limits

Fabric SKUs	Parquet files per table	Row groups per table	Rows per table (millions)	Max model size on disk/OneLake ¹ (GB)	Max memory (GB)
F2	1,000	1,000	300	10	3
F4	1,000	1,000	300	10	3
F8	1,000	1,000	300	10	3
F16	1,000	1,000	300	20	5
F32	1,000	1,000	300	40	10
F64/FT1/P1	5,000	5,000	1,500	Unlimited	25
F128/P2	5,000	5,000	3,000	Unlimited	50
F256/P3	5,000	5,000	6,000	Unlimited	100
F512/P4	10,000	10,000	12,000	Unlimited	200
F1024/P5	10,000	10,000	24,000	Unlimited	400
F2048	10,000	10,000	Author: Shanmukh Sattiraju https://in.linkedin.com/in/shanmukh-sattiraju	24,000	400

Fallback to direct query from Direct lake

Scenario 2:

- Using features that don't support Direct Lake mode, like SQL views in a warehouse, the query can fall back to DirectQuery mode.

Handling fallback behavior

Direct Lake models include the **DirectLakeBehavior** property, which has three options:

- **Automatic** - (Default) Specifies queries fall back to *DirectQuery* mode if data can't be efficiently loaded into memory.
- **DirectLakeOnly** - Specifies all queries use Direct Lake mode only. Fallback to DirectQuery mode is disabled. If data can't be loaded into memory, an error is returned. Use this setting to determine if DAX queries fail to load data into memory, forcing an error to be returned.
- **DirectQueryOnly** - Specifies all queries use DirectQuery mode only. Use this setting to test fallback performance.

Copy measures from Model1 to Model2

Semantic Model 1

Total_UnEmployed
Total_Employed
Total_Sales
Total_Attended



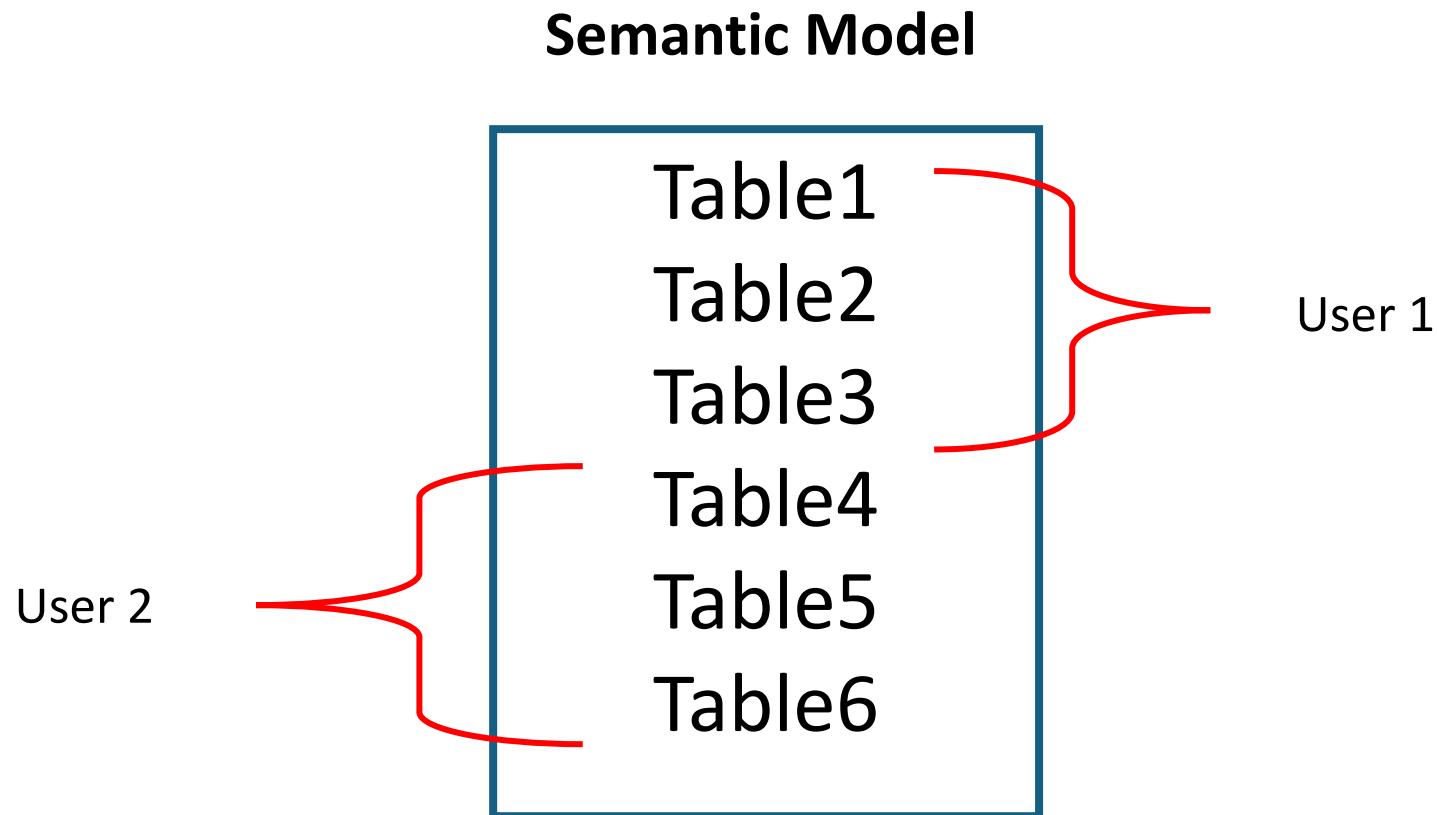
Semantic Model 2

Total_UnEmployed
Total_Employed
Total_Sales
Total_Attended

Row-level security

- Row-level security (RLS) with Power BI can be used to restrict data access for given users.
- RLS only restricts data access for users with **Viewer** permissions. It doesn't apply to Admins, Members, or Contributors.
- You can also configure RLS on semantic models that are using DirectQuery
- You can't define roles within Power BI Desktop for Analysis Services live connections. You need to do that within the Analysis Services model. (applies to DirectLake)
- `userprincipalname()` defines email address of user

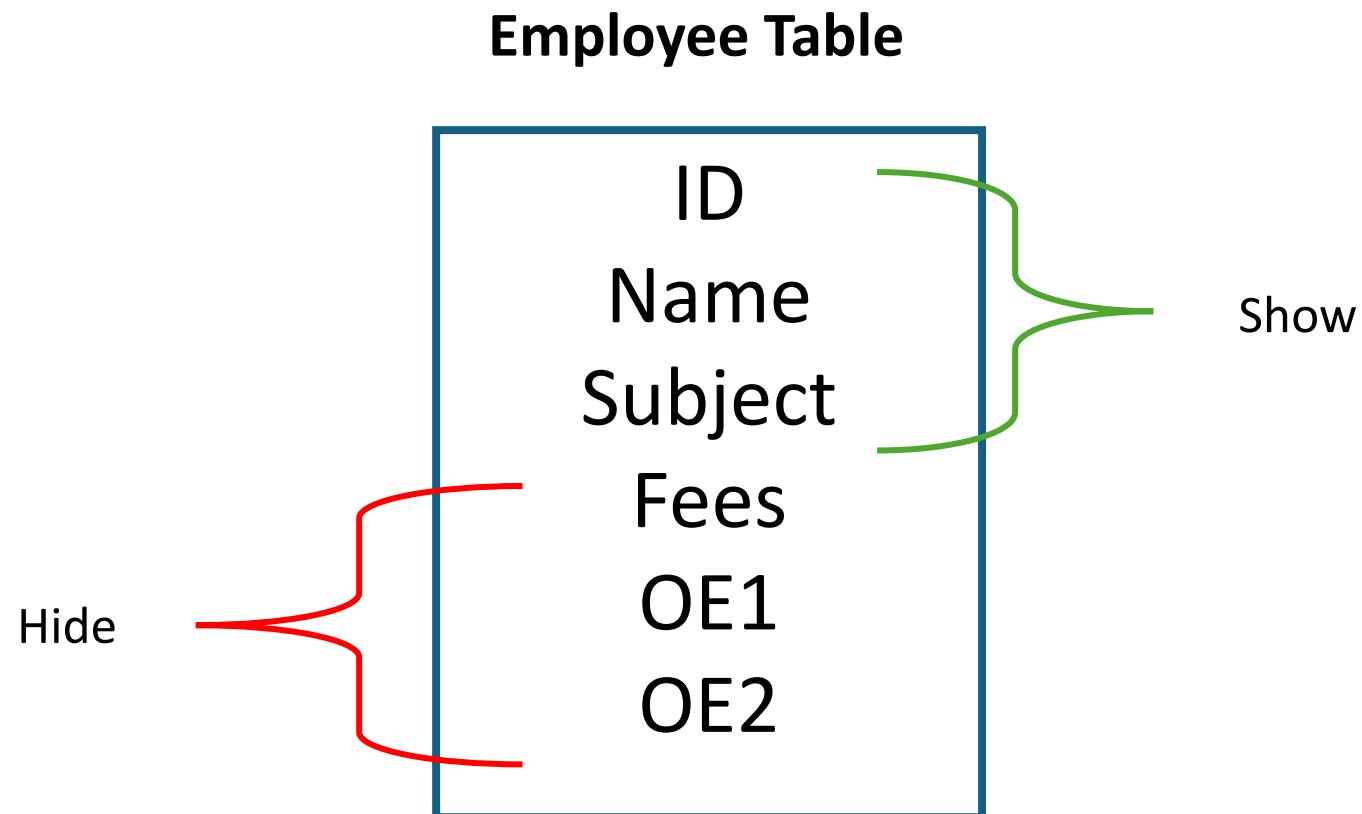
Object Level Security



There are 3 options

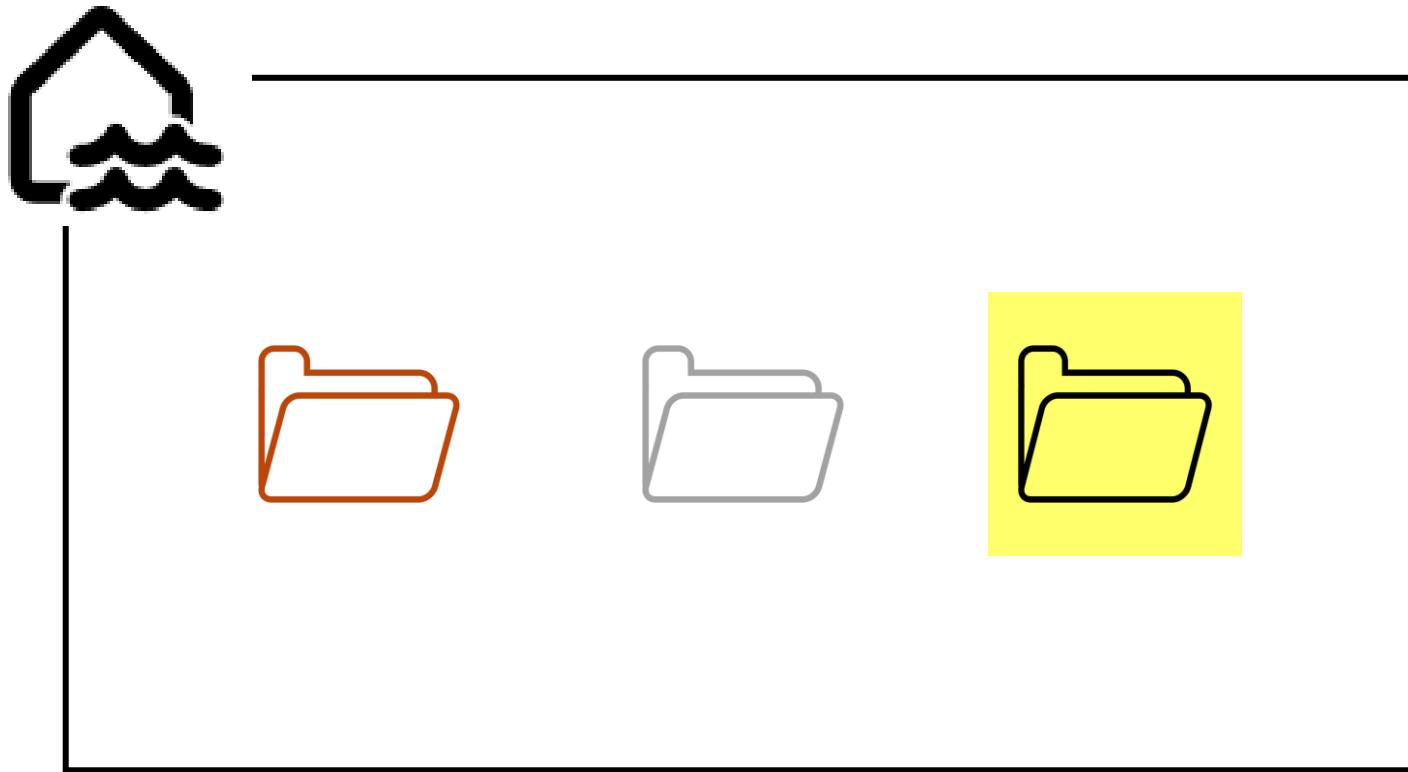
- Default = role by default see this table
- None = This will hide the table from this role
- Read = The table will be visible only to this role

Column Level Security



End to end project implementation in Fabric

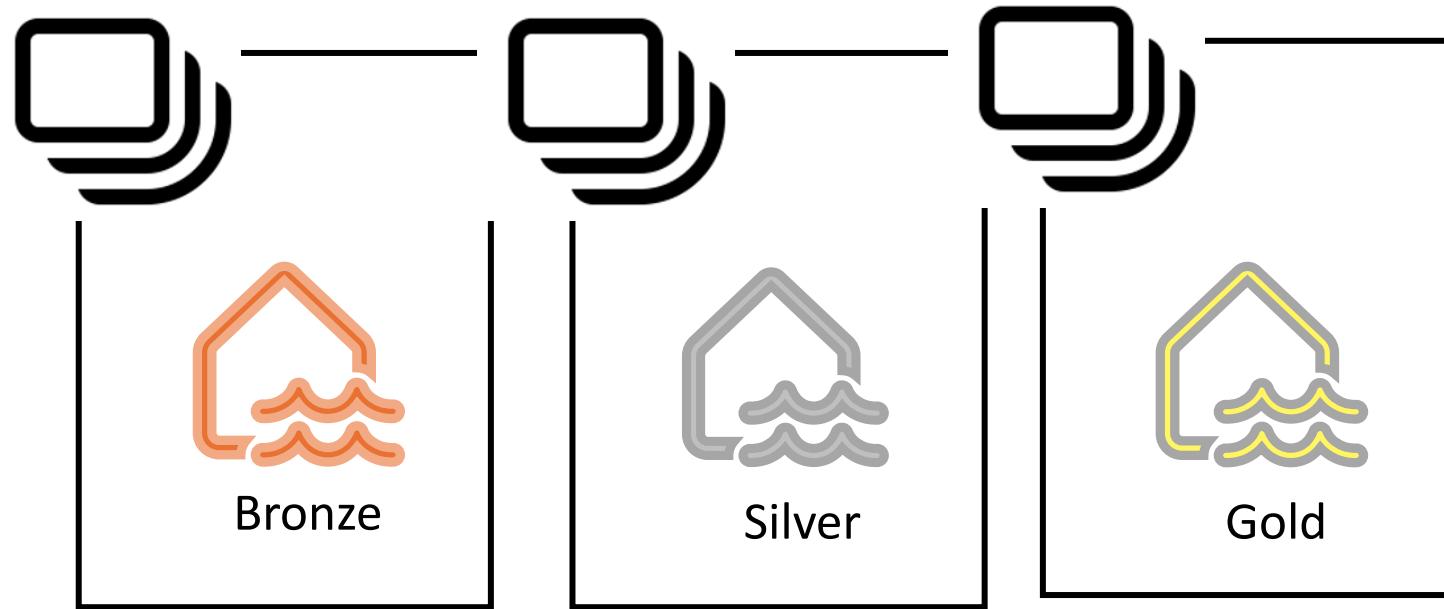
Medallion Architecture foundation



Medallion Architecture foundation



Medallion Architecture foundation



Medallion Architecture in Fabric

	Bronze	Silver	Gold
What happens in this layer?	Ingest raw data	Cleanse and validate data	Additional transformations and modelling
What tool is used?	Pipelines, dataflows and notebooks	Dataflows or notebooks	SQL Endpoint or semantic models

Incremental ingestion types

Timestamp-based incremental ingestion

- Ingest data based on timestamp column (e.g. last_updated or created_at)
- Fetch records where the timestamp is greater than the maximum timestamp from last ingestion

Supported Data Sources:

- Relational Databases: MySQL, PostgreSQL, Oracle, SQL Server
- NoSQL Databases: MongoDB, Cassandra, DynamoDB
- APIs: Rest APIs, SOAP APIs, GraphQL APIs
- Data warehouse: Snowflake, Redshift, Google BigQuery, Azure Synapse

Change Data Capture (CDC)

- Captures changes (inserts, updates, deletes) from source system
- Use database logs or triggers to track changes

Supported Data Sources:

- Relational Databases: MySQL, PostgreSQL, Oracle, SQL Server
- NoSQL Databases: MongoDB, Cassandra, DynamoDB
- Data warehouse: Snowflake, Redshift, Google BigQuery, Azure Synapse

Delta Ingestion

- Ingest only the new or changed data since the last ingestion
- Use unique identifiers to track and compare changes

Supported Data Sources:

- Relational Databases: MySQL, PostgreSQL, Oracle, SQL Server
- NoSQL Databases: MongoDB, Cassandra, DynamoDB
- File Systems: HDFS, S3, Azure Blob or ADLS, Cloud Storage

Batch Window Ingestion

- Ingest data in fixed time interval (e.g. hourly, daily)
- Fetch data within the defined time window

Supported Data Sources:

- Relational Databases: MySQL, PostgreSQL, Oracle, SQL Server
- NoSQL Databases: MongoDB, Cassandra, DynamoDB
- File Systems: HDFS, S3, Azure Blob or ADLS, Cloud Storage

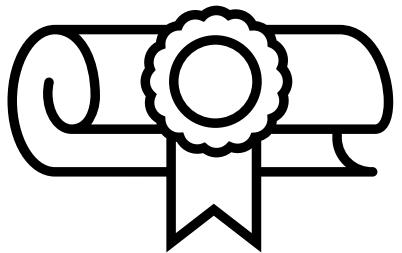
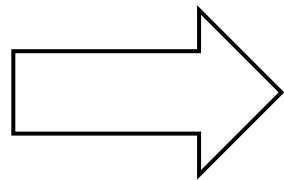
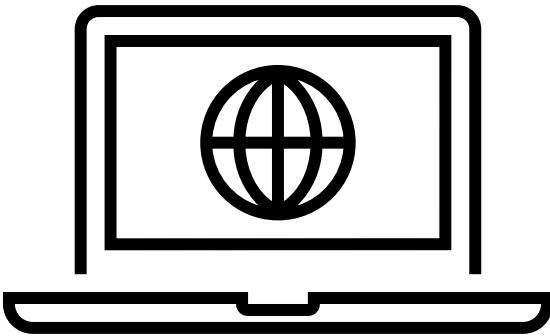
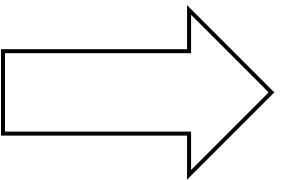
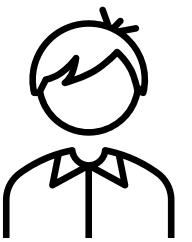
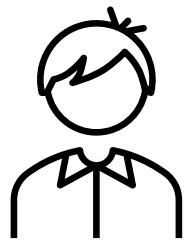
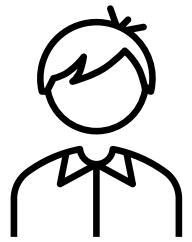
Partition-based ingestion

- Ingest data based on partitions (e.g. date, region)
- Ingest data for each partition incrementally

Supported Data Sources:

- Relational Databases: MySQL, PostgreSQL, Oracle, SQL Server
- NoSQL Databases: MongoDB, Cassandra
- File Systems: HDFS, S3, Azure Blob or ADLS, Cloud Storage
- Data warehouse: Snowflake, Redshift, Google BigQuery, Azure Synapse

End to End Fabric project



Online Learning platform

Dataset information

No.	Column	Description
1	Student_ID	Unique identifier for each student.
2	Name	Student's full name.
3	Age	Student's age.
4	Gender	Student's gender (M/F).
5	Grade_Level	Student's current grade level.
6	Course_ID	Unique identifier for each course.
7	Course_Name	Name of the course.
8	Enrollment_Date	Date the student enrolled in the course.
9	Completion_Date	Date the student completed the course.
10	Status	Current status of the student in the course (e.g., In Progress, Completed).
11	Final_Grade	Final grade obtained by the student in the course.
12	Attendance_Rate	Percentage of classes attended by the student.
13	Time_Spent_on_Course (hrs)	Total hours spent by the student on the course.

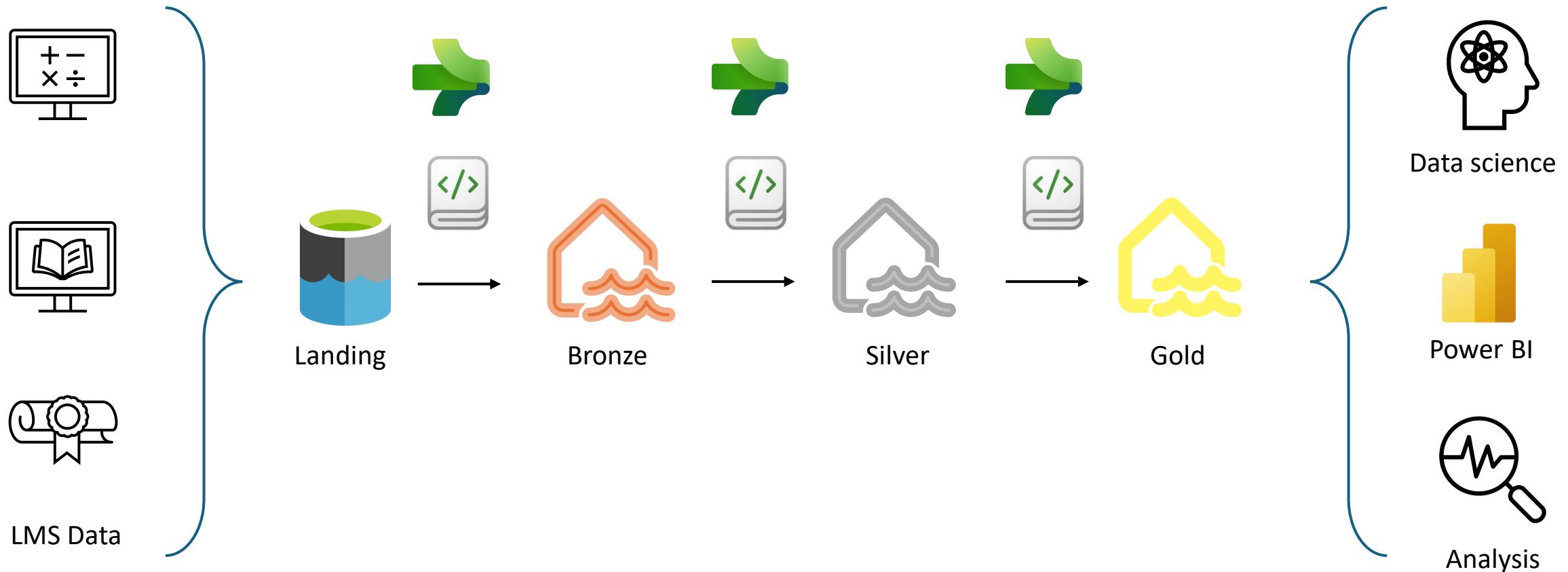
Dataset information

No	Column	Description
14	Assignments_Completed	Number of assignments completed by the student.
15	Quizzes_Completed	Number of quizzes completed by the student.
16	Forum_Posts	Number of forum posts made by the student.
17	Messages_Sent	Number of messages sent by the student.
18	Quiz_Average_Score	Average score of all quizzes taken by the student.
19	Assignment_Scores	Assignment scores by students
20	Assignment_Average_Score	Average score of all assignments completed by the student.
21	Project_Score	Score of the final project completed by the student.
22	Extra_Credit	Extra credit points earned by the student.
23	Overall_Performance	Overall performance score considering all aspects of the course.

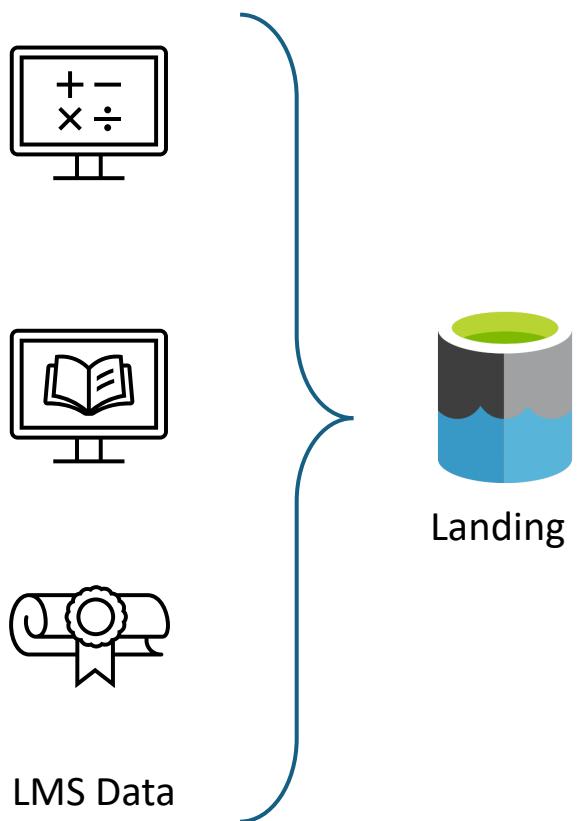
Dataset information

No	Column	Description
24	Feedback_Score	Average feedback score provided by the student for the course.
25	Parent_Involvement	Level of parent involvement in the student's education (e.g., High, Medium, Low).
26	Demographic_Group	Demographic group the student belongs to (e.g., Urban, Suburban, Rural).
27	Internet_Access	Whether the student has access to the internet at home (Yes/No).
28	Learning_Disabilities	Any learning disabilities the student may have.
29	Preferred_Learning_Style	Student's preferred learning style (e.g., Visual, Auditory, Kinesthetic).
30	Language_Proficiency	Proficiency level in the language of instruction (e.g., Beginner, Intermediate, Advanced).
31	Participation_Rate	Percentage of active participation in class activities.

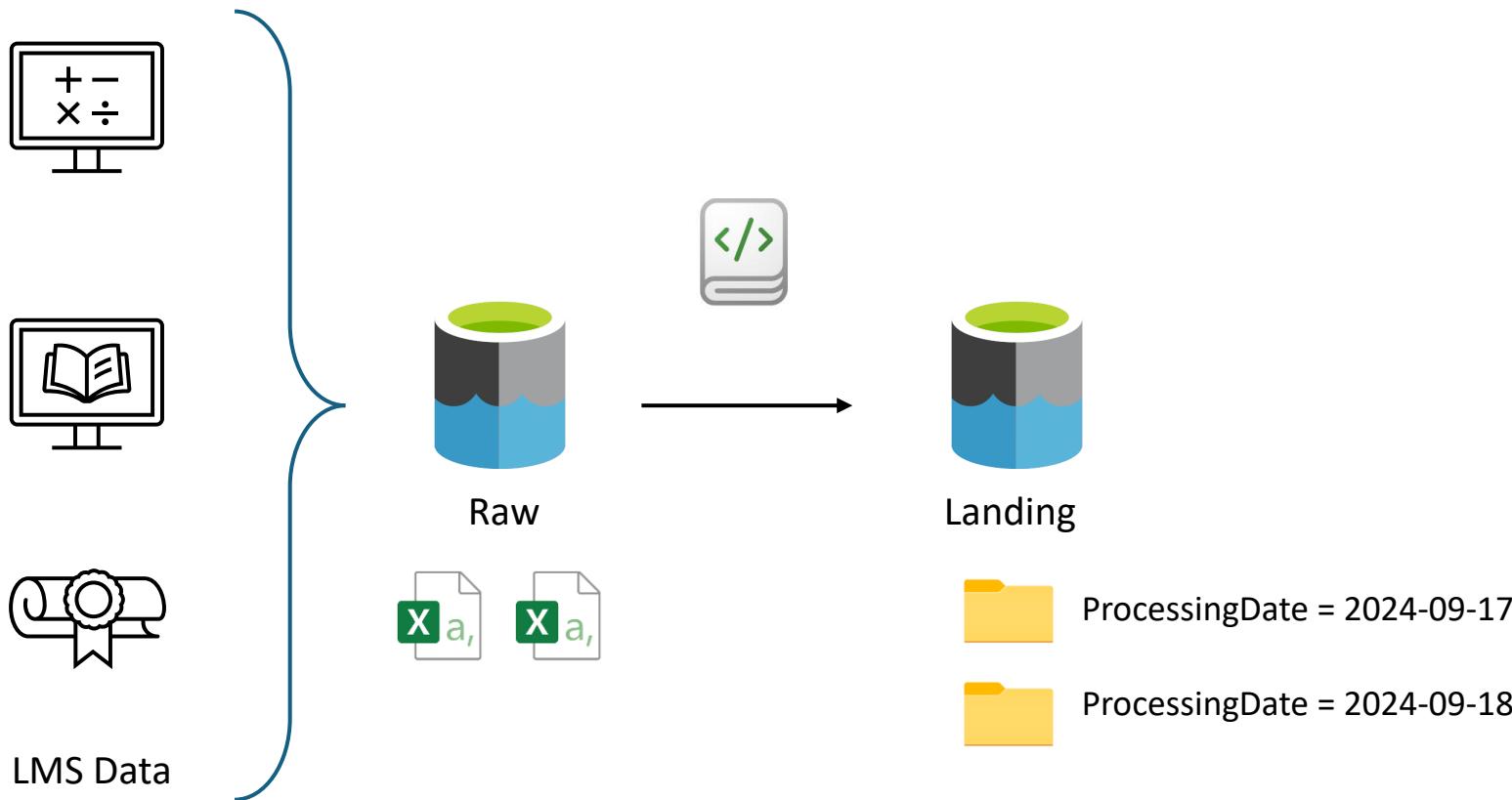
Fabric Project Architecture



Raw to landing layer



Raw to landing layer

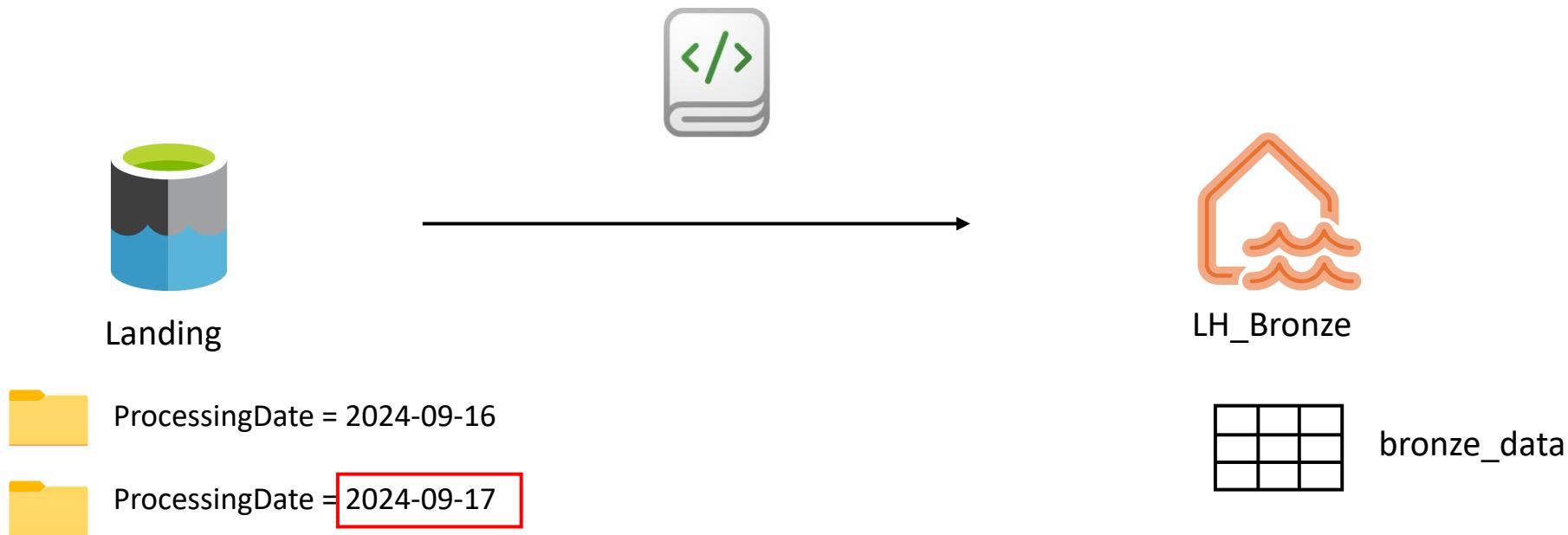


Steps for Raw to landing layer

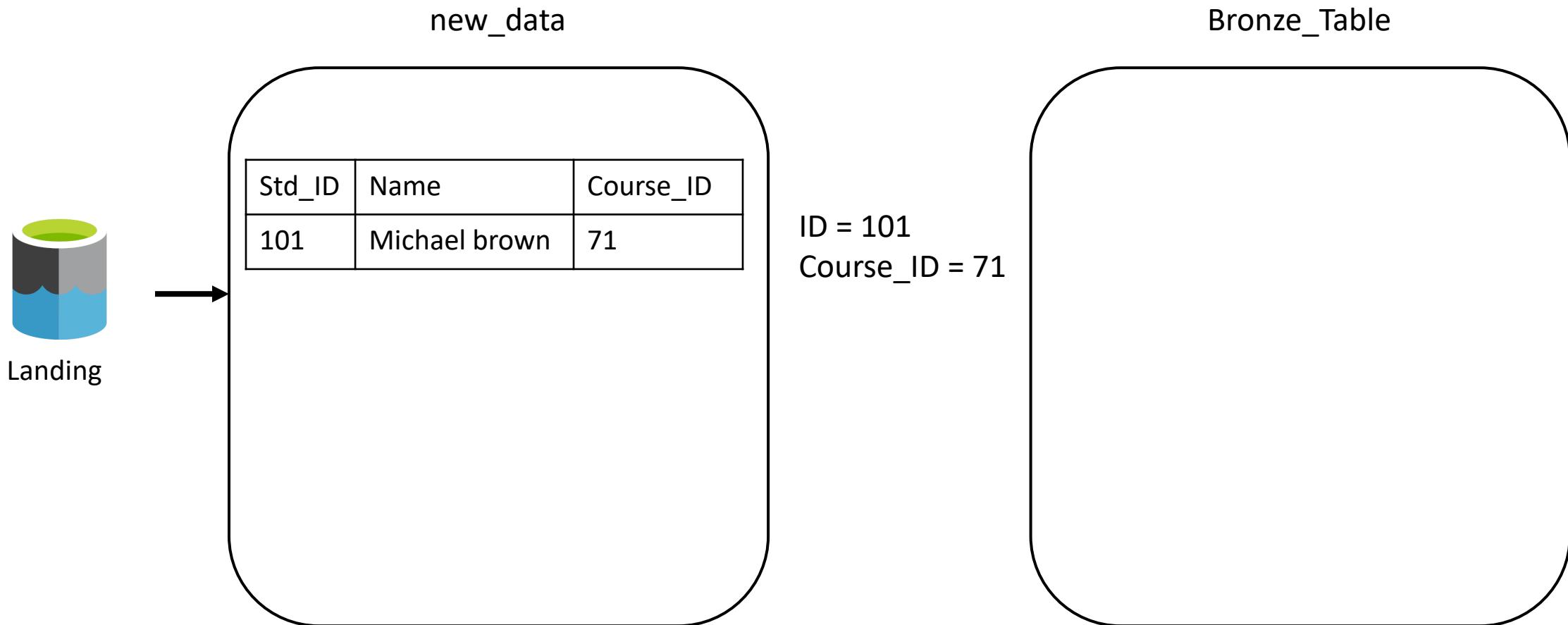
1. Data comes to Raw layer containing format of 'LMS_YYYY-MM-DD'
2. Only 1 file will be generated per day
3. We need to write the arrived data to Landing zone
4. While writing we need to partition the data based on today (E.g. 2024-09-17) as Processed_Date
5. This Processed_Date column will be used for partitioning the data

Landing to Bronze layer

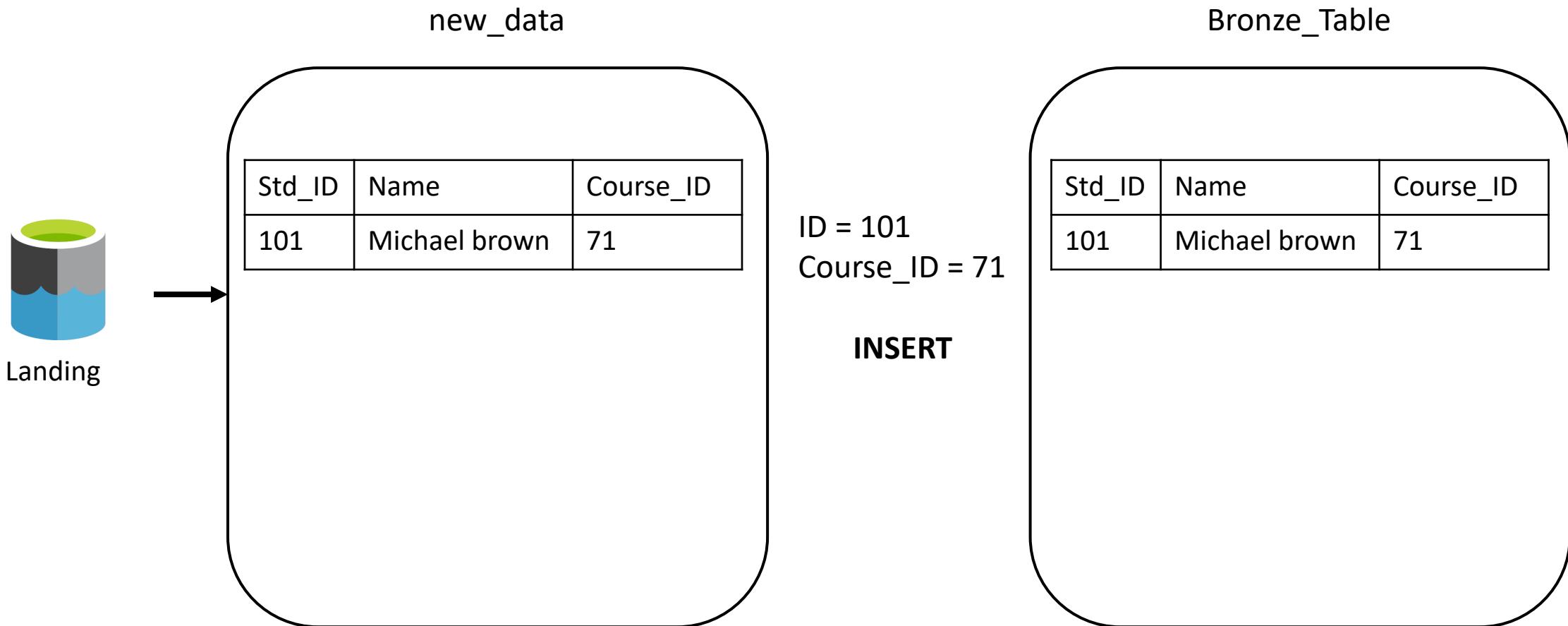
Today's Date = 2024-09-17



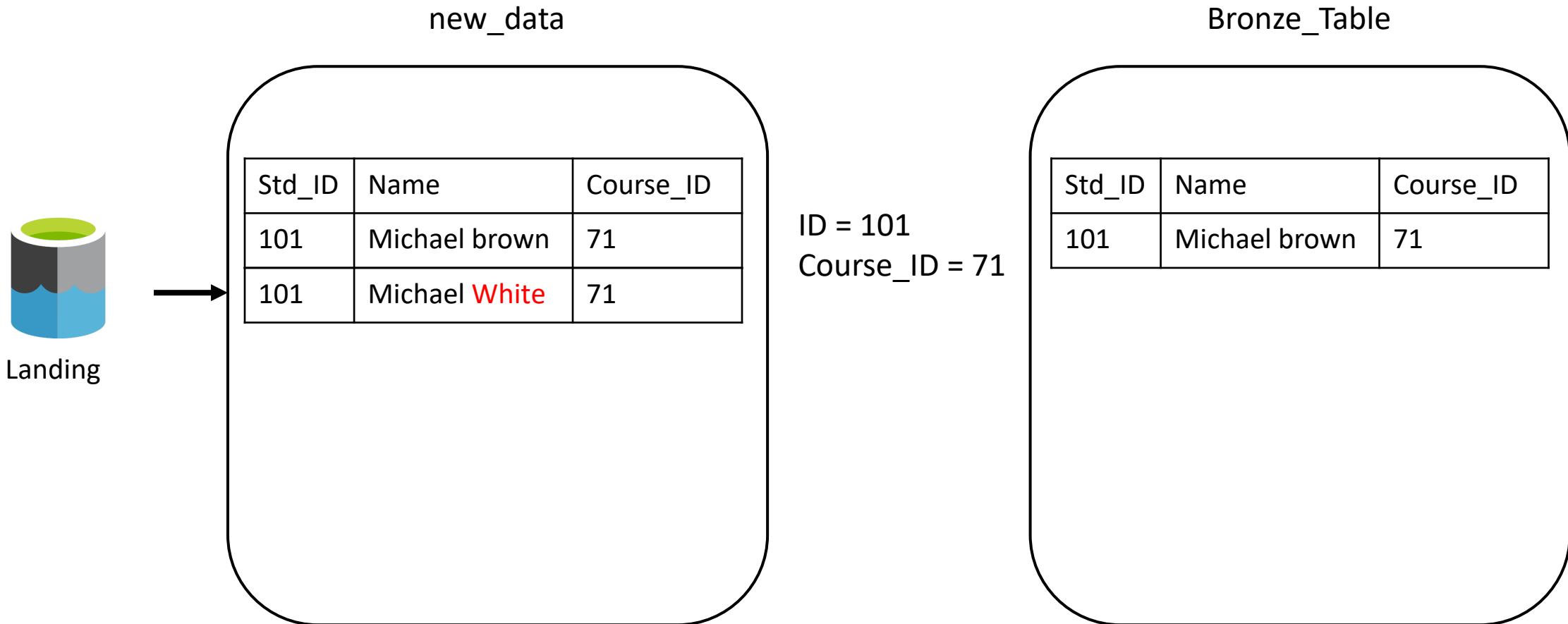
Understanding UPSERT to bronze



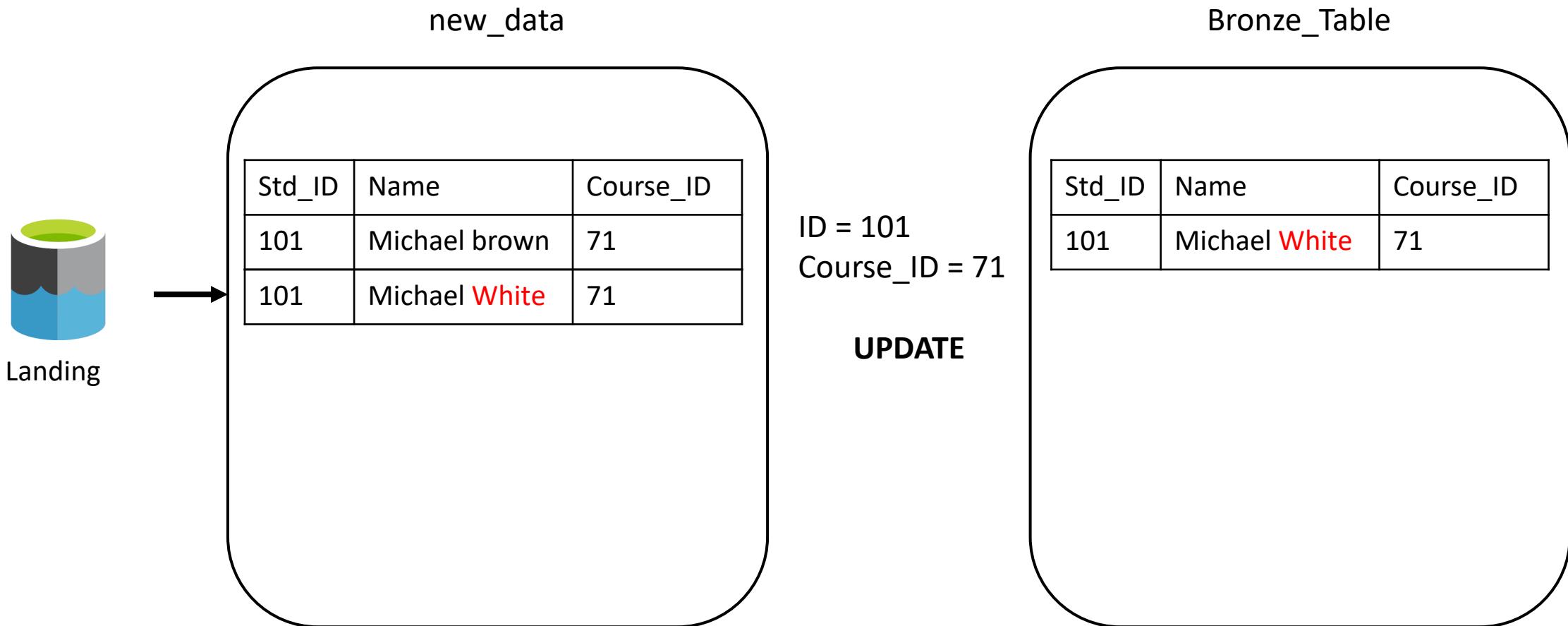
Understanding UPSERT to bronze



Understanding UPSERT to bronze

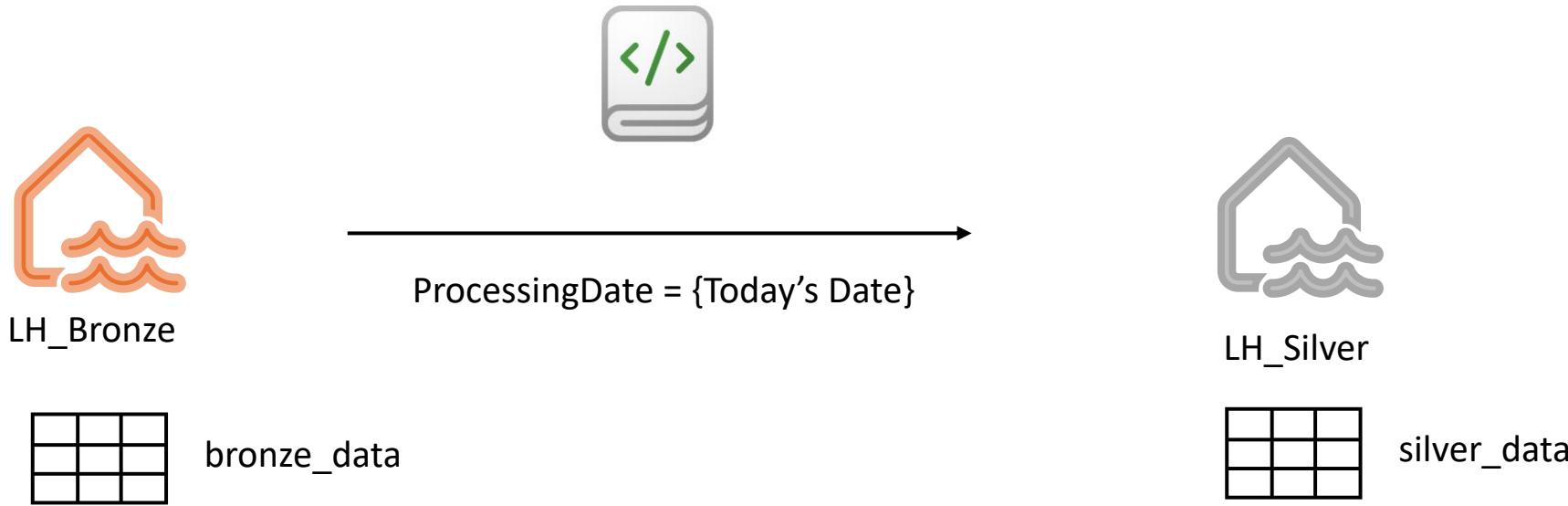


Understanding UPSERT to bronze



Bronze to Silver layer

Today's Date = 2024-09-17



Bronze to Silver layer

Data Cleaning

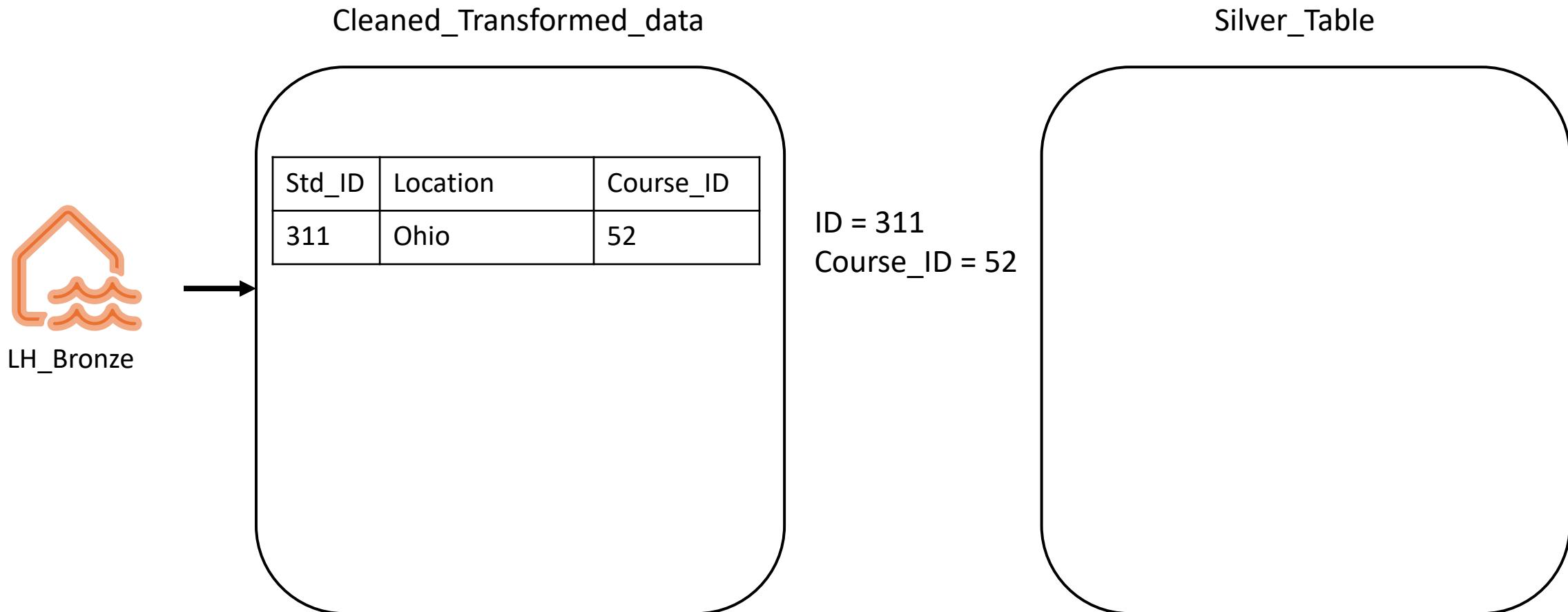
1. Handle duplicates
2. Handle Missing or NULL values
 1. Delete rows for missing critical values
 2. Fill rows with default values for other data
3. Standardize date formats
4. Check for logical consistency

Bronze to Silver layer

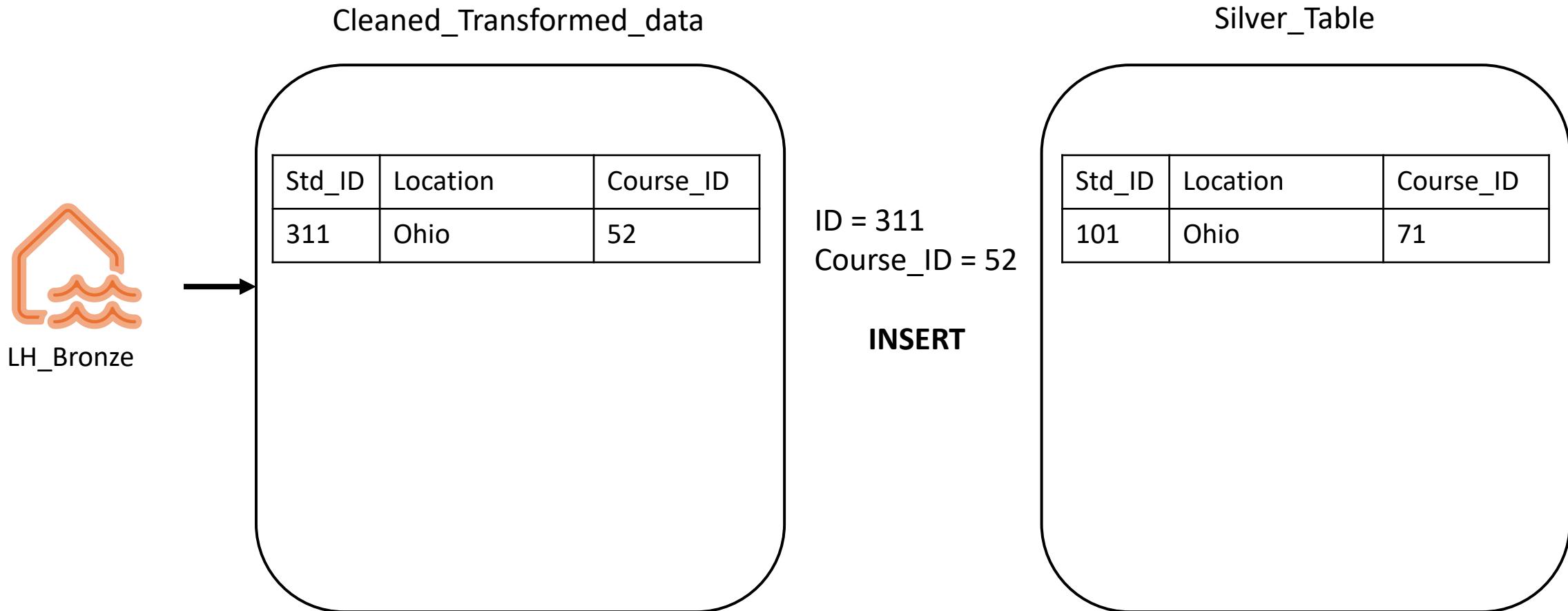
Business transformations

Column Name	Logic
Completion_Time_Days	Completion_date – Enrollment_Date
Performance_Score	$(Quiz_Average_Score * 0.2) + (Assignment_Average_score) * 0.2 + (Project_Score) * 0.1$
Course_Completion_Rate	If completion_time_Days <=90 then “On-time” Else “Delayed”

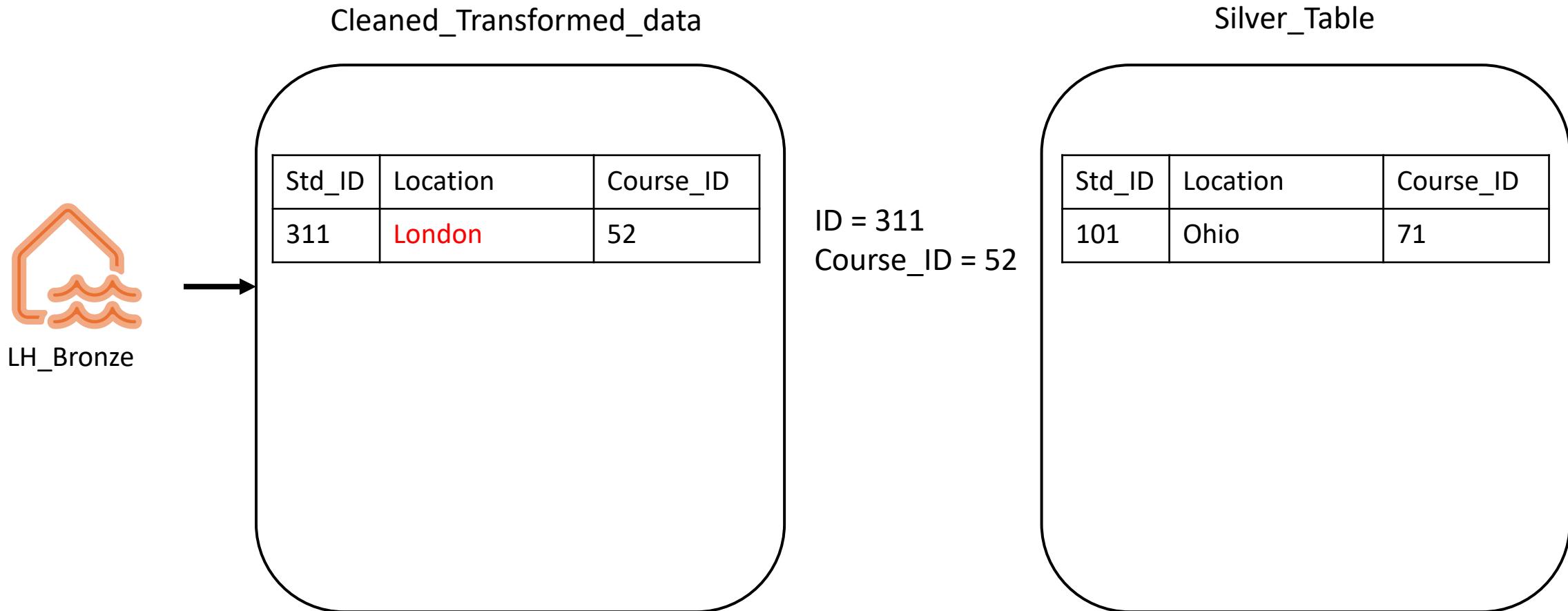
Understanding UPSERT to Silver



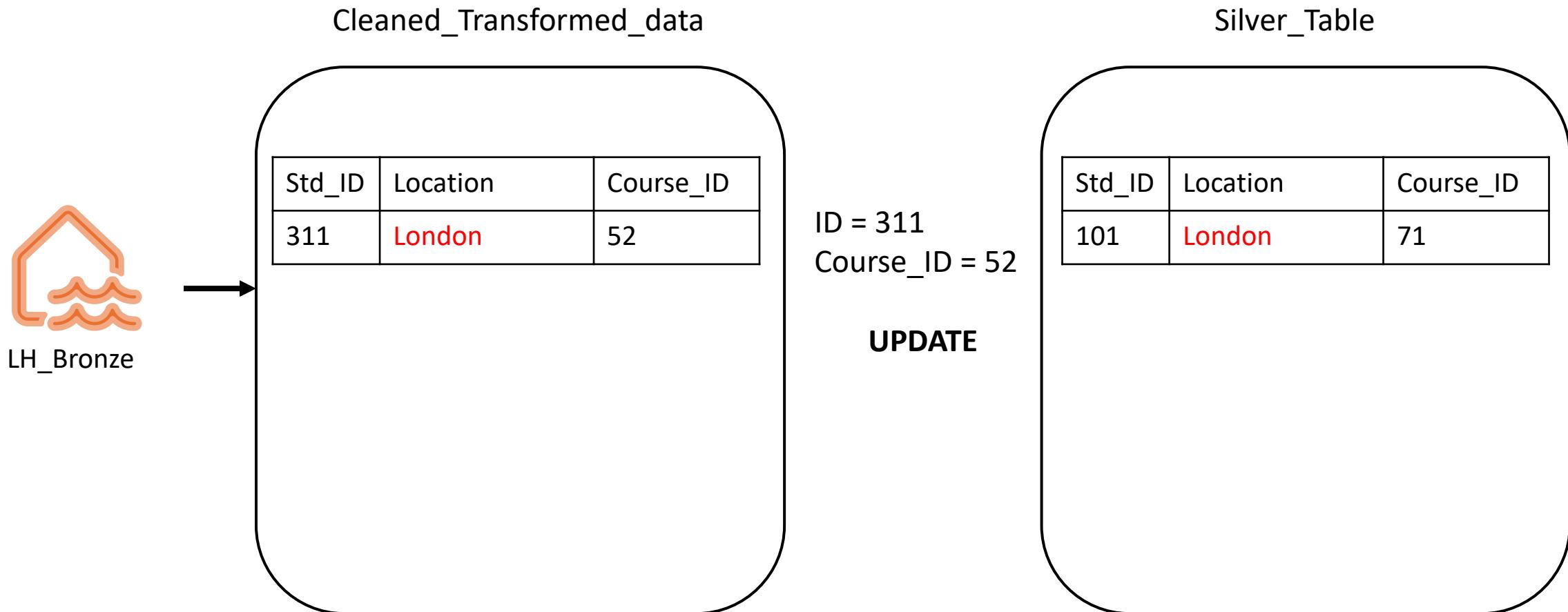
Understanding UPSERT to Silver



Understanding UPSERT to Silver

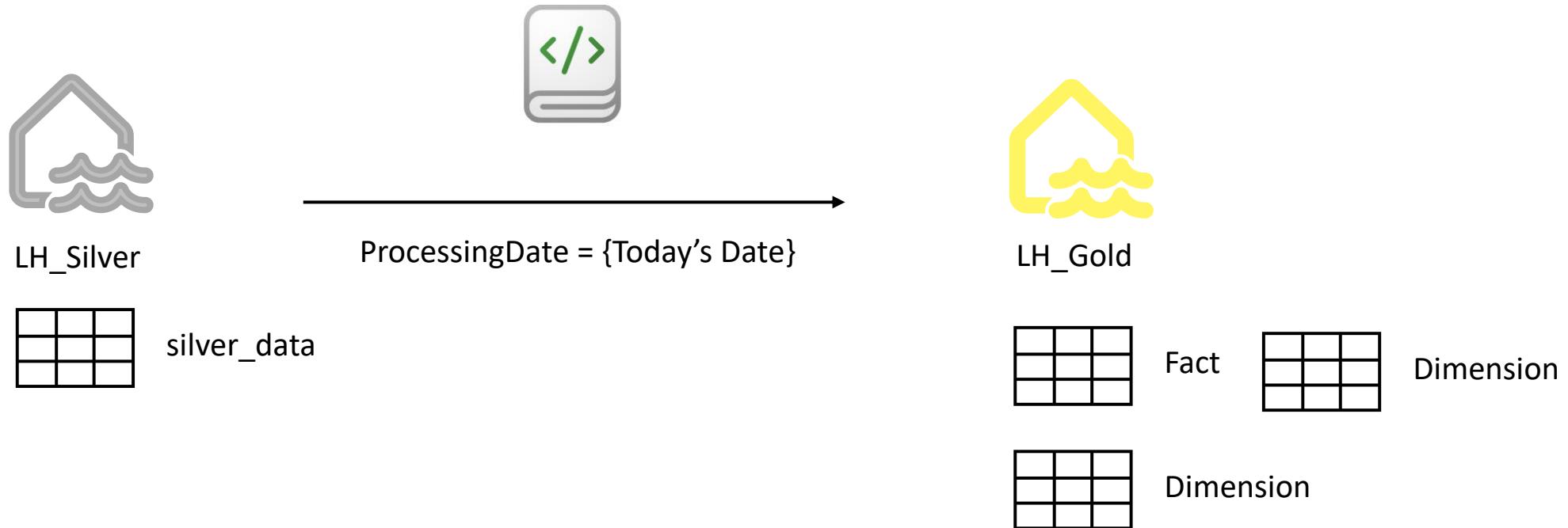


Understanding UPSERT to Silver

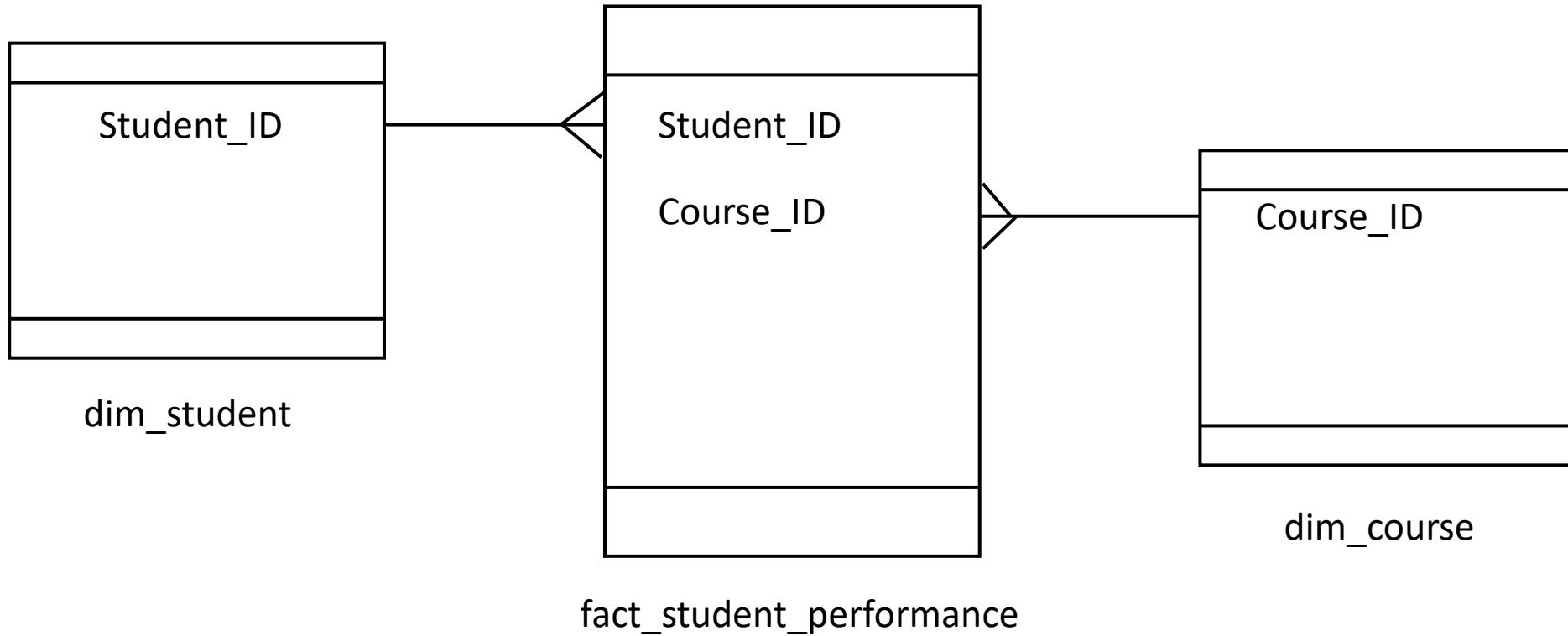


Silver to Gold layer

Today's Date = 2024-09-17

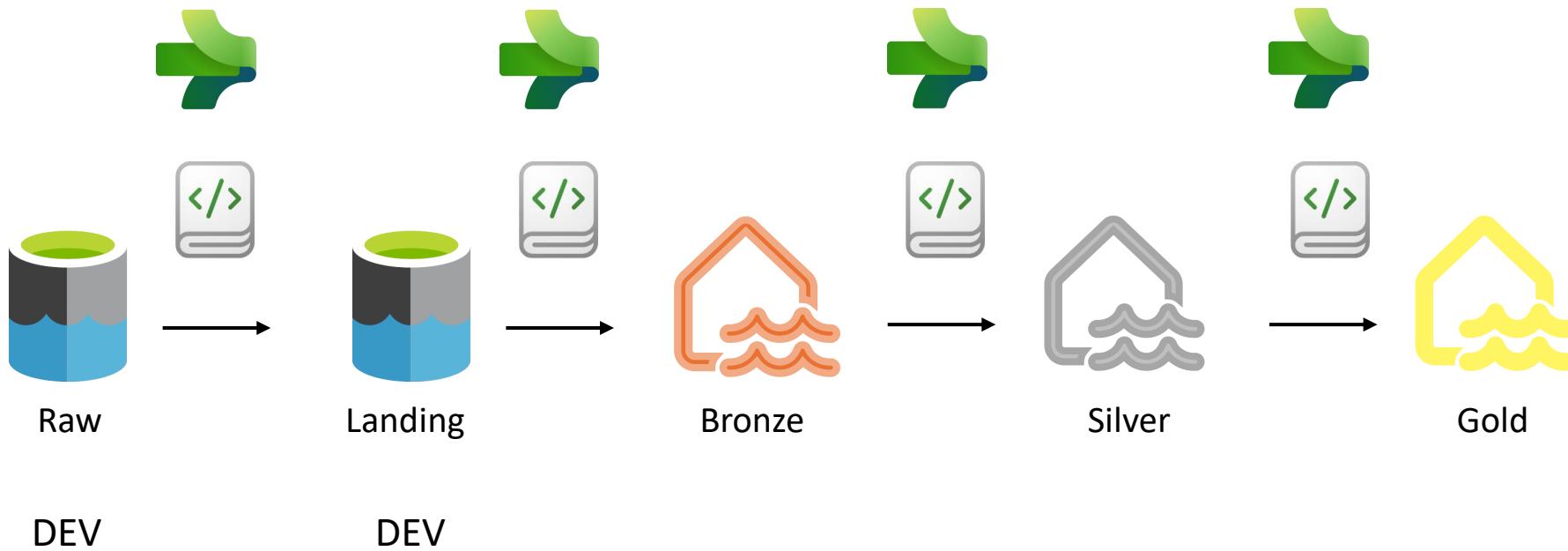


Data modelling

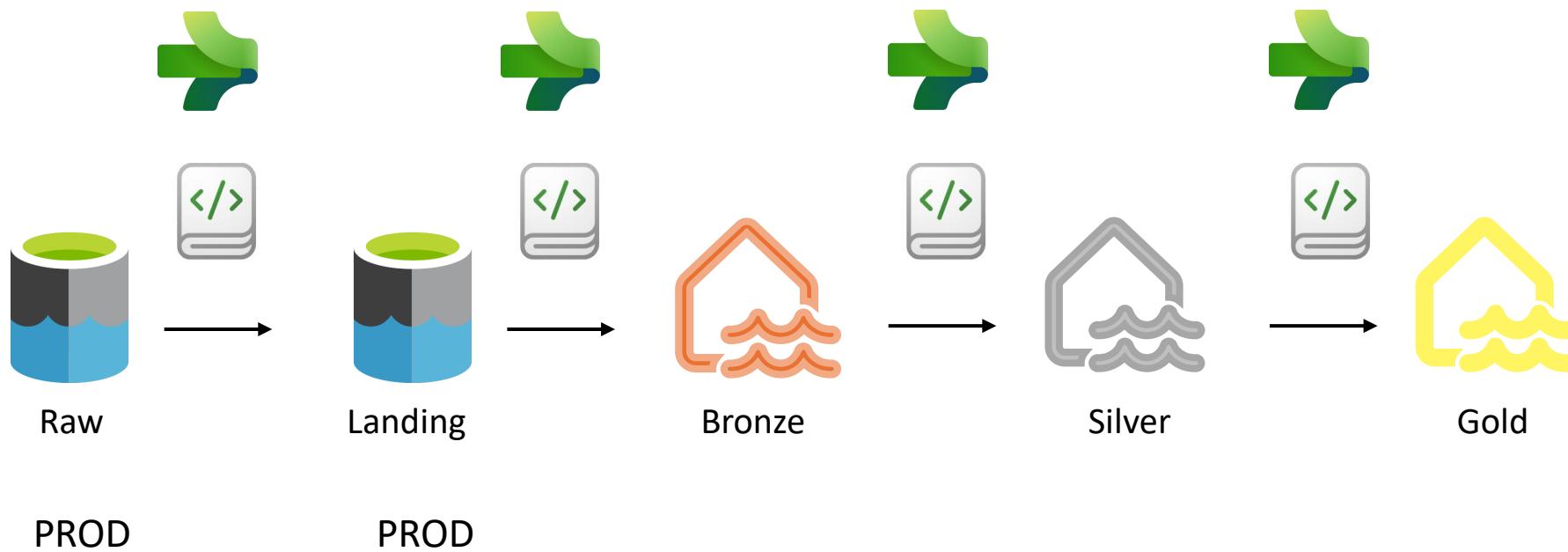


Git Integration

End to End flow



End to End flow



Supported items for Git integration

- Data pipelines
- Lakehouse
- Notebooks
- Paginated reports
- Reports (except reports connected to semantic models hosted in Azure Analysis Services, SQL Server Analysis Services or reports exported by Power BI Desktop that depend on semantic models hosted in MyWorkspace)
- Semantic models (except push datasets, live connections to Analysis Services, model v1).
- Spark Job Definitions
- Spark environment
- Warehouses

Limitations of enabling Git

- Currently, only Git in Azure Repos with the *same tenant* as the Fabric tenant is supported.
- If the workspace and Git repo are in two different geographical regions, the tenant admin must enable cross-geo exports.
- Azure DevOps **on-prem** isn't supported.
- Sovereign clouds aren't supported.

Permissions in workspace

- **Admin:** Can perform any operation on the workspace, limited only by their Azure DevOps role.
- **Member/Contributor:** Once they connect to a workspace, a member/contributor can commit and update changes, depending on their Azure DevOps role. For actions related to the workspace connection (for example, connect, disconnect, or switch branches) seek help from an Admin.
- **Viewer:** Can't perform any actions. The viewer can't see any Git related information in the workspace.

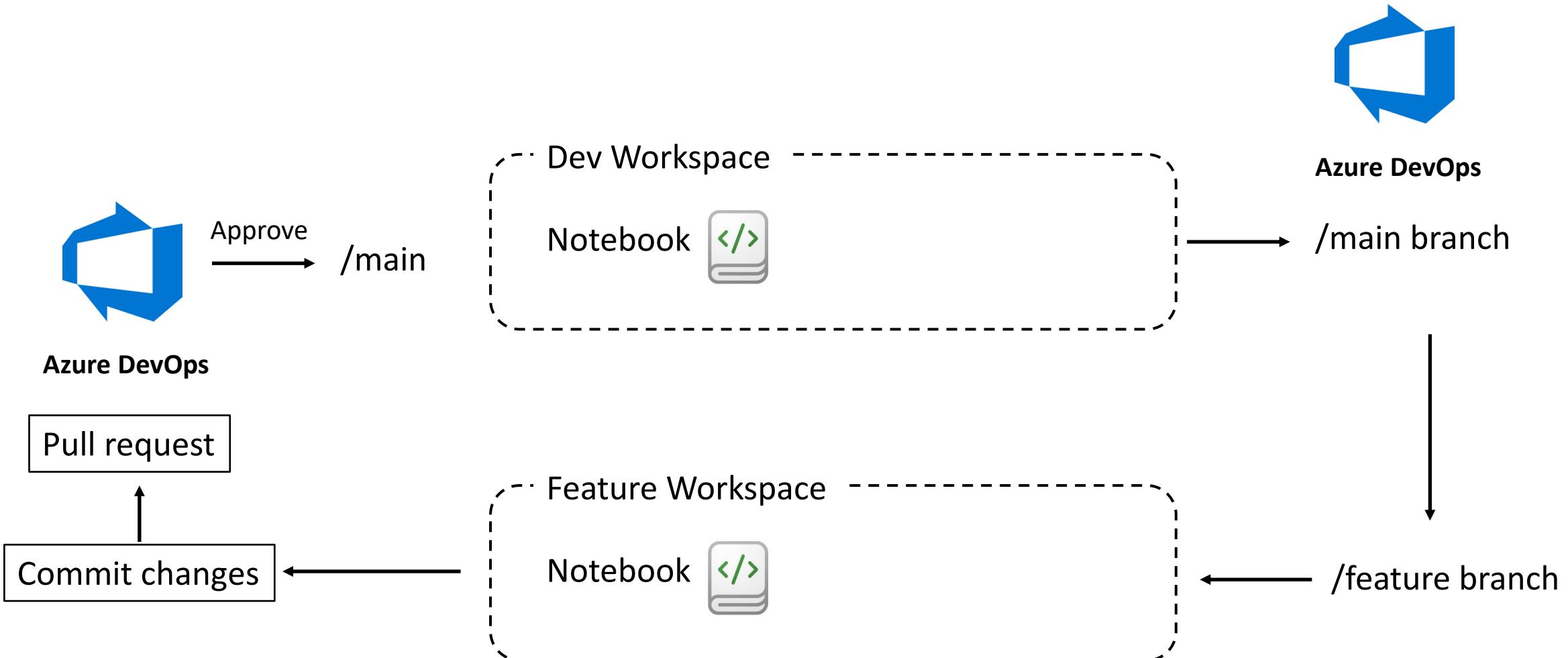
Operation	Workspace role	Git permissions
Connect workspace to Git repo	Admin	Read=Allow
Sync workspace with Git repo	Admin	Read=Allow
Disconnect workspace from Git repo	Admin	No permissions are needed
Switch branch in the workspace (or any change in connection setting)	Admin	Read=Allow (in target repo/directory/branch)
View Git connection details	Admin, Member, Contributor	Read or None
See workspace 'Git status'	Admin, Member, Contributor	Read=Allow

Operation	Workspace role	Git permissions
Update from Git	<p>All of the following:</p> <p>Contributor in the workspace (WRITE permission on all items)</p> <p>Owner of the item (if the tenant switch blocks updates for nonowners)</p> <p>BUILD on external dependencies (where applicable)</p>	Read=Allow
Commit workspace changes to Git	<p>All of the following:</p> <p>Contributor in the workspace (WRITE permission on all items)</p> <p>Owner of the item (if the tenant switch blocks updates for nonowners)</p> <p>BUILD on external dependencies (where applicable)</p>	Read=Allow Contribute=Allow branch policy should allow direct commit
Create new Git branch from within Fabric	Admin	Role=Write Create branch=Allow

Considerations with Azure DevOps

- The Azure DevOps account must be registered to the same user that is using the Fabric workspace.
- Power BI Datasets connected to Analysis Services aren't supported at this time.

Continuous Integration



Changed items are listed with an icon indicating the status:

 new

 modified

 deleted

 conflict

Source control

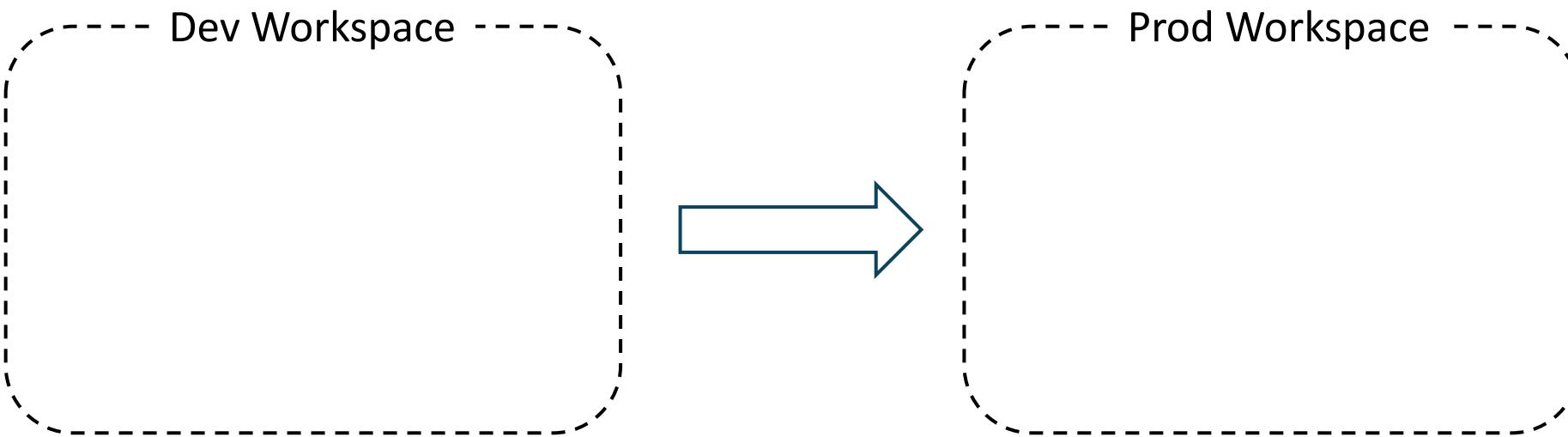
 TestGit1

Changes 3 • Updates 15 •

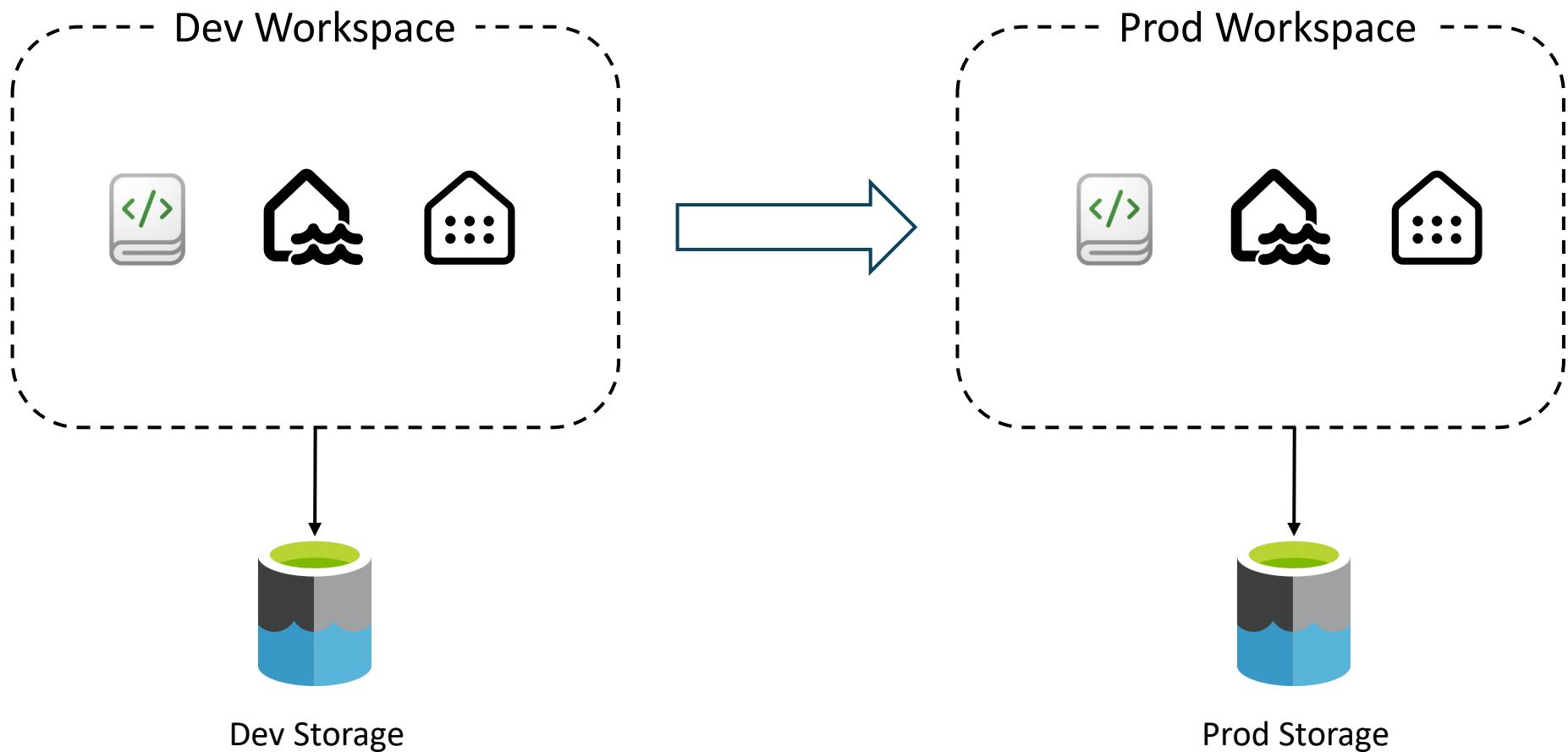
Item	Status
FamilyBM2M v3	
FamilyBM2M v3	
ParamsUsageV3	
Calcul Ent from DF	
Calcul Ent from DF	

Deployment pipeline (Continuous deployment)

Continuous Deployment



Continuous Deployment



Deployment rules

Change data source after deployment. Prod stage might be pointing to different datasets

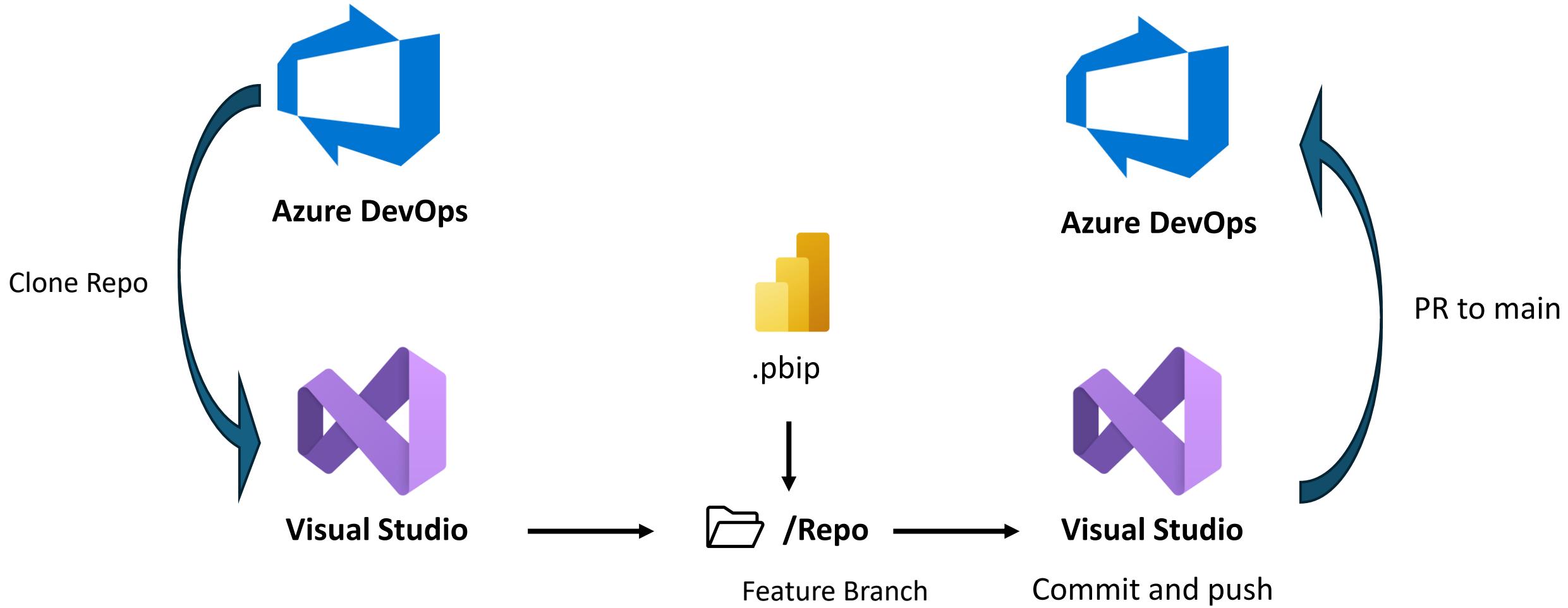
Item	Data source rule	Parameter rule	Default lakehouse rule	Details
Dataflow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Use to determine the values of the data sources or parameters for a specific dataflow.
Semantic model	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Use to determine the values of the data sources or parameters for a specific semantic model.
Datamart	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Use to determine the values of the data sources or parameters for a specific datamart.
Paginated report	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Defined for the data sources of each paginated report. Use to determine the data sources of the paginated report.
Notebook	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Use to determine the default lakehouse for a specific notebook.

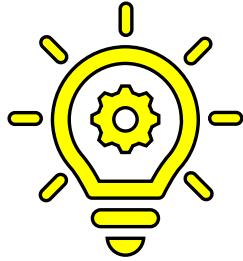
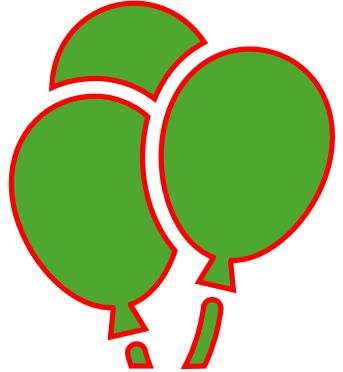
Supported Items for deployment

When you deploy content from one pipeline stage to another, the copied content can contain the following items:

- Data pipelines
- Dataflows Gen1
- Datamarts
- Lakehouse
- Notebooks
- Paginated reports
- Reports (based on supported semantic models)
- Spark environment
- Semantic models (except for DirectLake semantic models)
- Warehouses

Version control for Power BI items





Congratulations on completing the Course !

