



# Microsoft Fabric

Lakehouse vs Warehouse

James Serra

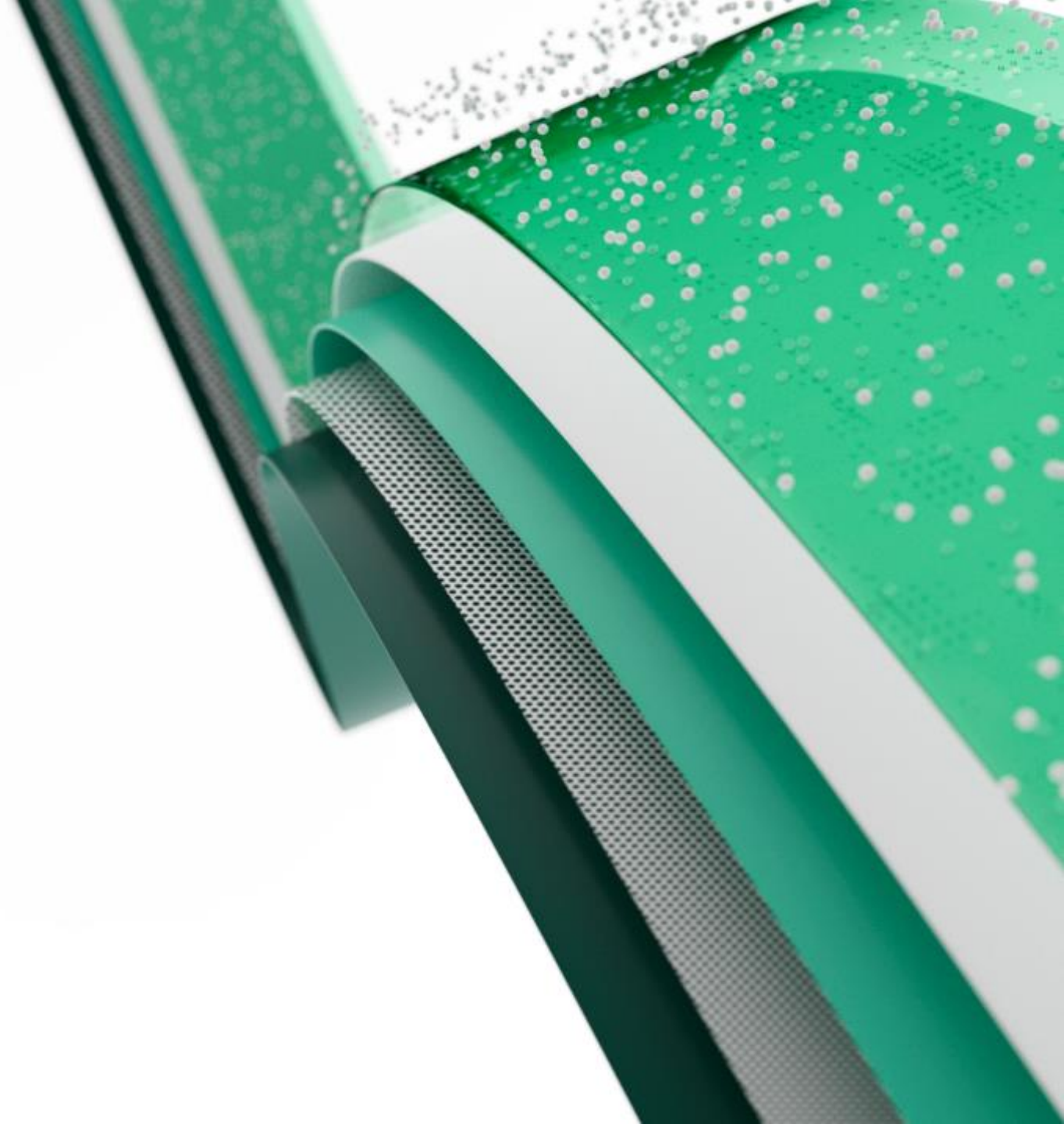
Industry Advisor

Microsoft, Federal Civilian

[jamesserra3@gmail.com](mailto:jamesserra3@gmail.com)

Blog: [JamesSerra.com](https://JamesSerra.com)

7/17/23



# About Me

- Microsoft, Data & AI Solution Architect in Microsoft Federal Civilian
- At Microsoft for most of the last nine years as a Data & AI Architect , with a brief stop at EY
- In IT for 35 years, worked on many BI and DW projects
- Worked as desktop/web/database developer, DBA, BI and DW architect and developer, MDM architect, PDW/APS developer
- Been perm employee, contractor, consultant, business owner
- Presenter at PASS Summit, SQLBits, Enterprise Data World conference, Big Data Conference Europe, SQL Saturdays, Informatica World
- Blog at [JamesSerra.com](http://JamesSerra.com)
- Former SQL Server MVP
- Author of book "Deciphering Data Architectures: Choosing Between a Modern Data Warehouse, Data Fabric, Data Lakehouse, and Data Mesh"

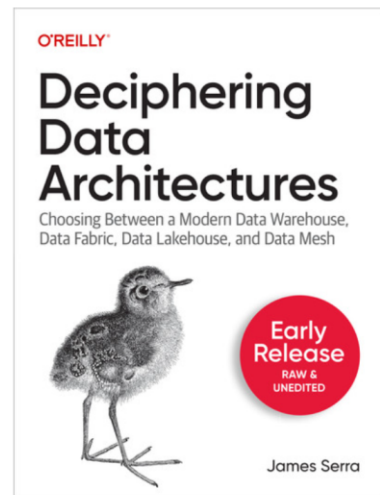


# My upcoming book

## Deciphering Data Architectures

Write the [first review](#)

By [James Serra](#)



TIME TO COMPLETE:  
49m

TOPICS:  
[Data Lake](#)

PUBLISHED BY:  
[O'Reilly Media, Inc.](#)

PUBLICATION DATE:  
September 2024

PRINT LENGTH:  
32 pages

[Table of contents](#)

Continue

Data fabric, data lakehouse, and data mesh have recently appeared as viable alternatives to the modern data warehouse. These new architectures have solid benefits, but they're also surrounded by a lot of hyperbole and confusion. This practical book provides a guided tour of each architecture to help data professionals understand its pros and cons.

In the process, James Serra, big data and data warehousing solution architect at Microsoft, examines common data architecture concepts, including how data warehouses have had to evolve to work with data lake features. You'll learn what data lakehouses can help you achieve, and how to distinguish data mesh hype from reality. Best of all, you'll be able to determine the most appropriate data architecture for your needs. By reading this book, you'll:

- Gain a working understanding of several data architectures
- Know the pros and cons of each approach
- Distinguish data architecture theory from the reality
- Learn to pick the best architecture for your use case
- Understand the differences between data warehouses and data lakes
- Learn common data architecture concepts to help you build better solutions
- Alleviate confusion by clearly defining each data architecture
- Know what architectures to use for each cloud provider

Five chapters available now:

[Deciphering Data Architectures \(oreilly.com\)](#)

- Foundation
  - 1. Big Data (available)
  - 2. Types of Data Architectures (available)
  - 3. The Architecture Design Session (available)
- Common data architecture concepts
  - 4. Relational Data Warehouse (available)
  - 5. Data Lake (available)
  - 6. Approaches to Data Stores
  - 7. Approaches to Design
  - 8. Approaches to Data Modeling
  - 9. Approaches to Data Ingestion
- Data Architectures
  - 10. Modern Data Warehouse (MDW)
  - 11. Data Fabric
  - 12. Data Lakehouse
  - 13. Data Mesh Foundation
  - 14. Data Mesh Adoption
- People, Process, and Technology
  - 15. People and process
  - 16. Technologies

What is the difference between lakehouse and warehouse?



Lakehouse

# Lakehouse

## Data Source



Shortcut Enabled



Structured /  
Unstructured

## Ingestion



Shortcuts



Pipelines &  
Dataflows

## Store



Lakehouse(s)



## Transform

Notebooks &  
Dataflows

## Expose



PBI



Lake Warehouse

OneLake -> Workspace -> Lakehouse

<https://onelake.dfs.fabric.microsoft.com/SerraFabricDemo/DemoLakehouse.Lakehouse/Tables>  
<https://onelake.dfs.fabric.microsoft.com/SerraFabricDemo/DemoLakehouse.Lakehouse/Files>



# Lakehouse – Lakehouse mode

The screenshot shows the Lakehouse explorer interface. On the left, the 'Lakehouse explorer' pane shows a tree view with 'wwilakehouse' expanded to show 'Tables' (containing 'Managed' and 'TestTable') and 'Files' (containing 'Unmanaged'). A tooltip for 'TestTable' shows 'Lakehouse' (Explore your data files and folders) and 'SQL endpoint' (Query data using SQL). The main pane displays a table with columns 'Column1' and 'Column2' and rows: (1, Team, Subject), (2, test3, test4), (3, test5, test99). On the right, another view shows 'Files > wwi-data' with a list of files: 'Australia.csv' and 'Canada.csv'. Below this, text reads: 'Double-click file to view it' and 'Right-click -> Load to Delta table'.

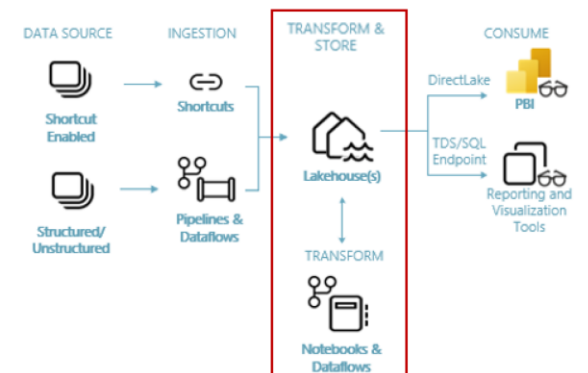
Right click -> View table files

**Table** - This is a virtual view of the managed area in your lake. This is the main container to host tables of all types (CSV, Parquet, Delta, Managed tables and External tables). All tables, whether automatically or explicitly created, will show up as a table under the managed area of the Lakehouse. This area can also include any types of files or folder/subfolder organizations.

**Files** - This is a virtual view of the unmanaged area in your lake. It can contain any files and folders/subfolder's structure. The main distinction between the managed area and the unmanaged area is the automatic delta table detection process which runs over any folders created in the managed area. Any delta format files (parquet + transaction log) will be automatically registered as a table and will also be available from the serving layer (TSQL)

## Automatic Table Discovery and Registration

Lakehouse Table Automatic discovery and registration is a feature of the lakehouse that provides a fully managed file to table experience for data engineers and data scientists. Users can drop a file into the managed area of the lakehouse and the file will be automatically validated for supported structured formats, *which is currently only Delta tables*, and registered into the metastore with the necessary metadata such as column names, formats, compression and more. Users can then reference the file as a table and use SparkSQL syntax to interact with the data. So don't need to explicitly call CREATE TABLE statement to create tables to use with SQL



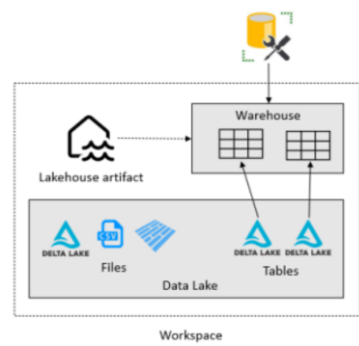
# Lakehouse – SQL endpoint mode

wwilakehouse	Warehouse (default)
wwilakehouse	Lakehouse

- Lakehouse  
Explore your data files and folders
- SQL endpoint  
Query data using SQL

NOTE: "Warehouse mode" was renamed "SQL endpoint"

Can query tables (not files).  
Cannot modify data



## SQL Query

**Explorer**

- Warehouses
- WorldWideImporters
  - Schemas
    - dbo
      - Functions
      - StoredProcedures
      - Tables
        - ag...
        - ag...
        - din
        - din
        - din
        - din
        - dimension\_sto...

**Data preview**

	Date	Calenr
1	2000-11-21T00:00:00.0000000	CY200
2	2000-07-15T00:00:00.0000000	CY200
3	2000-01-25T00:00:00.0000000	CY200
4	2000-11-30T00:00:00.0000000	CY200
5	2000-03-17T00:00:00.0000000	CY200
6	2000-06-20T00:00:00.0000000	CY200
7	2000-11-21T00:00:00.0000000	CY200
8	2000-08-03T00:00:00.0000000	CY200
9	2000-05-08T00:00:00.0000000	CY200
10	2000-03-28T00:00:00.0000000	CY200
11	2000-01-18T00:00:00.0000000	CY200
12	2000-04-22T00:00:00.0000000	CY200
13	2000-02-17T00:00:00.0000000	CY200
14	2000-10-07T00:00:00.0000000	CY200
15	2000-05-17T00:00:00.0000000	CY200
16	2000-05-17T00:00:00.0000000	CY200

## Visual Query

**Visual Query Tools**

- Choose columns, Remove columns (Manage columns)
- Keep rows, Filter rows (Reduce rows)
- Sort (Sort)
- Group by, Replace values, Merge queries, Append queries (Combine)

**Visual Query Diagram**

The diagram shows a data flow starting from a 'Source' table (gold\_aggregate\_sale\_by\_date\_city), moving through 'Merged queries', 'Expanded silver...', and finally to 'Grouped rows'.

**SQL Query Interface**

Home | SQL endpoint

New SQL query | New visual query

**Explorer**

- Warehouses
- DemoLakehouse
  - Schemas
    - dbo
      - Functions
      - StoredProcedures
      - Tables
        - canada
        - customer
        - customer2
      - Views
    - guest
    - INFORMATION\_SCHEMA
    - sys
    - Security
  - Queries
    - My Queries
    - SQL query 2

**SQL query 2**

```

1 SELECT TOP (100) [Country]
2     , [Date]
3     , [ProductID]
4     , [Revenue]
5     , [Units]
6     , [Zip]
7 FROM [DemoLakehouse].[dbo].[canada]
    
```

Run | Save as view

Messages | **Results** | Save as table | Download Excel file | Vis

	Country	Date	ProductID	Revenue
1	Canada	2017-02-26T00:00:00.0000000	1982	1763.68!
2	Canada	2016-07-10T00:00:00.0000000	2054	2729.73!
3	Canada	2018-12-29T00:00:00.0000000	1935	787.185
4	Canada	2018-01-21T00:00:00.0000000	862	1249.13!
5	Canada	2017-03-27T00:00:00.0000000	1006	634.882!
6	Canada	2018-12-29T00:00:00.0000000	1934	787.185
7	Canada	2017-02-27T00:00:00.0000000	764	1826.68!
8	Canada	2017-02-27T00:00:00.0000000	763	1826.68!
9	Canada	2019-07-20T00:00:00.0000000	862	1264.88!



# Lakehouse – shortcuts (to lakehouse)

Lakehouse explorer

- wwilakehouse
  - Tables
    - australia
    - TestTable
  - Unidentified
    - \_mashup\_temporary
    - testfolder1
  - Files
    - Folder1
    - book-recommendation
    - fraud-detection
    - wwi-data
      - Test

Files > wwi-data

Name
Australia.csv
Canada.csv


- Refresh
- New shortcut
- New subfolder
- Upload >
- Rename
- Delete
- Properties

## New shortcut

Use shortcuts to quickly pull data from internal and external locations into your lakehouses, warehouses, or datasets. Shortcuts can be updated or removed from your item, but these changes will not affect the original data and its source.


### Internal sources

Microsoft OneLake  
Fabric




### External sources

Azure Data Lake Storage Gen2  
Azure



Amazon S3  
File



# Lakehouse – shortcuts (to lakehouse)...cont

## Select a data source type



⚠ When accessing this shortcut using a dataset or T-SQL, the identity of the calling item's owner is used to authorize access rather than the user's identity.

ℹ **wwilakehouse** is located in the region **West Central US**. Any data sourced through this shortcut will be processed in the same region.

Find and connect to the data you want to use with your shortcut.

All My data Endorsed in your org

Filter by keyword Filter

Explorer >>

Name	Type	Capacity region	Owner	Location	Endorsement	Sensitivity
wwilakehouse	Lakehouse	West Central US	James Serra	SerraTridentTraining	-	Confidential\Micro... ⓘ
TestWarehouse	Warehouse	West Central US	James Serra	SerraTridentTraining	-	General ⓘ
rta-tutorial-db ⓘ	KQL Database	West Central US	James Serra	SerraTridentTraining	-	Confidential\Micro... ⓘ
WorldWideImporters	Warehouse	West Central US	James Serra	SerraTridentTraining	-	Confidential\Micro... ⓘ
serratestlakehouse1	Lakehouse	West Central US	James Serra	TridentSerra	-	Confidential\Micro... ⓘ
SerraTridentLakehouse	Lakehouse	West Central US	James Serra	TridentSerra	-	Confidential\Micro... ⓘ
test1dw	Warehouse	West Central US	James Serra	TridentSerra	-	-

Previous

Next

Cancel

## New shortcut

⚠ When accessing this shortcut using a dataset or T-SQL, the identity of the

ℹ **wwilakehouse** is located in the region **West Central US**. Any data sourced

Find and connect to the data you want to use with your shortcut.

OneLake

- ▼ serratestlakehouse1
  - > Tables
  - ▼ Files
    - >  testfolder1

# Lakehouse – shortcuts (to Data Warehouse)

You can create shortcuts to your Data Warehouse. This allows you to access tables in your Data Warehouse directly from your Lakehouse without having to orchestrate any data movement.

Home

Get data ▾ New Power BI

Lakehouse

- wwilakehouse
  - Tables
    - australia
    - TestTable
    - Unidentified 2
  - Files
    - Folder1

New data pipeline

New Dataflow Gen2

New shortcut

## New shortcut

Use shortcuts to quickly pull data from internal and external sources. Shortcuts do not affect the original data and its source.

### Internal sources

Microsoft OneLake  
Fabric

## Select a data source type

⚠ When accessing this shortcut using a dataset or T-SQL, the identity of the user will be the user who created the shortcut.

ℹ **wwilakehouse** is located in the region **West Central US**. Any data sources accessed through this shortcut will also be in this region.

Find and connect to the data you want to use with your shortcut.

All My data Endorsed in your org

»	Name	Type
🏠	TestWarehouse	Warehouse

## Lakehouse explorer

- wwilakehouse
  - Tables
    - australia
    - TestTable
  - Unidentified
  - SerraTestTable

## New shortcut

⚠ When accessing this shortcut using a dataset or T-SQL, the identity of the user will be the user who created the shortcut.

ℹ **wwilakehouse** is located in the region **West Central US**. Any data sources accessed through this shortcut will also be in this region.

Find and connect to the data you want to use

- OneLake
  - TestWarehouse
    - Tables
      - dbo
      - SerraTestTable

**Note:** Data Warehouse tables are automatically mirrored to OneLake using Delta Lake logs. This function is supported for tables that are Insert only. When a Delete or Update occurs, new data will not be available in OneLake or accessible via the shortcut.

Data Warehouse

# Data warehouse

## Data Source



Shortcut Enabled



Structured /  
Unstructured



## Ingestion



Mounts



Pipelines &  
Dataflows



## Store



Data Warehouse



## Transform

Procedures



## Expose



PBI



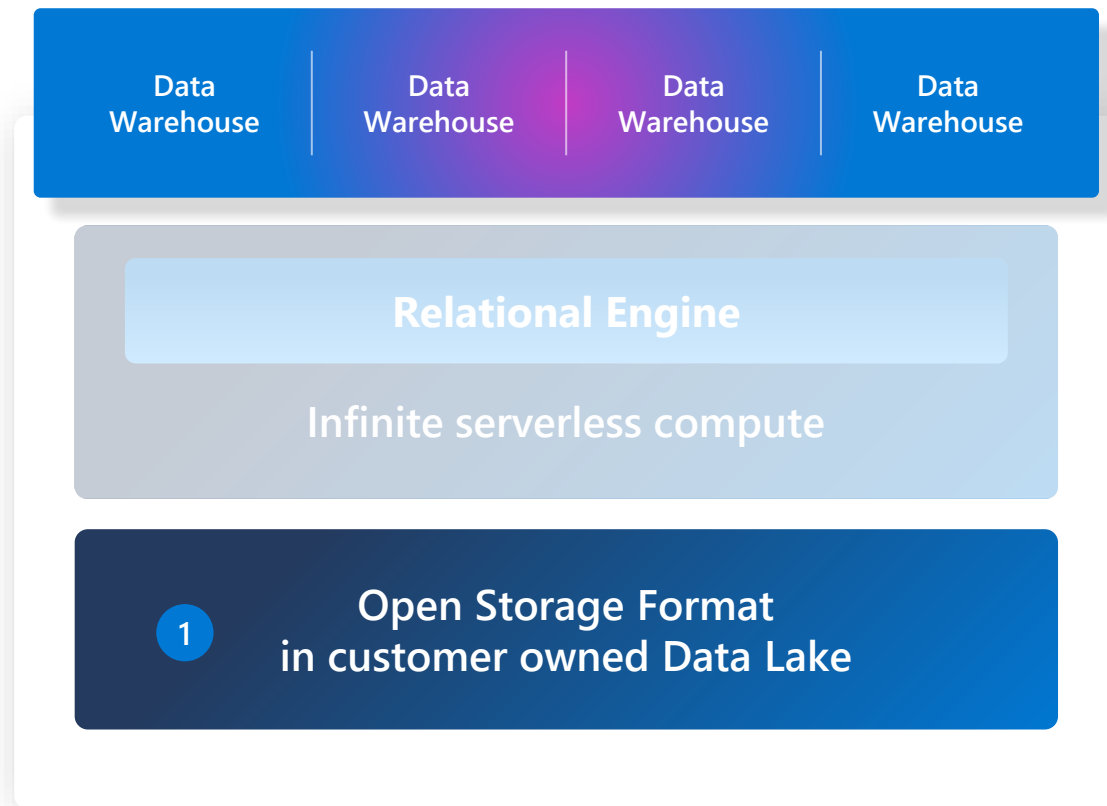
Warehouse

# Synapse Data Warehouse

Infinitely scalable and open



## Synapse Data Warehouse in Fabric



### 1 Open standard format in an open data lake replaces proprietary formats as the native storage

- First transactional data warehouse natively embracing an open standard format
- Data is stored in Delta – Parquet with no vendor lock-in
- Is auto-integrated and auto-optimized with minimal knobs
- Extends full SQL ecosystem benefits

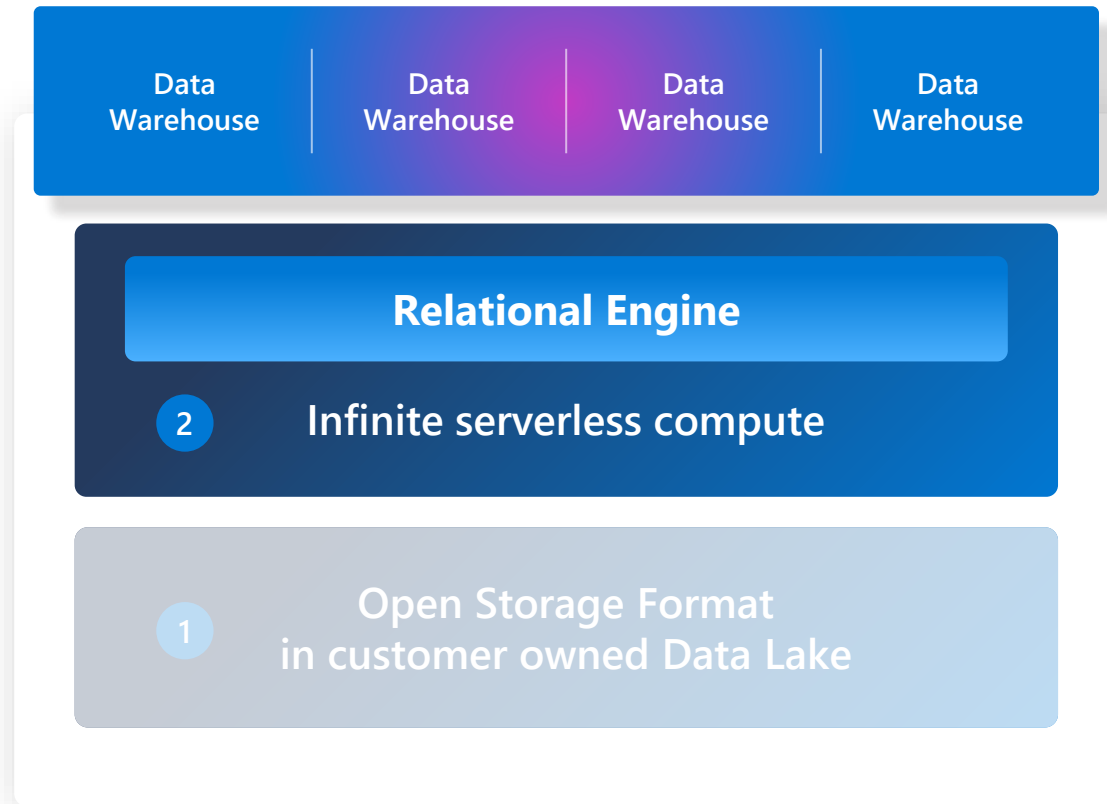


# Synapse Data Warehouse

Infinitely scalable and open



## Synapse Data Warehouse in Fabric



## 2 Dedicated clusters are replaced by serverless compute infrastructure

- Physical compute resources assigned within milliseconds to jobs
- Infinite scaling with dynamic resource allocation tailored to data volume and query complexity
- Instant scaling up/down with no physical provisioning involved
- Resource pooling providing significant efficiencies and pricing

Available now

## Virtual warehouses and cross database querying

- Build virtual warehouses by creating Lakehouses with shortcuts to data in the Lake
- Query across Warehouse and Lakehouse SQL Endpoint with zero data movement
- Write T-SQL query with 3-part-naming for cross-joining

The screenshot displays a data analytics application interface. On the left, the 'Explorer' pane is highlighted with a red box, showing a tree view of data sources including 'Warehouses', 'sales', 'marketing', 'tpcd1tb', and 'demolake'. The main area shows a SQL query window with the following code:

```
1 SELECT TOP 100 dt.d_year, item.i_brand_id brand_id, item.i_brand brand, SUM(ss_ext_sales_price) sum_agg
2 FROM date_dim dt, store_sales, item
3 WHERE dt.d_date_sk = store_sales.ss_sold_date_sk
4 AND store_sales.ss_item_sk = item.i_item_sk
5 AND item.i_manufact_id = 931
6 AND dt.d_moy=11
7 GROUP BY dt.d_year, item.i_brand, item.i_brand_id
8 ORDER BY dt.d_year, sum_agg desc, brand_id;
```

Below the query, the 'Results' tab shows a table with 13 rows and 5 columns: d\_year, brand\_id, brand, and sum\_agg. The results are as follows:

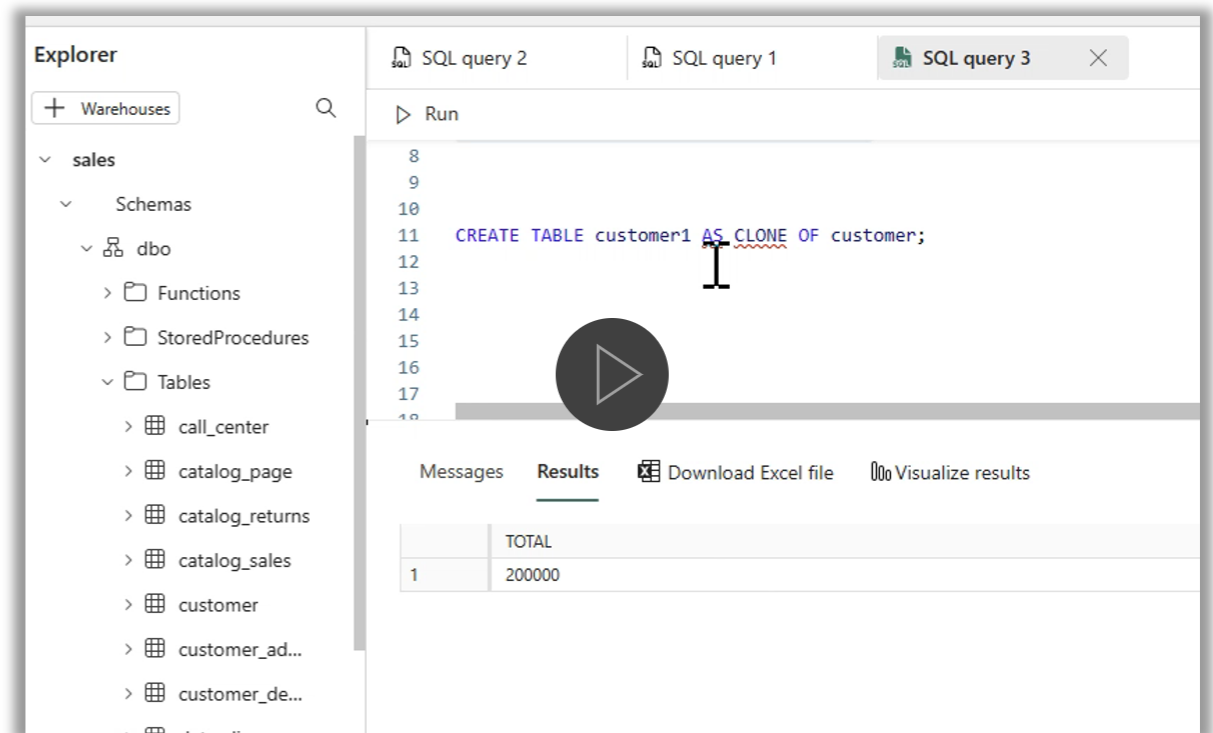
d_year	brand_id	brand	sum_agg
1998	3002002	importoexporti #2	57072.23
1998	2002001	importoimporto #1	55121.24
1998	5003002	exportischolar #2	43818.97
1998	5003001	exportischolar #1	43427.27
1998	5004001	edu packscholar #1	35833.07
1998	7005004	scholarbrand #4	35621.42
1998	8012006	importomaxi #6	33982.90
1998	1004002	edu packamalg #2	32570.71
1998	8004005	edu packnameless #5	31966.71
1998	3004002	edu packesporti #2	30722.59
1998	9012003	importounivamalg #3	28539.16
1998	7015007	scholarnameless #7	28369.62
1998	7016008	corpnameless #8	27960.94

The interface also shows a 'Messages' tab with a success message: 'Succeeded (2 sec 451 ms)'. The bottom status bar indicates 'Columns: 4 Rows: 100'.

Coming  
June '23

## Cloning

- Create zero-copy table clones as of current point-in-time
- Zero-copy Warehouse clones coming in CYQ3'23



# Data Warehouse – query Lakehouse tables

Anything in the tables section of the lakehouse (Delta today, potentially more formats will be available in the future) you can query this using a 3-part name from the data warehouse

Name	Type
Demo Notebook	Notebook
DemoLakehouse	Dataset (default)
DemoLakehouse	SQL endpoint
DemoLakehouse	Lakehouse
DemoWarehouse	Dataset (default)
DemoWarehouse	Warehouse

Home

Get data

New Pow

Explorer

- ▼ DemoLakehouse
  - ▼ Tables
    - > canada
      - > customer

Files

Explorer

- + Warehouses
- ▼ DemoWarehouse
  - > Schemas
  - > Security
- ▼ Queries
  - ▼ My Queries
    - SQL query 1
    - SQL query 2
    - SQL query 3
    - SQL query 4
    - SQL query 5
  - Shared Queries

SQL query 4

Run Save as view

```
1 select * from DemoLakehouse.dbo.canada
```

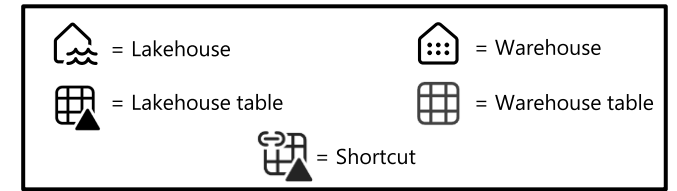
Messages Results Save as table Download Excel file

	ProductID	Date
1	1982	2017-02-26T00:00:00.0000000
2	2054	2016-07-10T00:00:00.0000000
3	1935	2018-12-29T00:00:00.0000000
4	862	2018-01-21T00:00:00.0000000

# Lakehouse vs Warehouse



# Microsoft Fabric



Spark Engine



Synapse Data Engineering



Synapse Data Warehousing

SQL Engine

Use Spark Notebooks



Use SQL Queries & Stored Procedures



SSMS  
Azure Data Studio



Full T-SQL support\*



Data pipeline



Dataflow Gen2



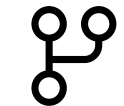
Write data into Lakehouse tables



Write data into Warehouse tables



Data pipeline



Dataflow Gen2

\* See limitations at <https://learn.microsoft.com/en-us/fabric/data-warehouse/tsql-surface-area>



# T-SQL limitations – SQL Endpoint or Warehouse

At this time, the following list of commands is NOT currently supported. Don't try to use these commands because even though they may appear to succeed, they could cause issues to your warehouse.

- ALTER TABLE ADD/ALTER/DROP COLUMN
- BULK LOAD
- CREATE ROLE
- CREATE SECURITY POLICY - Row Level Security (RLS)
- CREATE USER
- GRANT/DENY/REVOKE
- Hints
- Identity Columns
- Manually created multi-column stats
- MASK and UNMASK (Dynamic Data Masking)
- MATERIALIZED VIEWS
- MERGE
- OPENROWSET
- PREDICT
- Queries targeting system and user tables
- Recursive queries
- Result Set Caching
- Schema and Table names can't contain / or \
- SELECT - FOR (except JSON)
- SET ROWCOUNT
- SET TRANSACTION ISOLATION LEVEL
- sp\_showmemo\_xml
- sp\_showspaceused
- sp\_rename
- Temp Tables
- Triggers
- TRUNCATE

<https://learn.microsoft.com/en-us/fabric/data-warehouse/tsql-surface-area>

# Data warehouse and lakehouse properties

	Data warehouse	Lakehouse	Power BI Datamart
<b>Data volume</b>	Unlimited	Unlimited	Up to 100 GB
<b>Type of data</b>	Structured	Unstructured, semi-structured, structured	Structured
<b>Primary developer persona</b>	Data warehouse developer, SQL engineer	Data engineer, data scientist	Citizen developer
<b>Primary developer skill set</b>	SQL	Spark (Scala, PySpark, Spark SQL, R)	No code, SQL
<b>Data organized by</b>	Databases, schemas, and tables	Folders and files, databases and tables	Database, tables, queries
<b>Read operations</b>	Spark, T-SQL	Spark, T-SQL	Spark, T-SQL, Power BI
<b>Write operations</b>	T-SQL	Spark (Scala, PySpark, Spark SQL, R)	Dataflows, T-SQL
<b>Multi-table transactions</b>	Yes	No	No
<b>Primary development interface</b>	SQL scripts	Spark notebooks, Spark job definitions	Power BI
<b>Security</b>	Object level (table, view, function, stored procedure, etc.), column level, row level, DDL/DML	Row level, table level (when using T-SQL), none for Spark	Built-in RLS editor
<b>Access data via shortcuts</b>	Yes (indirectly through the lakehouse)	Yes	No
<b>Can be a source for shortcuts</b>	Yes (tables)	Yes (files and tables)	No
<b>Query across items</b>	Yes, query across lakehouse and warehouse tables	Yes, query across lakehouse and warehouse tables; query across lakehouses (including shortcuts using Spark)	No

Why two options?

Delta lake shortcomings:

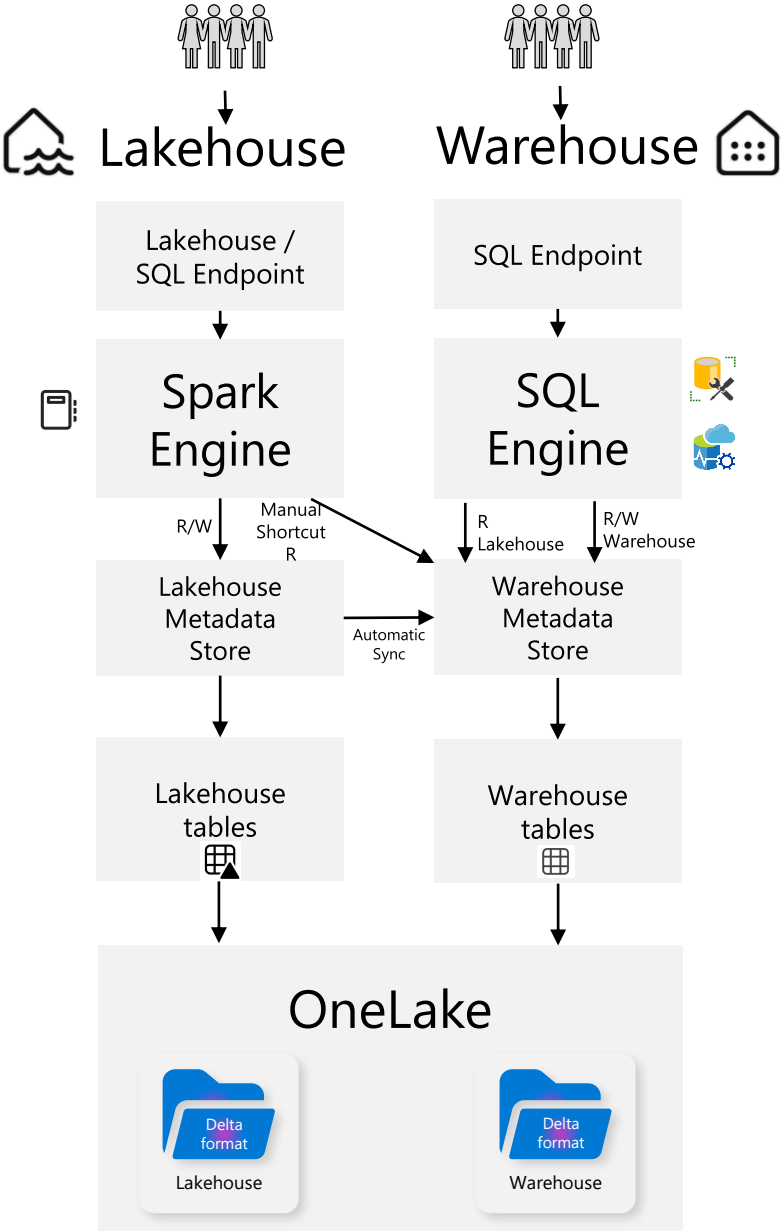
- No multi-table transactions
- Lack of full T-SQL support (can't update data with T-SQL, [limited T-SQL support for reads](#))
- Performance problem for trickle transactions (updates to a few rows in Spark)
- Does not support schemas

# Warehouse vs Lakehouse SQL Endpoint

	Warehouse and Lakehouse SQL Endpoint (SQL Engine)
Warehouse tables in same workspace	R/W, no shortcuts needed
Warehouse tables in different workspace	R, within Lakehouse need shortcut to Warehouse in different workspace (only see inserts) Note: Warehouses don't support shortcuts
Lakehouse tables in same workspace	R, no shortcuts needed (use Spark engine for writes)
Lakehouse tables in different workspace	R, within Lakehouse need shortcut to Lakehouse in different workspace

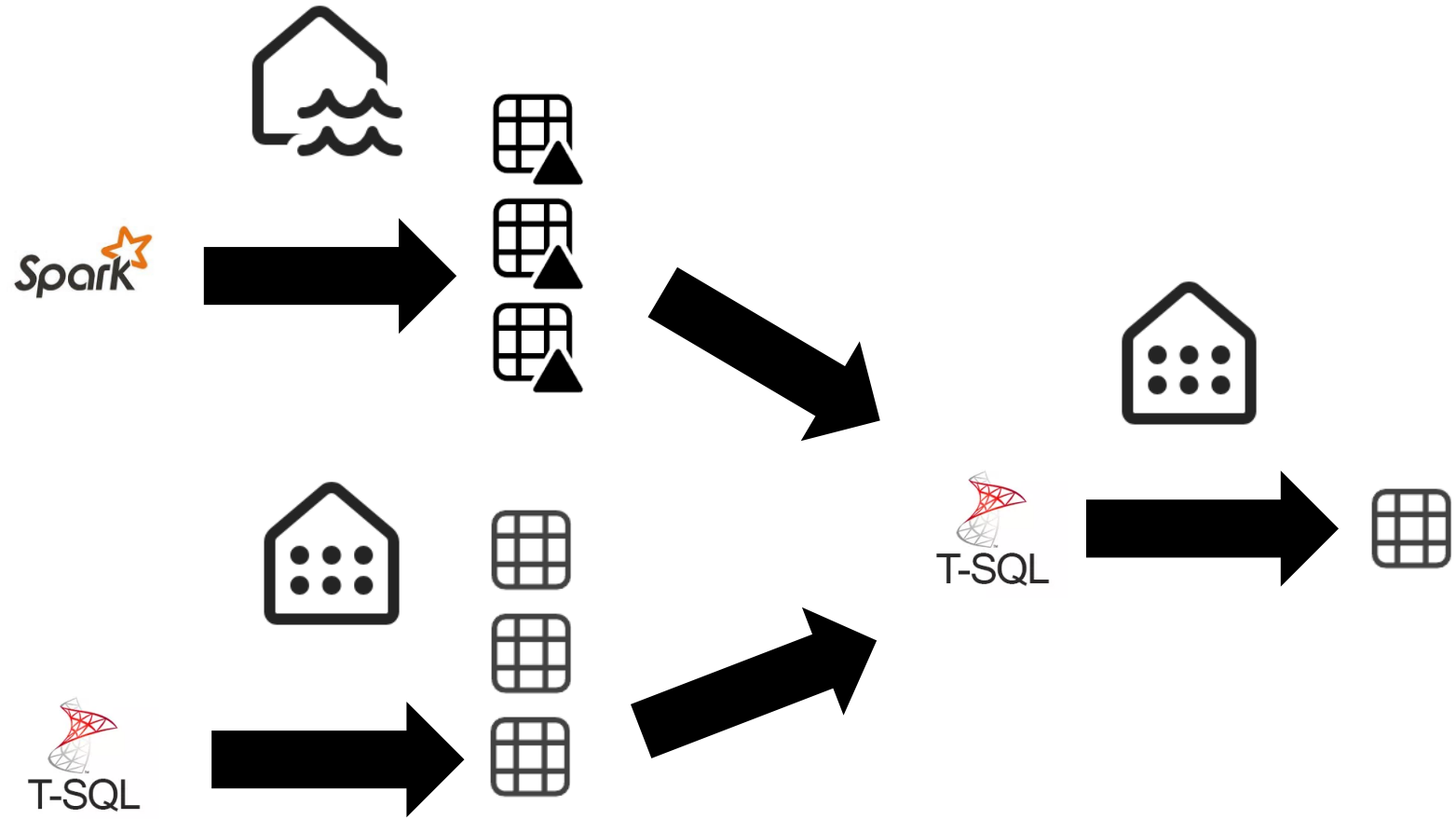
- T-SQL endpoint: one per workspace. Each warehouse and each lakehouse shows up as a database. This is the way you would connect using SSMS.
- SQL Endpoint: a workspace item, which has a SQL designer, editor, and saved queries. One per lakehouse and warehouse.

# Spark/Notebook vs SQL Engine/SSMS



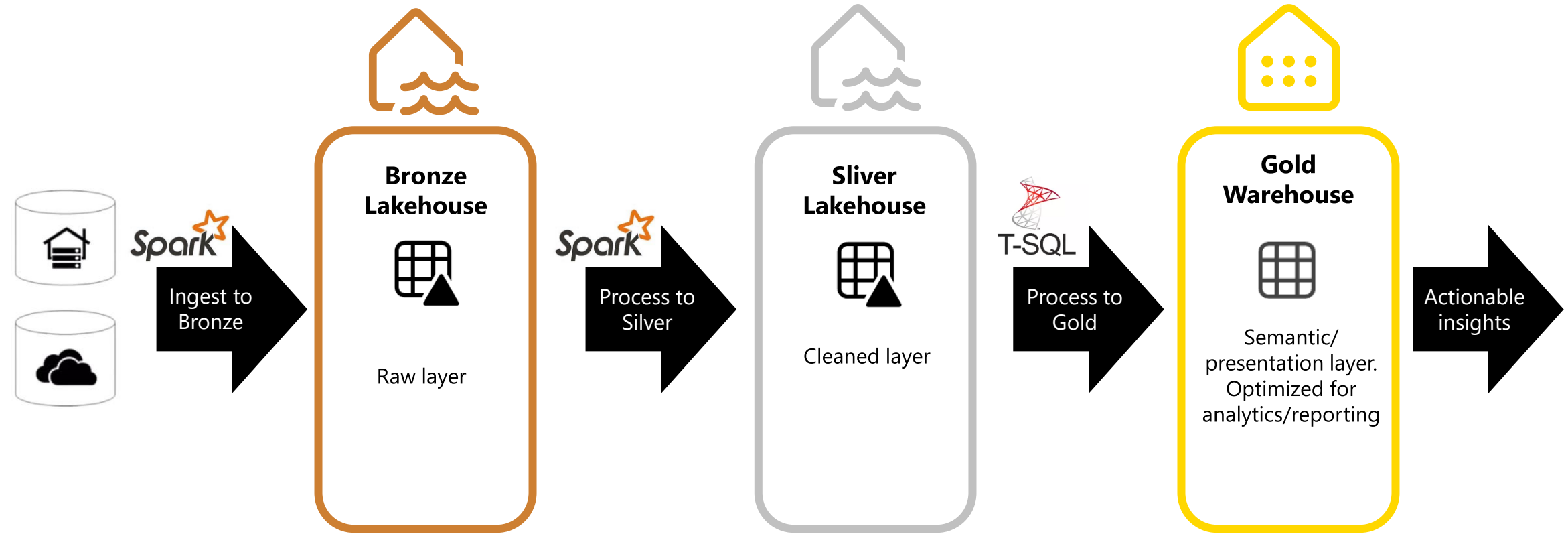


# Microsoft Fabric





# Microsoft Fabric





Q & A



James Serra, Microsoft, Industry Advisor

Email me at: [jamesserra3@gmail.com](mailto:jamesserra3@gmail.com)

Follow me at: @JamesSerra

Link to me at: [www.linkedin.com/in/JamesSerra](http://www.linkedin.com/in/JamesSerra)

Visit my blog at: [JamesSerra.com](http://JamesSerra.com)