Microsoft

DP-601: Implement a lakehouse with Microsoft Fabric

Presenter

Mohammed Arif, PhD

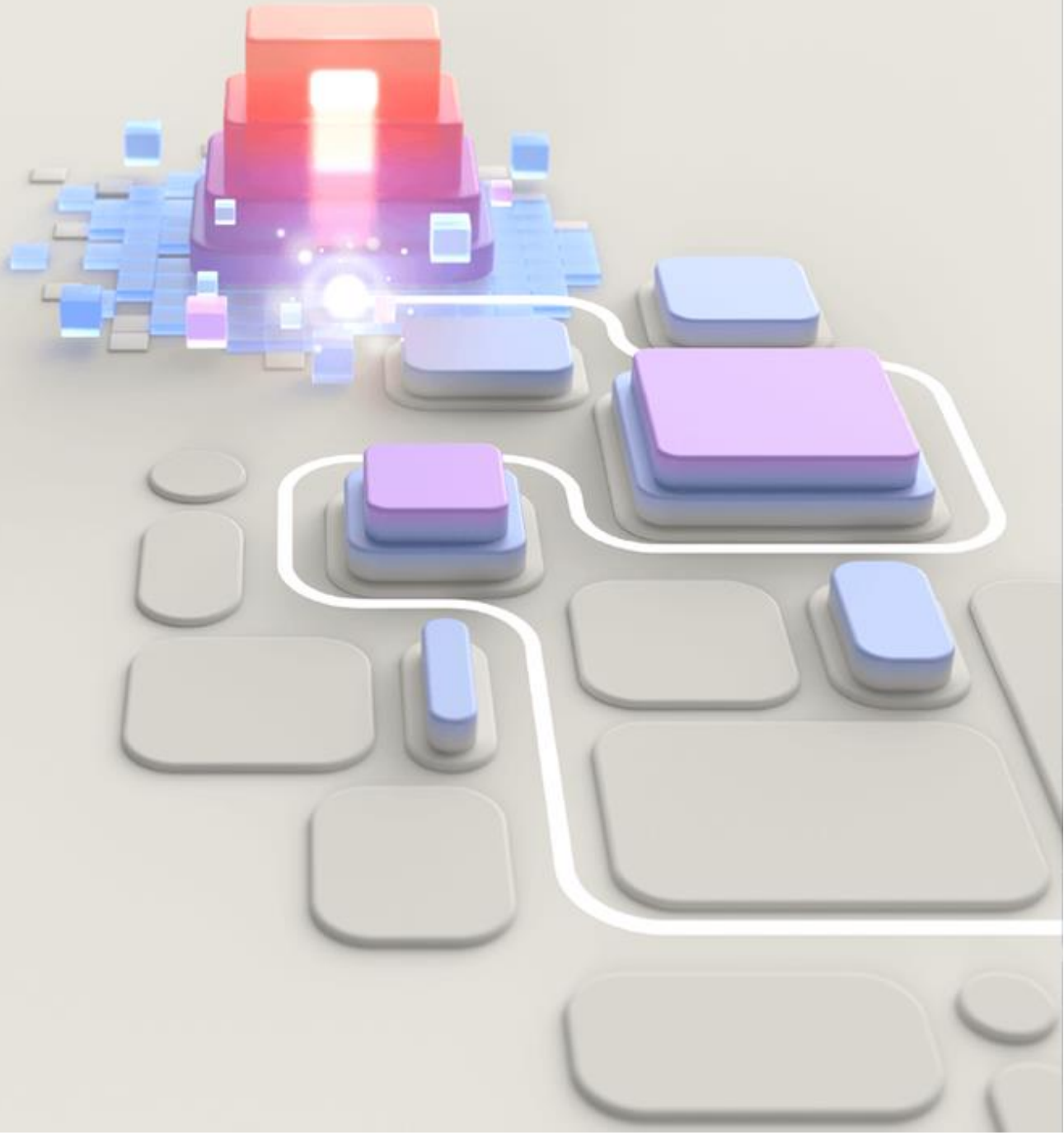
GenAI Architect & Data Scientist

Mohammed Arif has more than eighteen (18+) years of working experience in Information Communication and Technology (ICT) industry. The highlights of his career are more than nine (9) years of holding various senior management and/or C-Level and had six (6) years of international ICT consultancy exposure in various countries (APAC and Australia), specially on Big Data, Data Engineering, Machine Learning and AI arena.

He is also Certified Trainer for Microsoft & Cloudera.

/arifmazumder

Microsoft



https://arif.works/dp601

Explore end-to-end analytics with Microsoft Fabric

Learning objectives

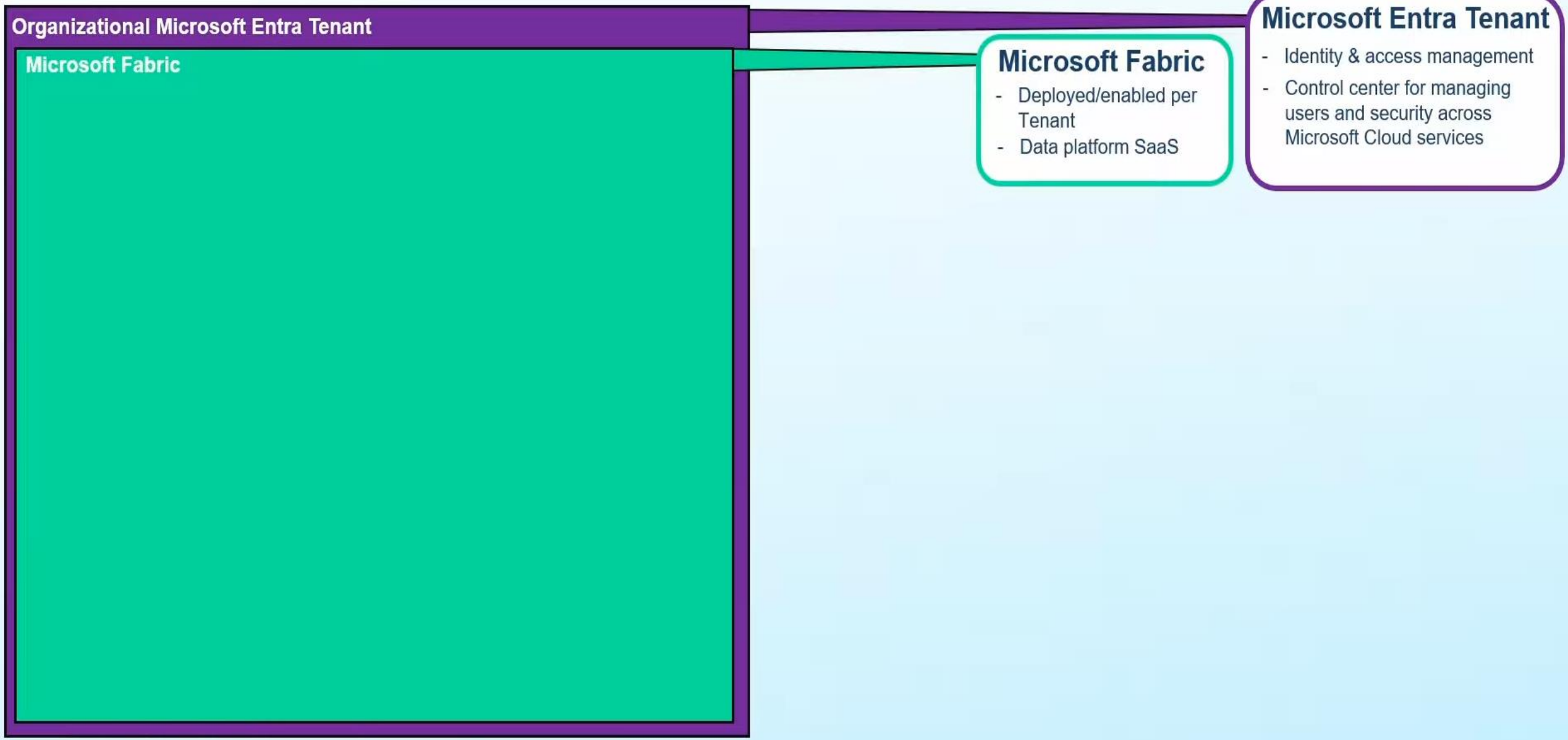
- Describe end-to-end analytics in Microsoft Fabric
- Understand data teams and roles that use Fabric
- Describe how to enable and use Fabric

But First ..

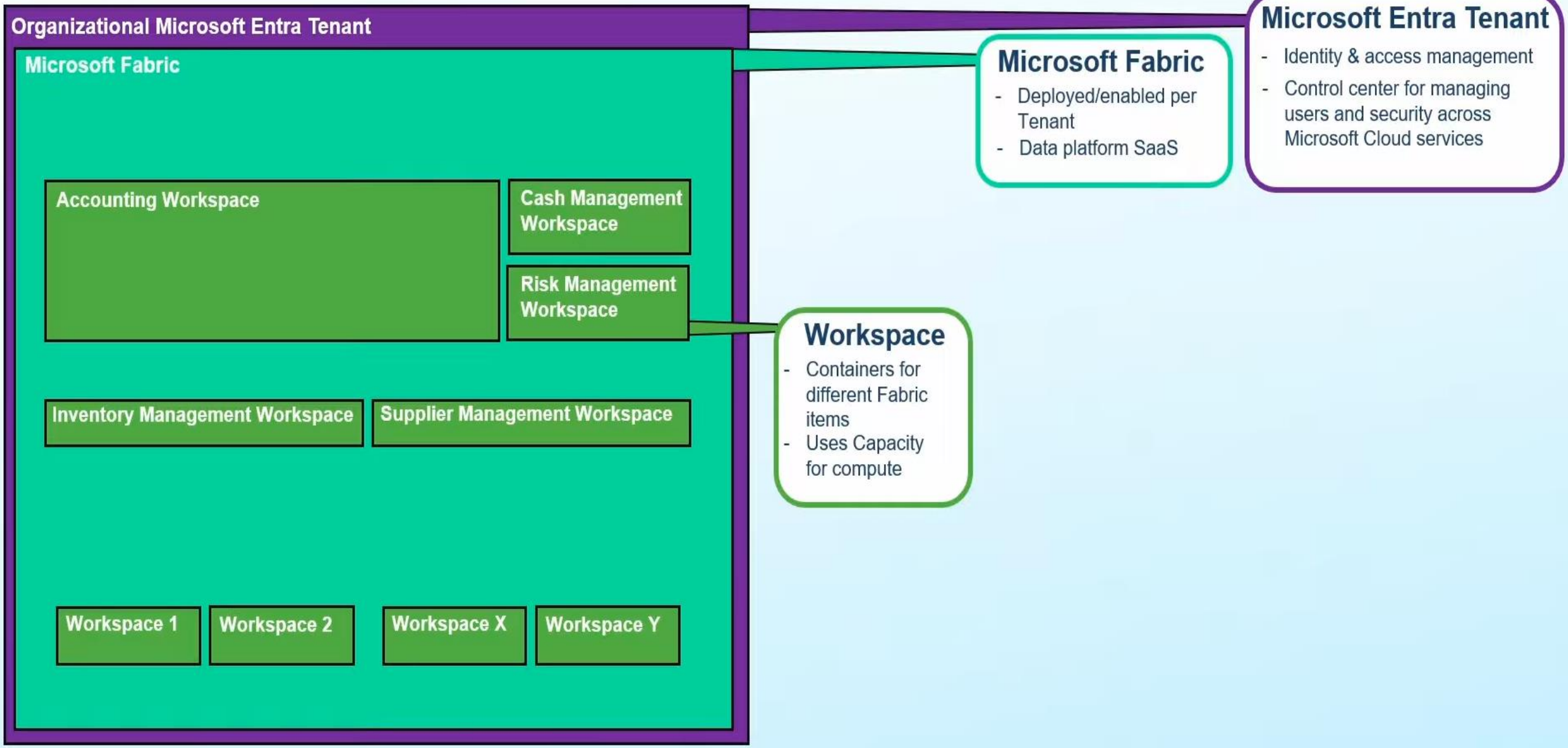
Microsoft Fabric Structure

Microsoft Entra Tenant

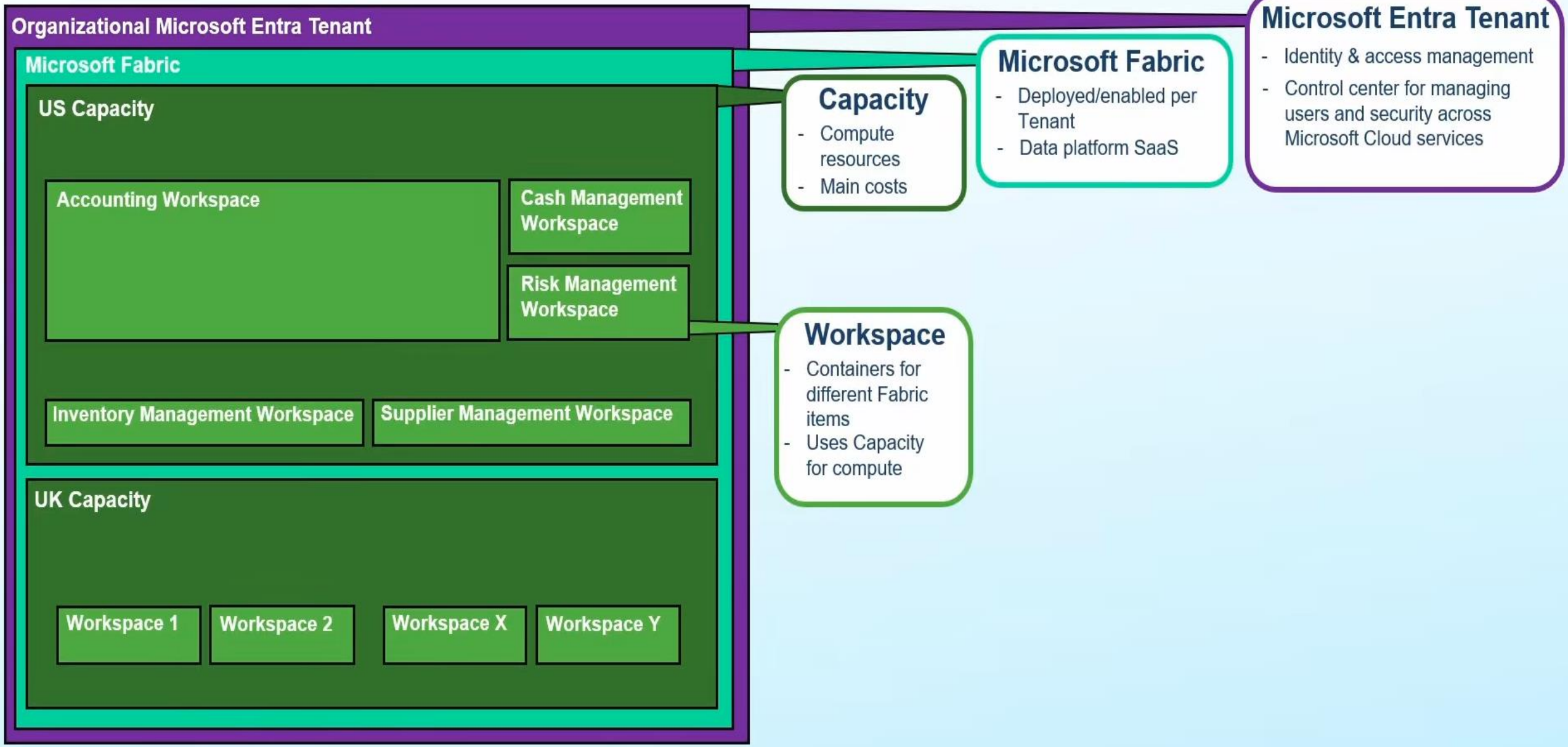
- Identity & access management
- Control center for managing users and security across Microsoft Cloud services



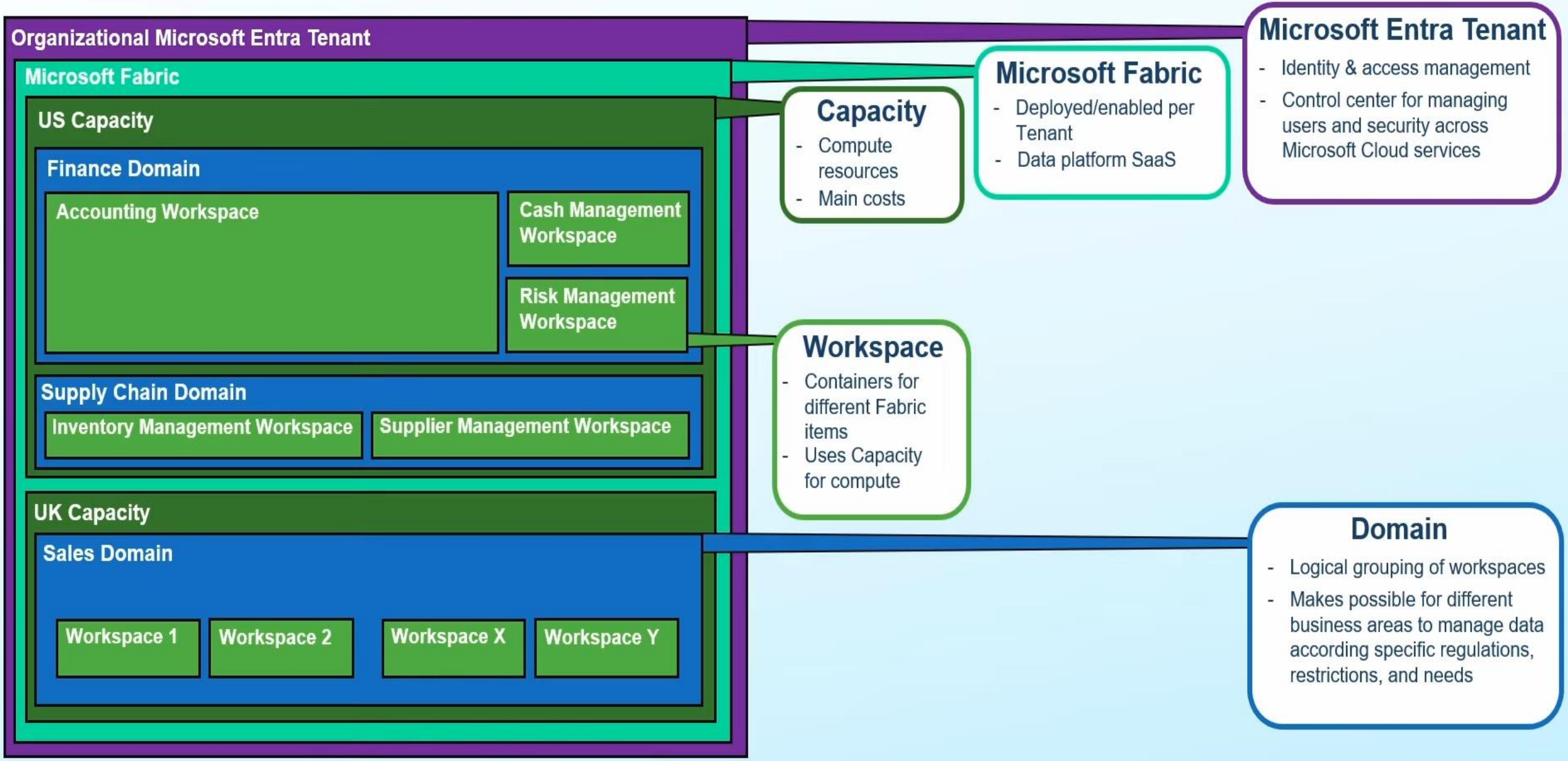
Microsoft Fabric Structure



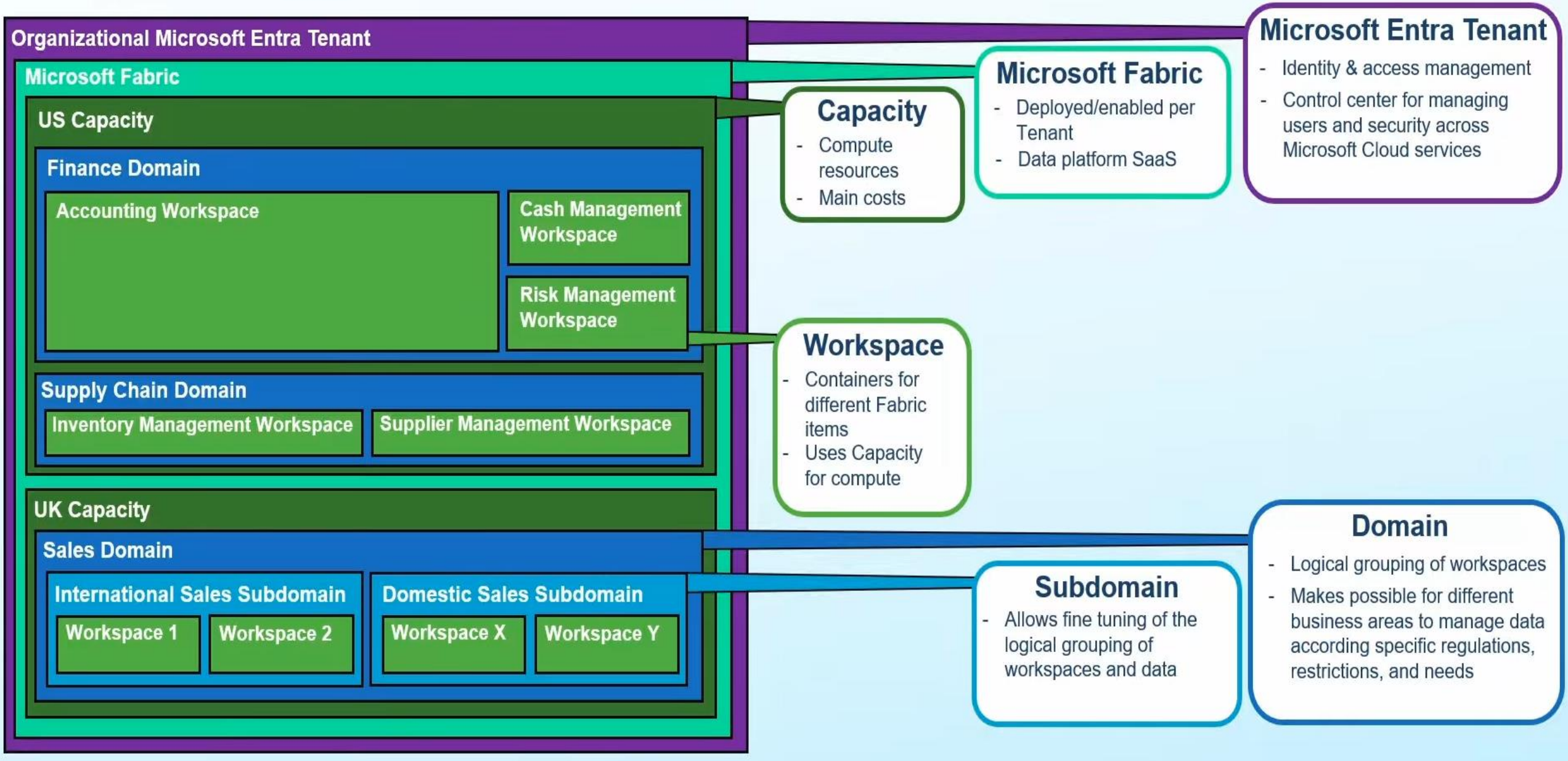
Microsoft Fabric Structure



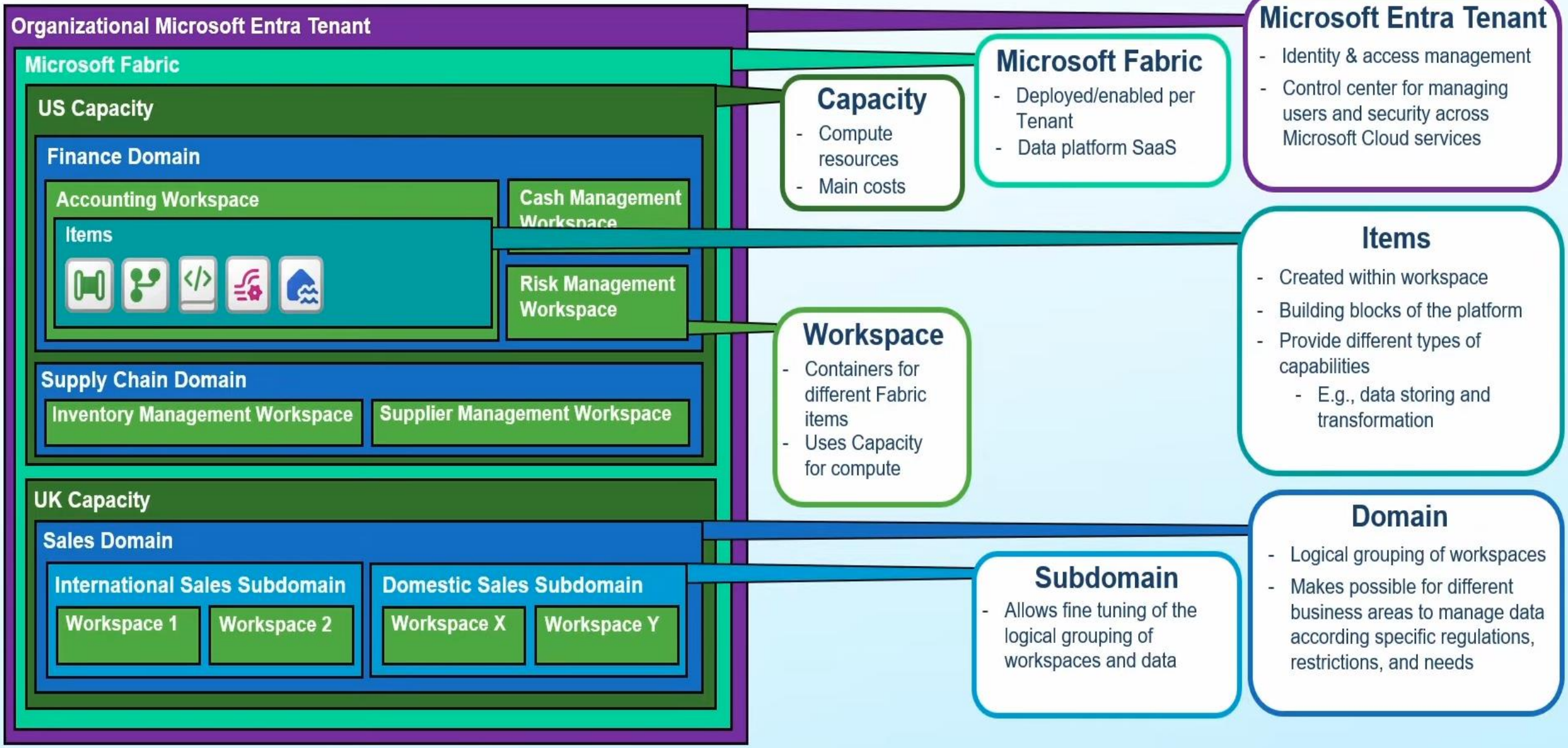
Microsoft Fabric Structure



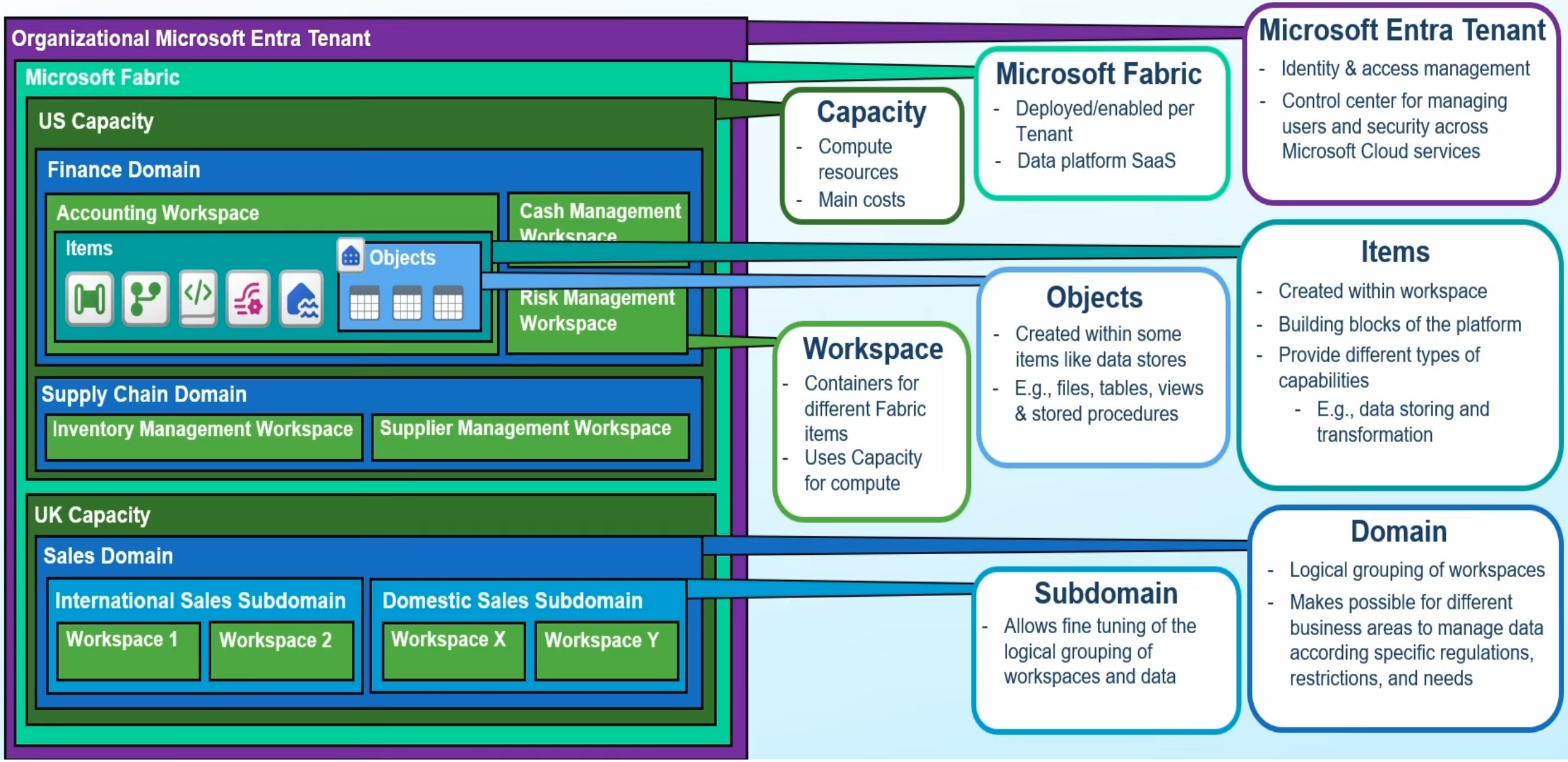
Microsoft Fabric Structure



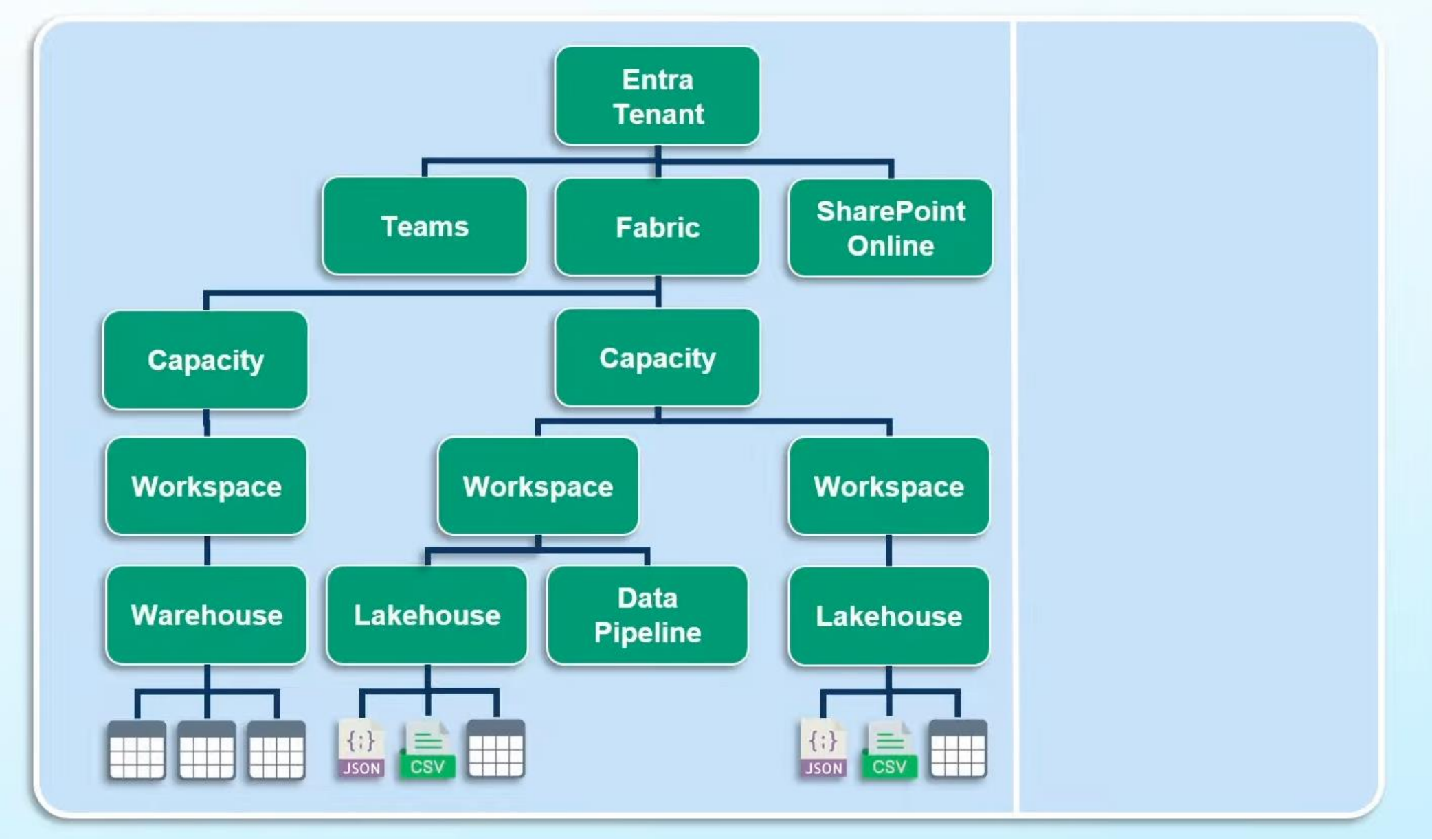
Microsoft Fabric Structure



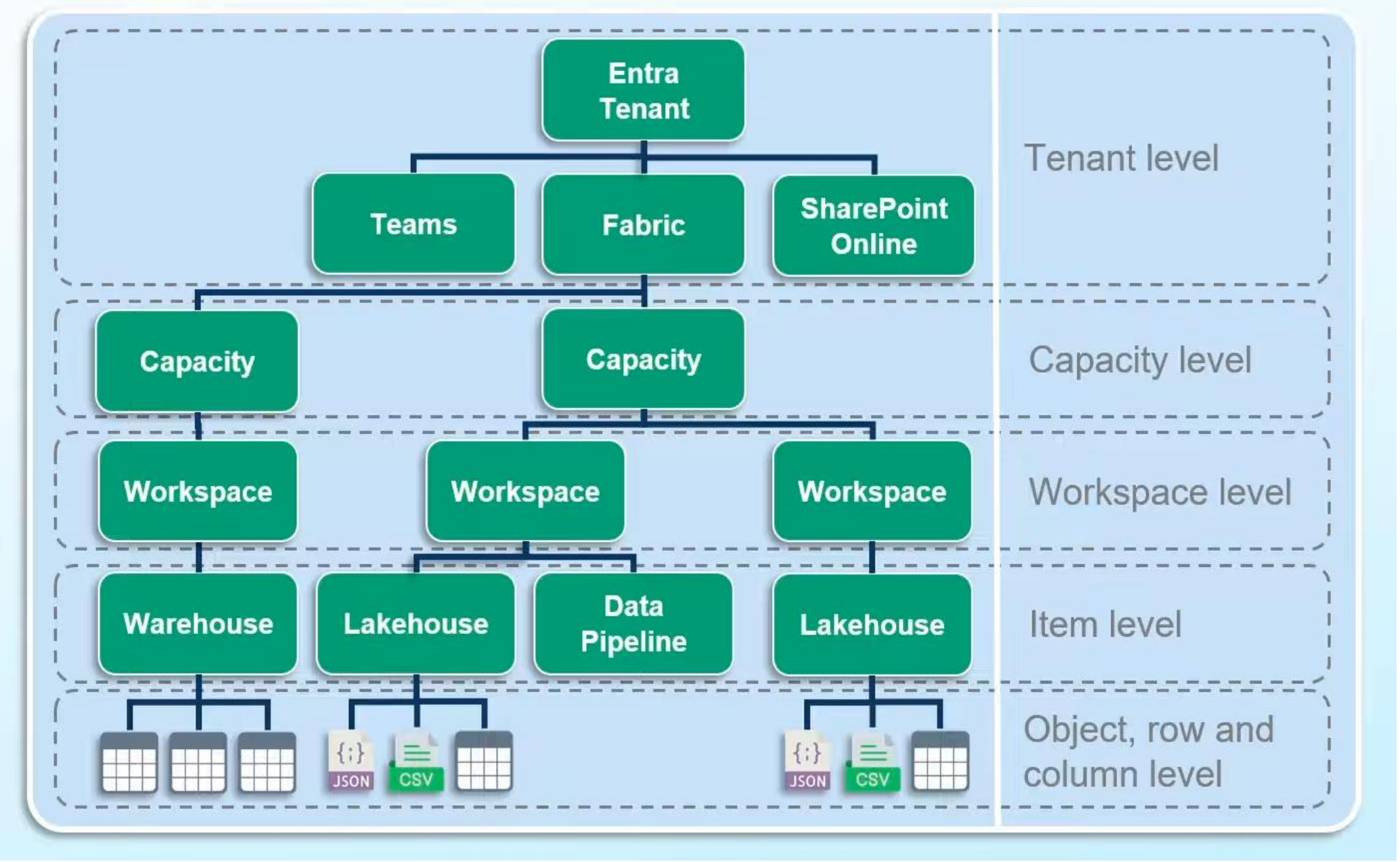
Microsoft Fabric Structure



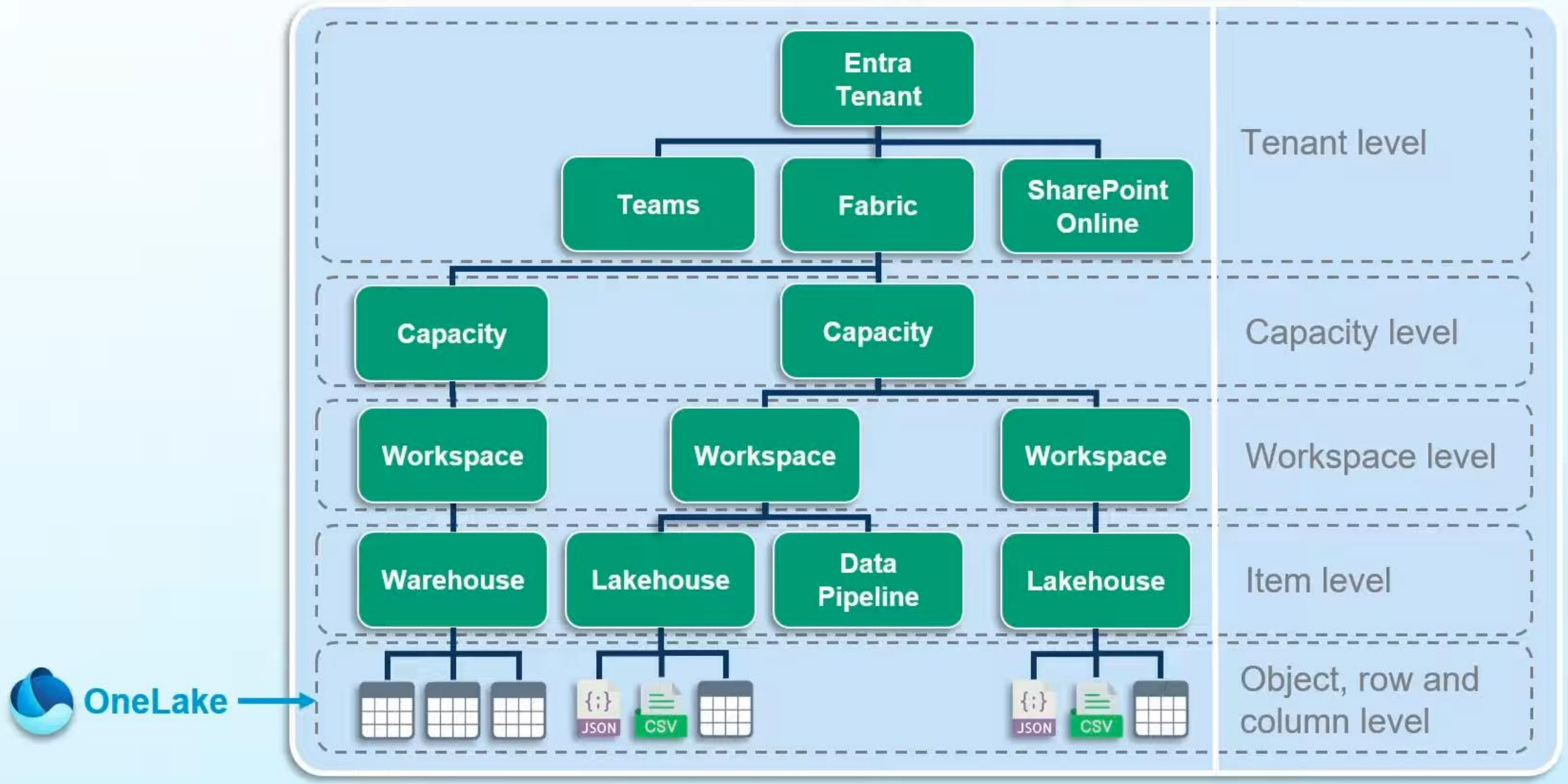
Microsoft Fabric Structure



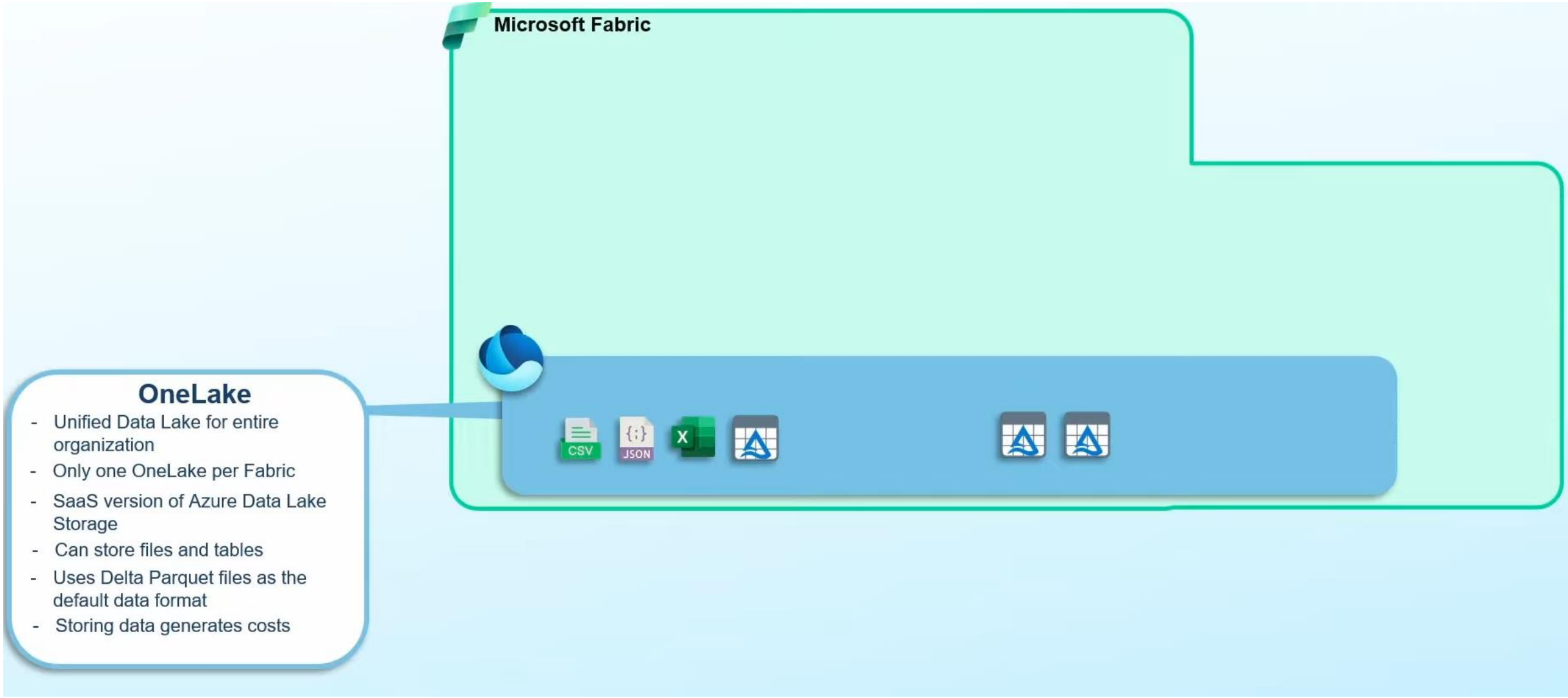
Microsoft Fabric Structure

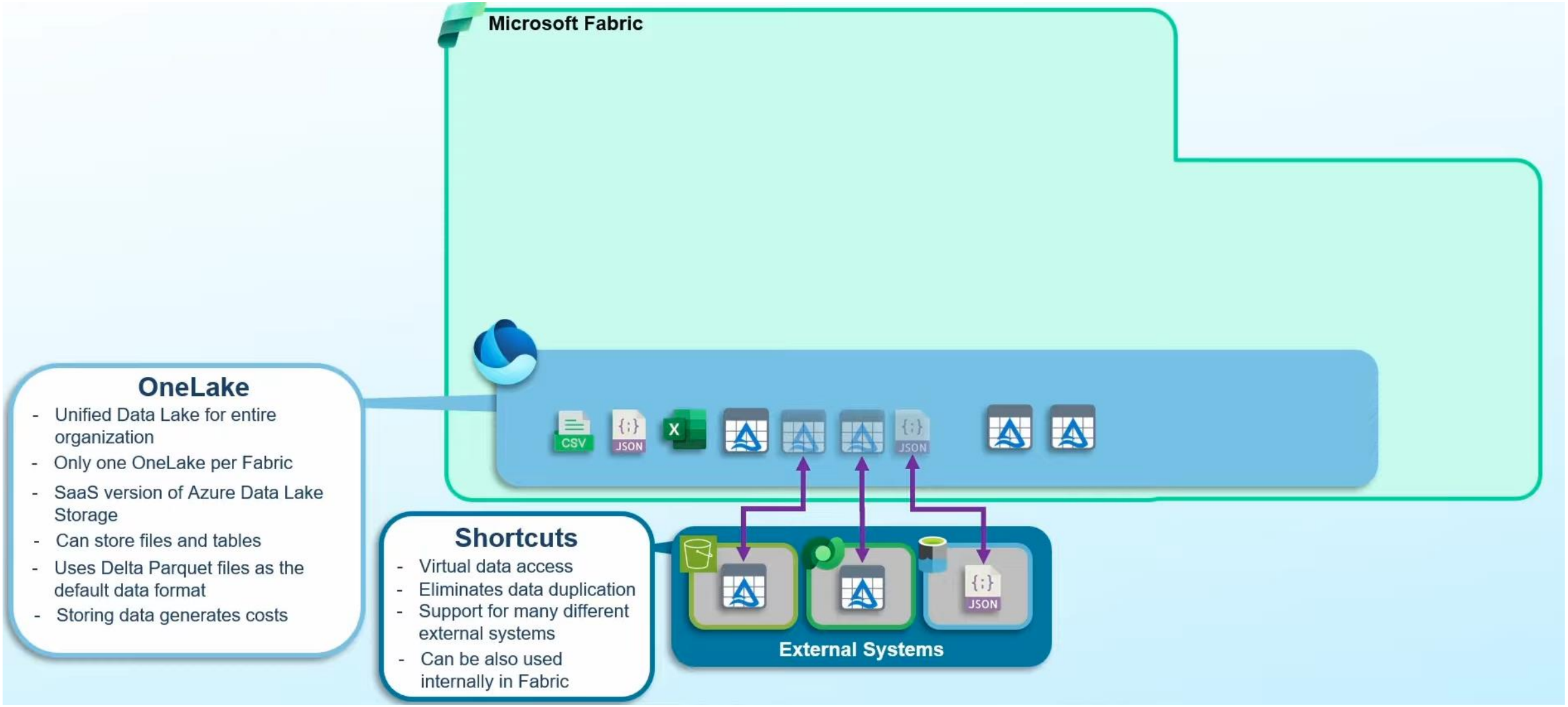


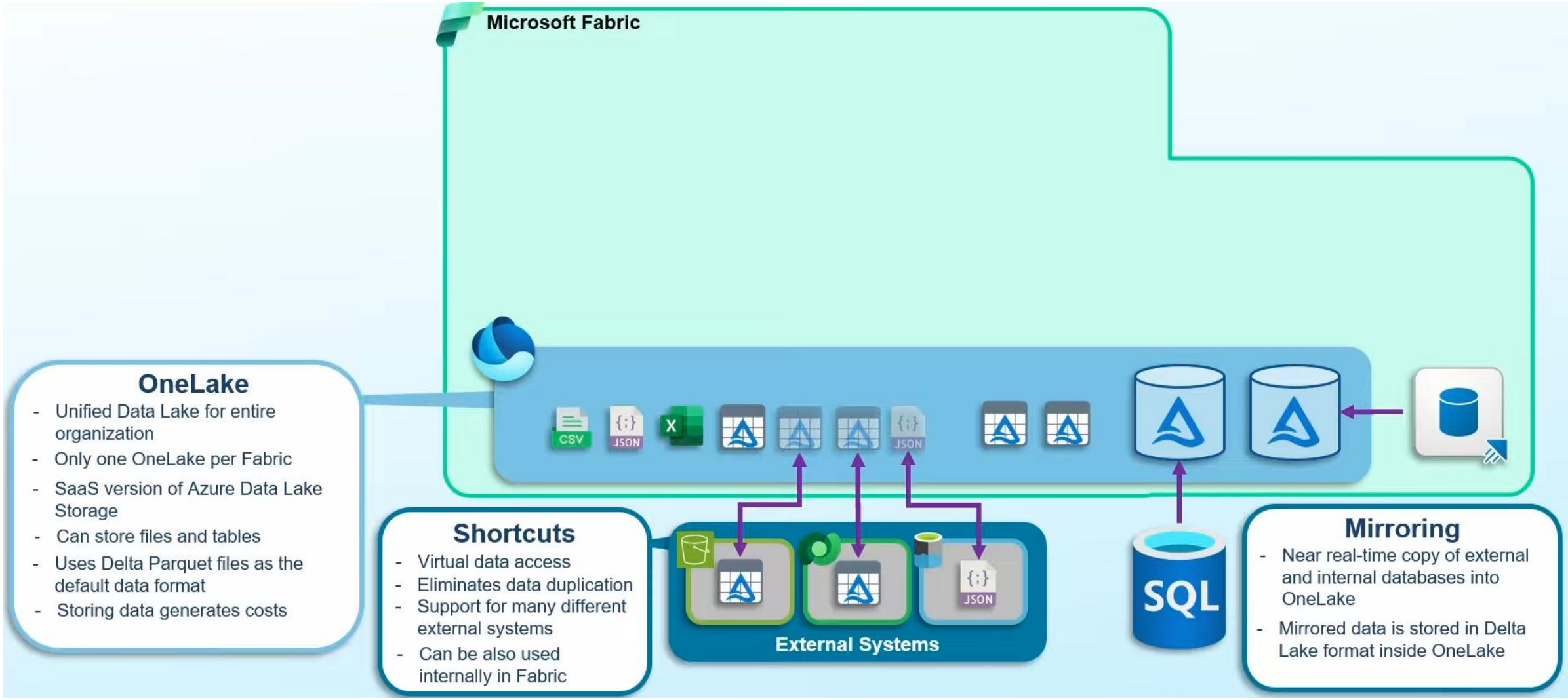
Microsoft Fabric Structure



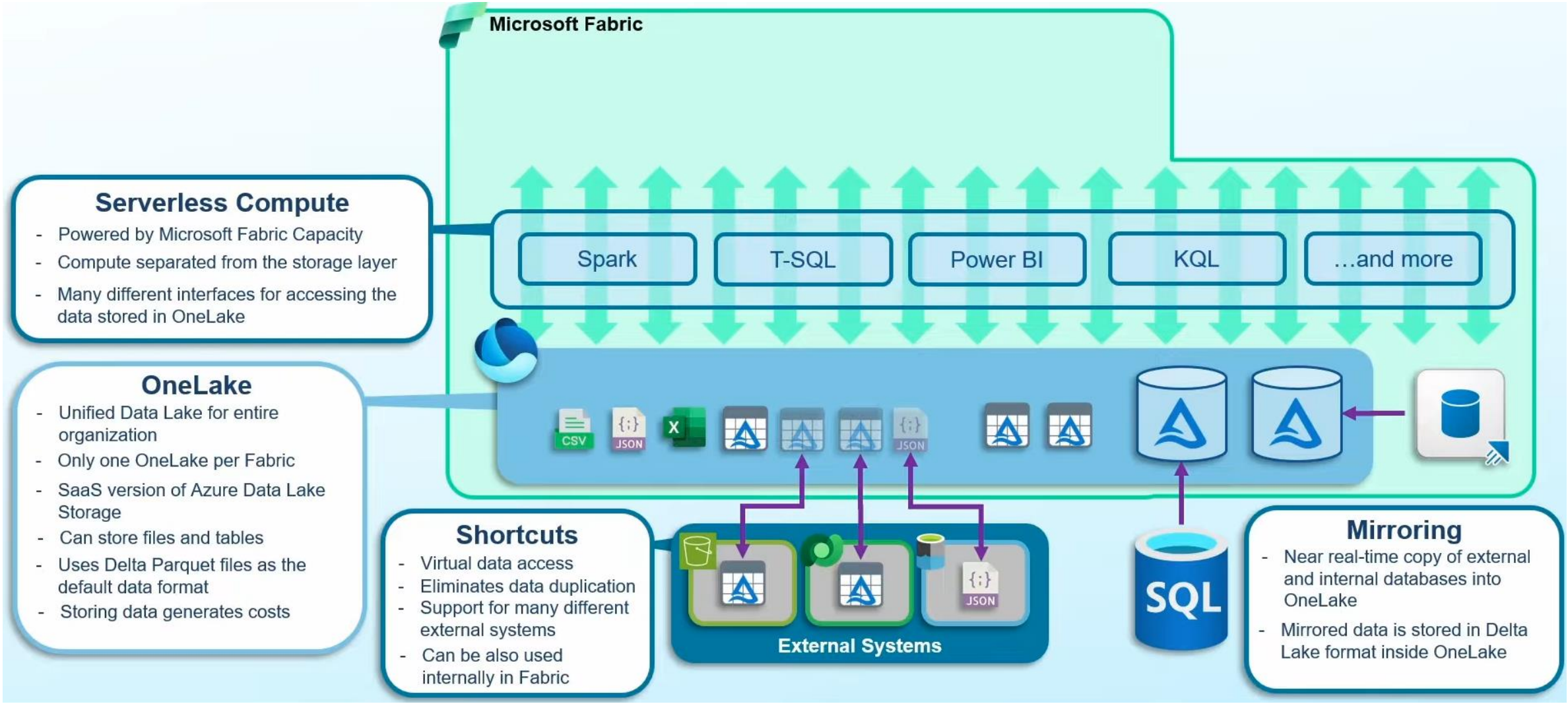
Microsoft Fabric Structure



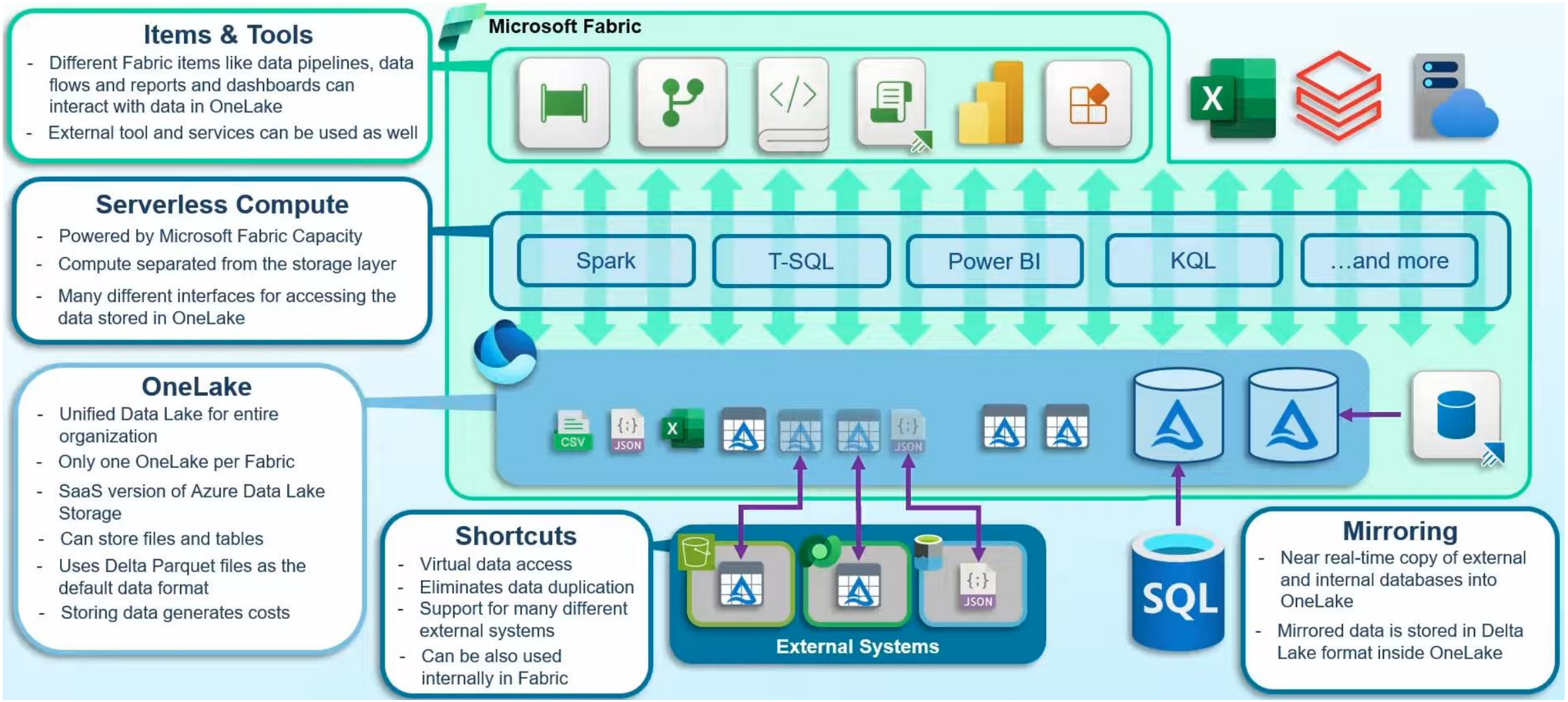




One Lake



One Lake



One Lake

Customer Success Team

Data Engineering Team

IT Team

Manufacturing Team

Data Science Team

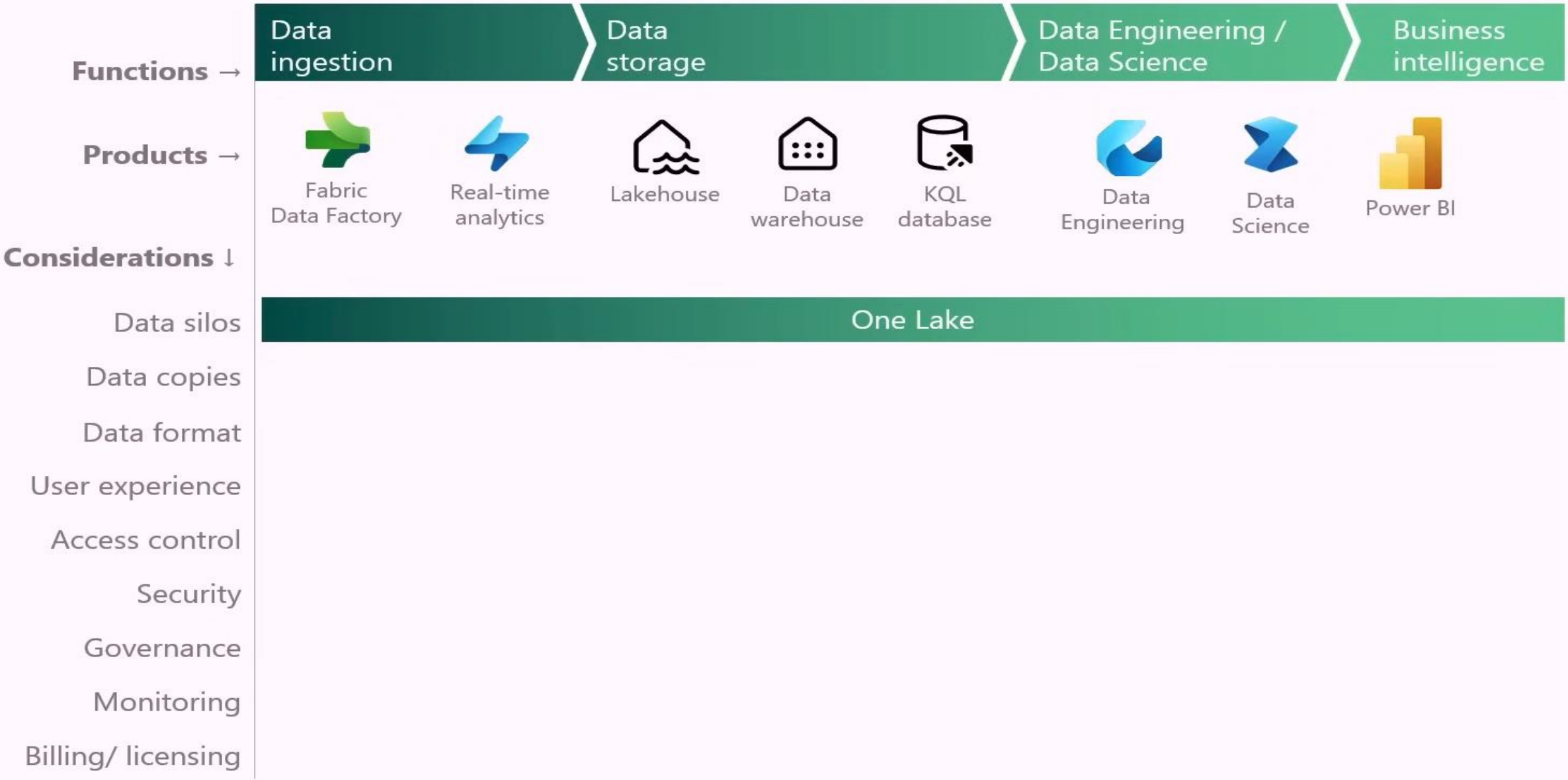
Step 1:
Customer leaves a review of a product on the website

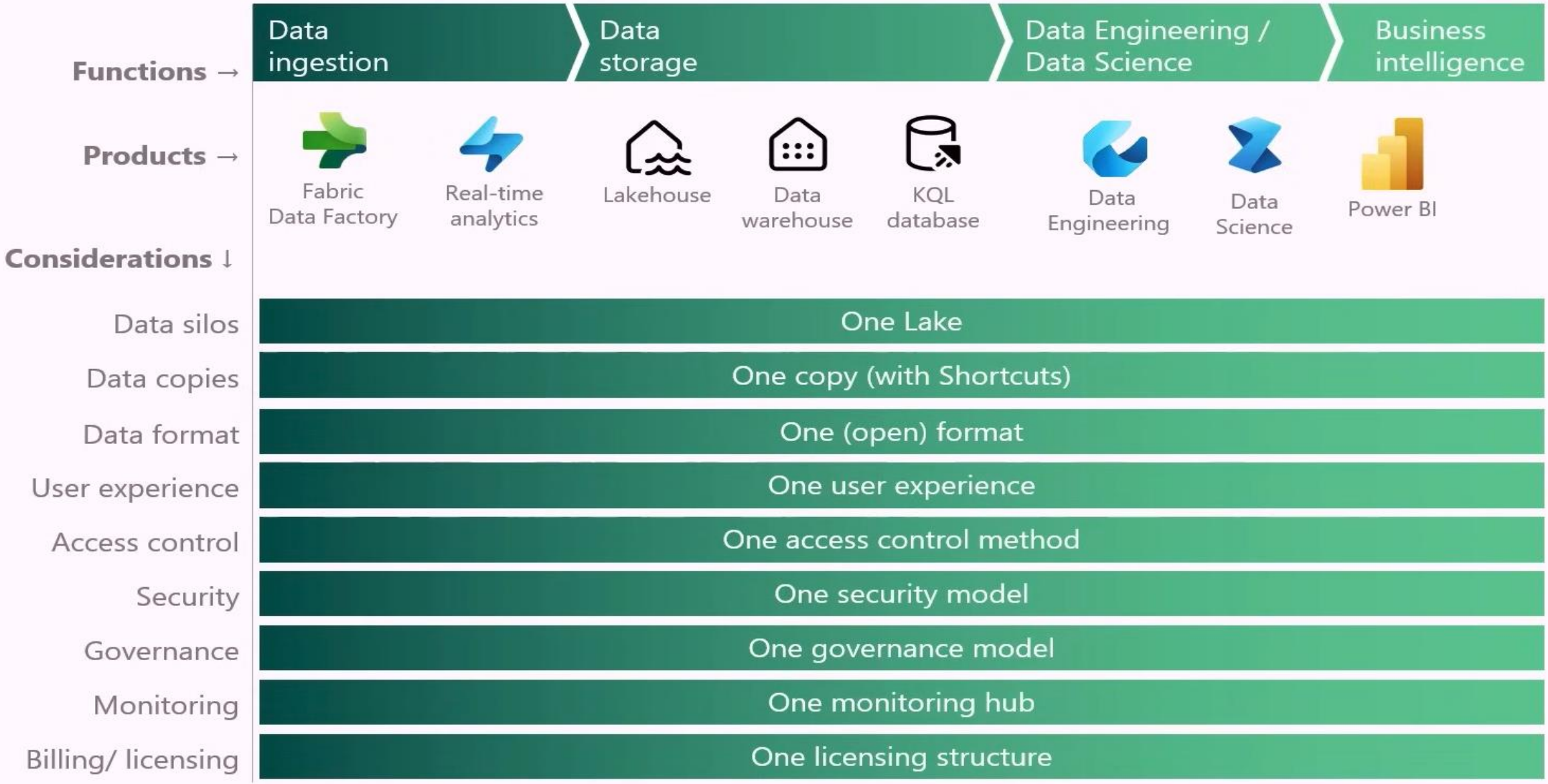
Step 2:
Data is stored in an Azure SQL database, managed by the customer success team

Step 3:
The data engineers have built a data pipeline that copies this data every morning to the data lake for data scientists to analyze

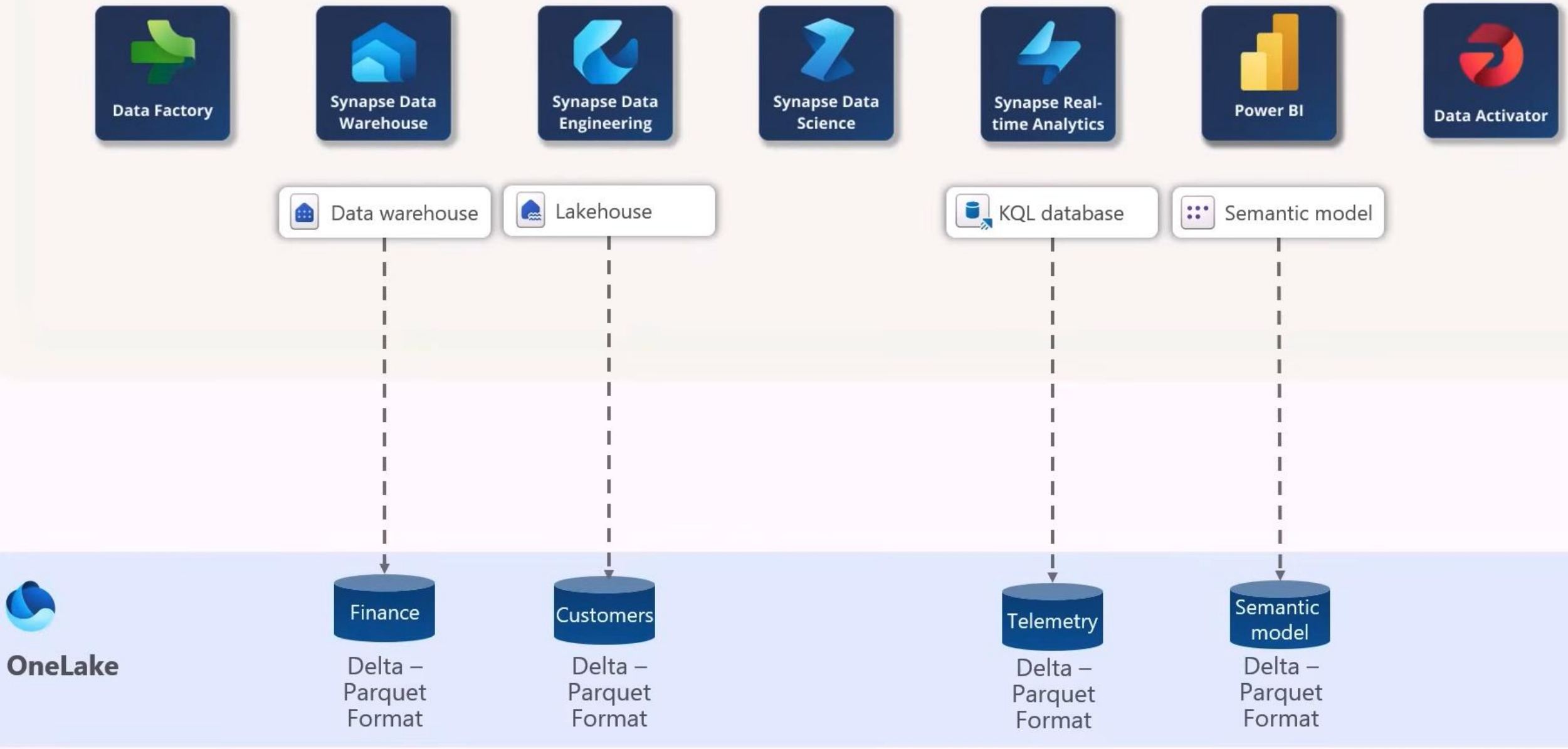
Step 4:
Data scientists have built a model that predicts the sentiment of customer reviews and writes the data into a data lake container.

Step 5:
A Power BI report communicates these findings back to product teams to improve products going forward

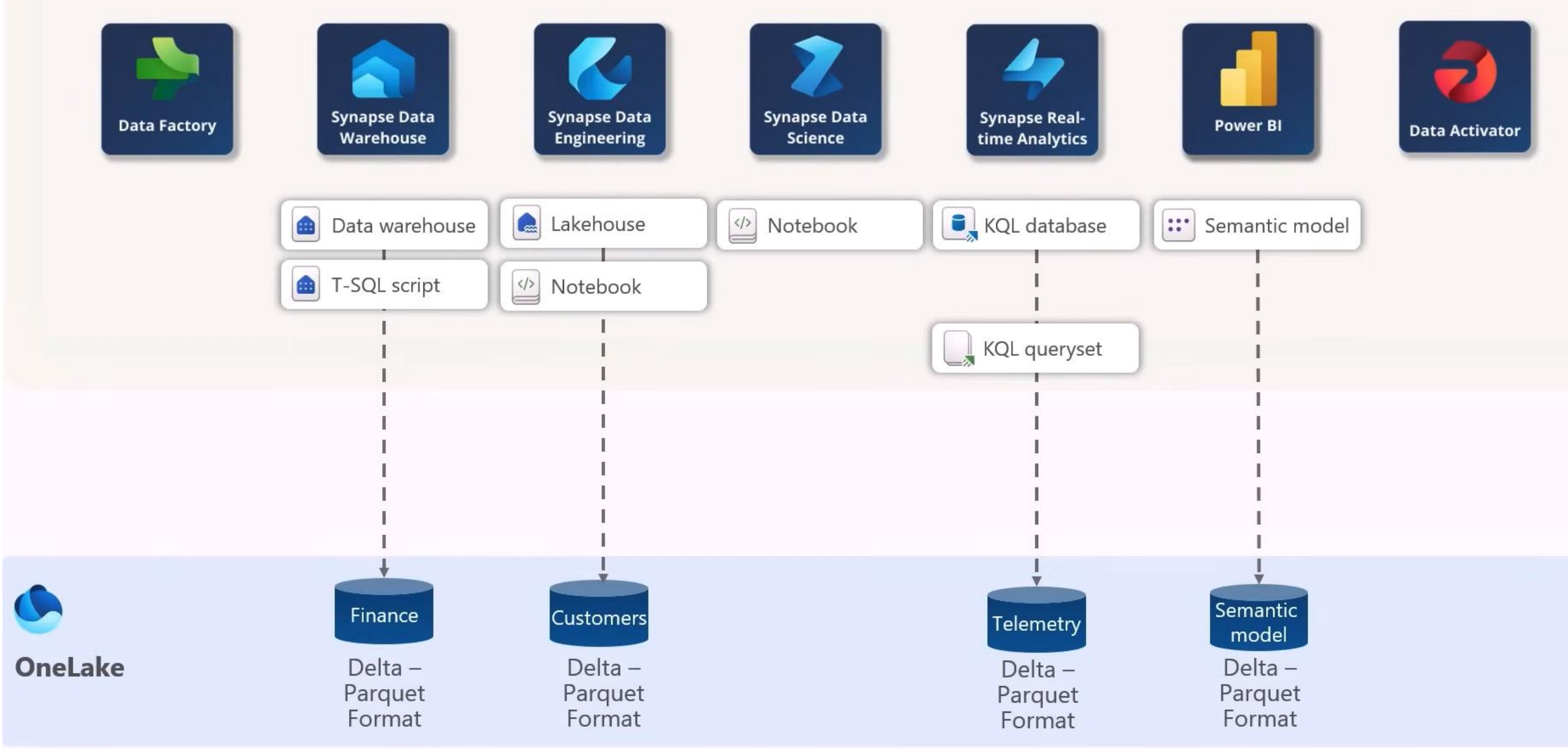


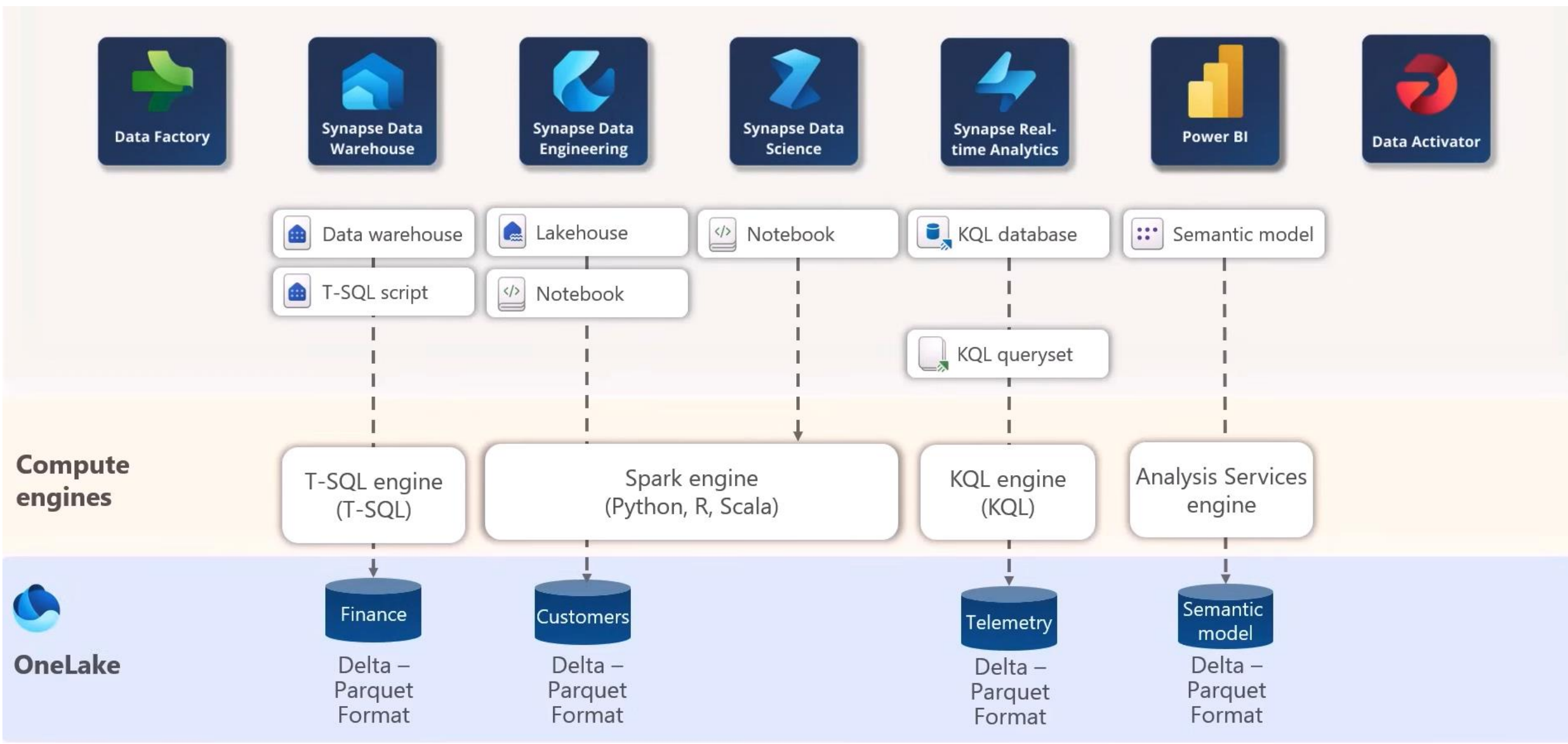


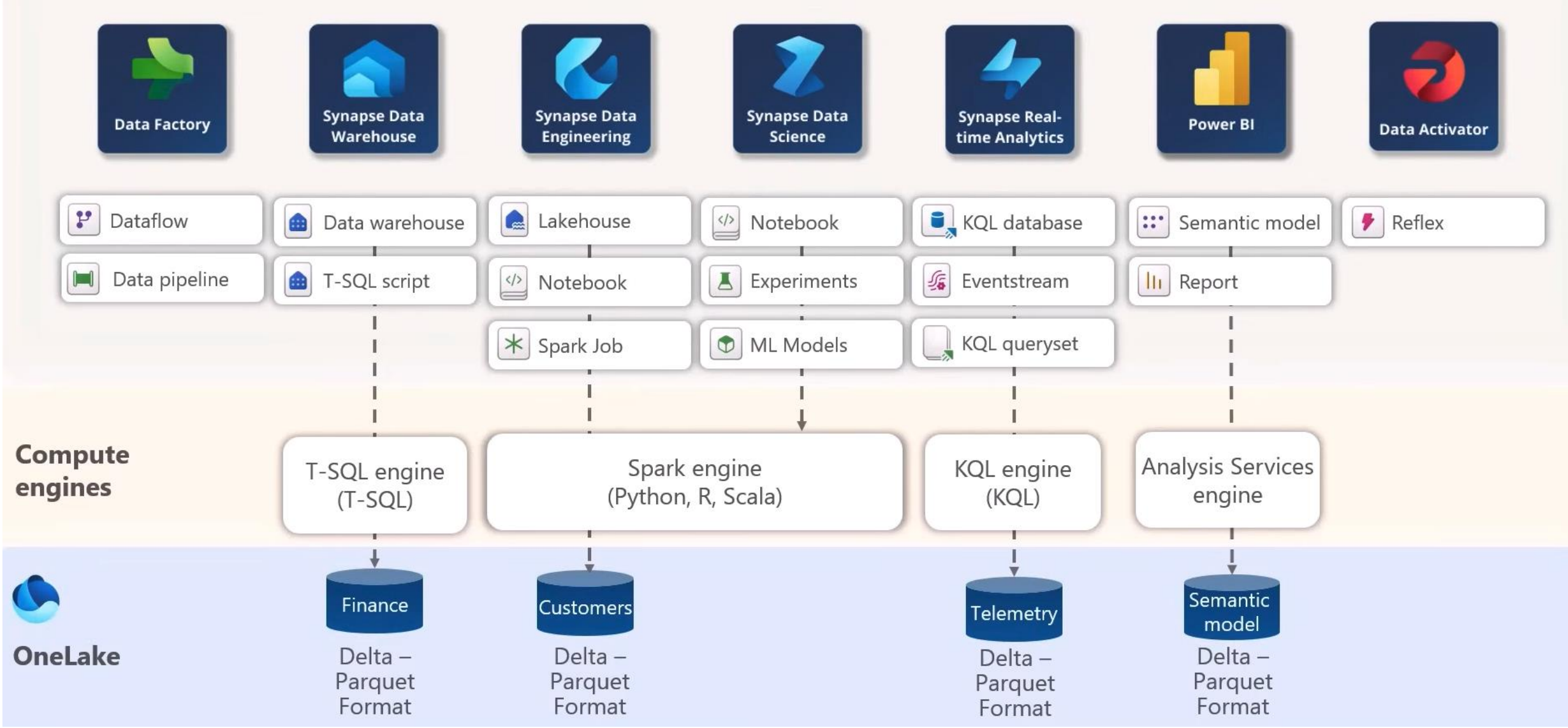
Fabric Experience



Fabric Experience







DATA FACTORY

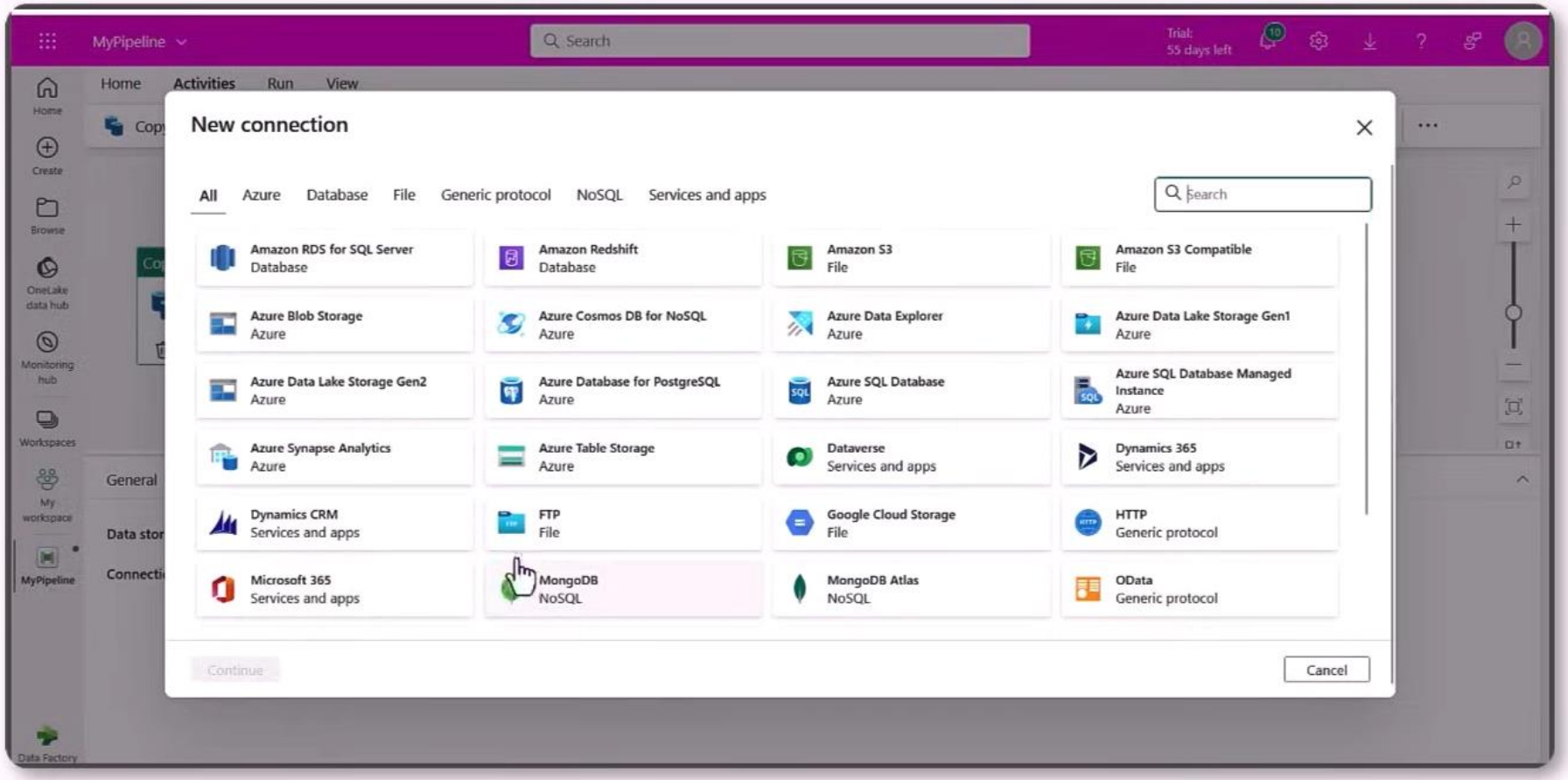
Core purpose:
Moving and transforming your data. A set of tools to help you with Extract, Transform and Load.

Fabric items:

- Dataflow
- Data pipeline

Similar to:

- Azure Data Factory
- Synapse Pipelines
- Power BI Dataflow Gen1



SYNAPSE DATA WAREHOUSE

Core purpose:
provides a familiar transactional data warehouse solution with tables, schemas, views, stored procedures etc. Query-able using Structured Query Language (T-SQL)

Fabric items:

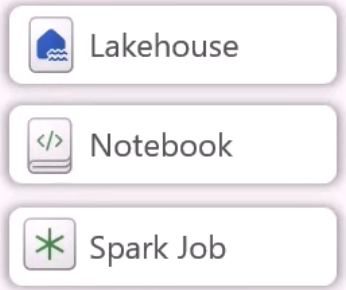
Similar to:

- SQL Server/ Azure SQL
- Synapse SQL Serverless/Dedicated
- Snowflake

SYNAPSE DATA ENGINEERING

Core purpose:
Enables users to design, build, and maintain **infrastructures and systems** that enable their organizations to collect, store, process, and analyze large volumes of data .

Fabric items:



Similar to:

- Azure Data Lake Storage*(ADLS Gen2)
- Databricks
- Snowflake

SYNAPSE DATA SCIENCE

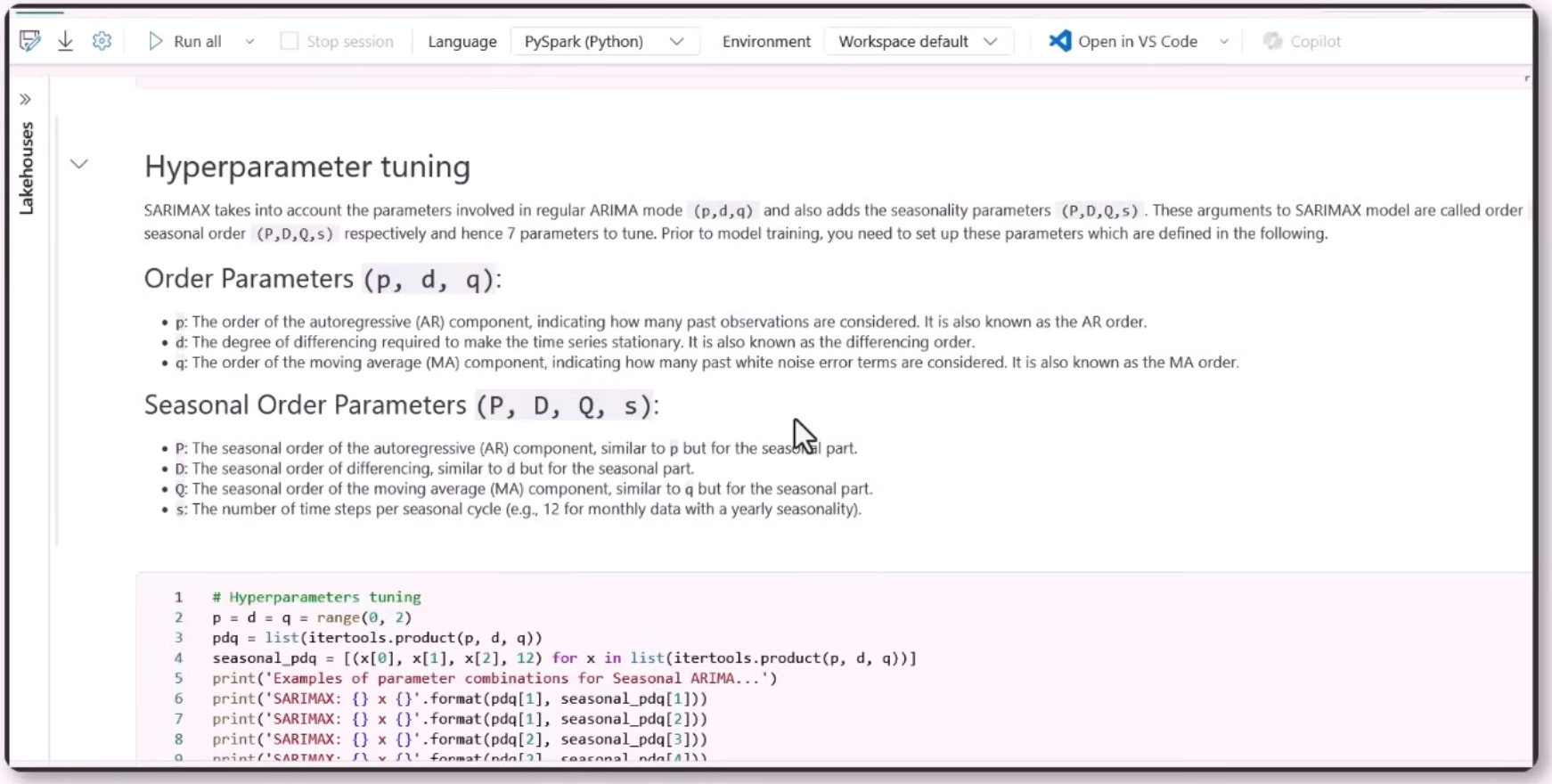
Core purpose:
Supports the entire data science workflow in an organisation, from data exploration, preparation and cleansing to experimentation, modeling, model scoring and serving of predictive insights to BI reports.

Fabric items:

- Notebook
- Experiments
- ML Models

Similar to:

- Azure Machine Learning
- Synapse notebooks
- Databricks notebooks



SYNAPSE REAL-TIME ANALYTICS

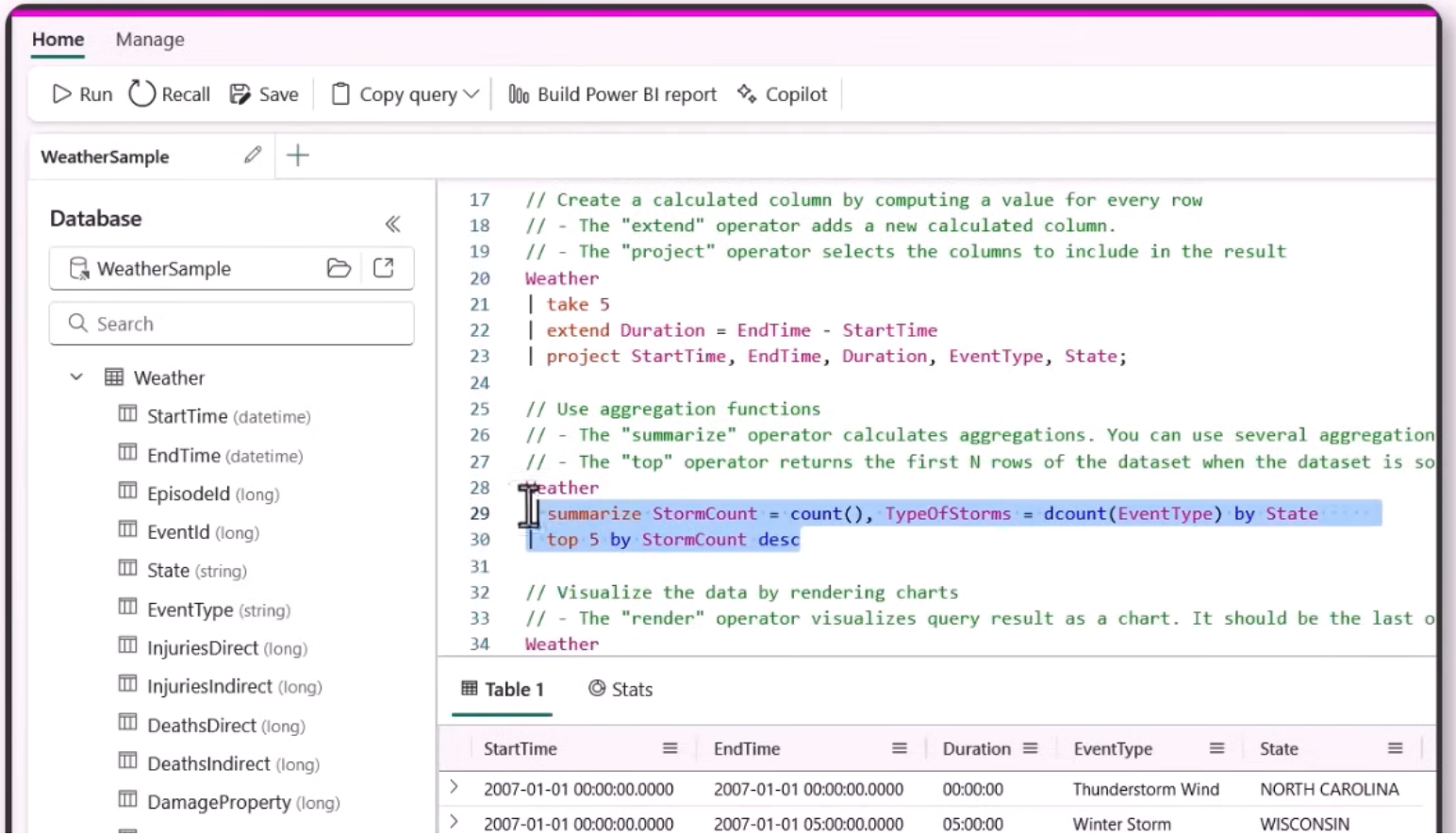
Core purpose:
Provides a set of tools to ingest, manage and analyze real-time event data

Fabric items:

- KQL database
- Eventstream
- KQL queryset

Similar to:

- Azure Data Explorer



POWER BI

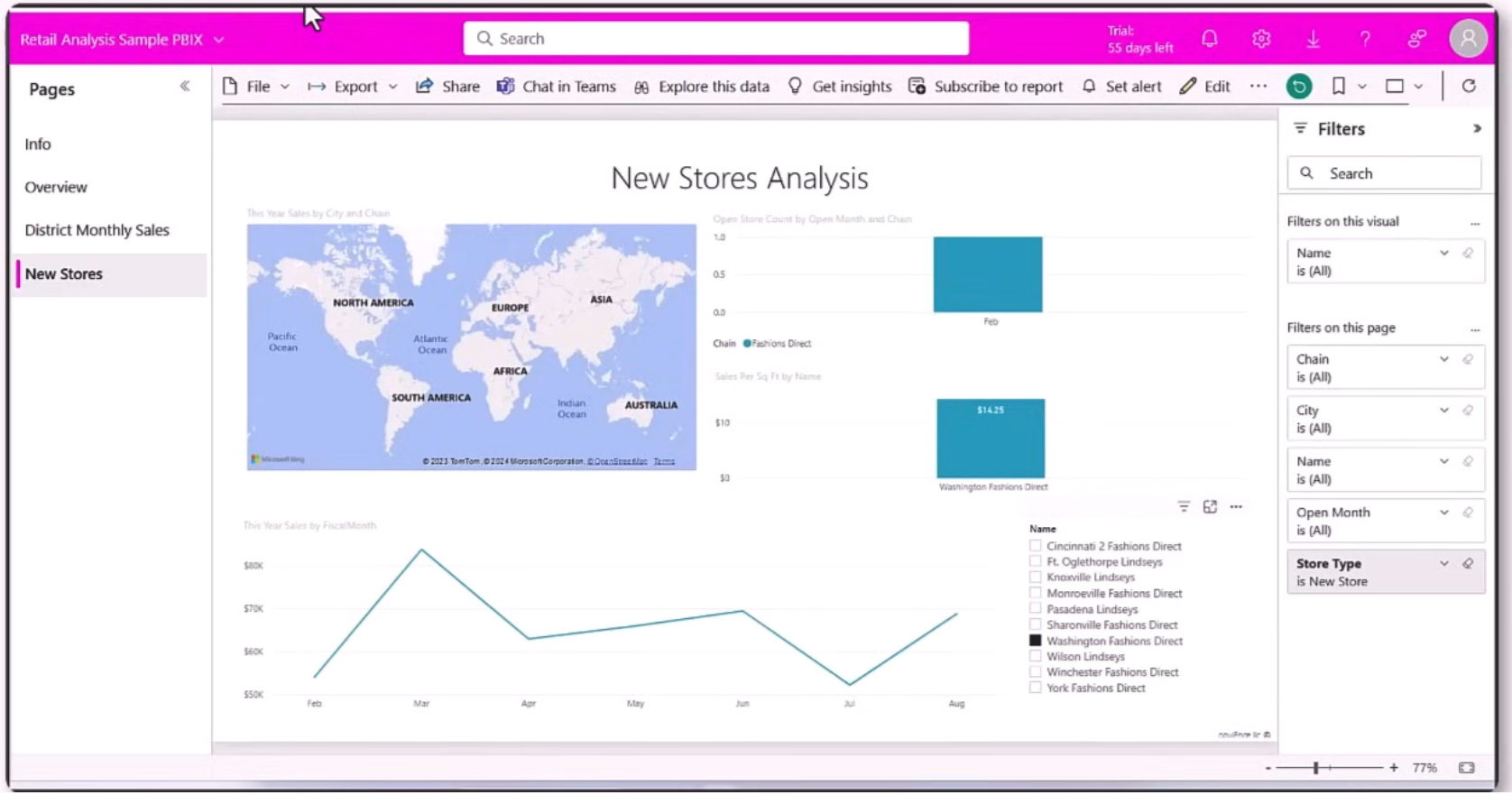
Core purpose:
Power BI is Microsoft's business intelligence solution that allows you to create reports to present visual insights to business users.

Fabric items:

- Report
- Semantic model

Similar to:

- Tableau
- Looker



DATA ACTIVATOR

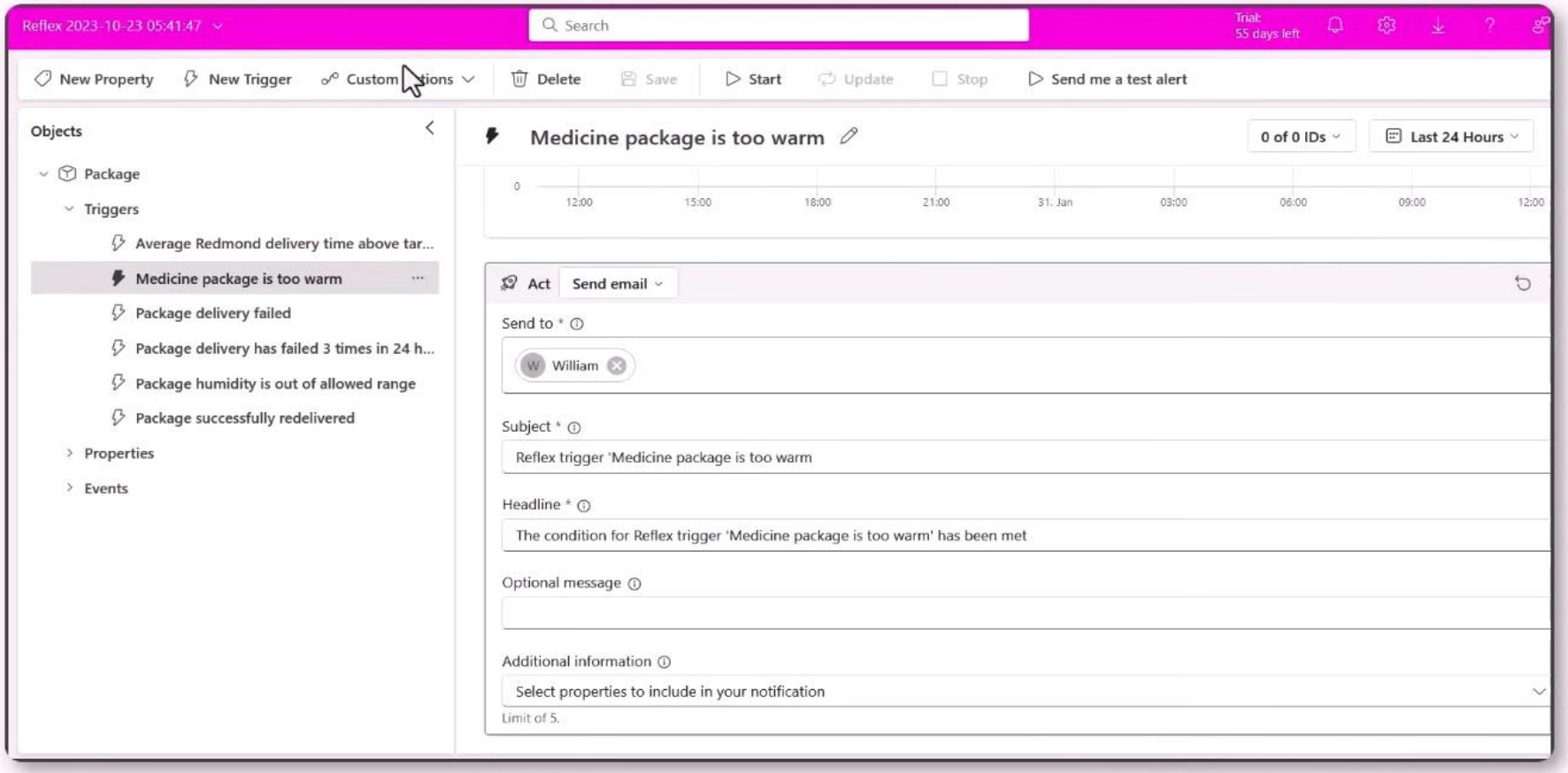
Core purpose:
Automatically taking actions (like running a Power Automate routine) when **patterns** or **conditions** are detected in changing data, such as data in Power BI reports and Eventstreams

Fabric items:

Reflex

Similar to:

- Power Automate
- Azure Functions



Microsoft Fabric

- **Data Factory**: Data integration combining Power Query with the scale of Azure Data Factory to move and transform data.
- **Data Engineering**: Data engineering with a Spark platform for data transformation at scale.
- **Data Warehouse**: Data warehousing with industry-leading SQL performance and scale to support data use.
- **Data Science**: Data science with Azure Machine Learning and Spark for model training and execution tracking in a scalable environment.
- **Real-Time Intelligence**: Real-time Intelligence to query and analyze large volumes of data in real-time.
- **Power BI**: Business intelligence for translating data to decisions.

Data teams and Fabric

Fabric's unified management and governance make it easier for data professionals to work together.

Data Engineers	Analytics Engineers	Data Scientists	Data Analysts	Decision Makers
 Data Factory	 Data Factory	 Data Science	 Power BI	 Power BI
 Data Warehouse	 Power BI	 Azure ML		 Microsoft 365
 Data Engineering	 Data Engineering			
 Real-time intelligence	 Data Warehouse			

Data Stewards

Enable and use Microsoft Fabric

Fabric must be enabled in your tenant by either:

- Fabric admin (formerly known as Power BI admin)
- Power Platform admin
- Microsoft 365 admin

Workspaces must then be assigned to Premium per capacity or Fabric license mode.

Knowledge check

1 **Which of the following is a key benefit of using Microsoft Fabric in data projects?**

- ❑ It allows data professionals to work on data projects independently, without the need for collaboration
- ❑ It requires duplication of data across different systems and teams to ensure data availability
- ☑ It provides a single, integrated environment for data professionals and the business to collaborate on data projects

2 **How do compute engines in Fabric interact with data stored in OneLake?**

- ☑ Compute engines can directly access data in OneLake without the need for duplication or movement
- ❑ OneLake automatically converts data into csv format for seamless integration with compute engines
- ❑ Compute engines in Fabric can only interact with a limited subset of data formats within OneLake

3 **Which of the following Fabric experiences is used to move and transform data?**

- ❑ Data Science
- ❑ Data Warehousing
- ☑ Data Factory

Recap

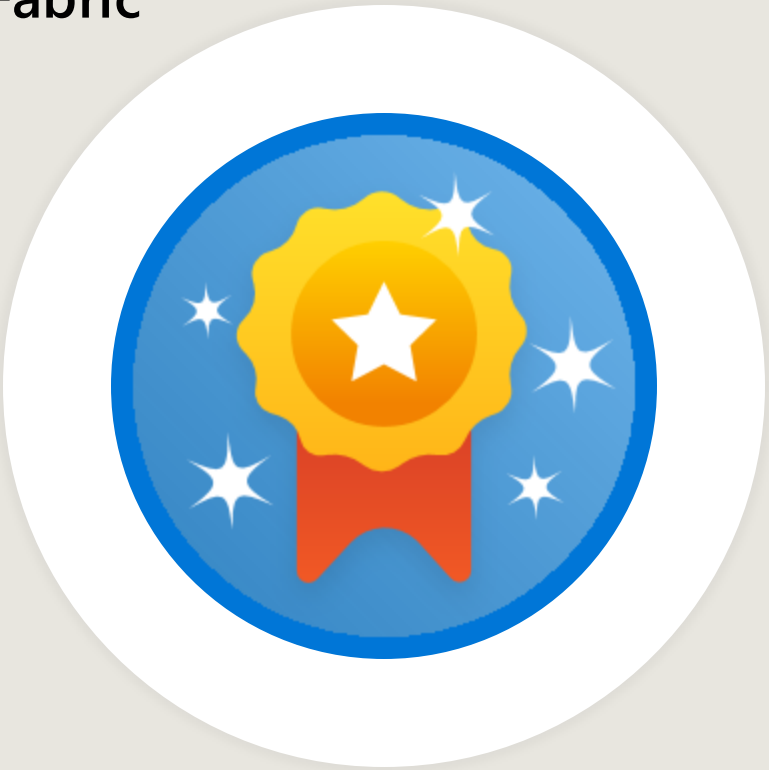
In this section, we covered:

- Different workloads in Microsoft Fabric.
- How Fabric offers flexibility for data teams to perform tasks.

Further reading

Introduction to end-to-end analytics using Microsoft Fabric

https://aka.ms/fabric-intro



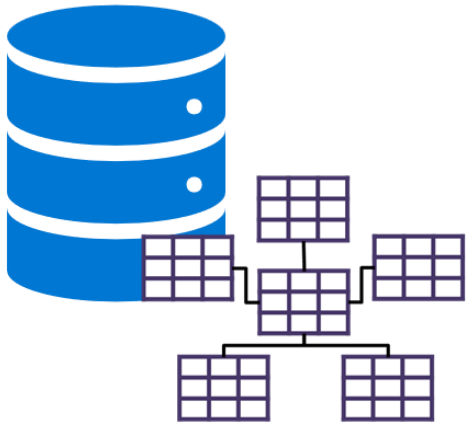
Get started with lakehouses in Microsoft Fabric

Learning objectives

- Describe end-to-end analytics in Microsoft Fabric
- Understand data teams and roles that use Fabric
- Describe how to enable and use Fabric

What is a lakehouse?

+



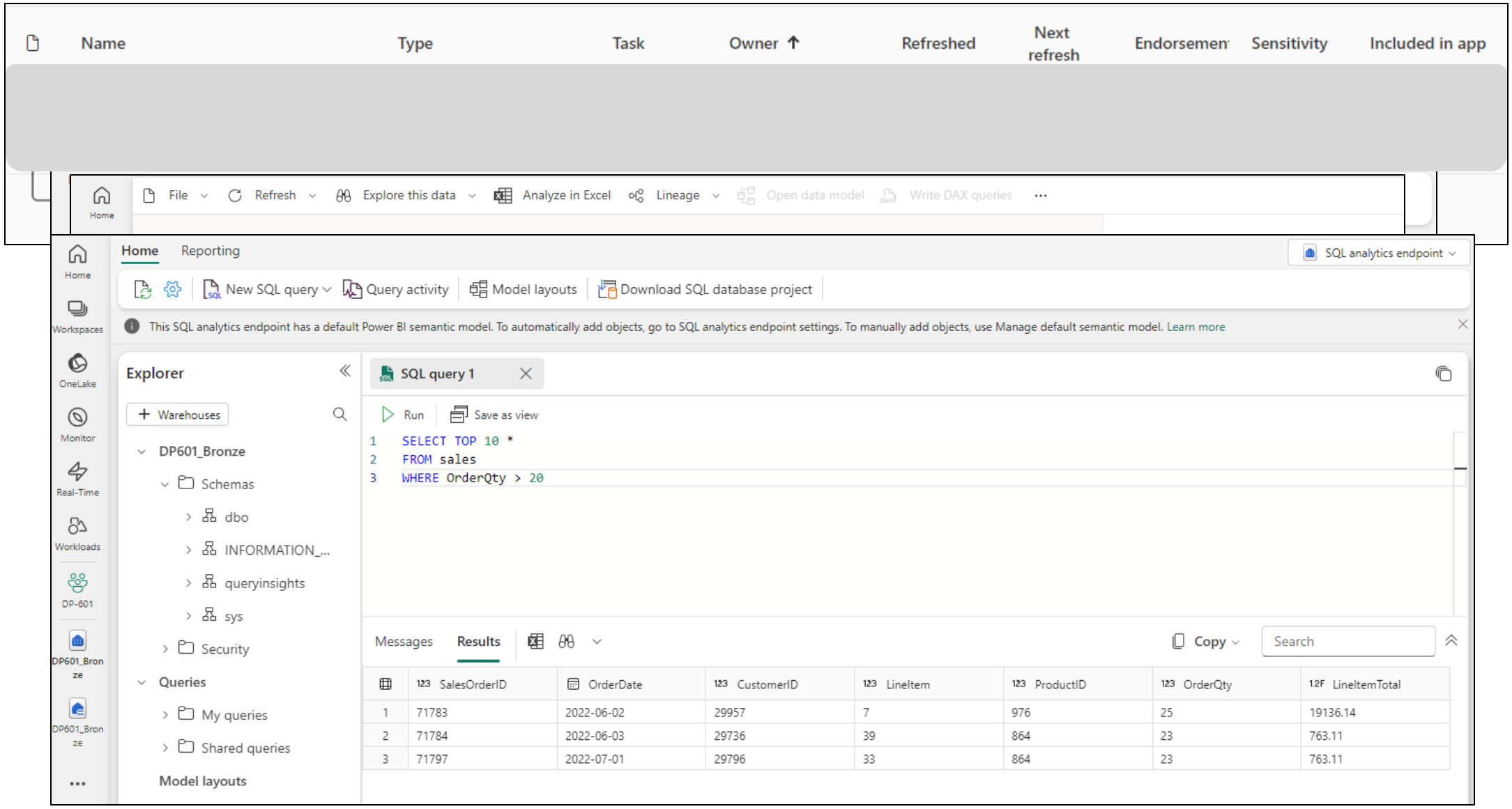
Data *Lake*

- Scalable, distributed file storage
- Flexible schema-on-read semantics
- Big data technology compatibility

Data Ware*house*

- Relational schema modeling
- SQL-based querying
- Proven basis for reporting and analytics

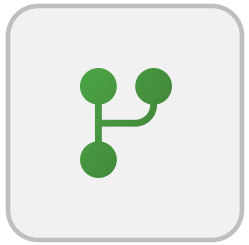
Work with a Fabric lakehouse



Load data into a lakehouse

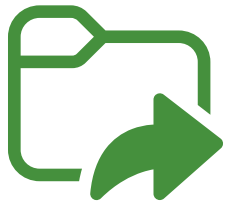
Ingest and connect to data sources to transform and load for analytics.

Notebooks



Dataflows Gen2

Pipelines



Shortcuts

Explore, transform, and visualize data in the lakehouse

Tools and techniques to explore and transform data:

- Apache Spark
  - Notebooks
  - Spark Job Definitions
- SQL analytics endpoint
- Dataflows Gen2
- Data Pipelines

Visualize lakehouse data using Power BI

Exercise

30 minutes

Create and ingest data with a Microsoft Fabric lakehouse

Knowledge check

1 **What is a Microsoft Fabric lakehouse?**

- ❑ A relational database based on the Microsoft SQL Server database engine.
- ❑ A hierarchy of folders and files in Azure Data Lake Store Gen2.
- ☑ An analytical store that combines the file storage flexibility of a data lake with the SQL-based query capabilities of a data warehouse.

2 **You want to include data in an external Azure Data Lake Store Gen2 location in your lakehouse, without the requirement to copy the data. What should you do?**

- ❑ Create a Data pipeline that uses a Copy Data activity to load the external data into a file.
- ☑ Create a shortcut.
- ❑ Create a Dataflow Gen2 that extracts the data and loads it into a table.

3 **You want to use Apache Spark to interactively explore data in a file in the lakehouse. What should you do?**

- ☑ Create a notebook.
- ❑ Switch to the SQL analytics endpoint mode.
- ❑ Create a Dataflow Gen2.

Recap

In this section, we covered:

- Lakehouse architecture.
- How to work with a lakehouse.
- Data ingestion, transformation, and exploration tools.

Further reading

Get started with lakehouses in Microsoft Fabric

https://aka.ms/fabric-lakehouse

Use Apache Spark in Microsoft Fabric

Learning objectives

- Describe end-to-end analytics in Microsoft Fabric
- Understand data teams and roles that use Fabric
- Describe how to enable and use Fabric

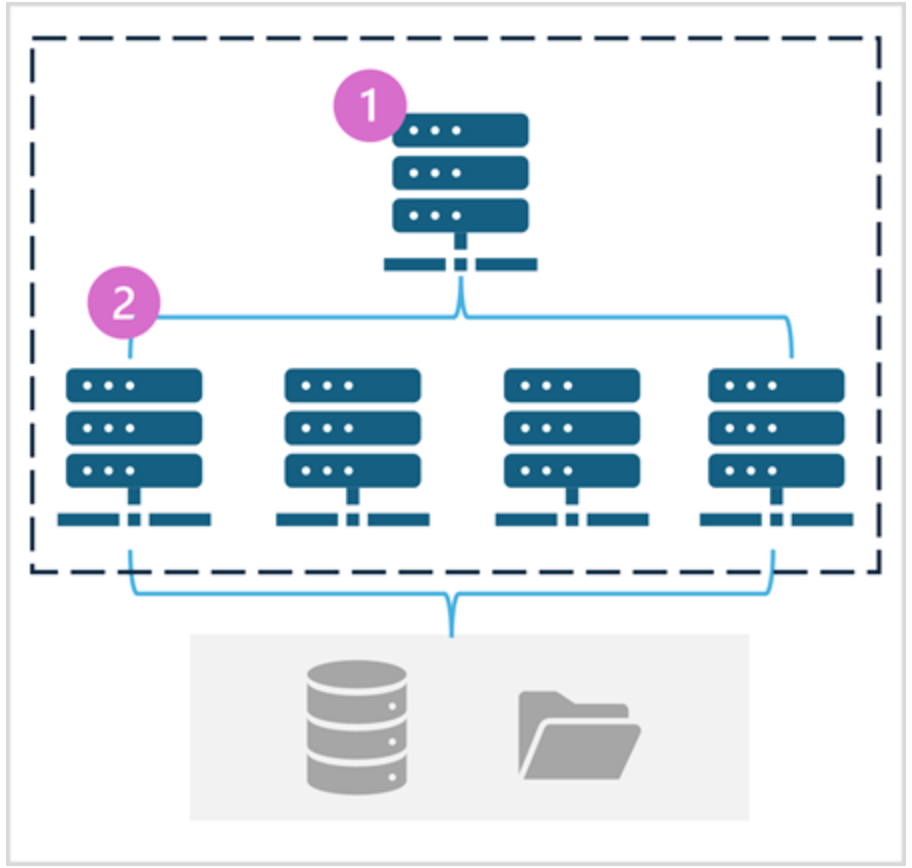
What is Apache Spark?

Apache Spark is a core technology for large-scale data analytics.

Microsoft Fabric provides support for Spark clusters, enabling you to analyze and process data in a lakehouse at scale.

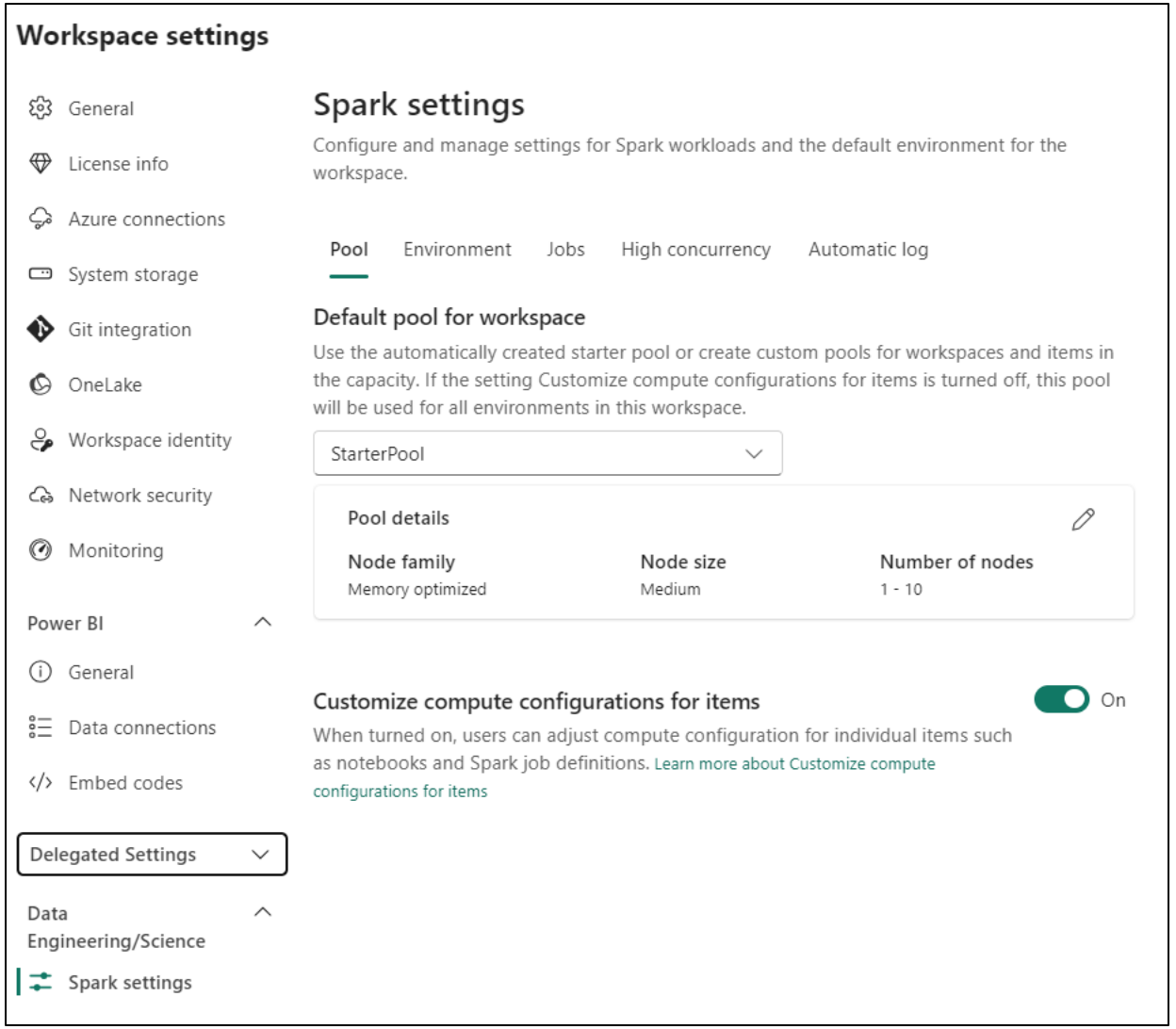
How to use Apache Spark

- Distributed data processing framework through Spark pools.
- Use notebooks to ingest, transform, load, analyze, and visualize data.
- Notebooks in Fabric support many programming languages, including PySpark and Spark SQL.



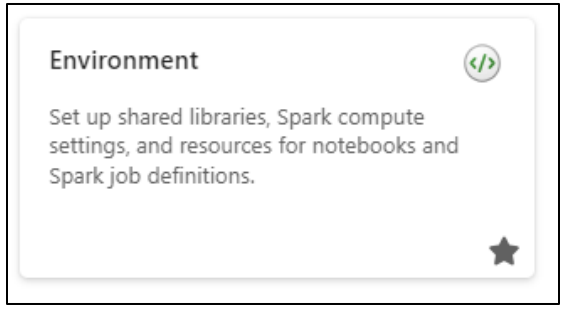
Prepare to use Apache Spark

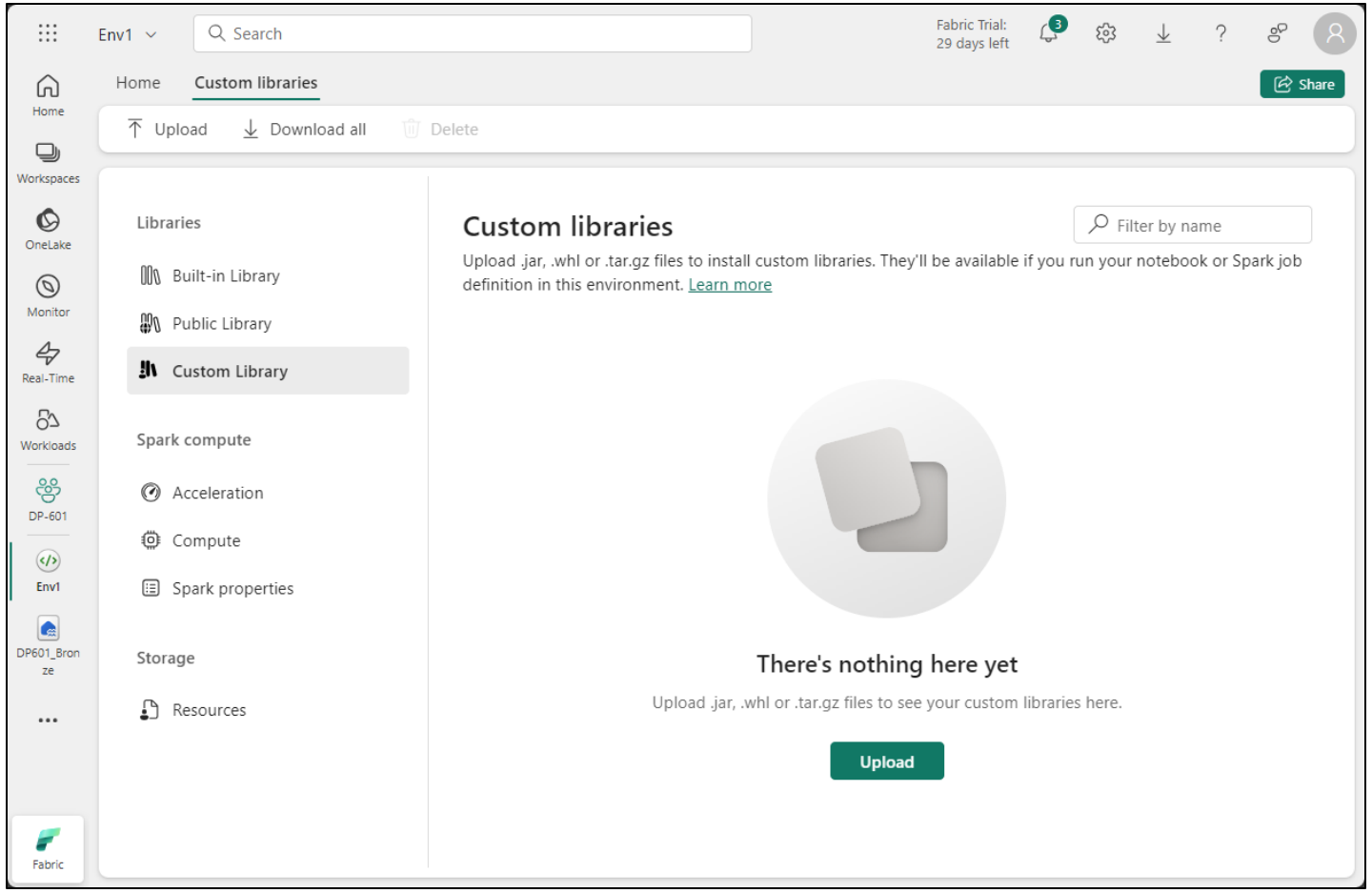
- Each workspace is assigned a Spark cluster.
- Workspace admins can manage settings for the Spark cluster in the Workspace settings.
- Specific configuration settings include:
  - Node family
  - Runtime version
  - Spark properties



Configure the Spark environment

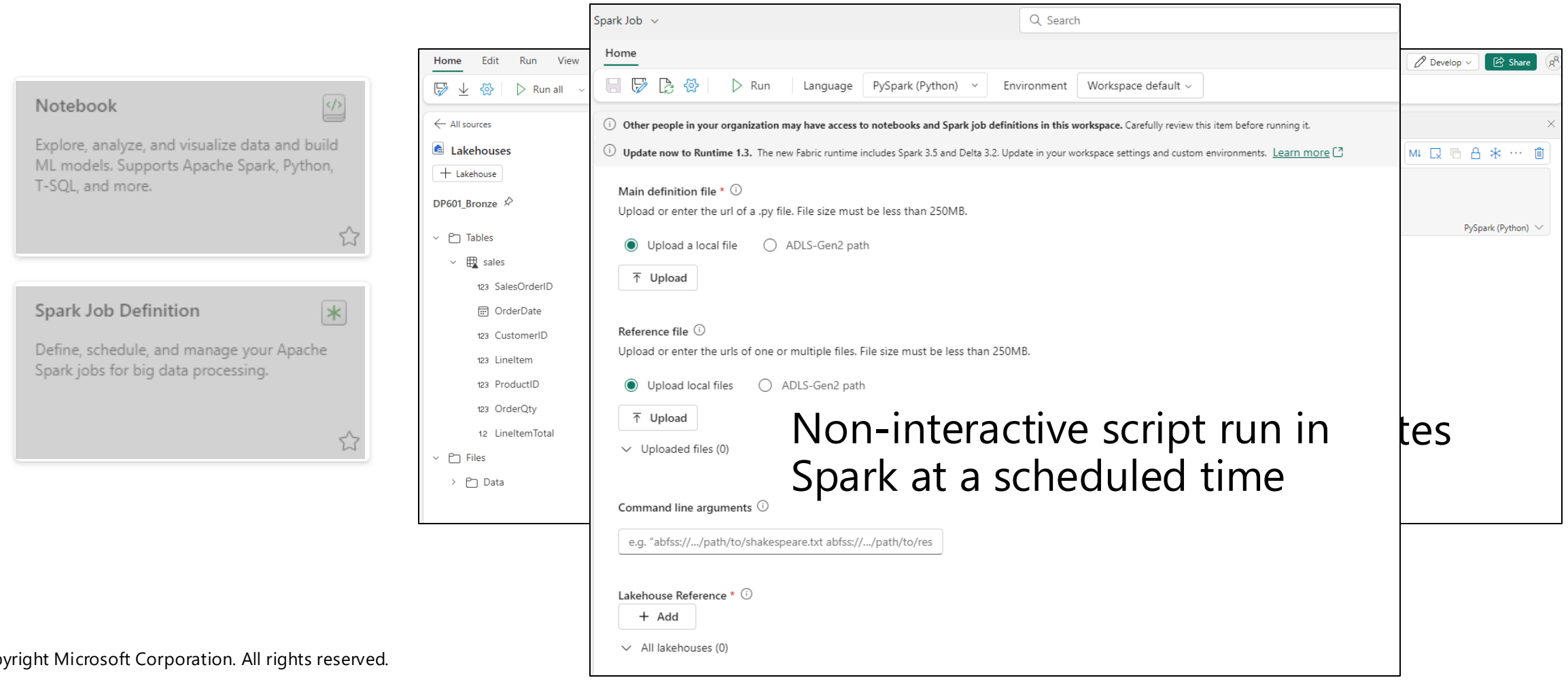
- Libraries
- Spark Compute
- Storage





Run Spark in Fabric

To edit and run Spark code in Fabric, use *notebooks* or create a *Spark Job Definition*



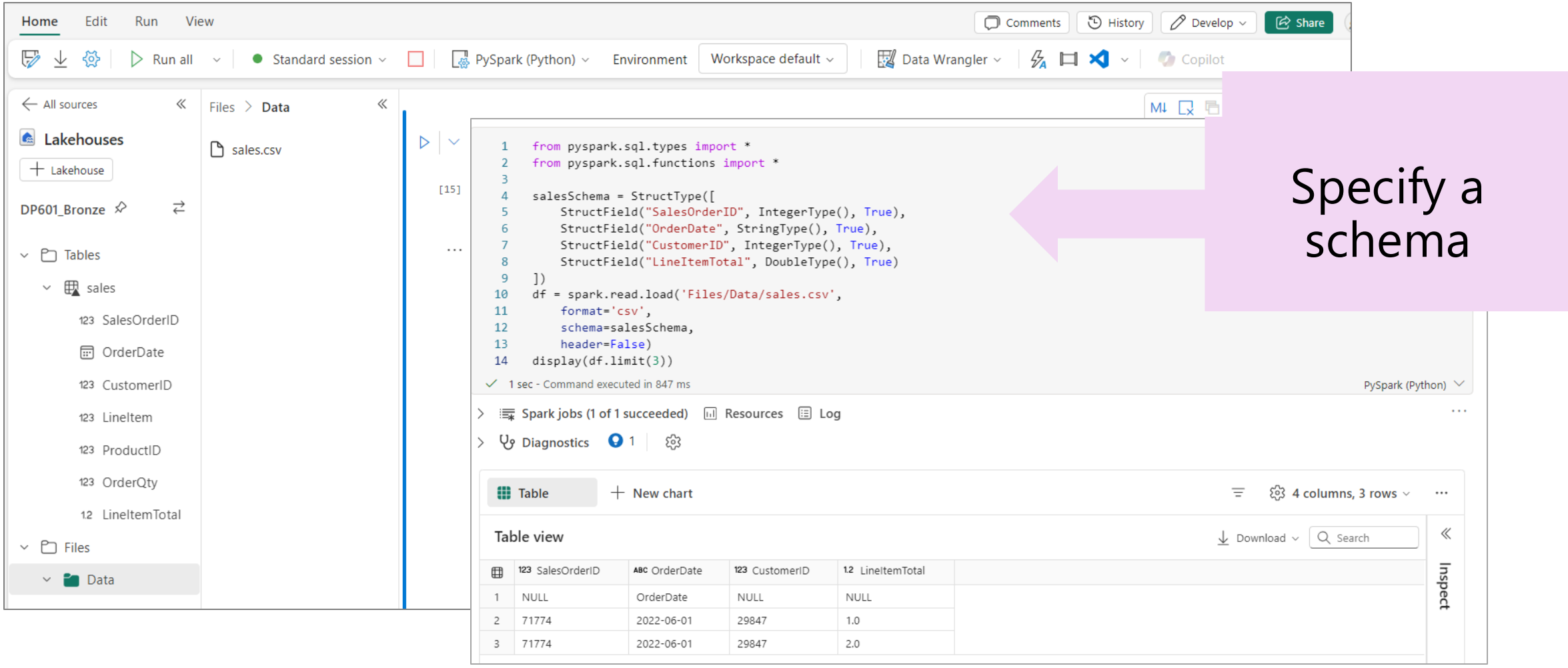
Ingest data with Spark

- Define data source
- Authentication type
- Credentials
- Display or load data

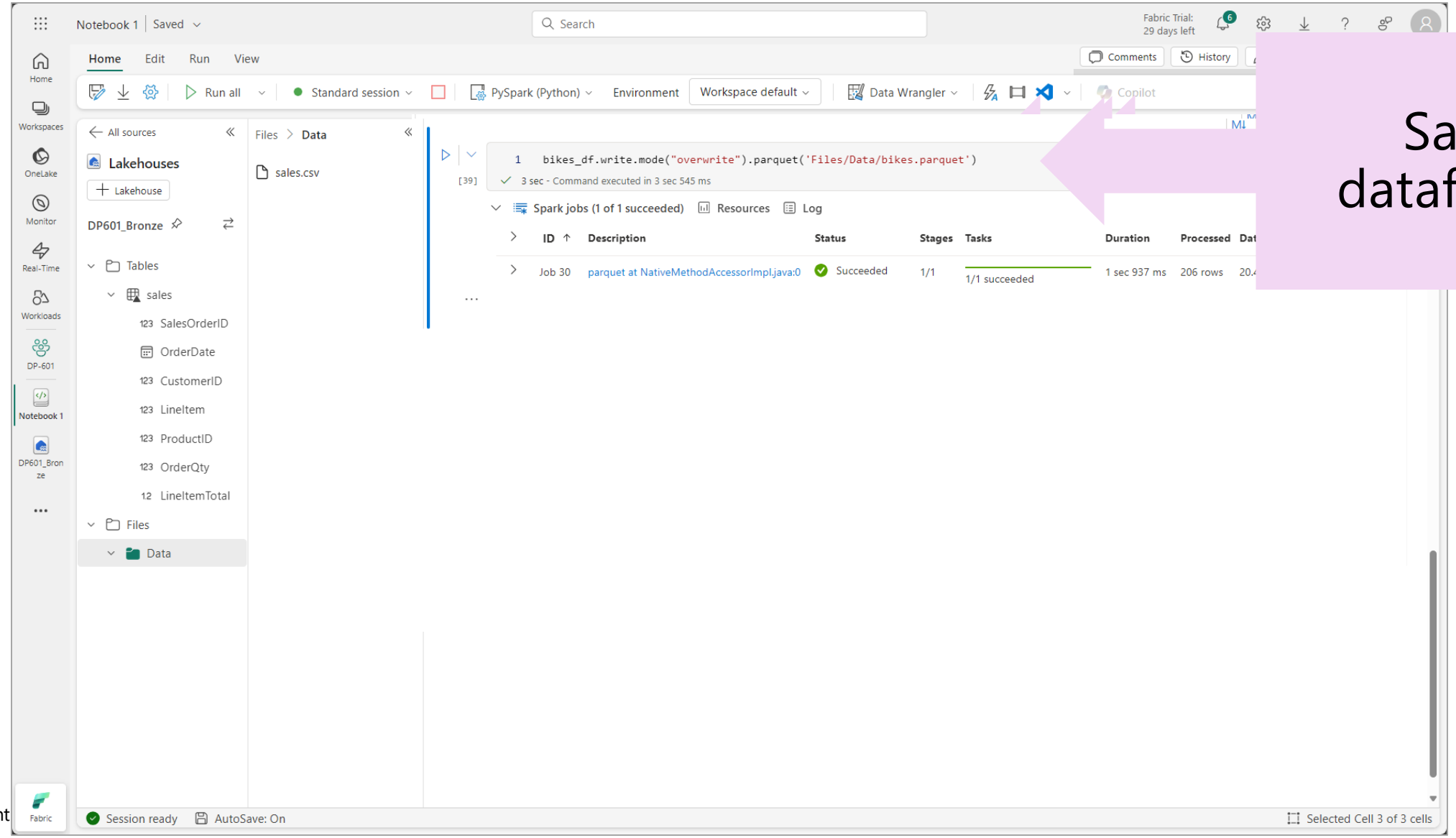
```
# Azure Blob Storage access info

blob_account_name = "azureopendatastorage"

blob_container_name = "nyctlc"

blob_relative_path = "yellow"

# blob_sas_token = "add your SAS token here" #
Construct the path for connection

 wasbs_path =
f'wasbs://{blob_container_name}@{blob_account_name}.
blob.core.windows.net/{blob_relative_path}’

# WASBS path for connection including SAS token

# wasbs_path =
f'wasbs://{blob_container_name}@{blob_account_name}.
blob.core.windows.net/{blob_relative_path}?{blob_sas
_token}’

# Read parquet data from Azure Blob Storage path

blob_df = spark.read.parquet(wasbs_path)

# Display the Azure Blob DataFrame display(blob_df)
```

Load data in a Spark Dataframe

Schema can be either inferred or specified



Transform data in a Spark Dataframe



Work with data using Spark SQL

Use the metastore to define tables and views

```
# Create a view in the metastore
df.createOrReplaceTempView("products_view")

# Save a dataframe as a new table
df.write.format("delta").saveAsTable("products")
```

Create external tables

```
# Create external table
df.write.format("delta").saveAsTable("myexternaltable",path="Files/myexternaltable")
```

Query data using the Spark SQL API

Use the Spark SQL API in code written in any language to query data in the catalog

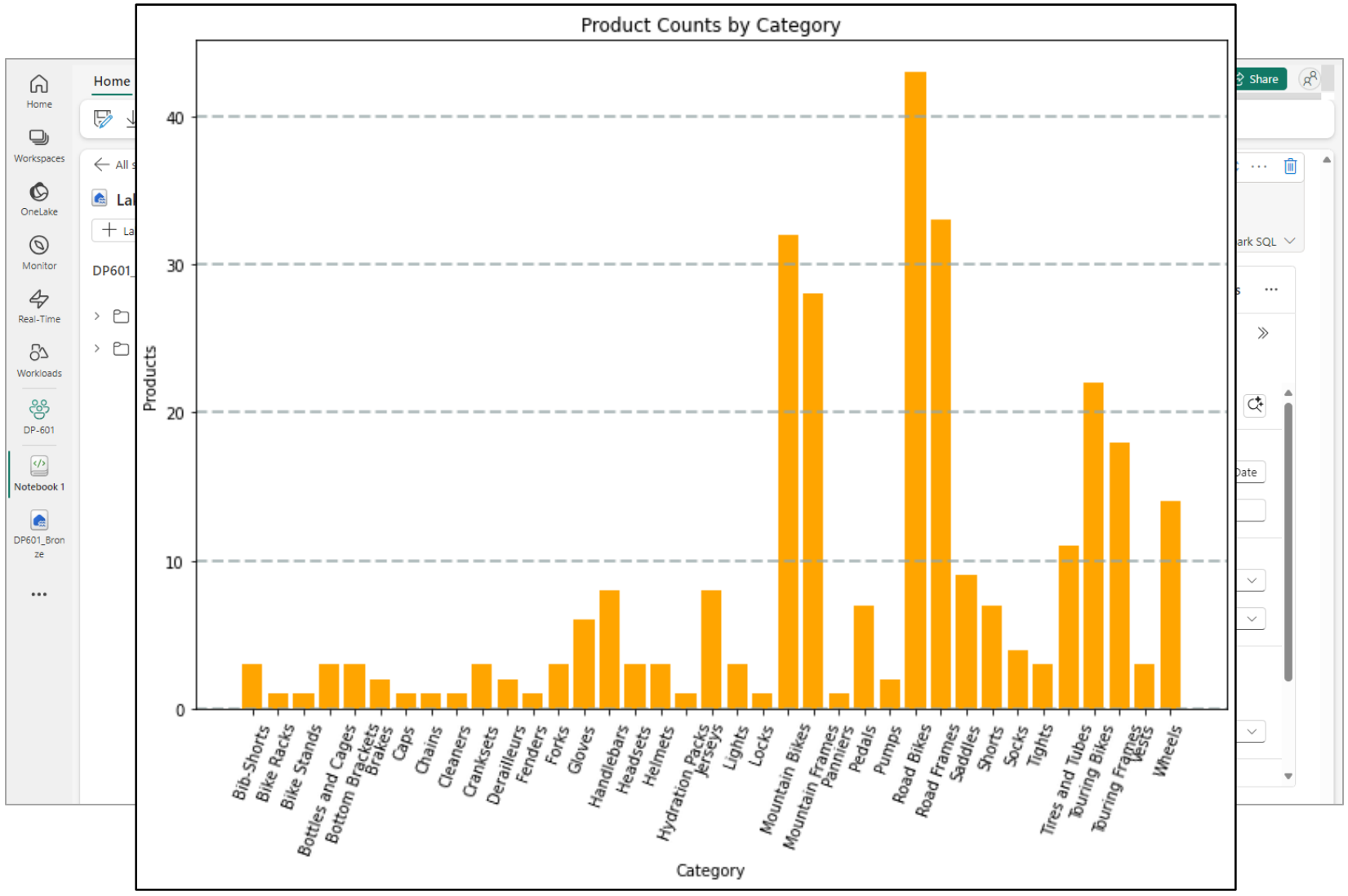
```
# Return data from the products table as a dataframe using PySpark

bikes_df = spark.sql("SELECT ProductID, ProductName, ListPrice \
                      FROM products \
                      WHERE Category IN ('Mountain Bikes', 'Road Bikes')")
display(bikes_df)
```

```
# Use %%sql magic to query objects in the catalog using native SQL

%%sql

SELECT Category, COUNT(ProductID) AS ProductCount
FROM products
GROUP BY Category
ORDER BY Category
```

Visualize data

Two ways to visualize data in notebooks:

1. Use built-in notebook charts
2. Use graphics packages in code
   - This example uses Matplotlib



Exercise

Analyze data with Apache Spark

45 minutes

Knowledge check

1 **You want to use Apache Spark to explore data interactively in Microsoft Fabric. What should you create?**

- ❑ A Spark job definition.
- ❑ A Data Factory pipeline.
- ☑ A notebook.

2 **You need to use Spark to analyze data in a CSV file. What's the simplest way to accomplish this goal?**

- ☑ Load the file into a dataframe.
- ❑ Import the data into a table in a warehouse.
- ❑ Convert the data to Parquet format.

3 **Which method is used to split the data across folders when saving a dataframe?**

- ❑ splitBy
- ❑ distributeBy
- ☑ partitionBy

Recap

In this section, we covered:

- How Apache Spark is integrated in Fabric.
- Ingest, transform, and load data into a lakehouse using Spark with notebooks.
- Worked with and visualized data using PySpark and Spark SQL.

Further reading

Use Apache Spark in Microsoft Fabric

https://aka.ms/fabric-spark

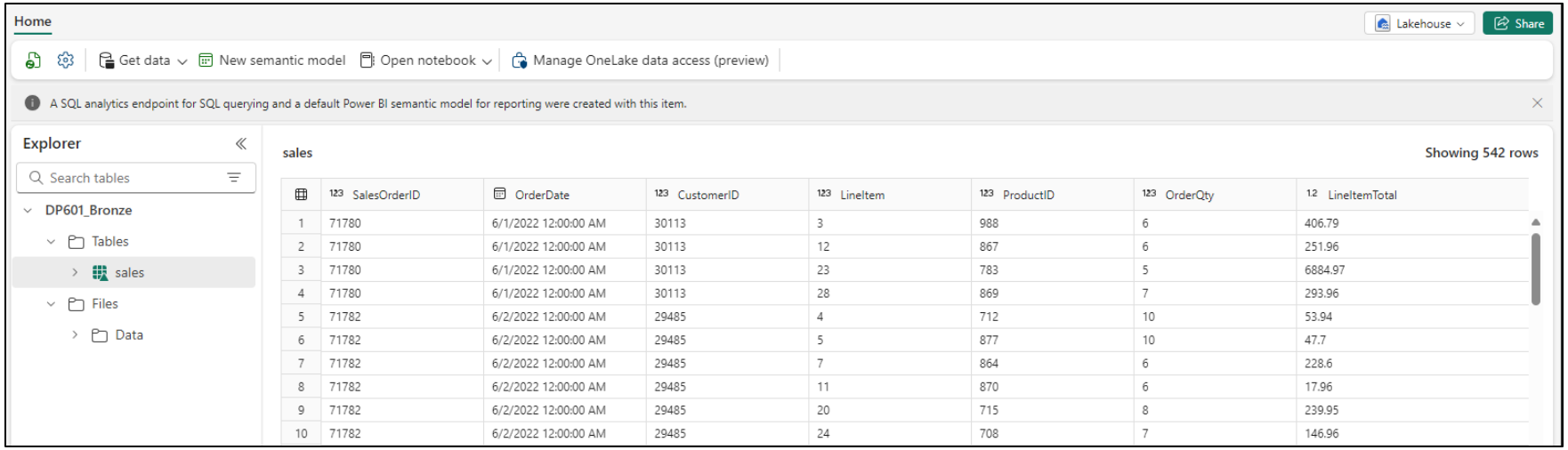
Work with Delta Lake tables in Microsoft Fabric

Learning objectives

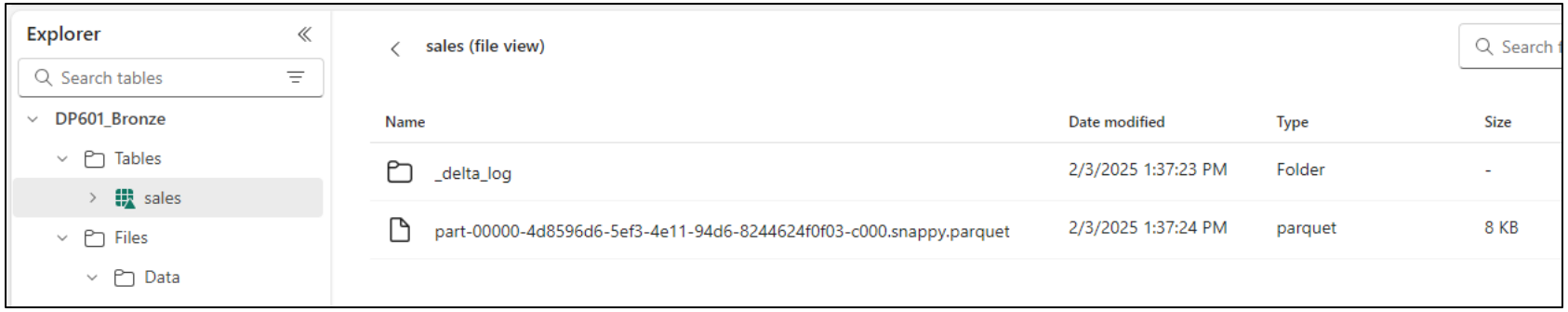
- Understand Delta Lake and delta tables in Microsoft Fabric
- Create and manage delta tables using Spark
- Optimize delta tables
- Use delta tables with Spark structured streaming

Understand Delta Lake

- Relational tables that support querying and data modification
- Support for ACID transactions
- Data versioning and time travel
- Standard formats and interoperability



	SalesOrderID	OrderDate	CustomerID	LineItem	ProductID	OrderQty	LineItemTotal
1	71780	6/1/2022 12:00:00 AM	30113	3	988	6	406.79
2	71780	6/1/2022 12:00:00 AM	30113	12	867	6	251.96
3	71780	6/1/2022 12:00:00 AM	30113	23	783	5	6884.97
4	71780	6/1/2022 12:00:00 AM	30113	28	869	7	293.96
5	71782	6/2/2022 12:00:00 AM	29485	4	712	10	53.94
6	71782	6/2/2022 12:00:00 AM	29485	5	877	10	47.7
7	71782	6/2/2022 12:00:00 AM	29485	7	864	6	228.6
8	71782	6/2/2022 12:00:00 AM	29485	11	870	6	17.96
9	71782	6/2/2022 12:00:00 AM	29485	20	715	8	239.95
10	71782	6/2/2022 12:00:00 AM	29485	24	708	7	146.96



Create Delta tables using code in Spark

1. Save a dataframe as a managed table

```
# Load a file into a dataframe
df = spark.read.load('Files/mydata.csv', format='csv', header=True)

# Save the dataframe as a delta table
df.write.format("delta").saveAsTable("mytable")
```

2. Use Spark SQL

```
%%sql

CREATE TABLE salesorders
(
    Orderid INT NOT NULL,
    OrderDate TIMESTAMP NOT NULL,
    CustomerName STRING,
    SalesTotal FLOAT NOT NULL
)
USING DELTA
```

3. Save a dataframe in delta format in an explicit path

```
delta_path = "Files/mydatatable"
df.write.format("delta").save(delta_path)
```

Managed vs external tables

Managed tables

- Defined without a specific location – Files are created in the default metastore folder (***Tables/...***)
- Dropping the table deletes the files

```
# Save a dataframe as a delta table
df.write.format("delta").saveAsTable("mytable")
```

External tables

- Defined with an explicit file location outside of the default metastore folder
- Dropping the table does not delete the files

```
df.write.format("delta").saveAsTable("myexternaltable",path="Files/myexternaltable")
```

Work with Delta tables in Spark

1. Use Spark SQL to embed a SQL statement in PySpark
   - Embed SQL statements in other languages using the **spark.sql** library.

```
spark.sql("INSERT INTO products VALUES (1, 'Widget', 'Accessories', 2.99)")
```

2. Native Spark SQL using %%sql magic
   - Use %%sql magic in a notebook to run SQL statements.

```
%%sql

UPDATE products
SET Price = 2.49 WHERE ProductId = 1;
```

3. Use the Delta API
   - Create an instance of a DeltaTable from a folder location containing files in delta format, and then use the API to modify the data in the table.

```
from delta.tables import *
from pyspark.sql.functions import *

# Create a DeltaTable object
delta_path = "Files/mytable"
deltaTable = DeltaTable.forPath(spark, delta_path)

# Update the table (reduce price of accessories by 10%)
deltaTable.update(
    condition = "Category == 'Accessories'",
    set = { "Price": "Price * 0.9" })
```

Spark Structured Streaming

Read data from many different kinds of streaming source, including:

- Network ports
- Real time message brokering services such as Azure Event Hubs or Kafka
- File system locations

Use Delta lake as a streaming source

Work with streaming data in a dataframe

```
%%sql
INSERT IN
Price)
VALUES
    (3001
    (3002
    (3003
    (3004
    (3005
```

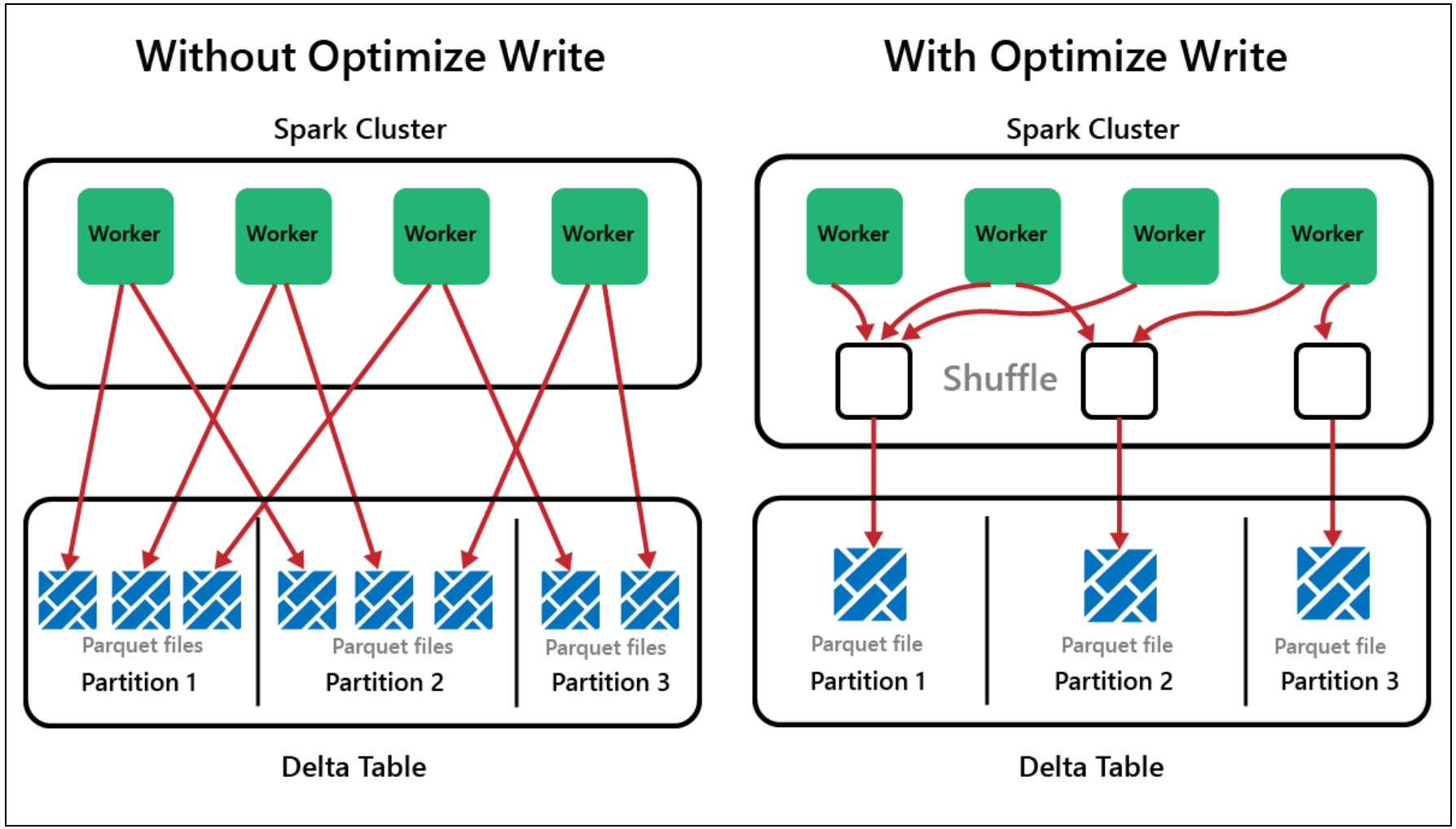
```
# Read and display the input table
df = spark.read.format("delta").table("orders_in")
display(df)

# Load a streaming DataFrame from the Delta table
stream_df = spark.readStream.format("delta") \
    .option("ignoreChanges", "true") \
    .table("orders_in")

# Verify that the stream is streaming
stream_df.isStreaming
```

Use Delta lake as a streaming sink

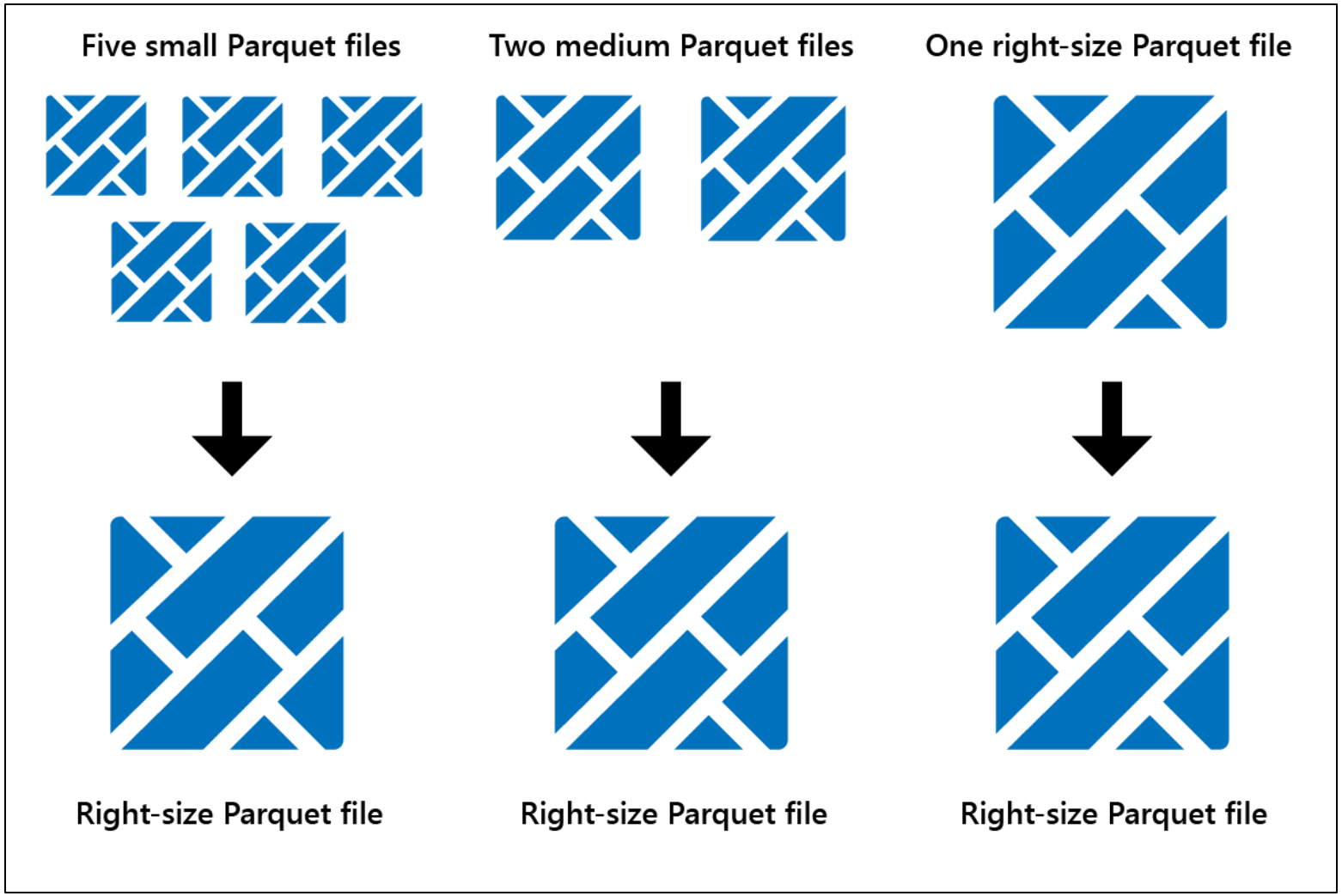
Write to a Delta table, query table, and stop stream.

```
# Write the stream to a delta table
output_table_path = 'Tables/orders_processed'
checkpointpath = 'Files/delta/checkpoint'
deltastream = transformed_df.writeStream.format("delta").option("checkpointLocation", checkpointpath).start(output_table_path)

print("Streaming to orders_processed...")

# Stop the streaming data to avoid excessive processing costs
delta_stream.stop()
```

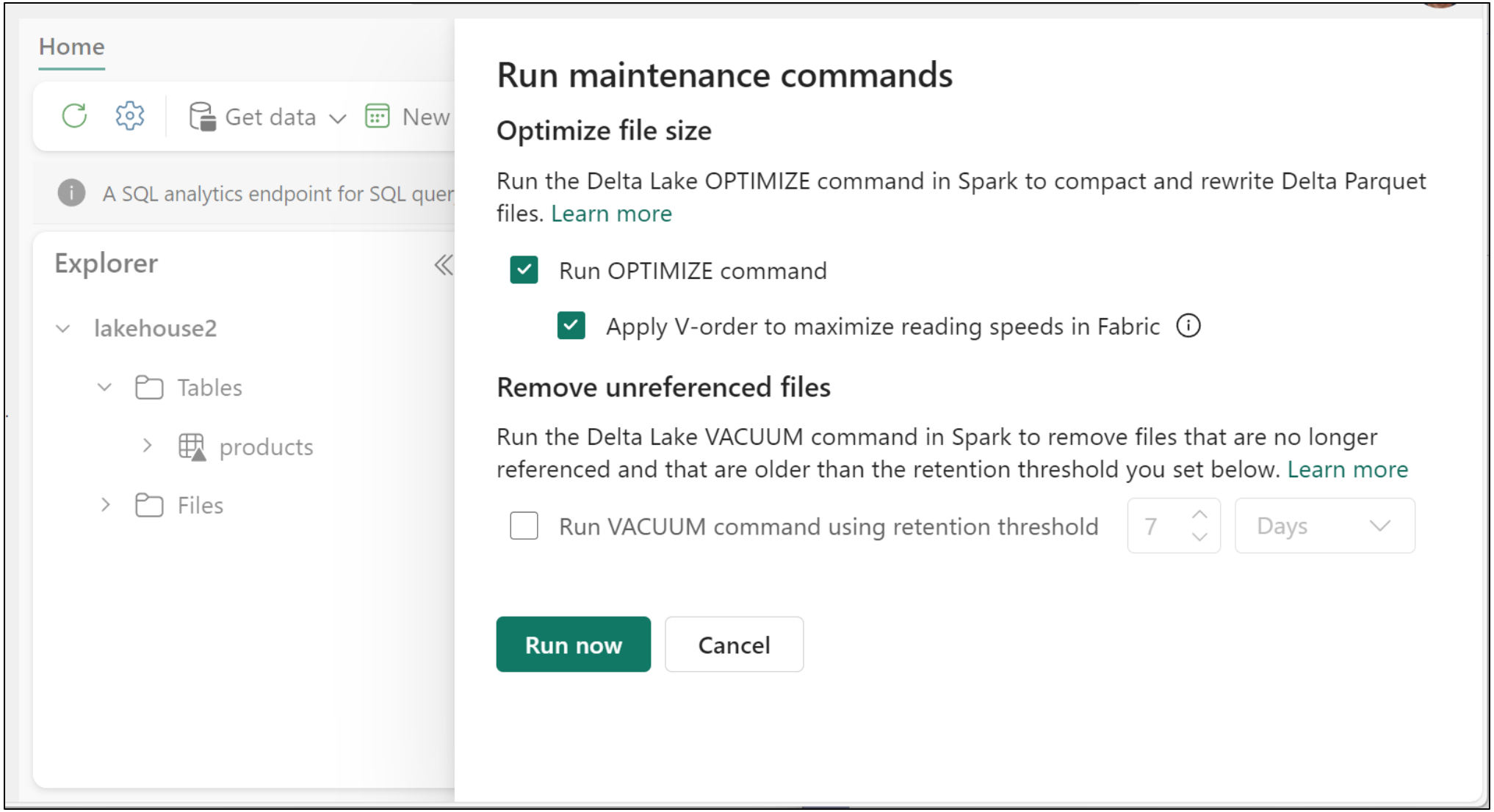
OptimizeWrite function



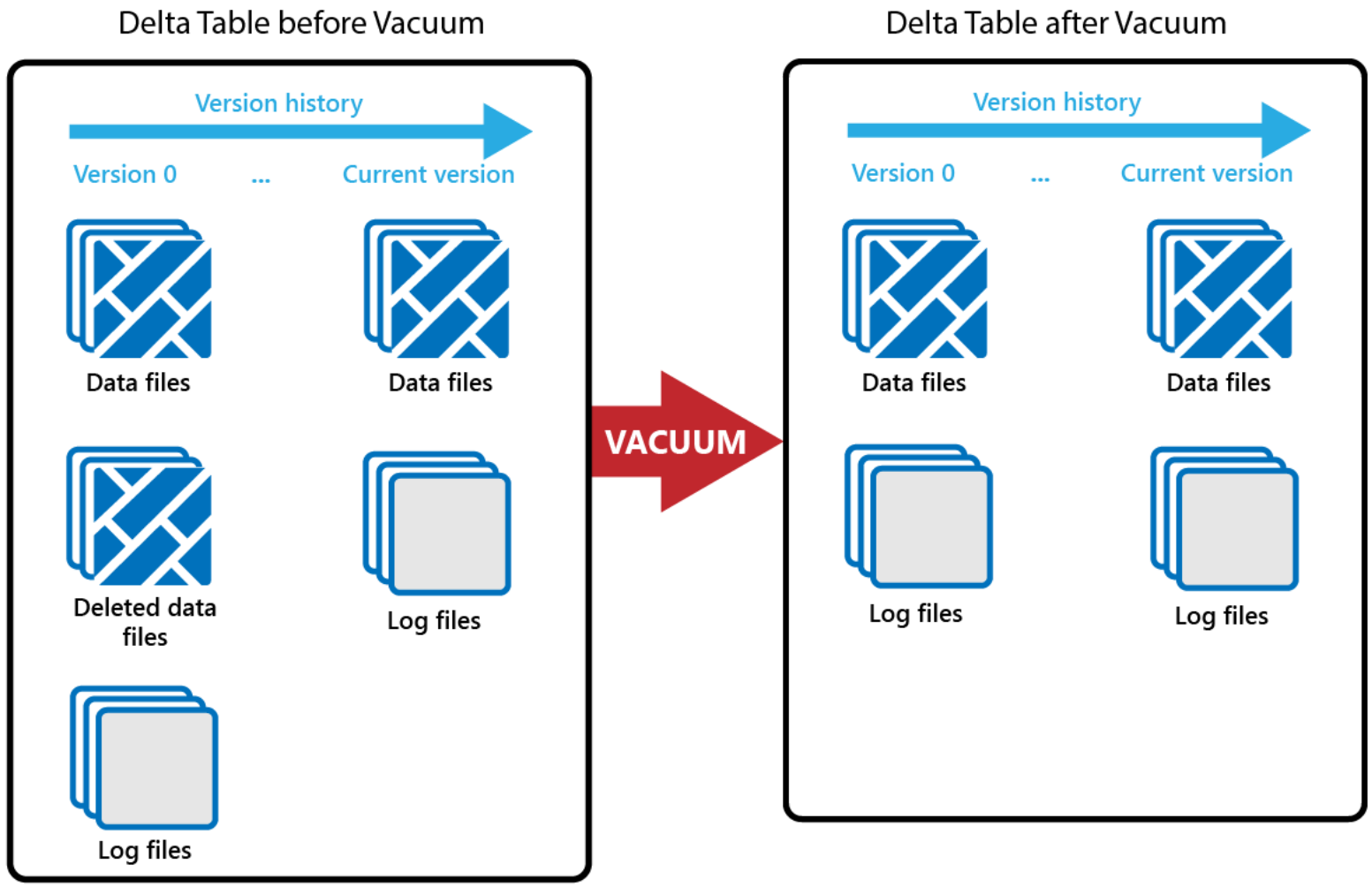
Optimize command



V-Order function



Vacuum command

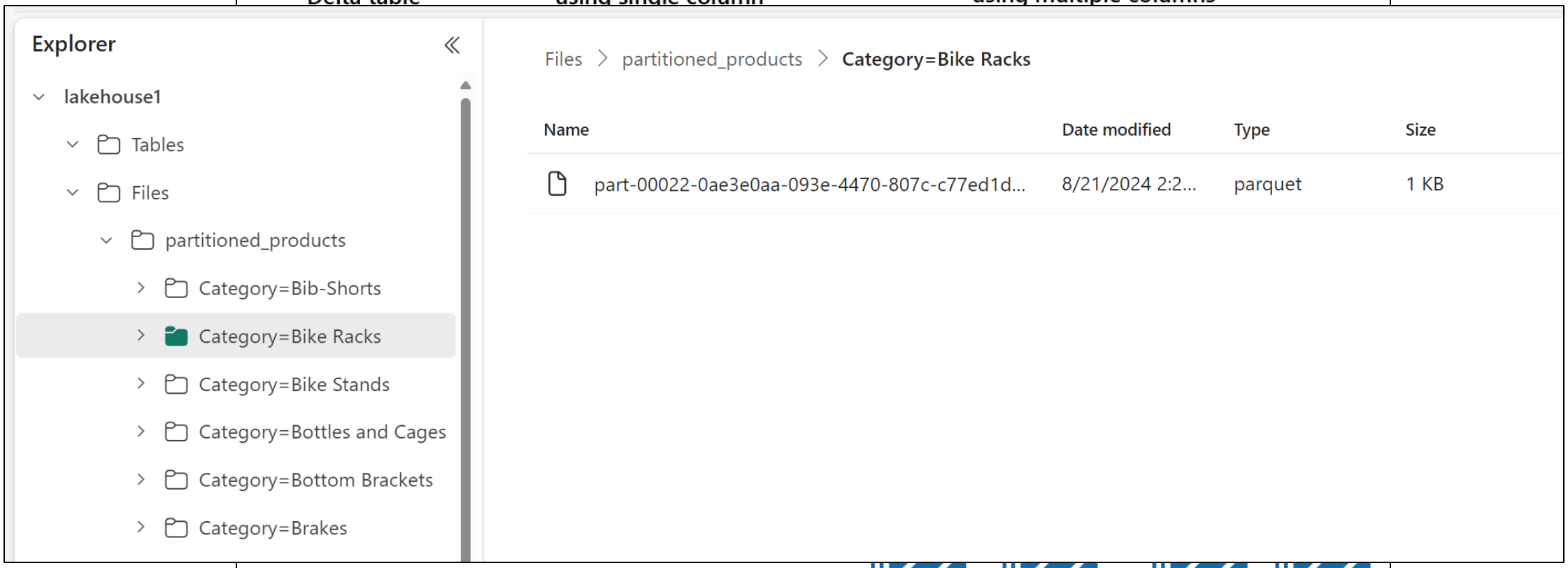


Partition files

Non-partitioned Delta table

Partitioned Delta table using single column

Partitioned Delta table using multiple columns



Parquet files

Data versioning and time travel

Delta tables have transaction logs to “time travel” to previous versions of data.

```
# Use SQL to see the history of a table
```

version	timestamp	operation	operationParameters
2	2023-04-04T21:46:43Z	UPDATE	{"predicate":"(ProductId = 1)"}
1	2023-04-04T21:42:48Z	WRITE	{"mode":"Append","partitionBy":"[]"}
0	2023-04-04T20:04:23Z	CREATE TABLE	{"isManaged":"true","description":null,"partitionBy":"[]","properties":"{}"}

Exercise

Use delta tables in Apache Spark

45 minutes

Knowledge check

1 **Which of the following descriptions best fits Delta Lake?**

- ❑ A Spark API for exporting data from a relational database into CSV files
- ☑ A relational storage layer for Spark that supports tables based on Parquet files
- ❑ A synchronization solution that replicates data between SQL Server and SPark

2 **You've loaded a Spark dataframe with data, that you now want to use in a delta table. What format should you use to write the dataframe to storage?**

- ❑ CSV
- ❑ PARQUET
- ☑ DELTA

3 **You have a managed table based on a folder that contains data files in delta format. If you drop the table, what happens?**

- ☑ The table metadata and data files are deleted
- ❑ The table definition is removed from the metastore, but the data files remain intact
- ❑ The table definition remains in the metastore, but the data files are deleted

Recap

In this section, we covered:

- Delta Lake adds relational database semantics to Apache Spark.
- Microsoft Fabric lakehouse tables are based on Delta Lake.
- Utilize advanced features and techniques through the Delta Lake API.

Further Reading

Work with Delta Lake tables in Microsoft Fabric

https://aka.ms/fabric-delta

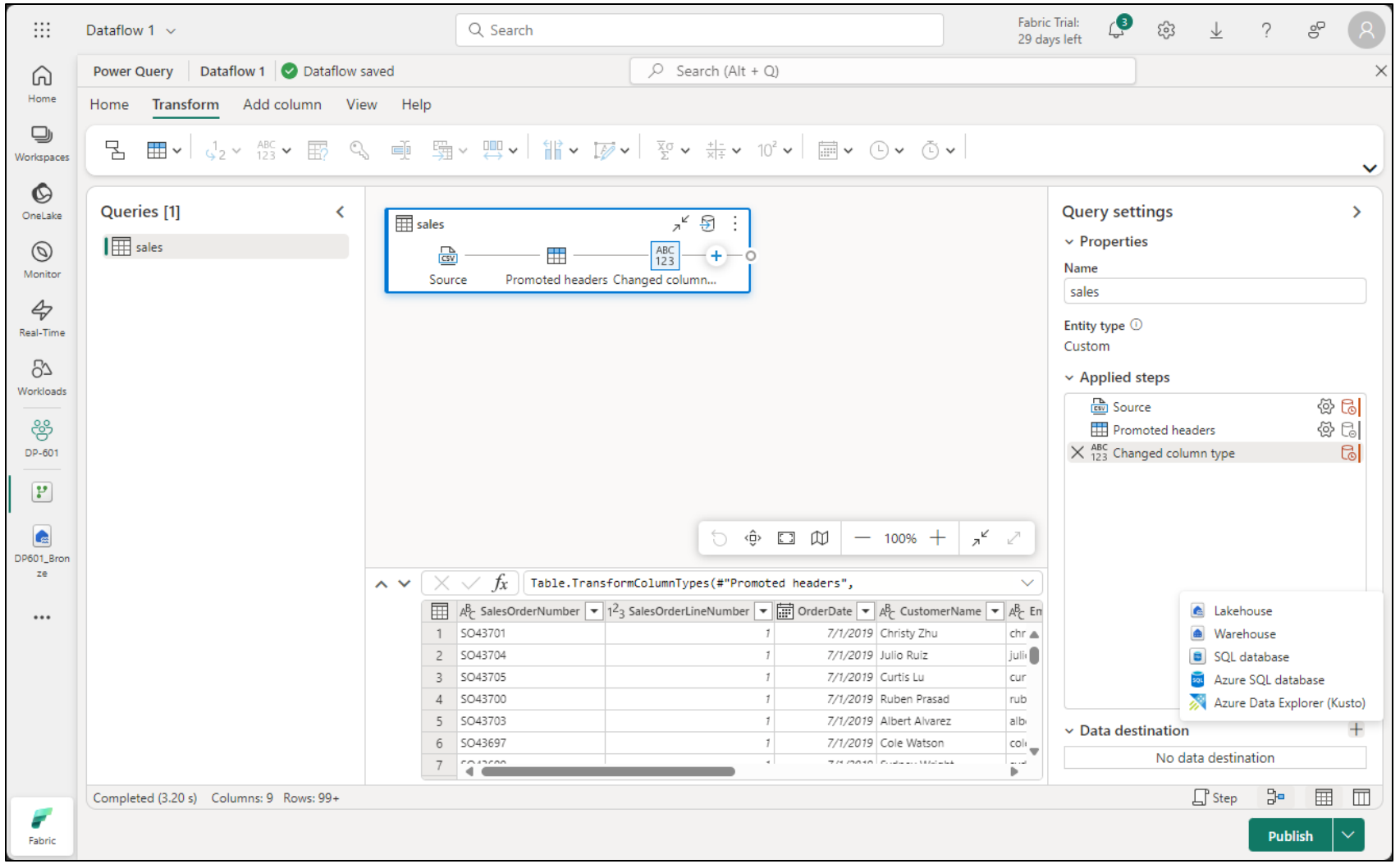
Ingest data with Dataflow Gen2 in Microsoft Fabric

Learning objectives

- Describe Dataflow Gen2 capabilities in Microsoft Fabric
- Create Dataflow solutions to ingest and transform data
- Include a Dataflow in a pipeline

Understand Dataflows Gen2

- Low-code graphical environment for defining ETL solutions
- Extract data from multiple sources, transform it, and load it into a destination
- Run dataflows independently or as an activity in a Pipeline



Dataflow Gen2 benefits and limitations

Benefits:

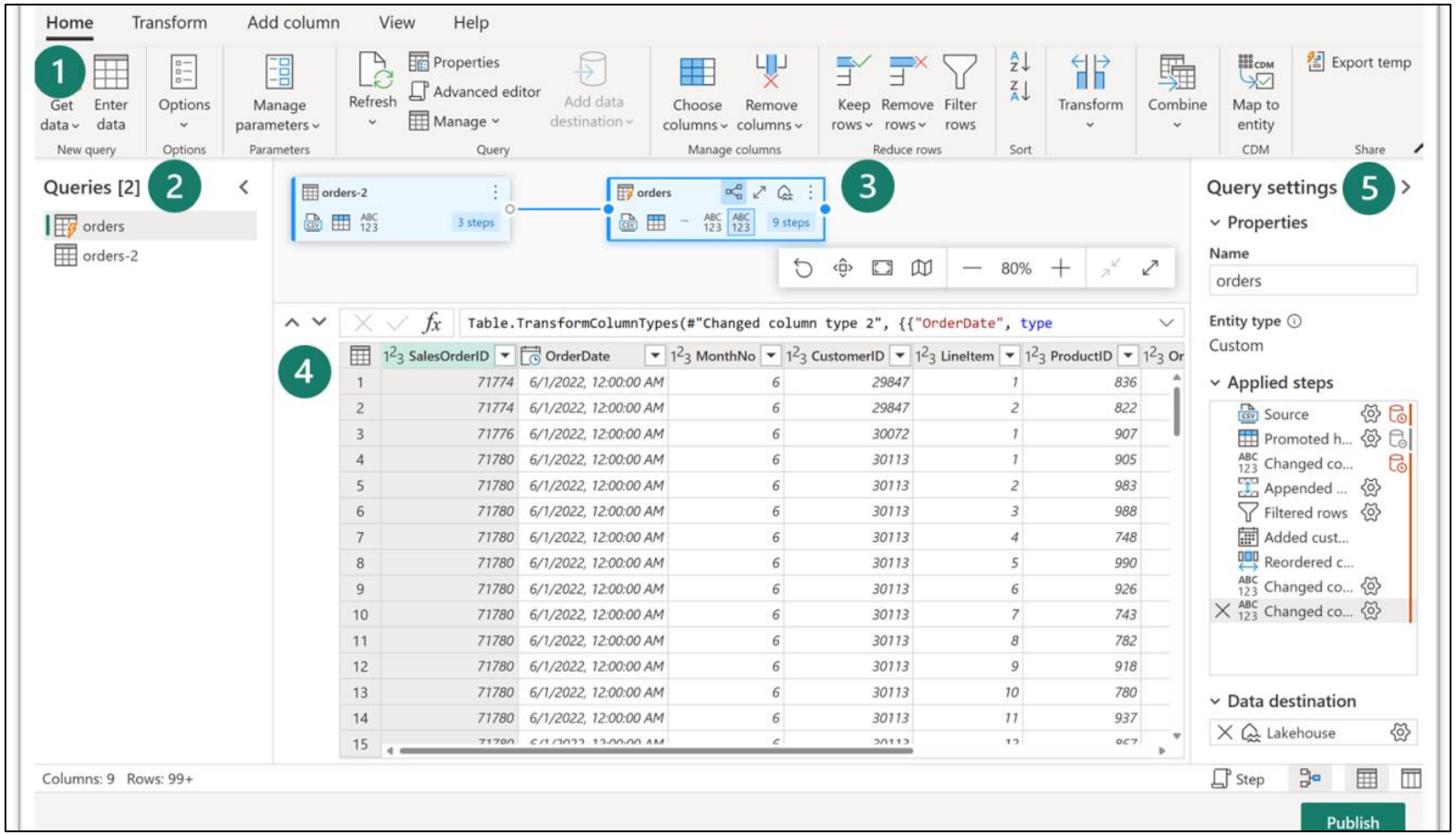
- Self-service access to data.
- Optimize performance with dataflows.
- Simplify data source complexity.
- Ensure data consistency and quality.

Limitations:

- Not a data warehouse replacement.
- No row-level security.
- Requires Fabric capacity workspace.

Explore Dataflow Gen2

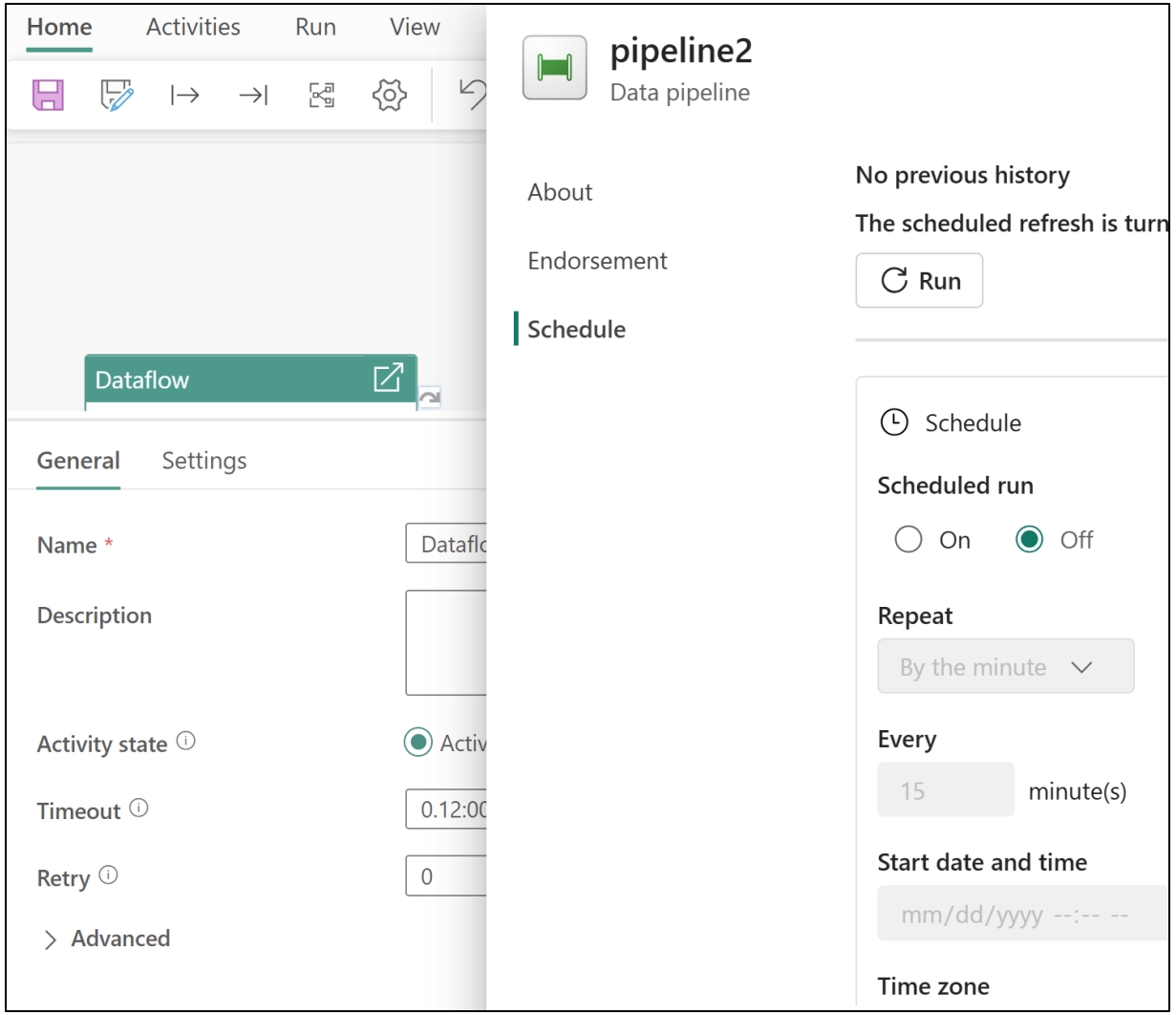
1. Power Query ribbon
2. Queries pane
3. Diagram view
4. Data preview
5. Query settings pane



Integrate Dataflow Gen2 and pipelines

Common activities include:

- Copy data
- Incorporate Dataflow
- Add Notebook
- Get metadata
- Execute a script or stored procedure



Exercise

30 minutes

Create and use a Dataflow Gen2 in Microsoft Fabric

Knowledge check

1 **What is a Dataflow Gen2?**

- ❑ A hybrid database that supports ACID transactions
- ❑ A way to export data to Power BI Desktop
- ☑ A way to import and transform data with Power Query Online

2 **Which Fabric experience lets you create a Dataflow Gen2?**

- ❑ Real-time analytics
- ❑ Data warehouse
- ☑ Data Factory

3 **How do Dataflow Gen2 in Microsoft Fabric facilitate collaboration?**

- ☑ By providing a shared and collaborative environment for data transformation and integration
- ❑ By enabling real-time data streaming and event processing
- ❑ By integrating with popular business intelligence tools for data visualization

Recap

In this section, we covered:

- Dataflow Gen2 capabilities in Microsoft Fabric.
- Dataflow Gen2 solutions to ingest and transform data.
- Using a Dataflow Gen2 in a pipeline.

Further reading

Ingest Data with Dataflow Gen2 in Microsoft Fabric

https://aka.ms/fabric-dataflow

Orchestrate processes and data movement with Microsoft Fabric

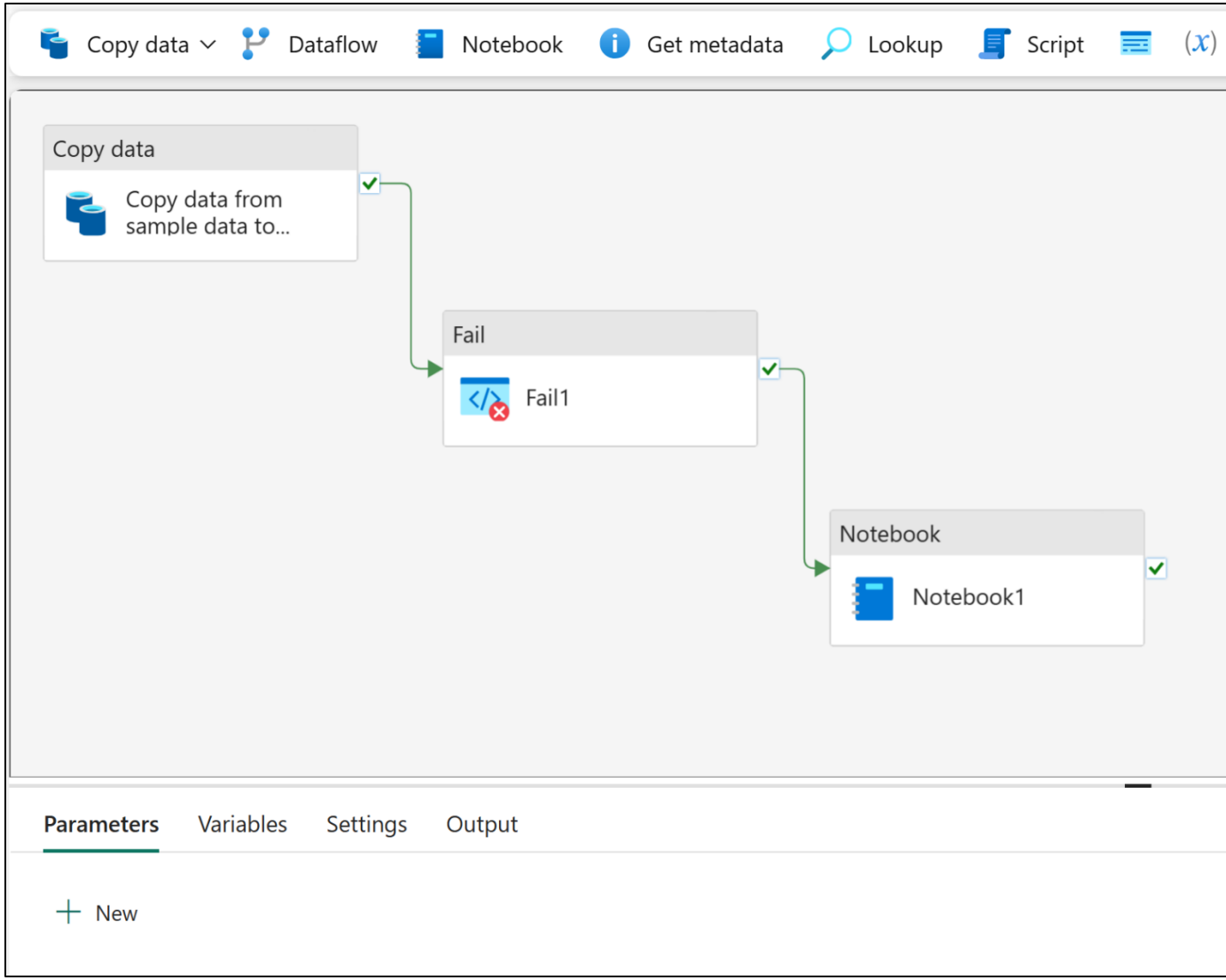
Learning objectives

- Describe pipeline capabilities in Microsoft Fabric.
- Use the Copy Data activity in a pipeline.
- Create pipelines based on predefined templates.
- Run and monitor pipelines.

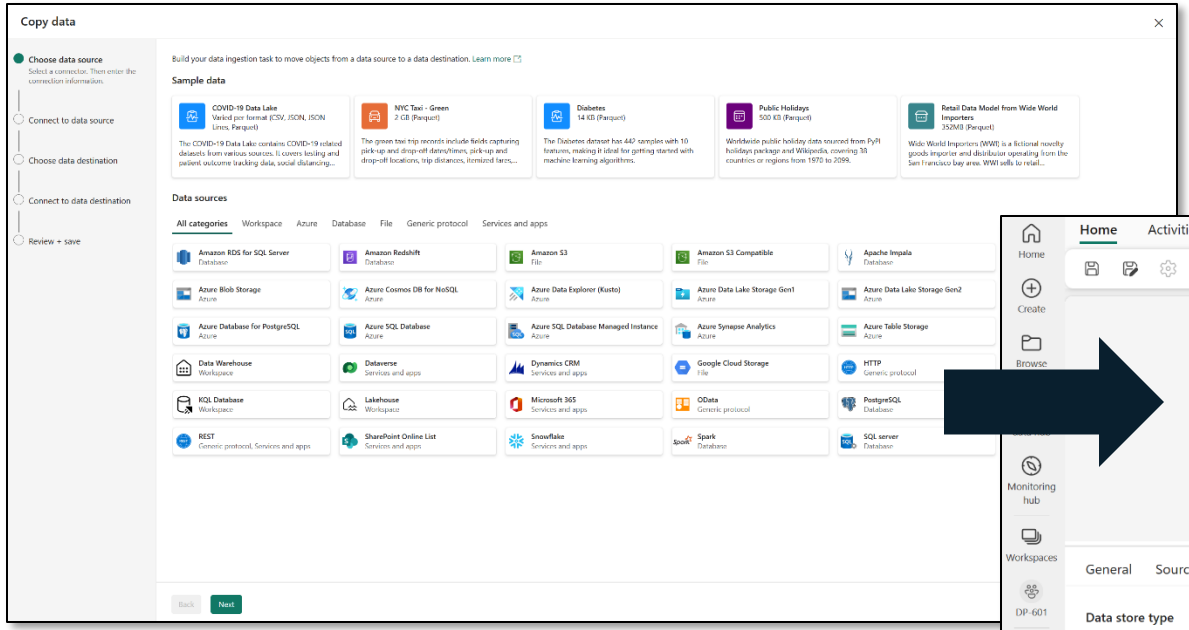
Pipelines in Microsoft Fabric

Pipeline concepts:

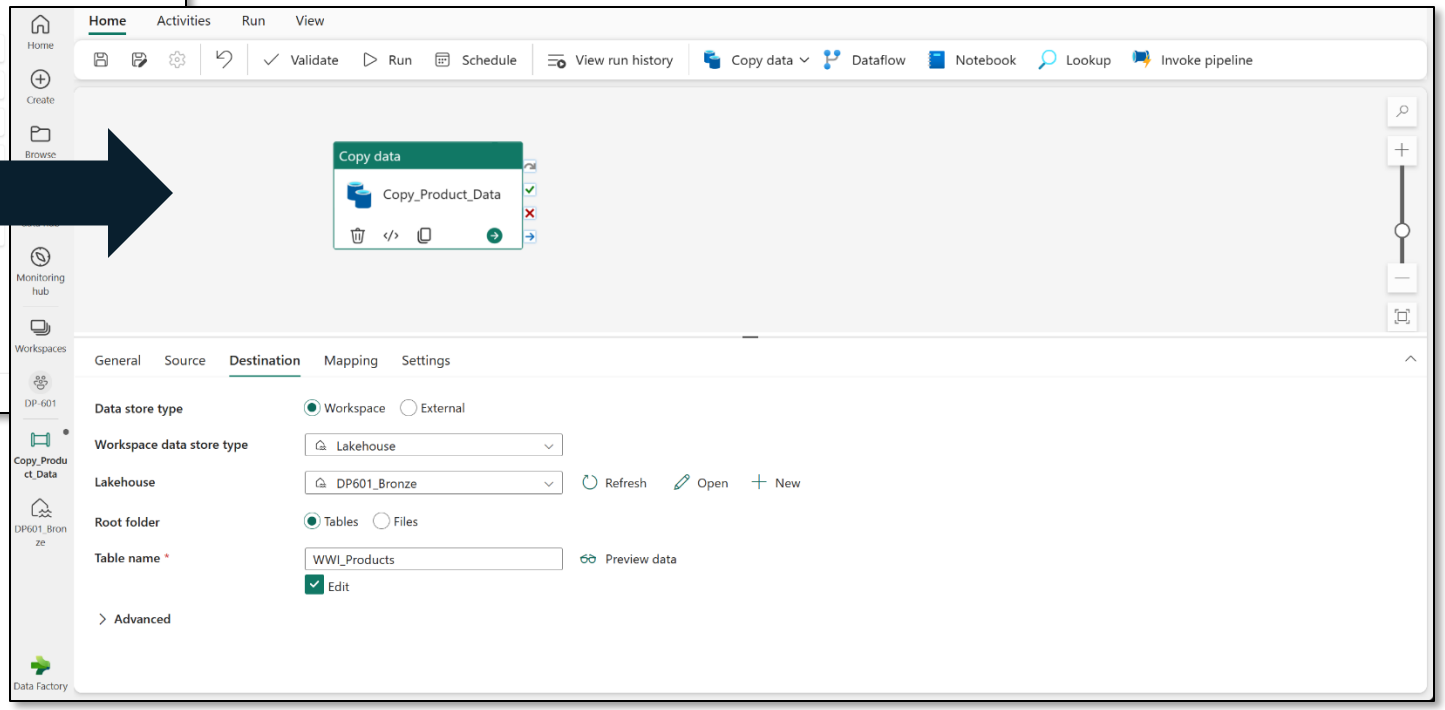
- Activities
  - Data transformation
  - Control flow
- Parameters
- Pipeline runs



Common Activities – copy data

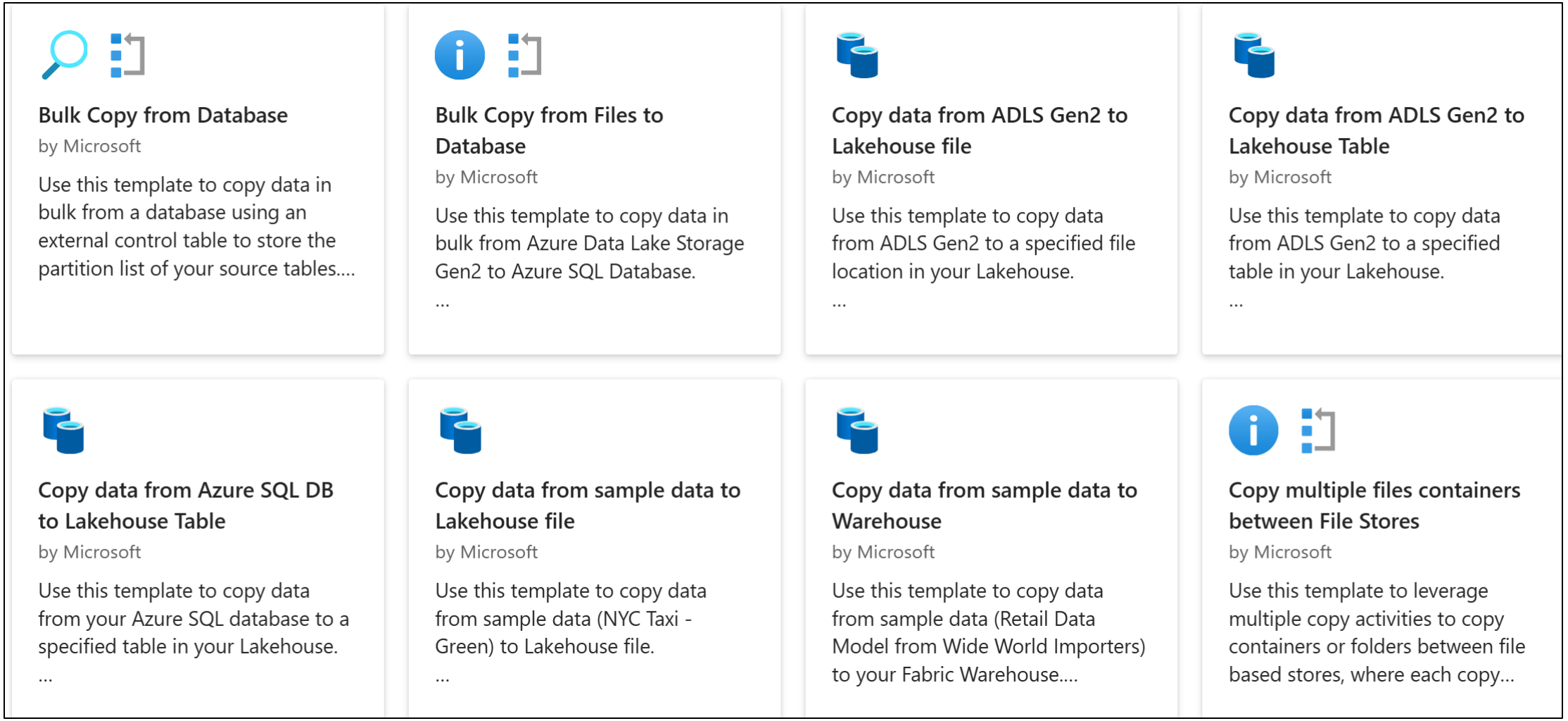


1. Use the copy data tool.



2. Edit the settings below the pipeline canvas.

Use templates for common activities



Bulk Copy from Database
by Microsoft
Use this template to copy data in bulk from a database using an external control table to store the partition list of your source tables....

Bulk Copy from Files to Database
by Microsoft
Use this template to copy data in bulk from Azure Data Lake Storage Gen2 to Azure SQL Database.
...

Copy data from ADLS Gen2 to Lakehouse file
by Microsoft
Use this template to copy data from ADLS Gen2 to a specified file location in your Lakehouse.
...

Copy data from ADLS Gen2 to Lakehouse Table
by Microsoft
Use this template to copy data from ADLS Gen2 to a specified table in your Lakehouse.
...

Copy data from Azure SQL DB to Lakehouse Table
by Microsoft
Use this template to copy data from your Azure SQL database to a specified table in your Lakehouse.
...

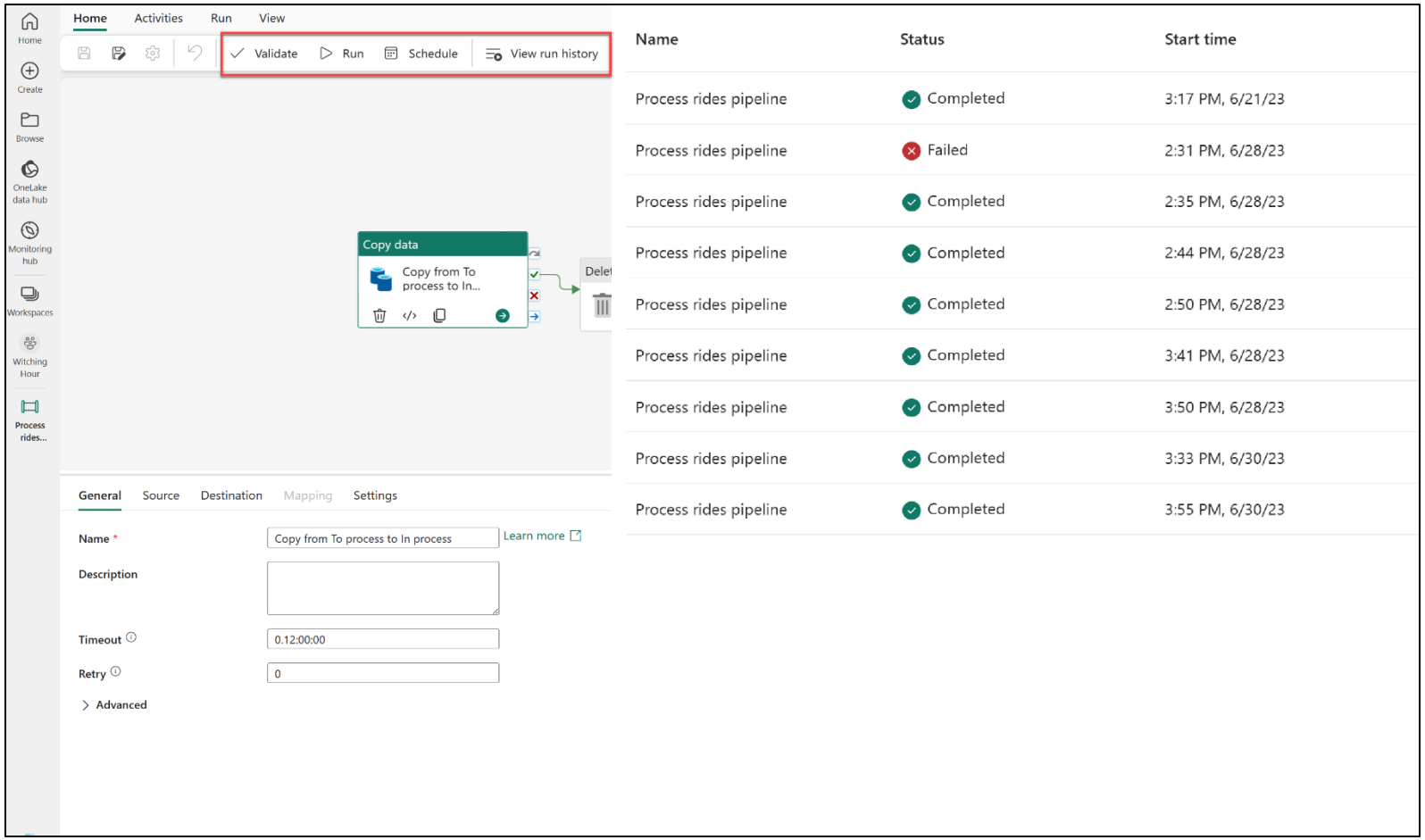
Copy data from sample data to Lakehouse file
by Microsoft
Use this template to copy data from sample data (NYC Taxi - Green) to Lakehouse file.
...

Copy data from sample data to Warehouse
by Microsoft
Use this template to copy data from sample data (Retail Data Model from Wide World Importers) to your Fabric Warehouse....

Copy multiple files containers between File Stores
by Microsoft
Use this template to leverage multiple copy activities to copy containers or folders between file based stores, where each copy...

Run and monitor pipelines

1. Validate
2. Run
3. Schedule
4. View run history



Exercise

Ingest data with a pipeline

30 minutes

Knowledge check

1 **What is a data pipeline?**

- ☐ A special folder in OneLake storage where data can be exported from a lakehouse.
- ☑ A sequence of activities to orchestrate a data ingestion or transformation process.
- ☐ A saved Power Query.

2 **You want to use a pipeline to copy data to a folder with a specified name for each run. What should you do?**

- ☐ Create multiple pipelines – One for each folder name.
- ☐ Use a Dataflow Gen2.
- ☑ Add a parameter to the pipeline and use it to specify the folder name for each run.

3 **You have previously run a pipeline containing multiple activities. What's the best way to check how long each individual activity took to complete?**

- ☐ Rerun the pipeline and observe the output, timing each activity.
- ☑ View the run details in the run history.
- ☐ View the refreshed value for your lakehouse's default semantic model.

Recap

In this section, we covered:

- How pipelines can be used to orchestrate processes and data movement.
- Common activities and ready to use templates.
- Running and monitoring pipelines.

Further Reading

Orchestrate processes and data movement with Microsoft Fabric

https://aka.ms/fabric-pipeline