



EUROPEAN **MiCROSOFT** **FABRiC**

Community Conference

STOCKHOLM 24-27 SEPTEMBER 2024

► JOIN THE CONVERSATION

#FABCONEUROPE





Drive Efficiency and Reliability Using Metadata-Driven Frameworks



Erwin de Kreuk

Principal Consultant Data & AI, InSpark, Netherlands



We Are InSpark

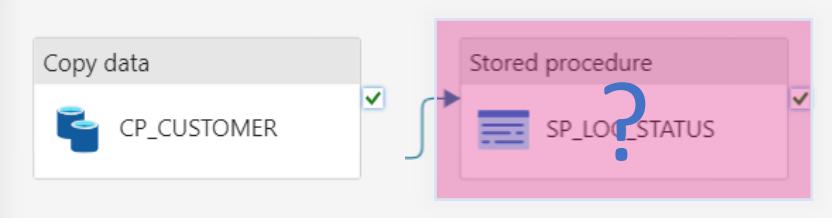
We help organizations
accelerating their **digital**
transformation with impactful
Microsoft solutions & expertise

Problem statement

The screenshot shows the Power BI FabricDemo workspace. On the left is a navigation sidebar with icons for Home, Create, Browse, OnLake data hub, Apps, Metrics, Monitor, Learn, Real-Time hub, Workspaces, and FabricDemo. The main area displays a list of pipelines under the 'FabricDemo > Chaos' folder. The pipelines listed are:

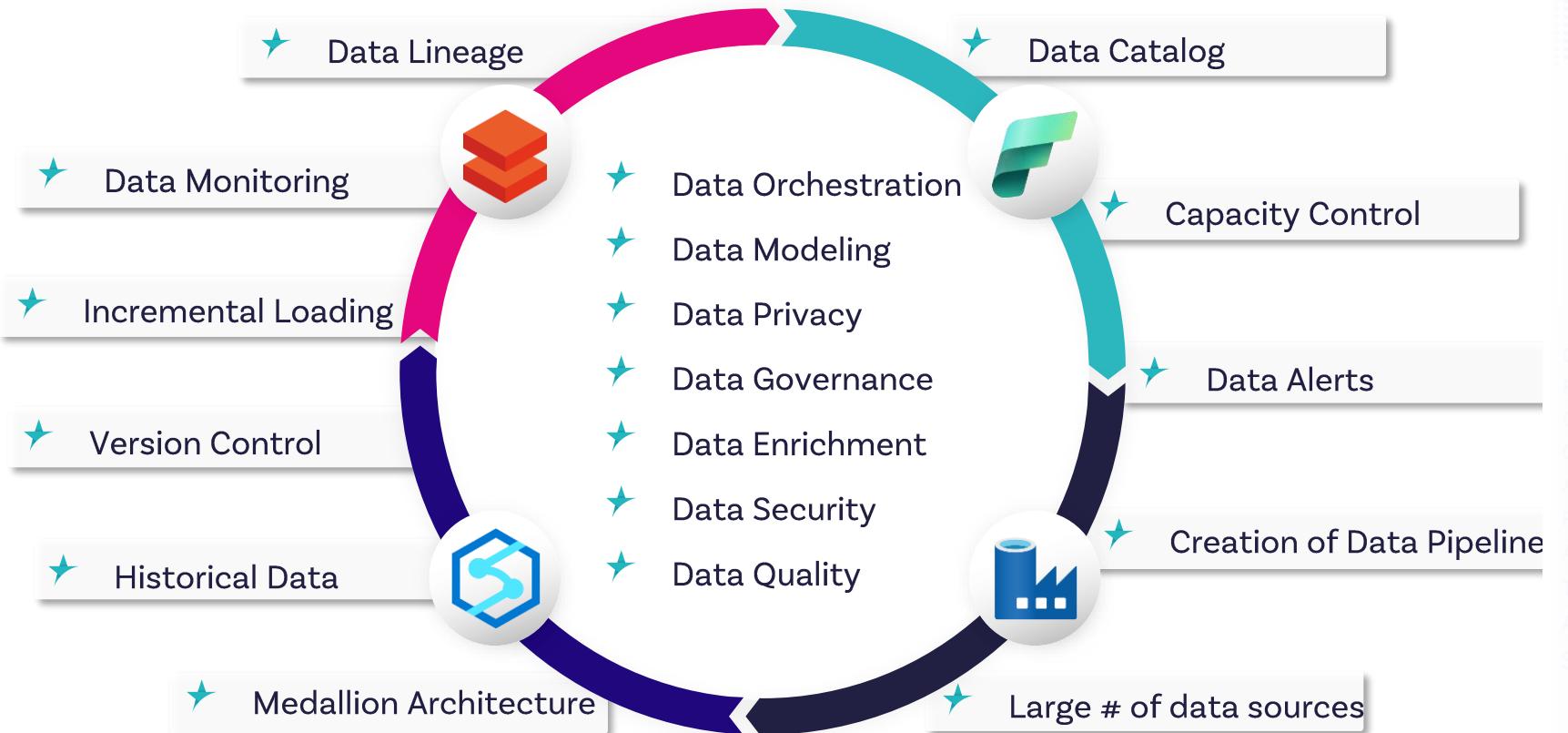
Name	Type
PL_EXTRACT_AW_CUSTOMER	Data pipeline
PL_EXTRACT_AW_PRODUCT_CATEGORY	Data pipeline
PL_EXTRACT_AW_PRODUCT_DESCRIPTION	Data pipeline
PL_EXTRACT_AW_PRODUCT_DESCRIPTION_LINEAGE	Data pipeline
PL_EXTRACT_AW_PRODUCT_MODEL	Data pipeline
PL_EXTRACT_AW_PRODUCT_MODEL_PRODUCT_DESCRIPTION	Data pipeline
PL_EXTRACT_AW_PRODUCTS	Data pipeline
PL_EXTRACT_AW_SALES_ORDER_DETAIL	Data pipeline
PL_EXTRACT_AW_SALES_ORDER_HEADER	Data pipeline
PL_EXTRACT_AW_SALES_ORDER_LINE	Data pipeline
PL_EXTRACT_WWI_ADDRESS	Data pipeline
PL_EXTRACT_WWI_CITIES	Data pipeline
PL_EXTRACT_WWI_COUNTRIES	Data pipeline
PL_EXTRACT_WWI_DELIVERY_METHODS	Data pipeline
PL_EXTRACT_WWI_PAYMENT_METHODS	Data pipeline
PL_EXTRACT_WWI_PEOPLE	Data pipeline
PL_EXTRACT_WWI_STATE_PROVINCES	Data pipeline
PL_EXTRACT_WWI_TRANSACTION_TYPES	Data pipeline

Content of each pipeline



Data platform Challenges

'From data source to data model' to report



“Simplify,

I am the Chief Data Information officer
and don't want to be the
Chief Integration Officer

Help me to simplify and automate my
Data Orchestration.”

Every CDO, Every organisation



What would help

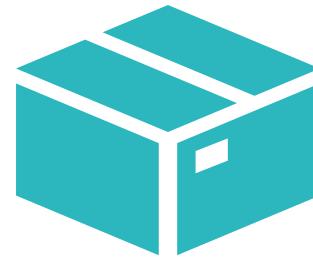
- Simplicity in connecting data sources
- Creating results within hours
- Focus on business value instead of data integration
- Meta Data Driven
 - Standardized data pipelines, Notebooks, orchestration and Way of Work
- Overview of data process flows
- Detailed logging information
- Easy Integration of solution like:
 - Data Privacy
 - Data Quality
 - Data Governance
- From data “Spaghetti to Lasagna”
 - Building a uniform data architecture

*I am the Chief Data Information Officer
and don't want to be the Chief Integration Officer*

*Help me to simplify and automate my Data
Orchestration*



Why



Scalability



Automation



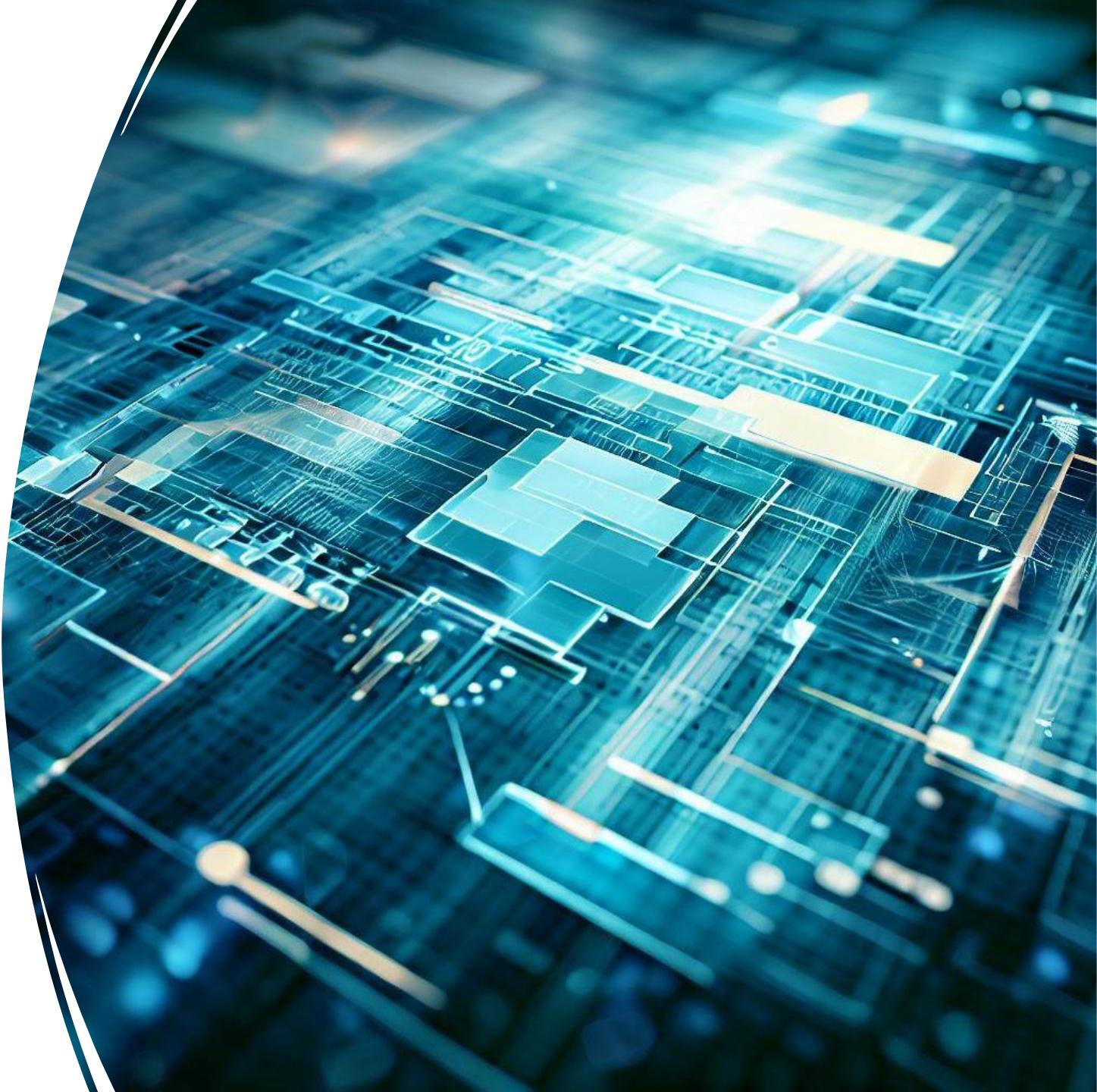
Traceability



Flexibility

Out-of-the-Box Framework

- Ready-to-use.
- Rapid implementation.
- Limited customization.
- Lower development effort.
- Lower upfront costs.
- Ongoing support and updates.



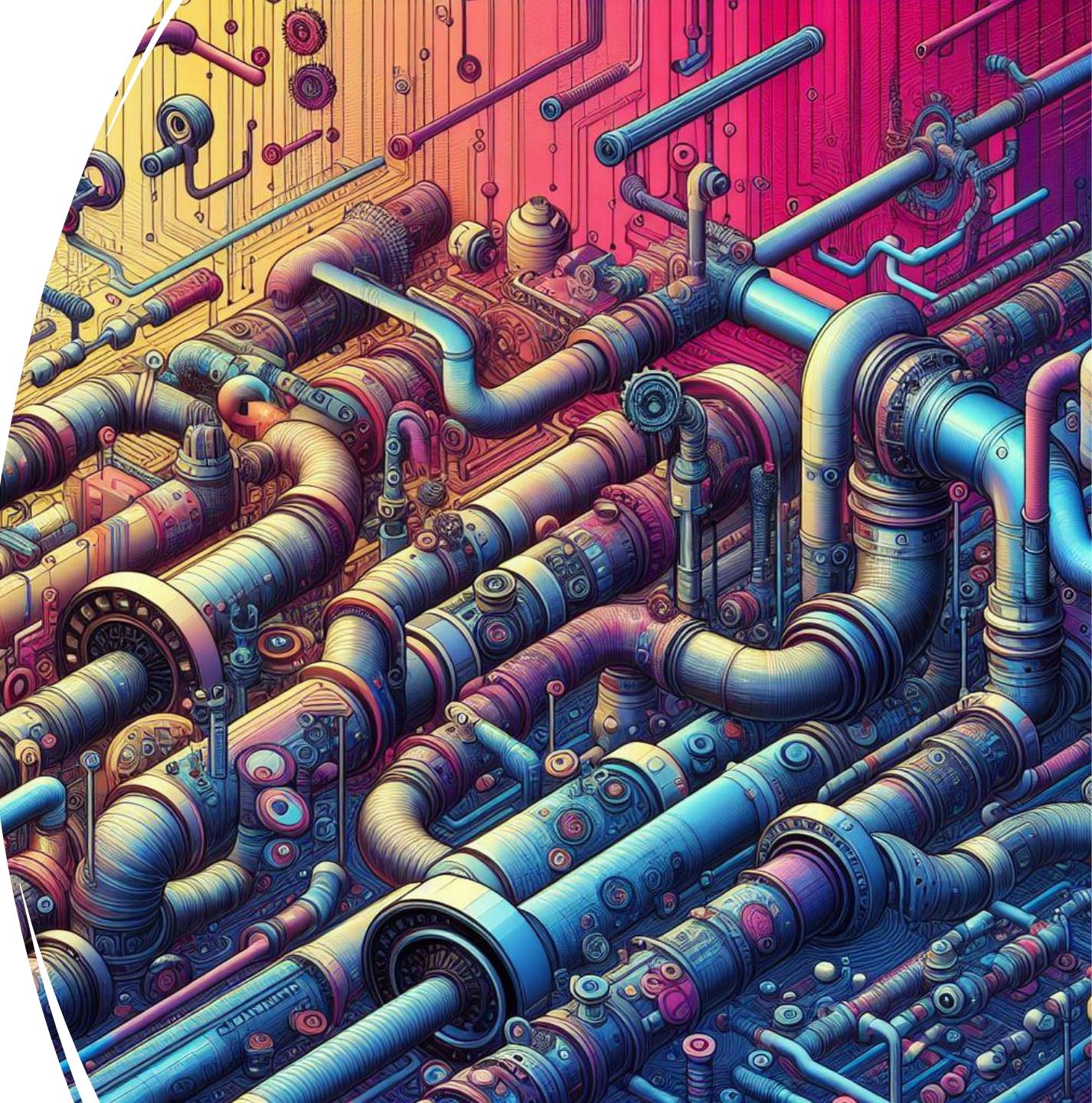
Custom-Made Framework

- Tailored to specific needs.
- Full control over design and features.
- Higher development effort.
- Flexibility and extensibility.
- Higher upfront costs.



Custom-Made Framework

- Based on parameters
- Meta data => Azure SQL Database / Json /
- Microsoft Fabric but also on Azure Synapse Analytics and Azure Data Factory
- Based on the Medallion Architecture



Who is already using Parameters?





Microsoft Fabric

The unified data platform for the era of AI



Data
Factory



Synapse Data
Engineering



Synapse Data
Science



Synapse Data
Warehousing



Synapse Real
Time Analytics



Power BI



Data
Activator



AI



OneLake



Purview

Unified
architecture

Unified
experience

Unified
governance

Unified
business model



Microsoft Fabric

The unified data platform for the era of AI



Data
Factory



Synapse Data
Engineering



Synapse Data
Science



Synapse Data
Warehousing



Synapse Real
Time Analytics



Power BI



Data
Activator



AI



OneLake



Purview

Unified
architecture

Unified
experience

Unified
governance

Unified
business model

Parameters

- Pipeline Parameters
 - Copy Activity Parameters
 - Foreach Parameters
- Notebook Parameters

IMPLEMENTING
DEFAULT PARAMETERS
THAT DEPEND ON
OTHER PARAMETERS

Understand Parameters

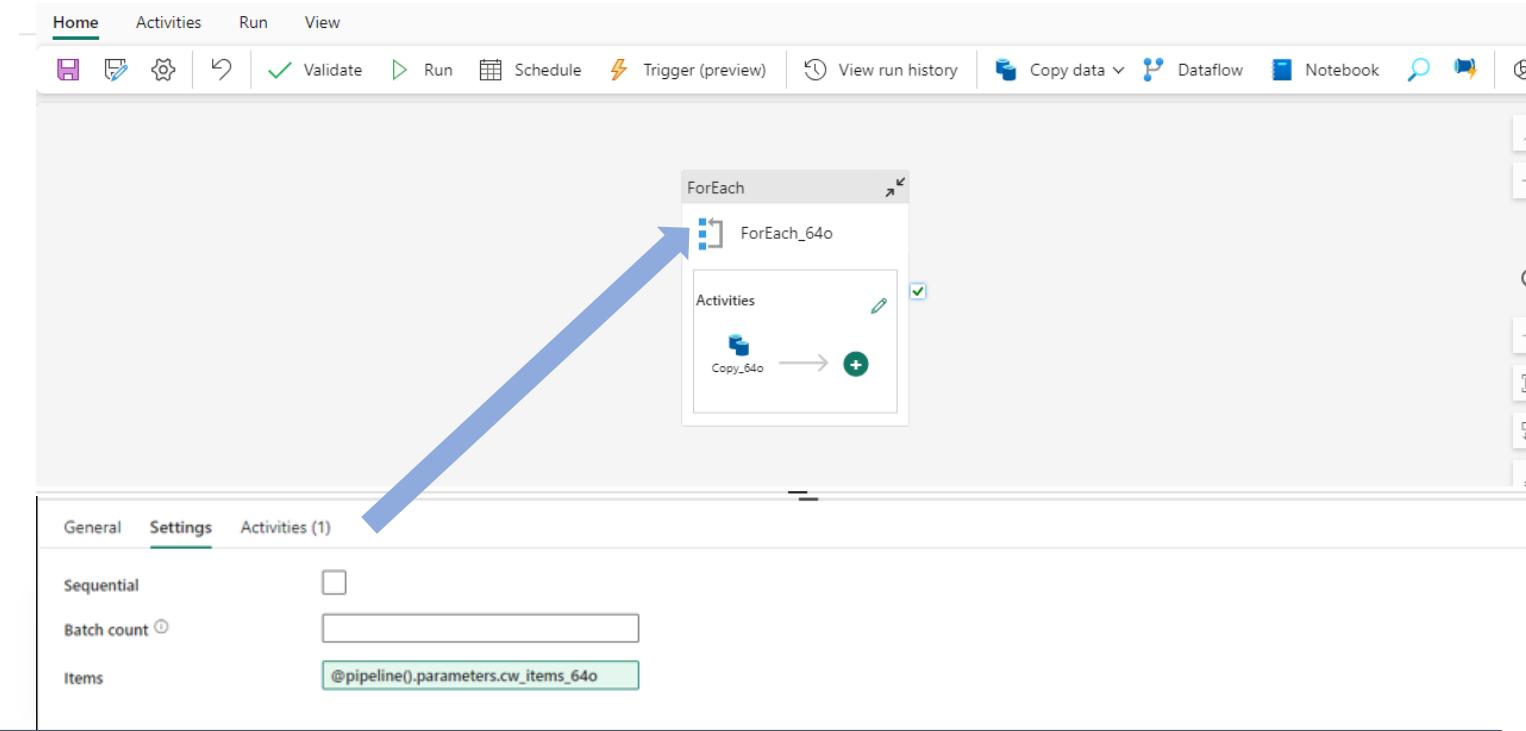
- Copy data assistant
 - Select Data Source
 - Select Tables
 - Select Destination
 - Select Filetypes
 - Review actions

The screenshot shows the Microsoft Fabric Copy Data Assistant interface. At the top, there's a navigation bar with Home, Activities, Run, View, Validate, Run, Schedule, and Trigger (preview) buttons. Below the navigation bar is a toolbar with icons for New, Delete, and other operations. A sidebar on the right lists activities: ForEach, For, Activities, and Copy_64. The main area is titled "Parameters" and contains a table with columns: Name, Type, and Default value. There is one row with the name "cw_items_64o", type "Array", and default value "[{"source": {"table": "Department"}, "destination": {"fileName": "HumanResourcesDepartment.par"}}, {"source": {"table": "Employee"}, "destination": {"fileName": "HumanResourcesEmployee.par"}}, {"source": {"table": "EmployeeDepartmentHistory"}, "destination": {"fileName": "HumanResourcesEmployeeDepHistory.par"}}, {"source": {"table": "EmployeePayHistory"}, "destination": {"fileName": "HumanResourcesEmployeePayHistory.par"}}]". A blue arrow points from the "Default value" field in the table to the JSON code on the right. The JSON code is a list of four objects, each defining a copy activity with a source table and a destination file name.

```
[{"source": {"table": "Department"}, "destination": {"fileName": "HumanResourcesDepartment.par"}}, {"source": {"table": "Employee"}, "destination": {"fileName": "HumanResourcesEmployee.par"}}, {"source": {"table": "EmployeeDepartmentHistory"}, "destination": {"fileName": "HumanResourcesEmployeeDepHistory.par"}}, {"source": {"table": "EmployeePayHistory"}, "destination": {"fileName": "HumanResourcesEmployeePayHistory.par"}}]
```

Pipeline Parameters

- Pass through from
 - Pipeline to Activity

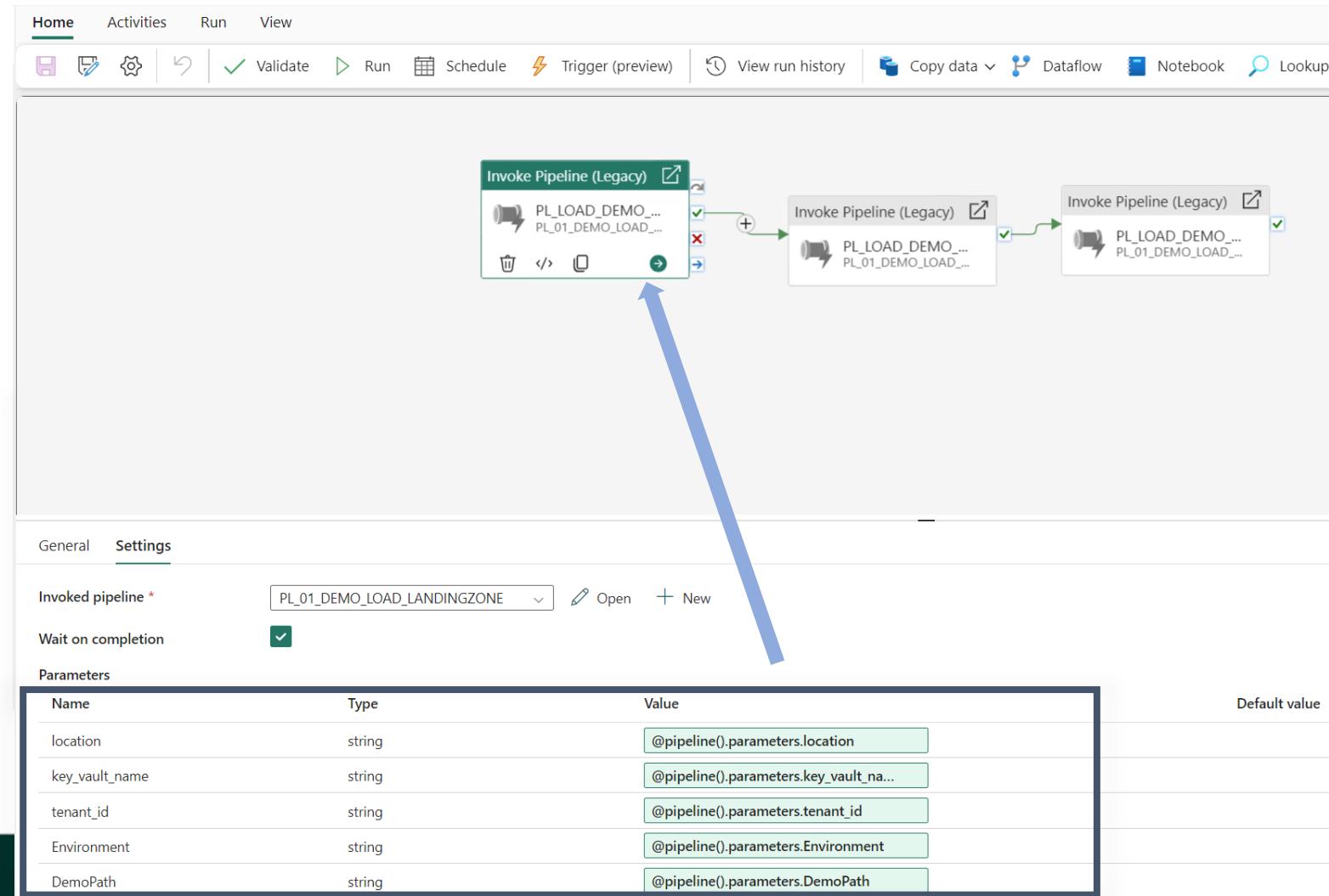


Items

`@pipeline().parameters.cw_items_64o`

Pipeline Parameters

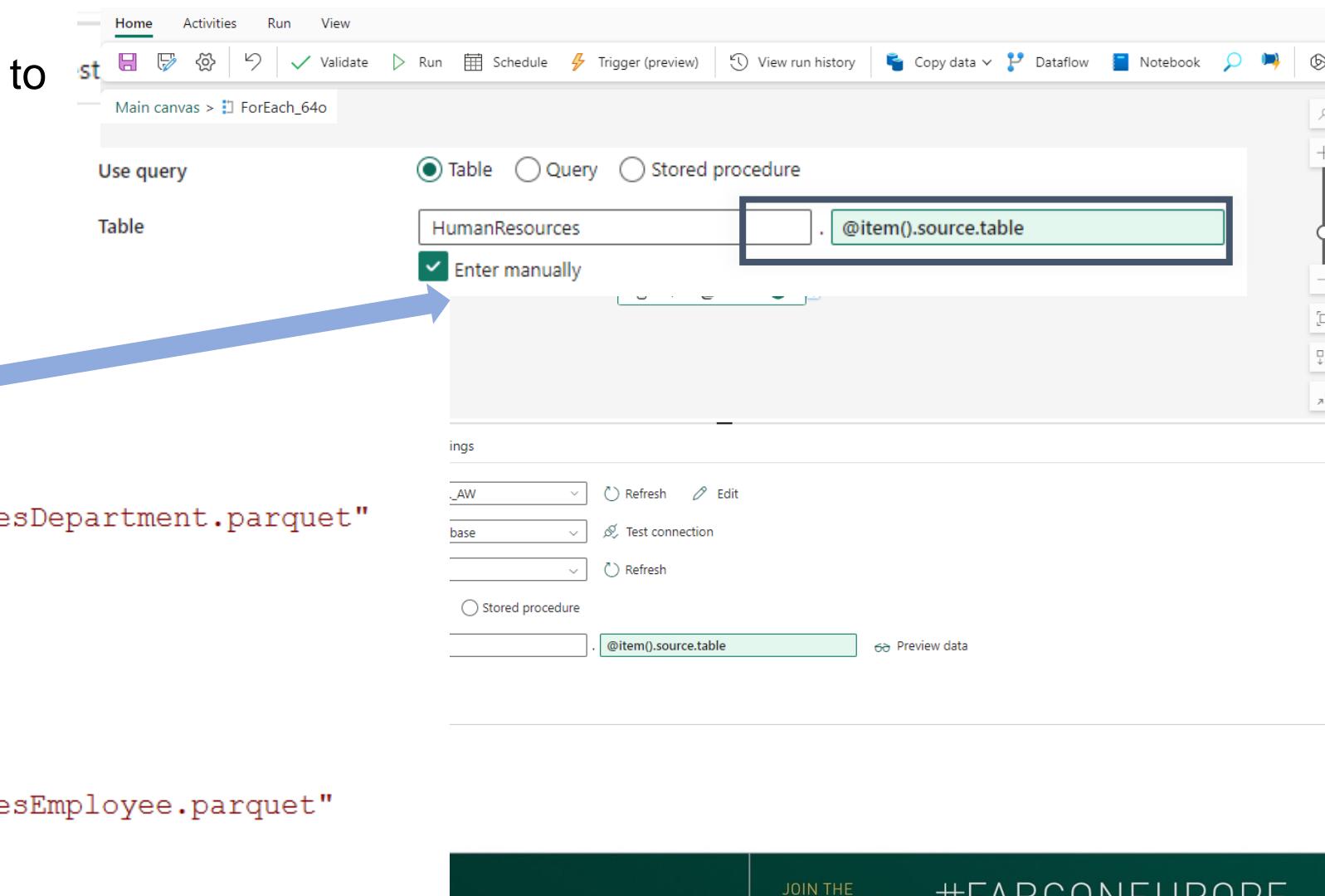
- Pass through from
 - Pipeline to Pipeline



Copy Activity Parameters

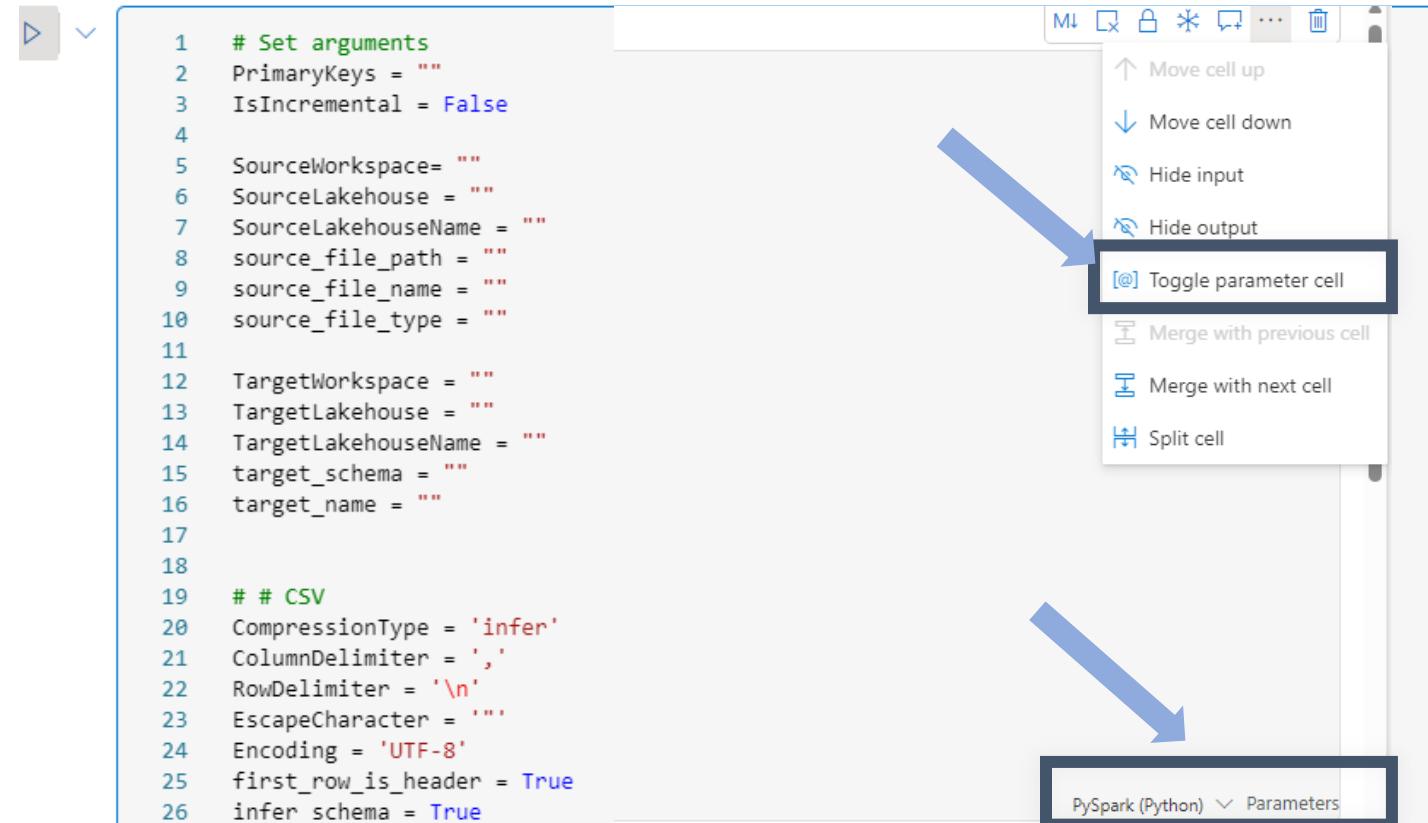
- Pass Parameters from Pipeline to Copy activity
- Use Parameters from For Each Activity

```
[  
  {  
    "source": {  
      "table": "Department"  
    },  
    "destination": {  
      "fileName": "HumanResourcesDepartment.parquet"  
    }  
  },  
  {  
    "source": {  
      "table": "Employee"  
    },  
    "destination": {  
      "fileName": "HumanResourcesEmployee.parquet"  
    }  
  }]
```



Notebook Parameters

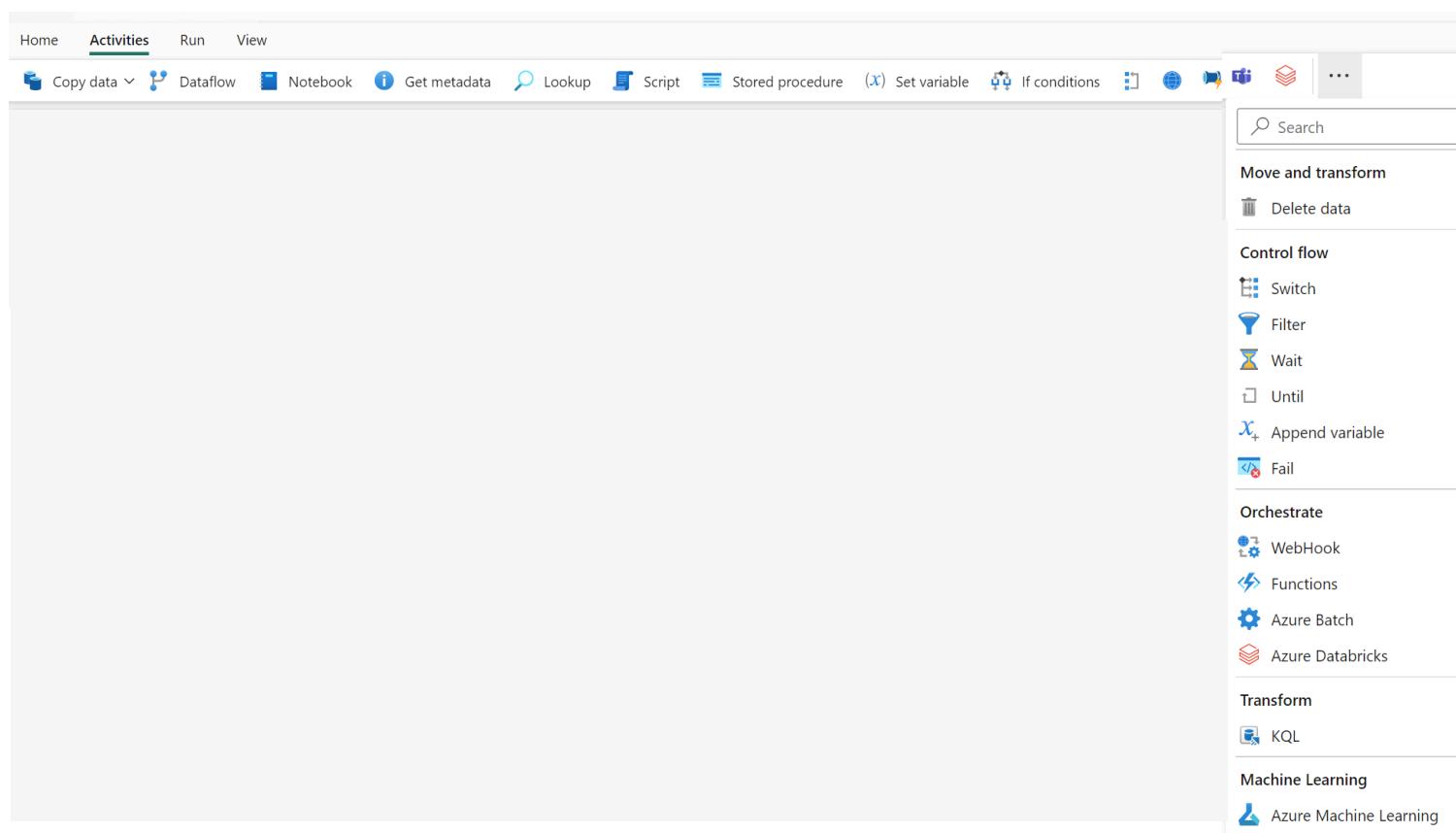
- Pass Parameters from Data Pipeline to Notebook Activity
 - Toggle parameter cell



```
1 # Set arguments
2 PrimaryKeys = ""
3 IsIncremental = False
4
5 SourceWorkspace= ""
6 SourceLakehouse = ""
7 SourceLakehouseName = ""
8 source_file_path = ""
9 source_file_name = ""
10 source_file_type = ""
11
12 TargetWorkspace = ""
13 TargetLakehouse = ""
14 TargetLakehouseName = ""
15 target_schema = ""
16 target_name = ""
17
18
19 # # CSV
20 CompressionType = 'infer'
21 ColumnDelimiter = ','
22 RowDelimiter = '\n'
23 EscapeCharacter = ''
24 Encoding = 'UTF-8'
25 first_row_is_header = True
26 infer schema = True
```

Blank canvas

- Everything build from scratch with no predefined options



Templates

- Templates are pre-defined pipelines that allow you to get started quickly with Data Factory.
- These templates help to reduce development time by providing an easy way to create pipelines.
- Templates are available for common data integration scenarios.
- Templates can be customized to meet specific requirements

Copy data from Azure SQL DB to Lakehouse Table

About this template

Use this template to copy data from your Azure SQL database to a specified table in your Lakehouse.

If you want to copy data from a number of tables, please use the "Copy assistant" to create your pipeline.

Inputs

Connection *

For Copy from SQL to LH (Copy source)

Select... + New

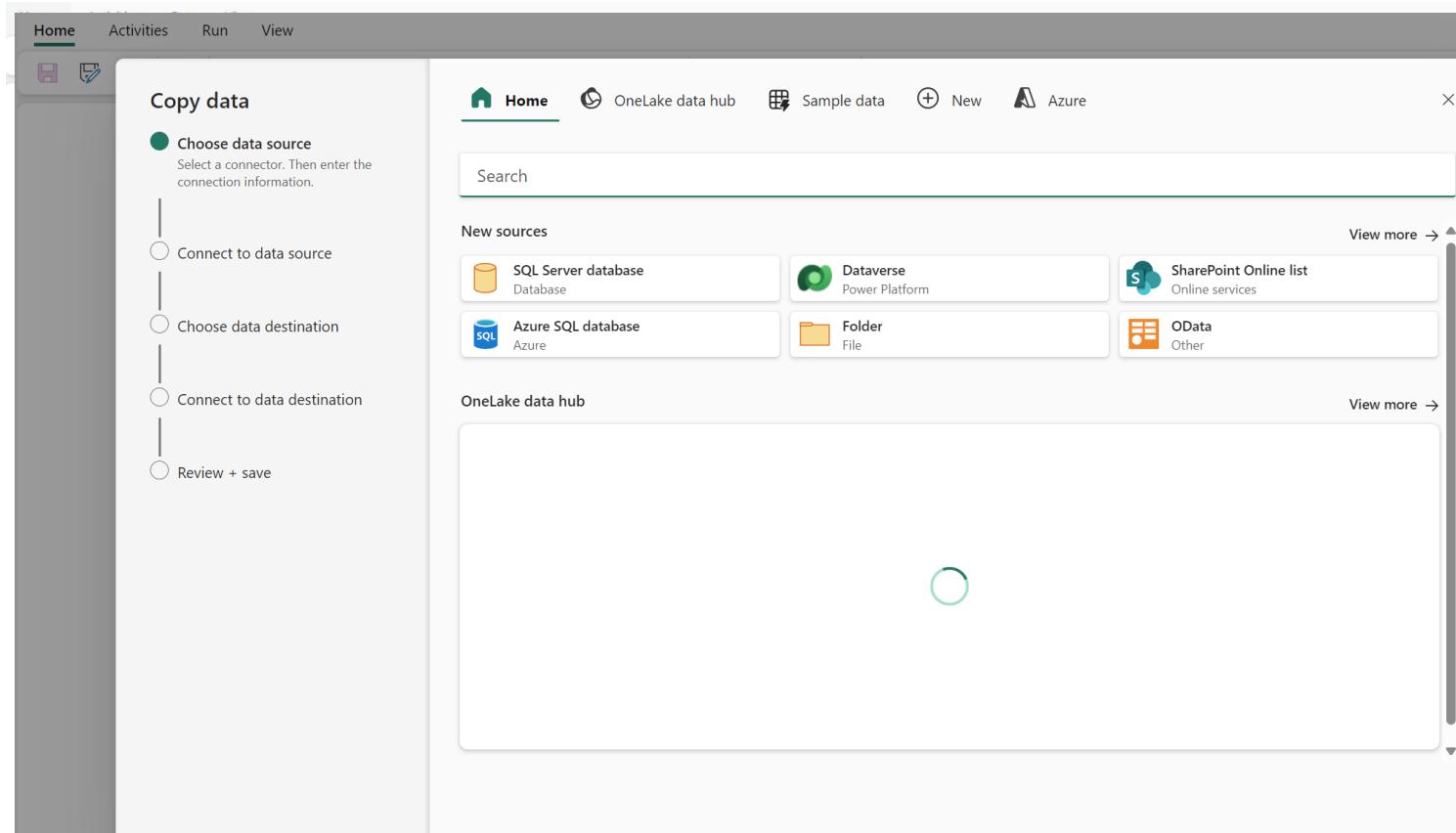
Lakehouse *

For Copy from SQL to LH (Copy destination)

Select... + New

Copy Assistant

- Reads data from a source data store.
- Performs serialization/deserialization, compression/decompression, column mapping, and so on.
- Writes data to the destination data store.

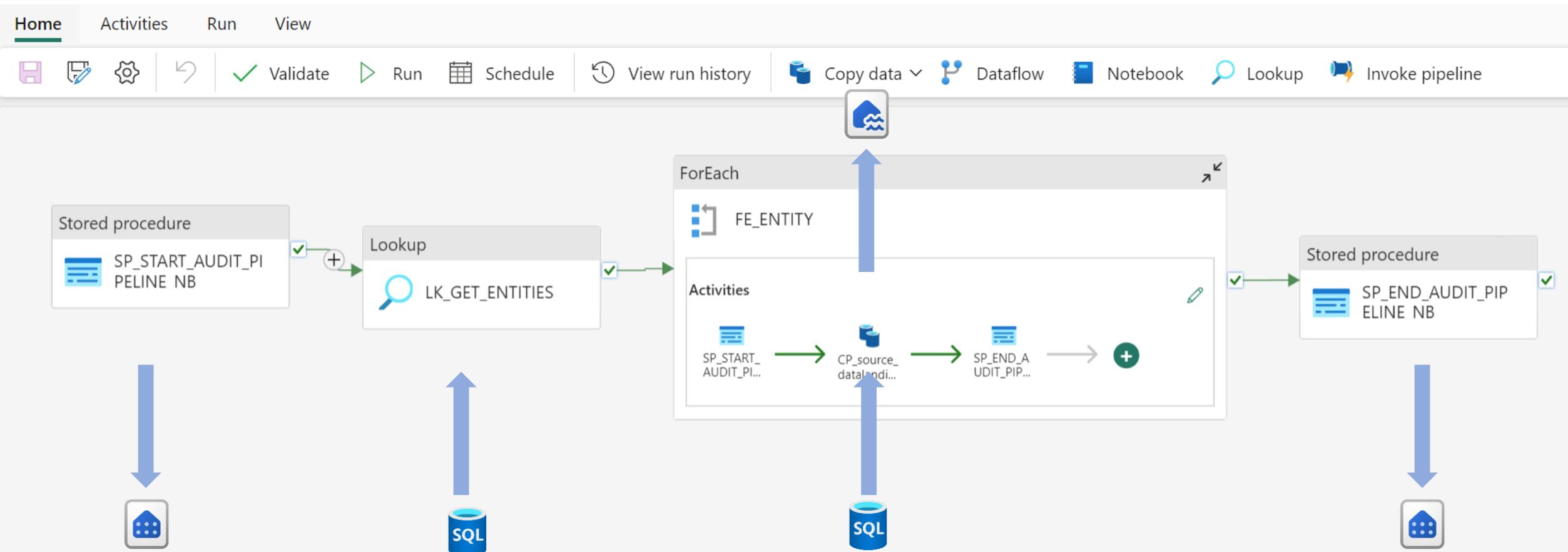


DEMO

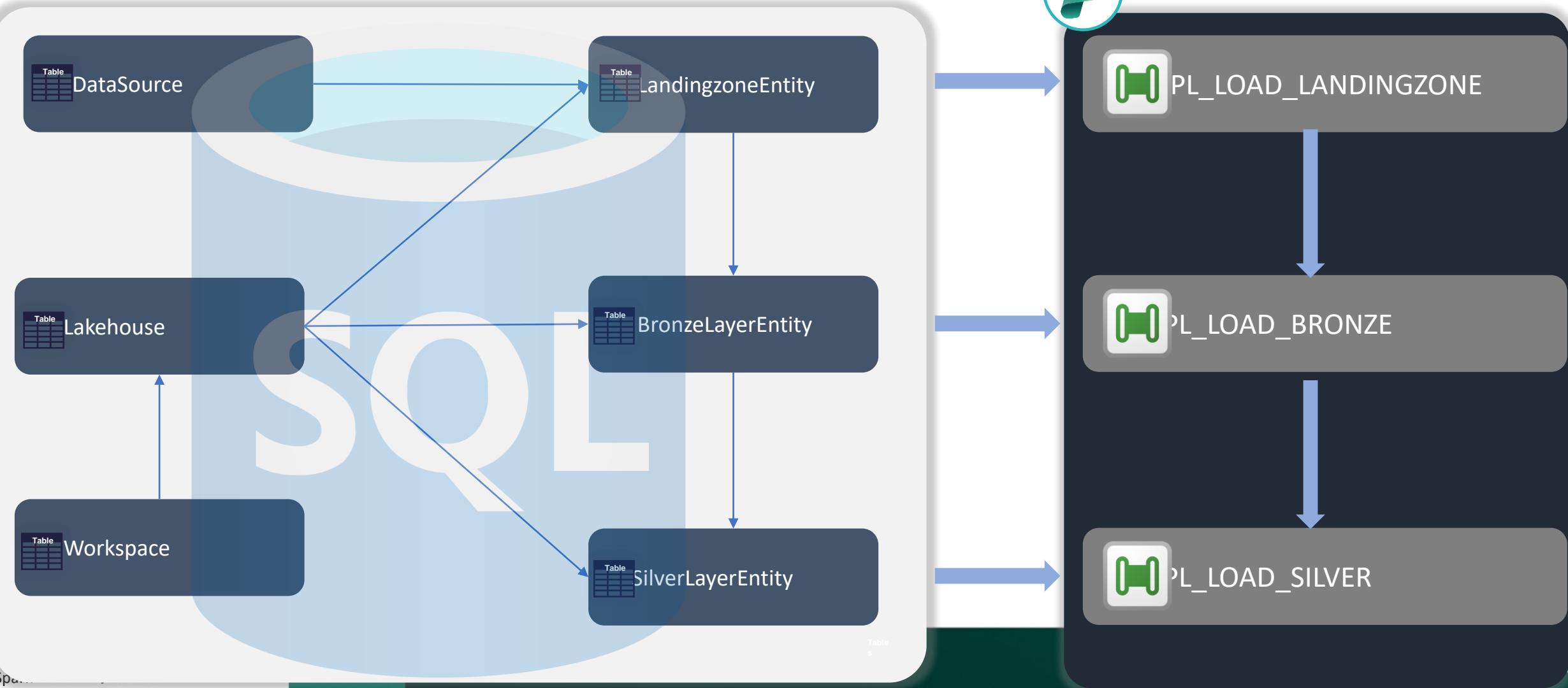


Framework in Azure SQL

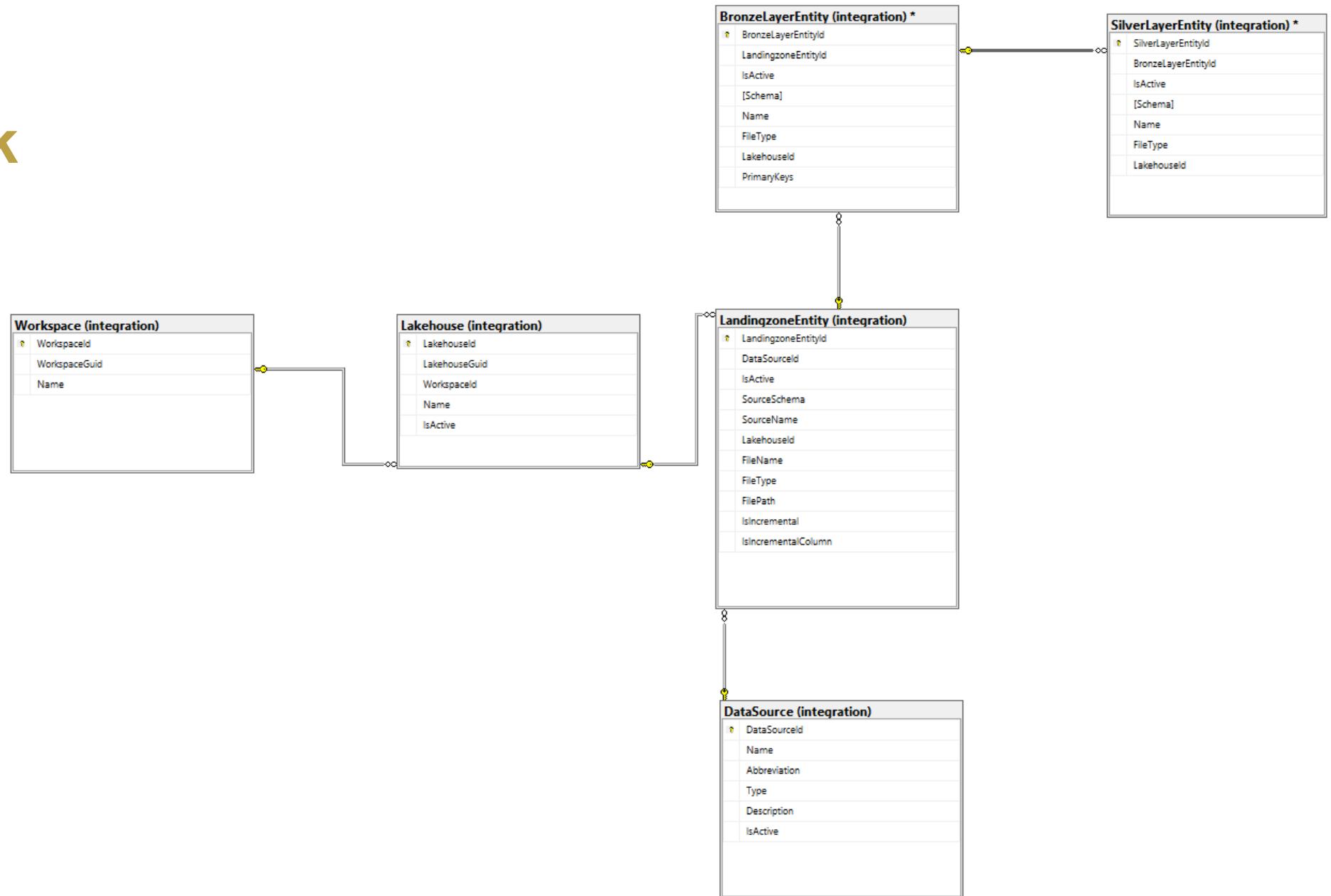




Framework



Framework



Who is using a Medallion architecture?



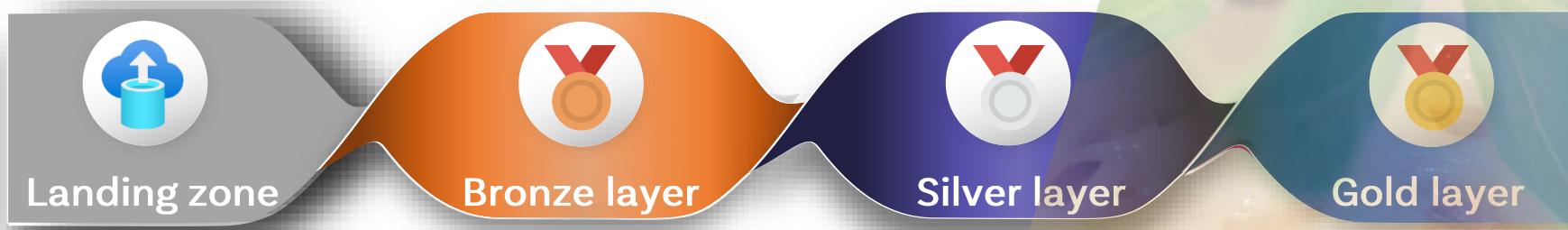
'Uniform data architecture'
From data "Spaghetti to Lasagna"



Medallion Architecture

- *'Data processing in different stages'*

Stages



'Data processing in different stages'

Stage:

Medallion Architecture



- Dimensions & Facts (Star Schema)
- Historical Analysis
- Business rules
- Documentation
- Aggregated data
- Logical table names



SPRO
View



- Historical Data (Type 1 or 2)
- Data quality rules
- Data Cleansing
- Validated data
- No business model/data



- Deduplicate data
- Add datatypes
- Data can be inconsistent
- Mostly a copy of the source
- Schema



- Structured data
- Unstructured data
- Incremental loads
- Data as is
- Stored in Datetime folder structure
- No Schema



Definition:

STOCK
HOLM
2 0 2 4

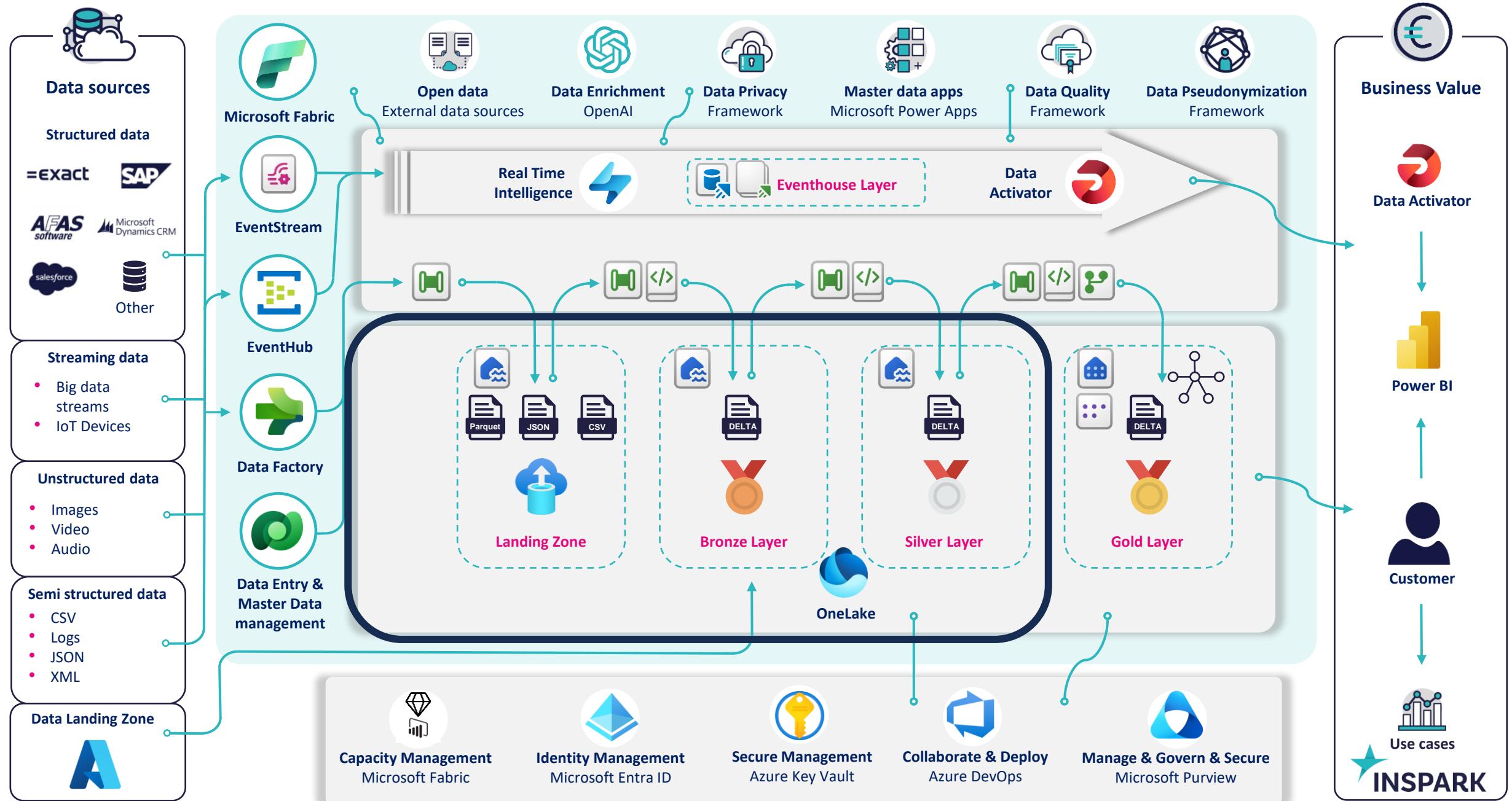
Filetype:

Files/Tables:

Fabric:

JOIN THE CONVERSATION

#FABCONEUROPE



Lakehouse

	Name	Type
	LH_Bronze_Layer	Lakehouse
	LH_Bronze_Layer	Semantic model (default)
	LH_Bronze_Layer	SQL analytics endpoint
	LH_Data_Landingzone	Lakehouse
	LH_Data_Landingzone	Semantic model (default)
	LH_Data_Landingzone	SQL analytics endpoint
	LH_Silver_Layer	Lakehouse
	LH_Silver_Layer	Semantic model (default)
	LH_Silver_Layer	SQL analytics endpoint

Copy Activity Parameters

- Pass Parameters from Pipeline to Copy activity
- Use Parameters from For Each Activity

Preview data

ceName	DataSourceAbbreviation	DataSourceType	IsActive	SourceSchema	SourceName	TargetFilePath	TargetFileName	TargetFileType	T
Jwdvlmdl01	ADLS	ADLS	true		customers.csv	demo/2024/09/10	customers_2024091015.csv	csv	C
Jwdvlmdl01	ADLS	ADLS	true		organizations.csv	demo/2024/09/10	organizations_2024091015.csv	csv	C
Jwdvlmdl01	ADLS	ADLS	true		people.csv	demo/2024/09/10	people_2024091015.csv	csv	C

General Source **Destination** Mapping Settings

Connection * `@item().TargetLakehouseGuid`

Connection type * `Lakehouse`

Workspace ID `@item().WorkspaceGuid`

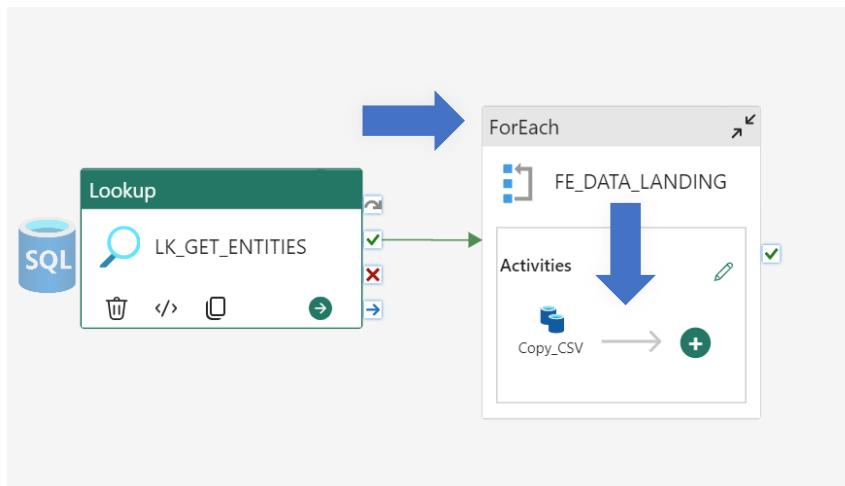
Root folder Tables Files

File path `@item().TargetFilePath` / `@item().TargetFileName`

File format * `Binary`

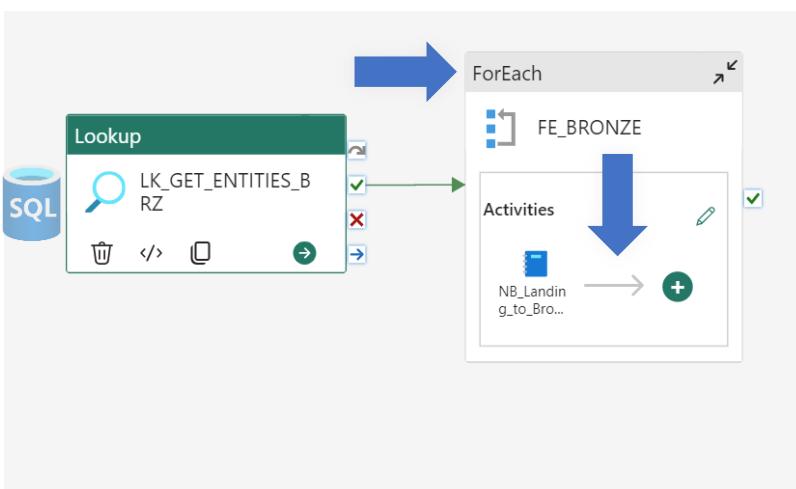
> Advanced

Two blue arrows point upwards from the 'File path' and 'File format' fields towards the 'Destination' tab of the pipeline editor.



Notebook Parameters

- Pass Parameters from Data Pipeline to Notebook
 - Toggle parameter cell



Preview data

#	EntityId	SourceFilePath	SourceFileName	SourceFileType	TargetSchema	TargetName	TargetWorkspaceId	SourceWorkspaceId	TargetLakehouseId	SourceLakehouseId	TargetLakehouseName	SourceLakehouseNa
1	1	demo/2024/09/10	customers_2024091017.csv	csv	demo	Customers	586fc19d-fa6a-4cb1-9ca3-e518a524f5da	586fc19d-fa6a-4cb1-9ca3-e518a524f5da	e130dba7-c8c3-438a-85ad-2cd4c9d59a09	009058ac-b71c-4774-a8ee-7c7d945c3972	LH_Bronze_Layer	LH_Data_Landingzor
2	3	demo/2024/09/10	organizations_2024091017.csv	csv	demo	Organizations	586fc19d-fa6a-4cb1-9ca3-e518a524f5da	586fc19d-fa6a-4cb1-9ca3-e518a524f5da	e130dba7-c8c3-438a-85ad-2cd4c9d59a09	009058ac-b71c-4774-a8ee-7c7d945c3972	LH_Bronze_Layer	LH_Data_Landingzor
3	4	demo/2024/09/10	people_2024091017.csv	csv	demo	People	586fc19d-fa6a-4cb1-9ca3-e518a524f5da	586fc19d-fa6a-4cb1-9ca3-e518a524f5da	e130dba7-c8c3-438a-85ad-2cd4c9d59a09	009058ac-b71c-4774-a8ee-7c7d945c3972	LH_Bronze_Layer	LH_Data_Landingzor

Notebook * NB_LANDING_BRONZE

Base parameters

+ New	- Delete		
<input type="checkbox"/>	Name	Type	Value
<input type="checkbox"/>	SourceLakehouse	String	@item().SourceLakehouseId
<input type="checkbox"/>	source_file_path	String	@item().SourceFilePath
<input type="checkbox"/>	source_file_name	String	@item().SourceFileName
<input type="checkbox"/>	PrimaryKeys	String	@item().PrimaryKeys
<input type="checkbox"/>	TargetLakehouse	String	@item().TargetLakehouseId
<input type="checkbox"/>	target_schema	String	@item().TargetSchema
<input type="checkbox"/>	target_name	String	@item().TargetName
<input type="checkbox"/>	SourceWorkspace	String	@item().SourceWorkspaceId
<input type="checkbox"/>	TargetWorkspace	String	@item().TargetWorkspaceId
<input type="checkbox"/>	source_file_type	String	@item().SourceFileType

DEMO

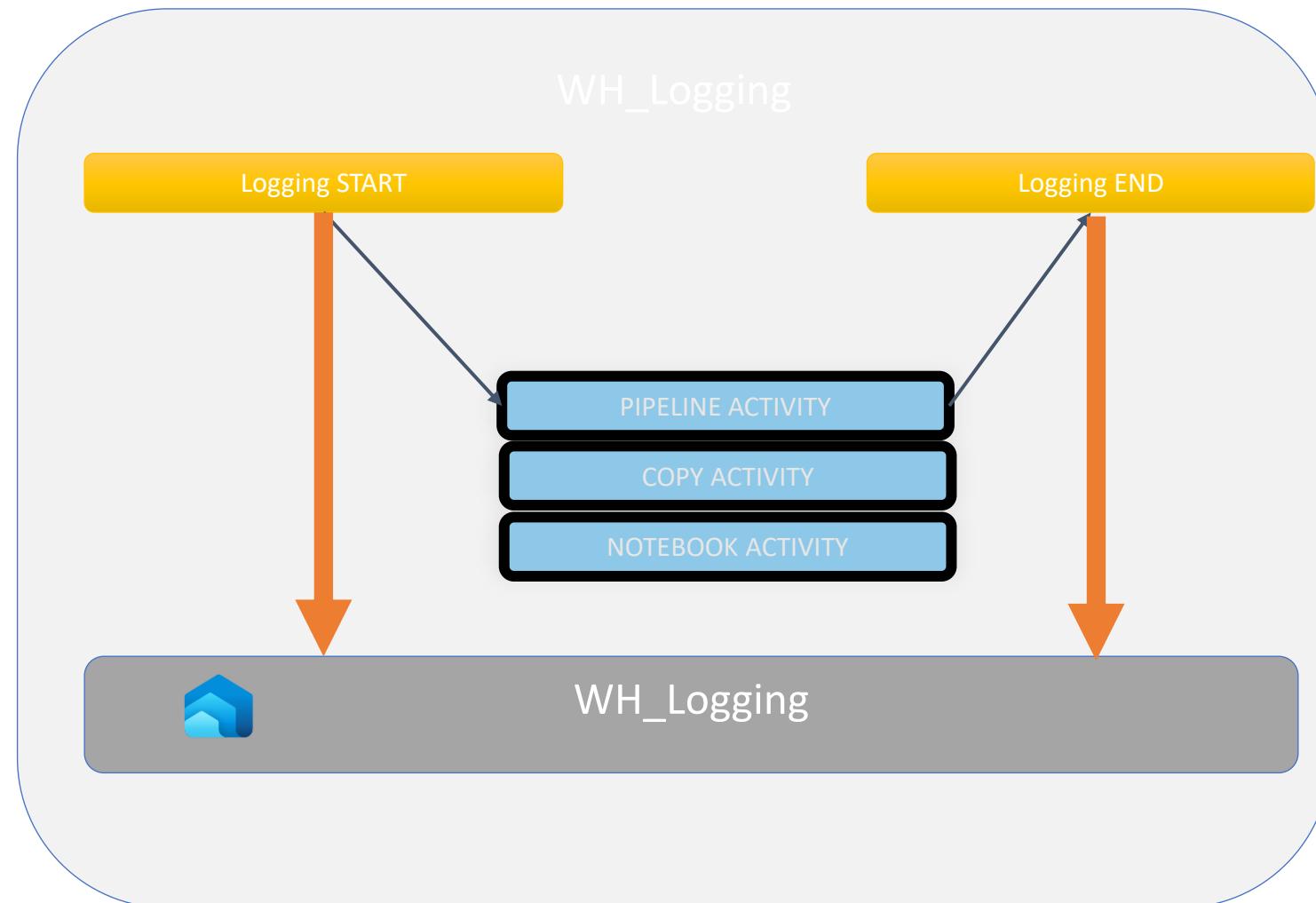


Logging



Logging

- Log Start and End Time of records
- Log Extracted Records
- Log Execution Failure



Logging

- Log Start and End Time of records
- Log Extracted Records
- Log Execution Failure

Pipeline expression builder 

Add dynamic content below using any combination of expressions, functions and system variables.

```
@pipeline().Pipeline
```

Clear contents

General Settings

Data store type Workspace External
Warehouse WH_Logging
Stored procedure name * `[logging].[sp_AuditPipeline]`

Stored procedure parameters 

<input type="checkbox"/>	Name	Type
<input type="checkbox"/>	LogData	String
<input type="checkbox"/>	LogType	String
<input type="checkbox"/>	PipelineGuid	Guid
<input type="checkbox"/>	PipelineName	String
<input type="checkbox"/>	PipelineParameters	String
<input type="checkbox"/>	PipelineParentRunGuid	Guid
<input type="checkbox"/>	PipelineRunGuid	Guid
<input type="checkbox"/>	TriggerGuid	Guid
<input type="checkbox"/>	TriggerTime	DateTime
<input type="checkbox"/>	TriggerType	String
<input type="checkbox"/>	WorkspaceGuid	Guid

Parameters System variables Functions Variables

Search 

Pipeline ID
ID of the pipeline

Pipeline Name
Name of the pipeline

Pipeline group ID
ID of the group to which the pipeline run belongs

Pipeline run ID
ID of the specific pipeline run

Pipeline trigger ID
ID of the trigger that invokes the pipeline

Pipeline trigger time
Time when the trigger that invoked the pipeline. The trigger time is the actual fired time, not the sched...

Pipeline trigger type
Type of the trigger that invoked the pipeline (Manual, Scheduler)

Pipeline triggered by pipeline ID
ID of the pipeline that triggered this pipeline. Applicable when a pipeline run is triggered by an Execut...

Pipeline triggered by pipeline name
Name of the pipeline that triggered this pipeline. Applicable when a pipeline run is triggered by an Ex...

Pipeline triggered by pipeline run ID
Run ID of the pipeline that triggered this pipeline. Applicable when a pipeline run is triggered by an Ex...

Workspace ID
ID of the workspace the pipeline run is running within

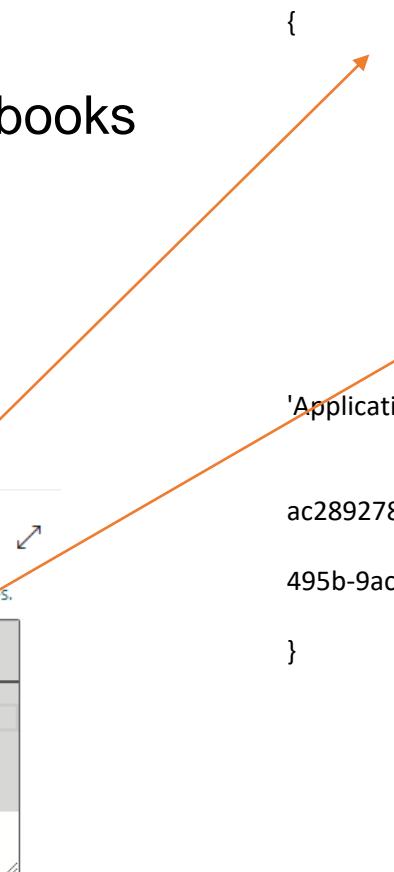
Logging

- Add Information about pipelines
- Adding System Variables
- Add Information about Notebooks

Pipeline expression builder

Add dynamic content below using any combination of [expressions](#), [functions](#) and [system variables](#).

```
{  
  "Action": "End",  
  @activity('NB_Landing_to_Bronze').output.result.exitValue  
}
```



```
{  
  "status": "Succeeded",  
  "result": {  
    "runId": "f0c69e6c-e28e-4db4-a8af-ac2892780xx2",  
    "runStatus": "Succeeded",  
    "sessionId": "fc55xx38-6fd8-47b0-863b-65c8c8db9878",  
    "sparkPool": "3xx61f99-edc7-4d6c-a866-f3bf70bc7235",  
    "error": null,  
    "lastCheckedOn": "2024-01-23T15:44:25.2733333Z",  
    "metadata": null,  
    "exitValue": {"CopyOutput": {"Total Runtime": "0:00:30.493626", "TargetSchema":  
      "Application", "TargetName": "People"}},  
    "message": "Notebook execution is in Succeeded state, runId: f0c69e6c-e28e-4db4-a8af-  
ac2892780442",  
    "SparkMonitoringURL": "https://app.powerbi.com/workloads/de-ds/sparkmonitor/fecxxff4-3a26-  
495b-9ac9-475bbc59fae7/fc55be38-6fd8-47b0-863b-65c8c8db9878?trident=1&experience=power-bi&tab=data",  
    "executionDuration": 49  
  }  
}
```

DEMO



Recap

- Can we build Pipelines dynamically?
- Can we extract data from my sources based on MetaData?
- Can we load the active(current) or historical records to a Lakehouse?
- Can we build history from extracted data based on MetaData?

Challenges



Out of the box fast and easy, less flexible



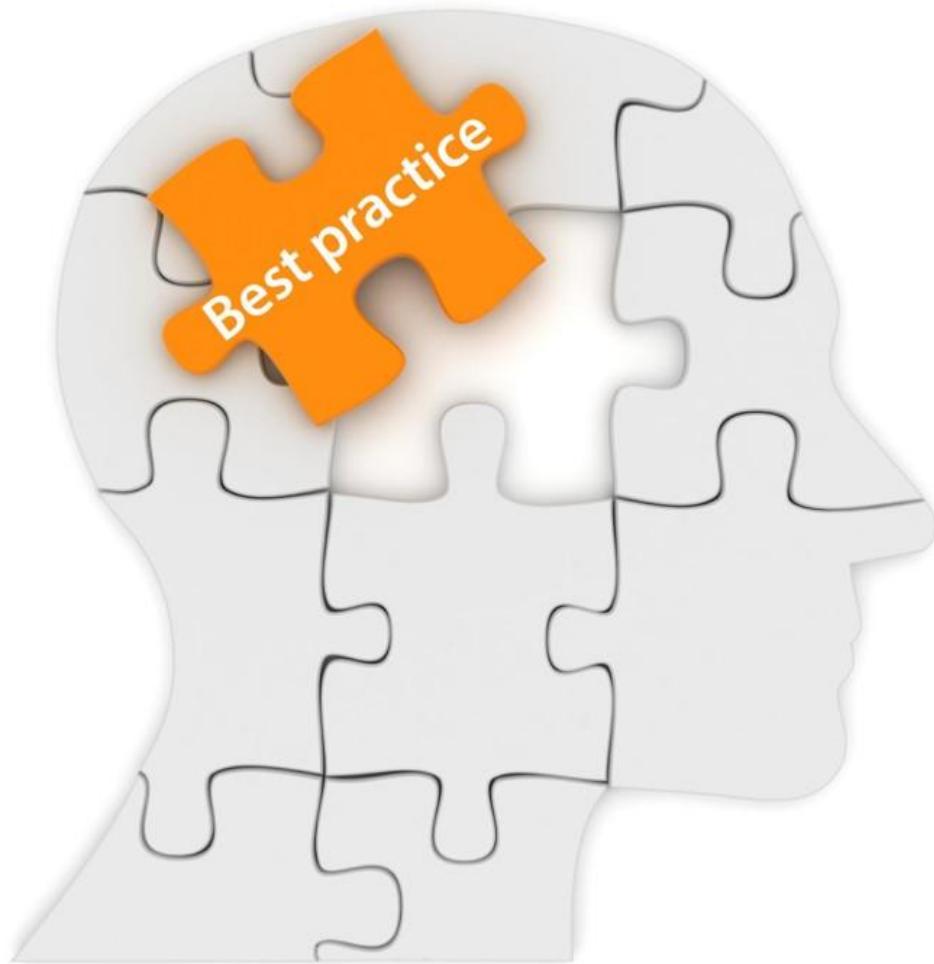
Parameterize of connections with Azure Key Vault (Like ADF/Synapse with Linked Services)



Schedule can't be Parameterized like in ADF/Synapse



Build in retry to Notebook Activity



Key Learnings and Best Practices



Metadata-driven approach is a best practice for managing data pipelines



Medallion Lakehouse architecture is a proven framework for implementing metadata-driven approach



Start small and gradually expand the metadata-driven approach across the organization





@erwindekreuk



linkedin.com/in/erwindekreuk



erwindekreuk.com



github.com/edkreuk



<https://sessionize.com/erwin-de-kreuk/>



Let's connect



FREE Microsoft Fabric



Certification Exam

Are you ready to get Fabric certified by the end of October?

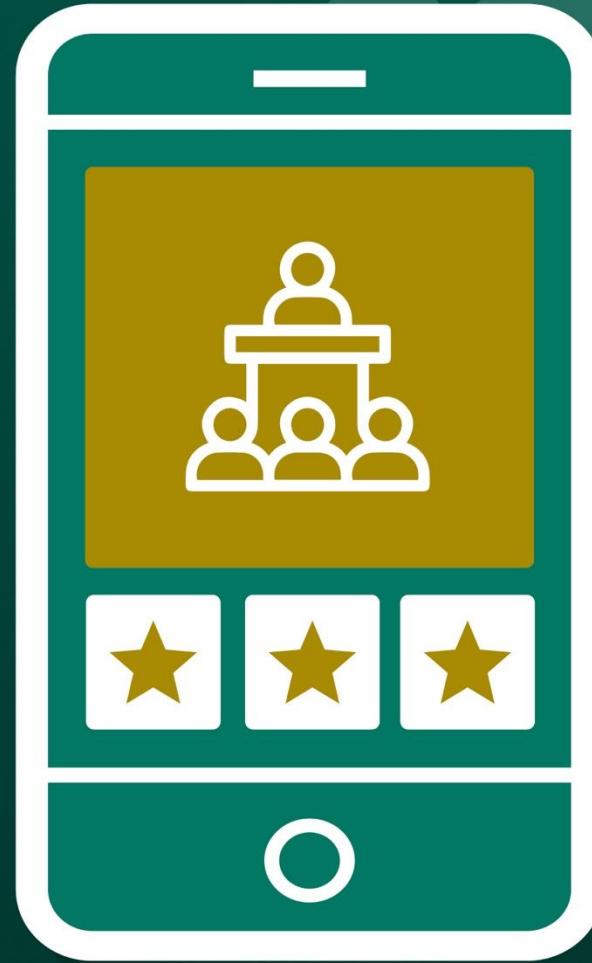
Claim your **100% discount voucher** for Exam DP-600: Fabric Analytics Engineer.

Come find us in the Community Lounge under the Get Certified banner to learn more!

aka.ms/FabCon24/IAmReady



Please rate
this session
on the app



cvent

