



# ETL in Azure Made Easy

## with Data Factory Data Flows

Paul Andrew

Principal Consultant & Solution Architect



@MrPaulAndrew





# ETL in Azure Made Easy

with Data Factory Data Flows



Paul Andrew

Principal Consultant & Solution Architect





# ETL in Azure Made Easy

with Data Factory Mapping Data Flows

Extract Transform Load

Paul Andrew

Principal Consultant & Solution Architect



 @MrPaulAndrew

# GitHub



<https://github.com/mrpaulandrew>

## CommunityEvents

Demo code, content and slides from various community events.

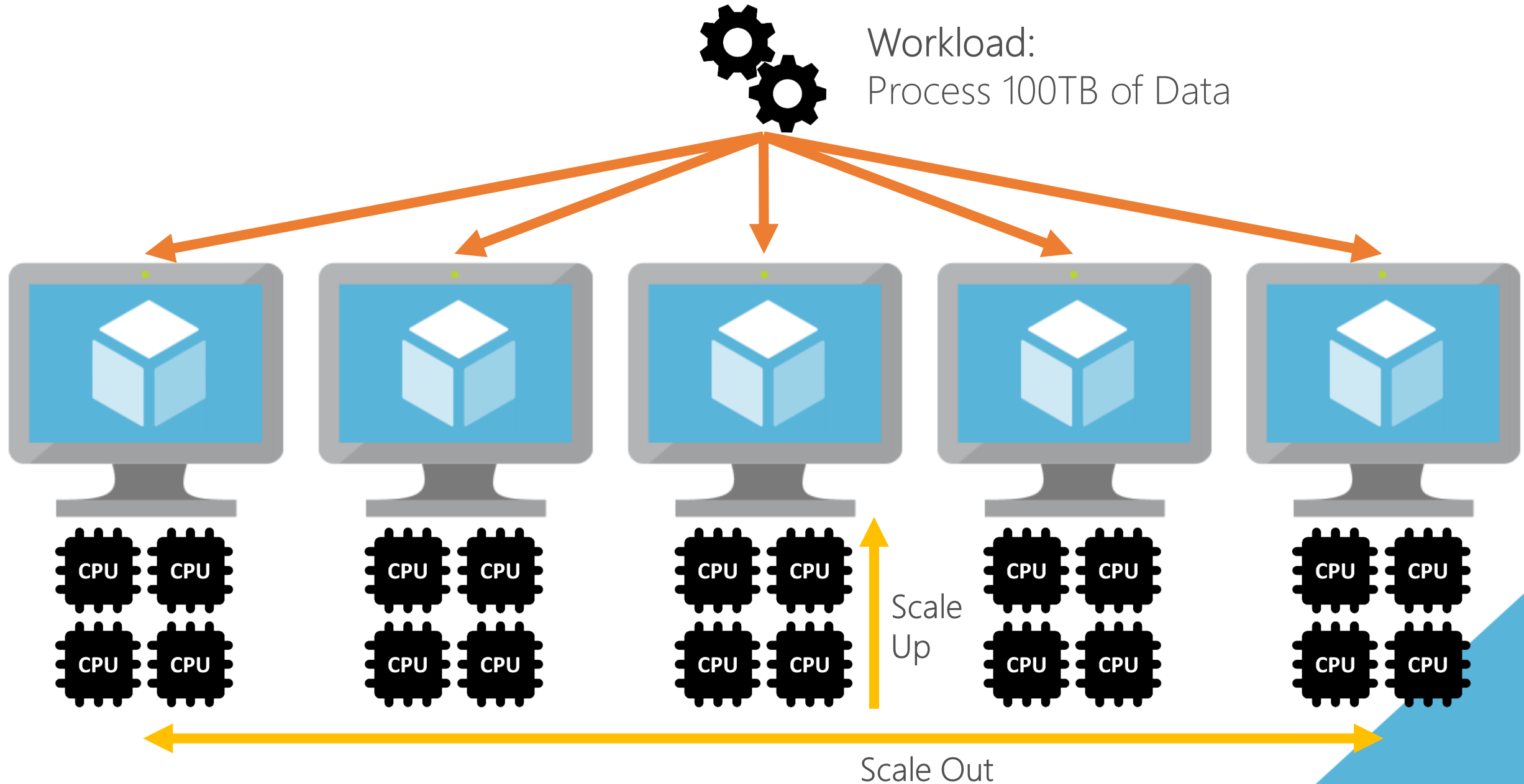
● C++

[{Event/Location}-{Month}-{Year}](#)

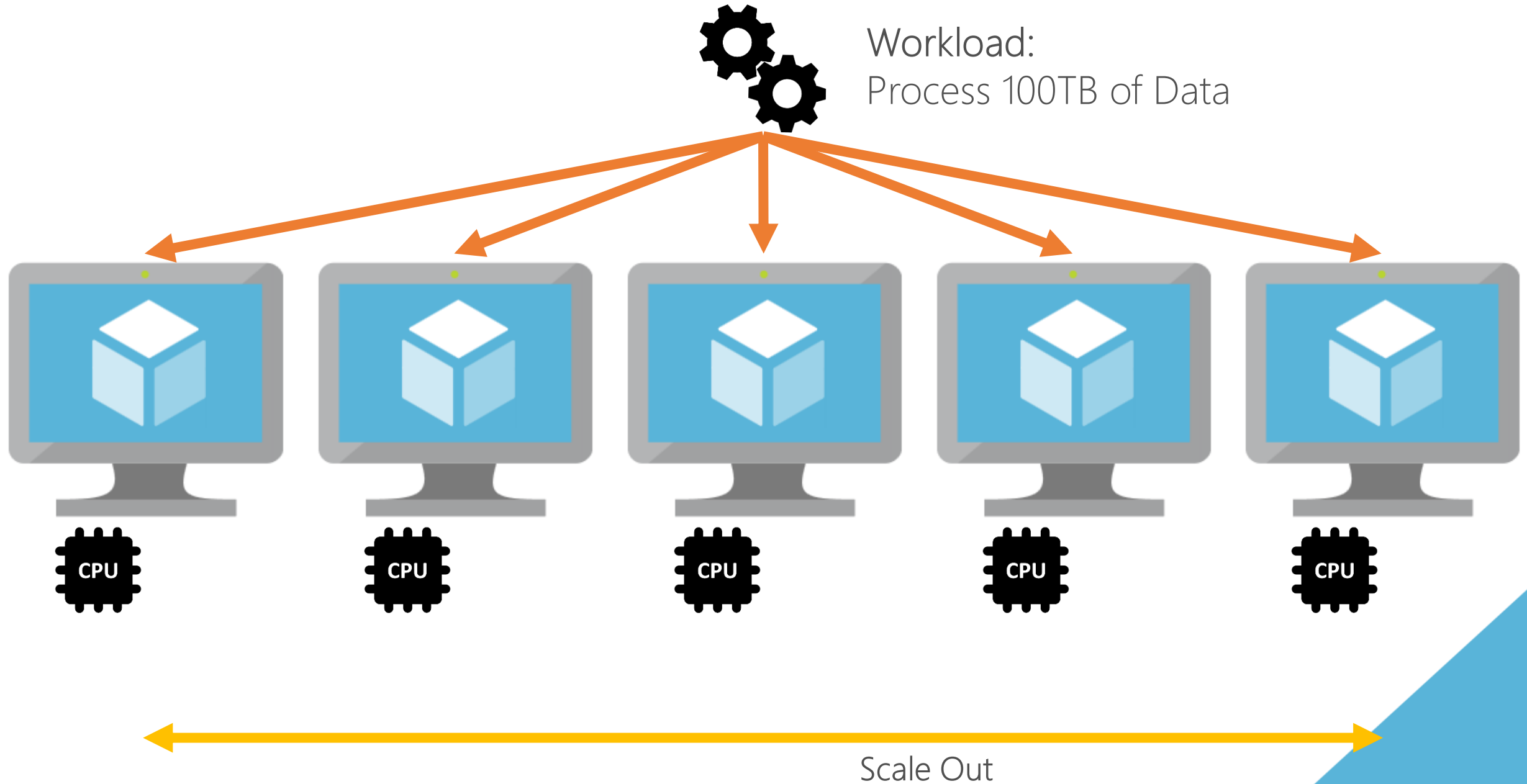
# Scale Up vs Scale Out

A large, solid blue triangle is positioned in the bottom right corner of the slide, pointing towards the top right.

# Scale Up and Scale Out

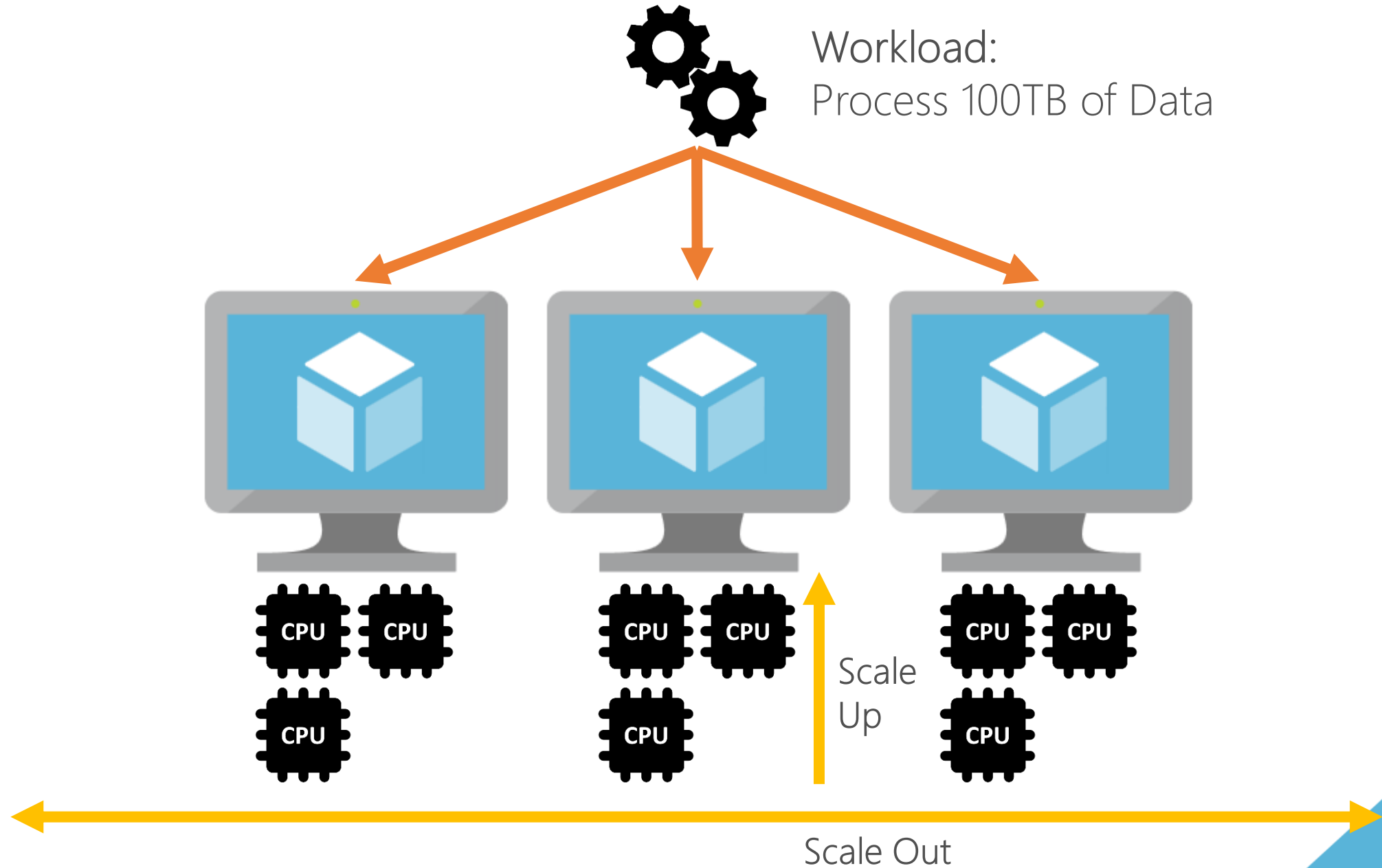


# Scale Up and Scale Out





# Scale Up and Scale Out



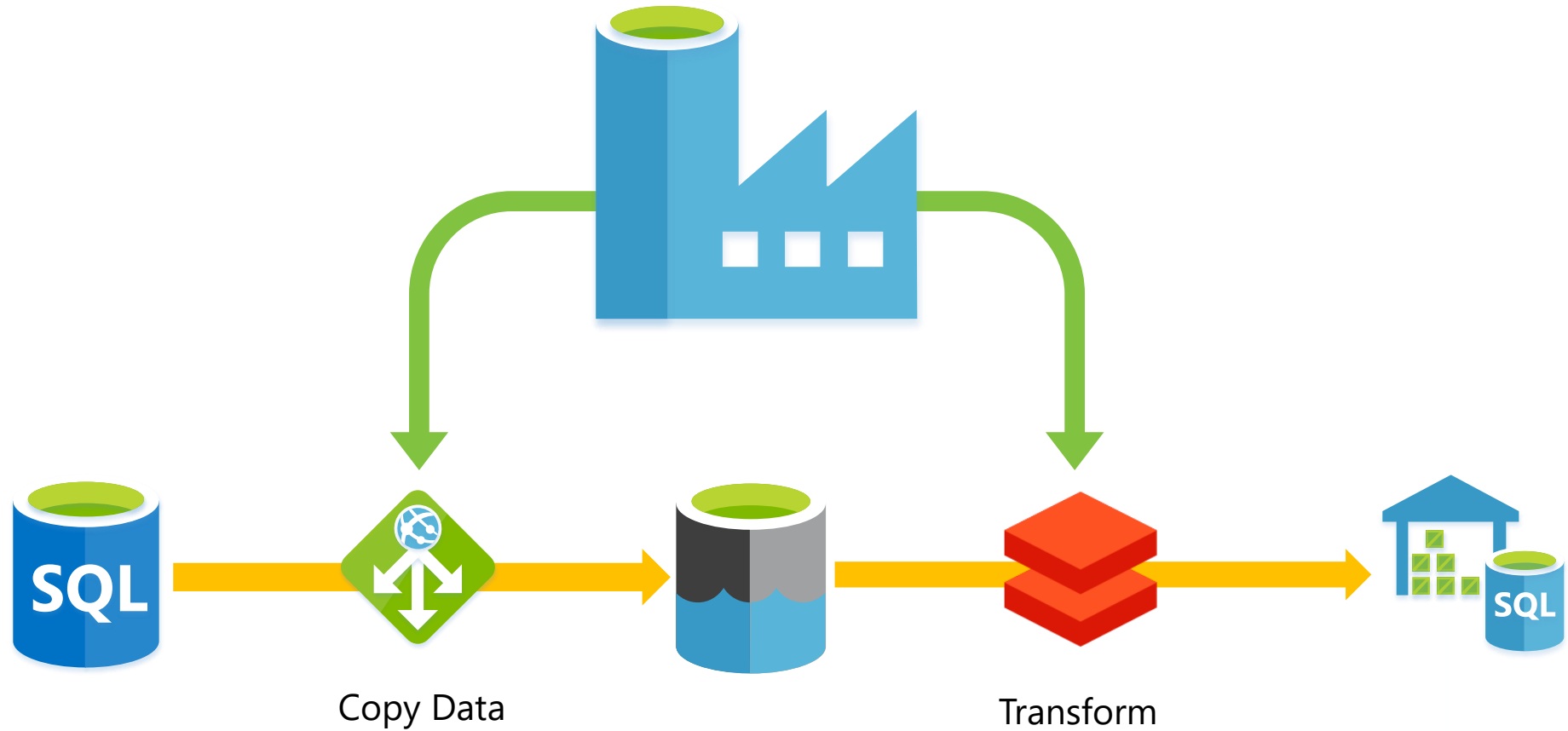
# Azure Data Factory

The background of the slide features a large, solid blue triangle on the right side, pointing towards the top right corner. A thin, light blue vertical line runs parallel to the left edge of this triangle, extending from the bottom to the top of the slide.

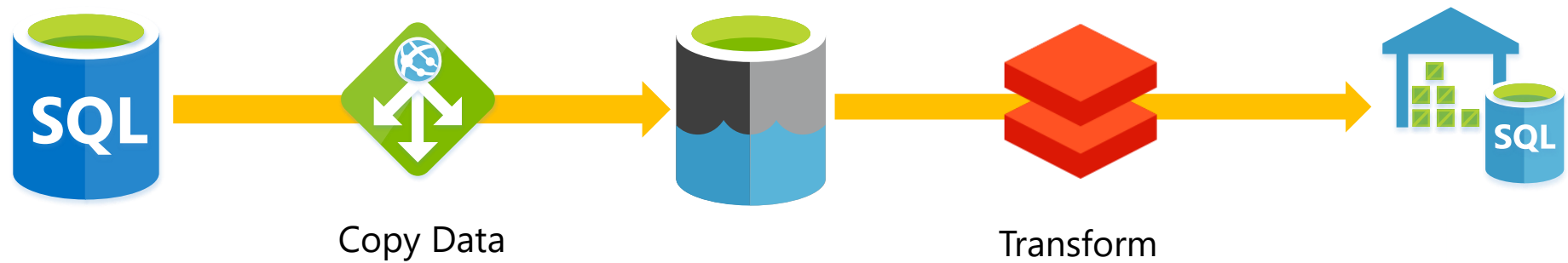
# What is Azure Data Factory?



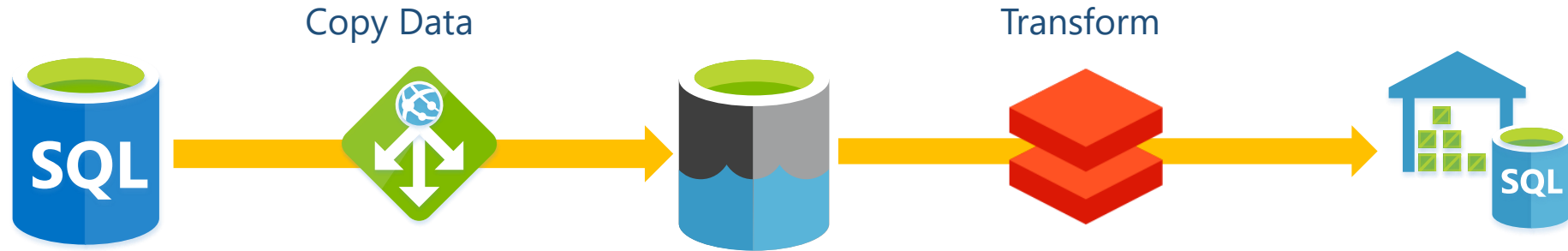
# What is Azure Data Factory?



# What is Azure Data Factory?



# Data Factory Components



1

**Linked Services** – How and what to connect to. Like the SSIS connection manager.



# Data Factory Components



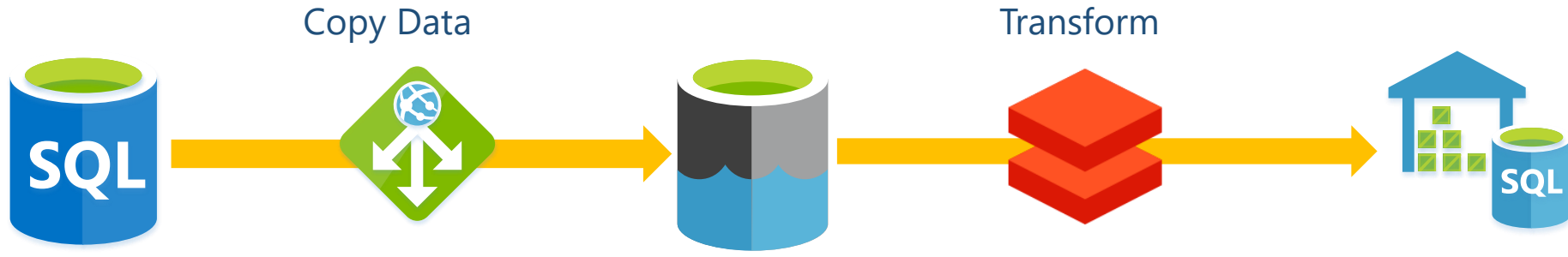
1

## Linked Services –



 Amazon Marketplace Web Service (Preview)	 Amazon Redshift	 Amazon S3	 HDFS	 HTTP	 Hive	 Netezza	 ODBC	 OData	 Azure Batch	 Azure Data Lake Analytics	 Azure Databricks
 Apache Impala (Preview)	 Azure Blob Storage	 Azure Cosmos DB (MongoDB API)	 HubSpot (Preview)	 Informix	 Jira (Preview)	 Office 365 (Preview)	 Oracle	 Oracle Eloqua (Preview)	 Azure Function	 Azure HDInsight	 Azure ML
 Azure Cosmos DB (SQL API)	 Azure Data Explorer (Kusto)	 Azure Data Lake Storage Gen1	 Magento (Preview)	 MariaDB	 Marketo (Preview)	 Oracle Responsys (Preview)	 Oracle Service Cloud (Preview)	 Paypal (Preview)	 ServiceNow	 Shopify (Preview)	 Spark
 Azure Data Lake Storage Gen2 (Preview)	 Azure Database for MariaDB	 Azure Database for MySQL	 Microsoft Access	 MongoDB	 MySQL	 Phoenix	 PostgreSQL	 Presto (Preview)	 Square (Preview)	 Sybase	 Teradata
 Azure Database for PostgreSQL	 Azure File Storage	 Azure Key Vault	 DB2	 Drill (Preview)	 Dynamics 365	 QuickBooks (Preview)	 REST	 SAP BW Open Hub	 Vertica	 Web Table	 Xero (Preview)
 Azure SQL Data Warehouse	 Azure SQL Database	 Azure SQL Database Managed Instance	 Dynamics AX (Preview)	 Dynamics CRM	 FTP	 SAP BW via MDX	 SAP Cloud For Customer	 SAP ECC	 Zoho (Preview)		
 Azure Search	 Azure Table Storage	 Cassandra	 File System	 Google AdWords (Preview)	 Google BigQuery	 SAP HANA	 SFTP	 SQL Server			
 Common Data Service for Apps	 Concur (Preview)	 Couchbase (Preview)	 Google Cloud Storage (S3 API)	 Greenplum	 HBase	 Salesforce	 Salesforce Marketing Cloud (Preview)	 Salesforce Service Cloud			

# Data Factory Components

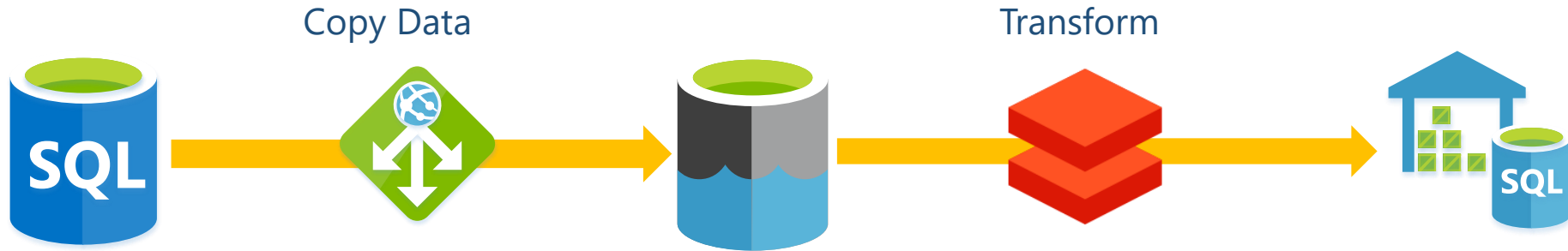


1

## Linked Services



# Data Factory Components



1

## Linked Services

2

**Data Sets** – Where is my data? What format? What file path/table do I need?

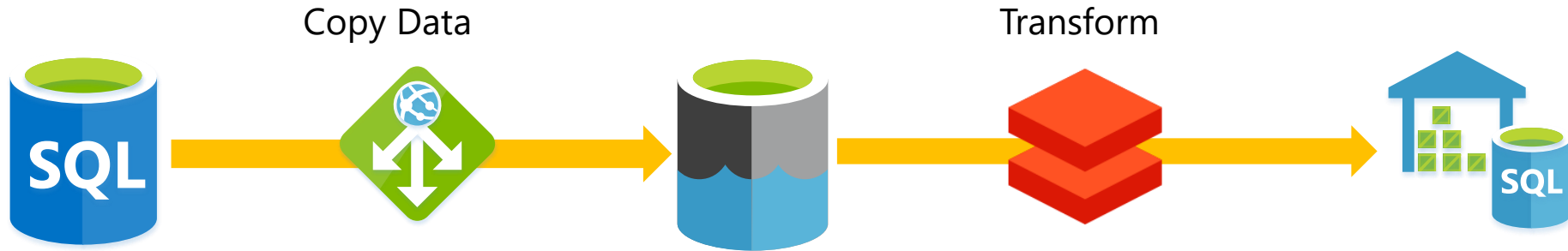


dbo.DimCustomer



/RAW/Orders/2018/01/01/Orders.csv

# Data Factory Components



1

**Linked Services**

2

**Data Sets**

3

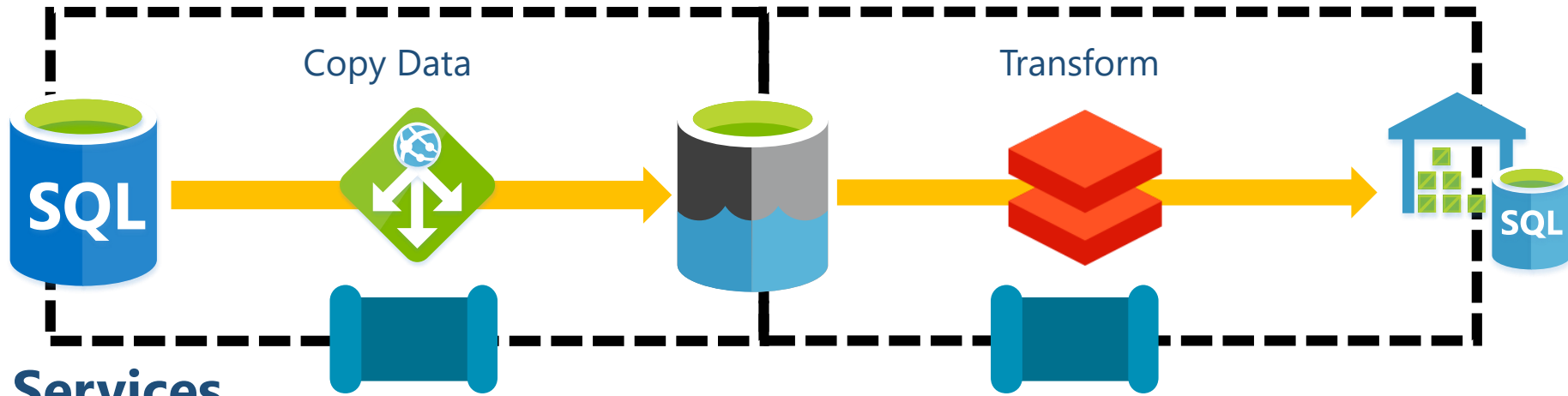
**Activities** – What do we want to happen?  
With what conditions?



## Databricks Notebook Activity

```
notebookPath: /Playground/Playing  
baseParameters: Testing  
libraries[jar]: dbfs:/lib1.jar  
linkedServiceName: BricksOfData01
```

# Data Factory Components



1 **Linked Services**

2 **Data Sets**

3 **Activities**

4 **Pipelines** – What groups of work do I want to do?



Sequence Container

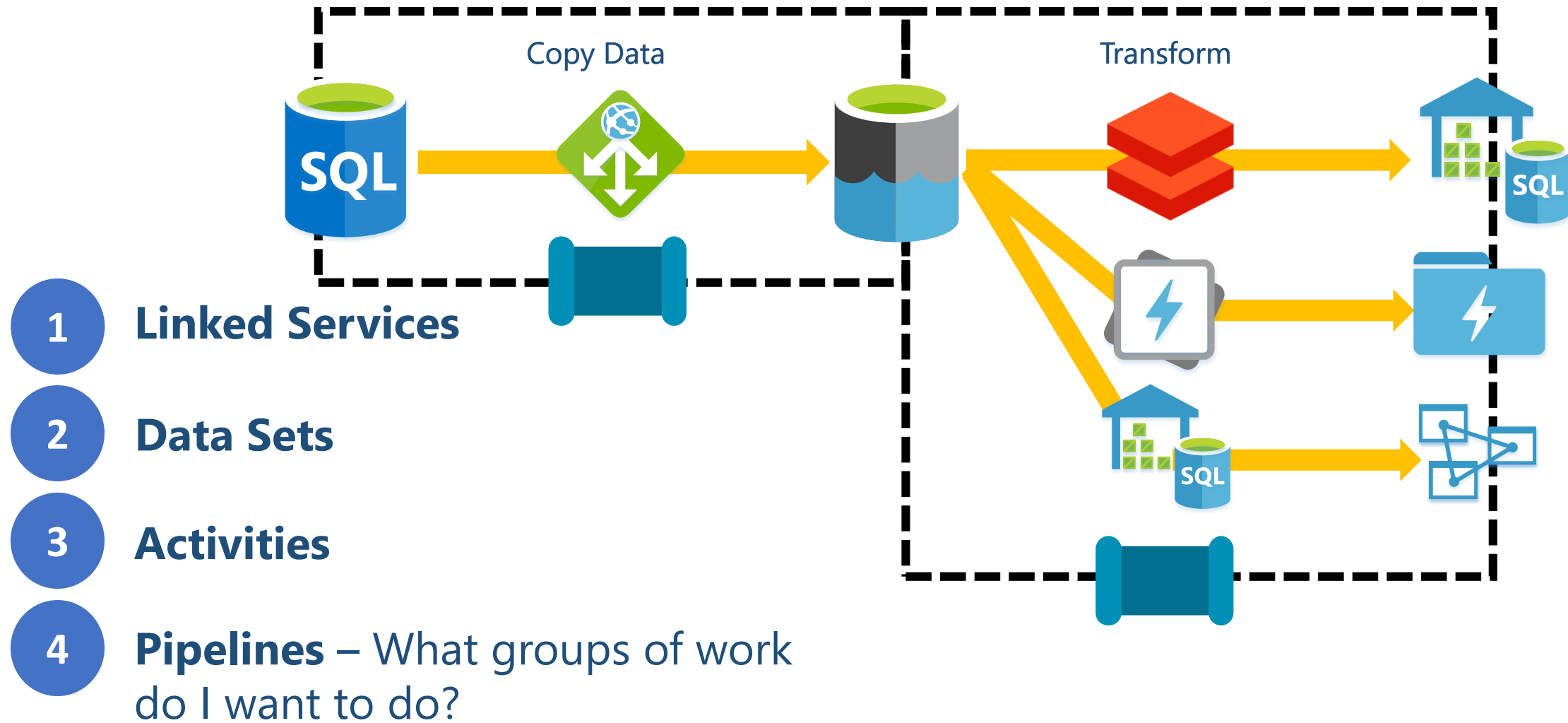


Execute Package Task

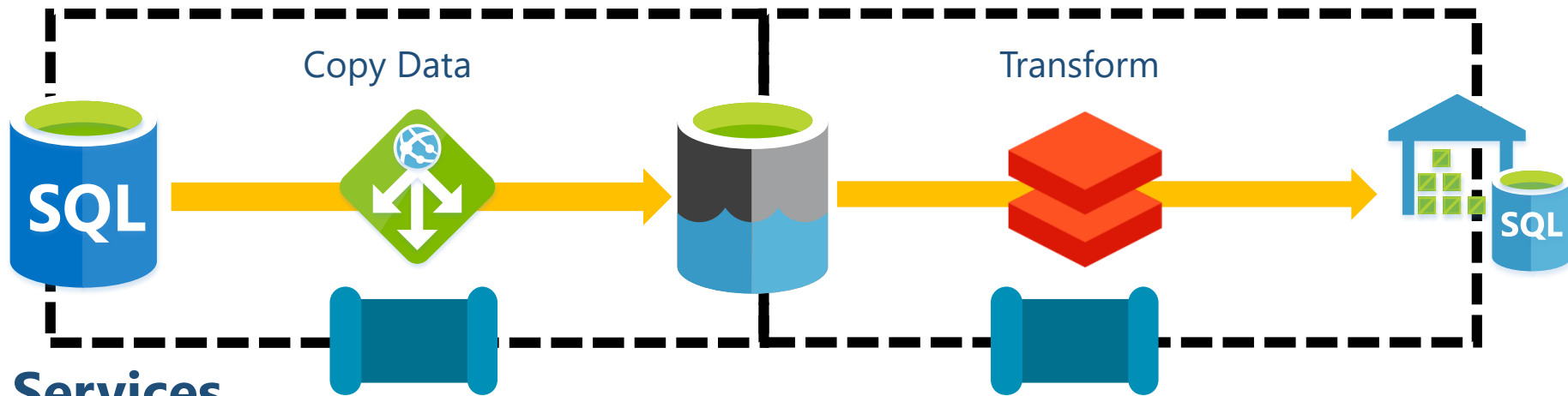


Execute Pipeline Activity

# Data Factory Components



# Data Factory Components



1

**Linked Services**

2

**Data Sets**

3

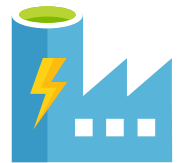
**Activities**

4

**Pipelines**

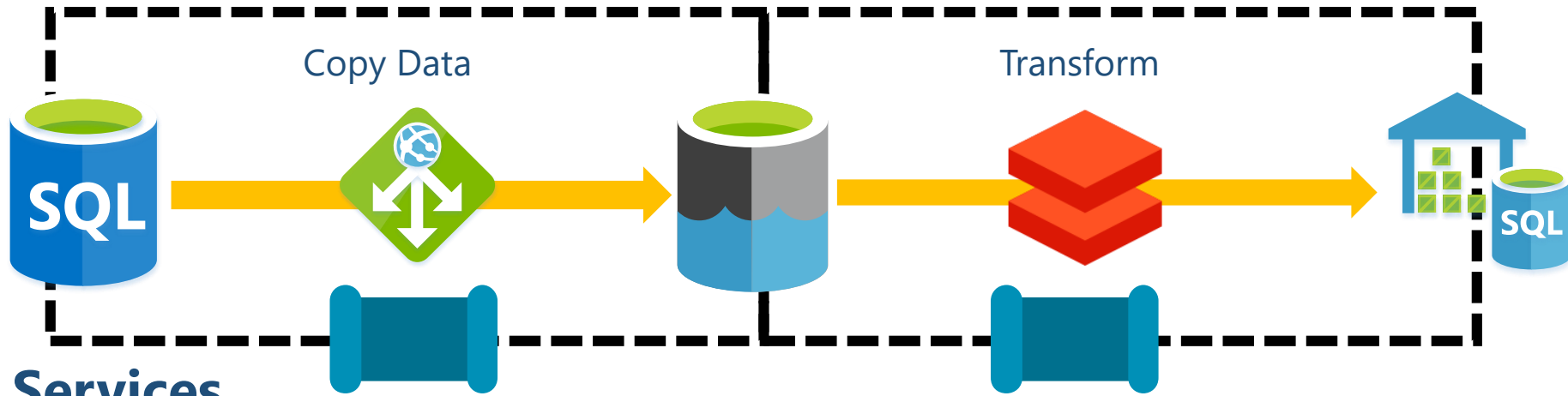
5

**Triggers** – How are we going to tell our pipeline(s) to execute?



- Manual via UI
- Tumbling Windows
- Scheduled
- Blob File Events
- Logic App Calls

# Data Factory Components



1

Linked Services

2

Data Sets

3

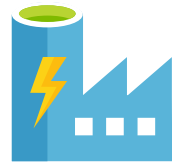
Activities

4

Pipelines

5

Triggers

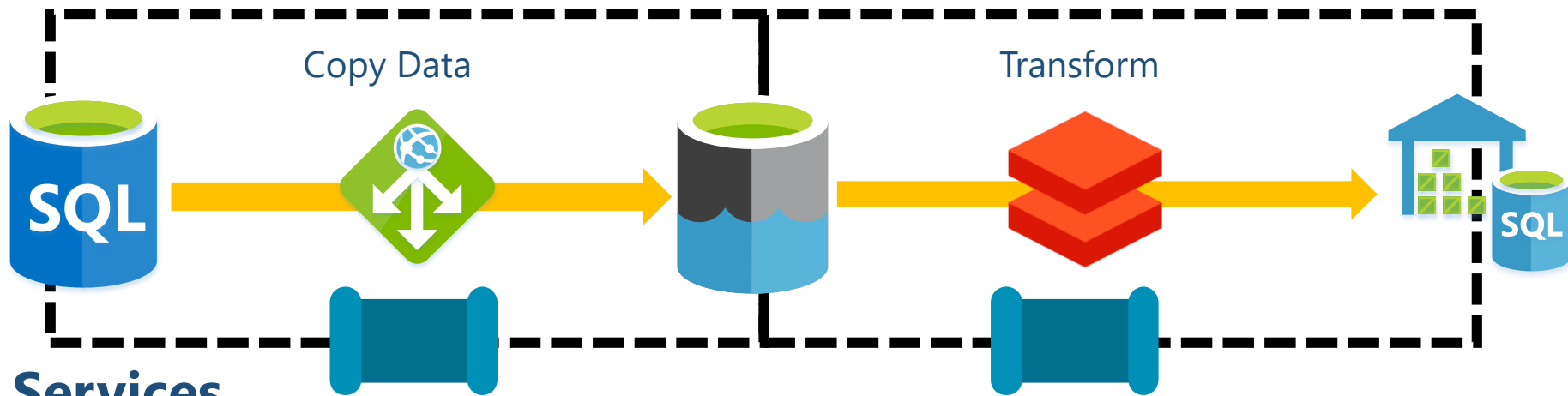


- **Manual**
- Tumbling Windows
- Scheduled
- Blob File Events
- Logic App Calls



```
Invoke-AzureRmDataFactoryV2Pipeline  
-DataFactoryName $dataFactoryName  
-ResourceGroupName $resourceGroupName  
-PipelineName $pipelineName
```

# Data Factory Components



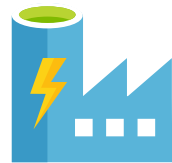
1 Linked Services

2 Data Sets

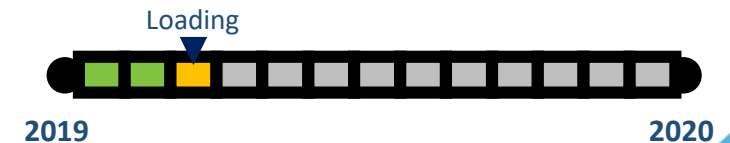
3 Activities

4 Pipelines

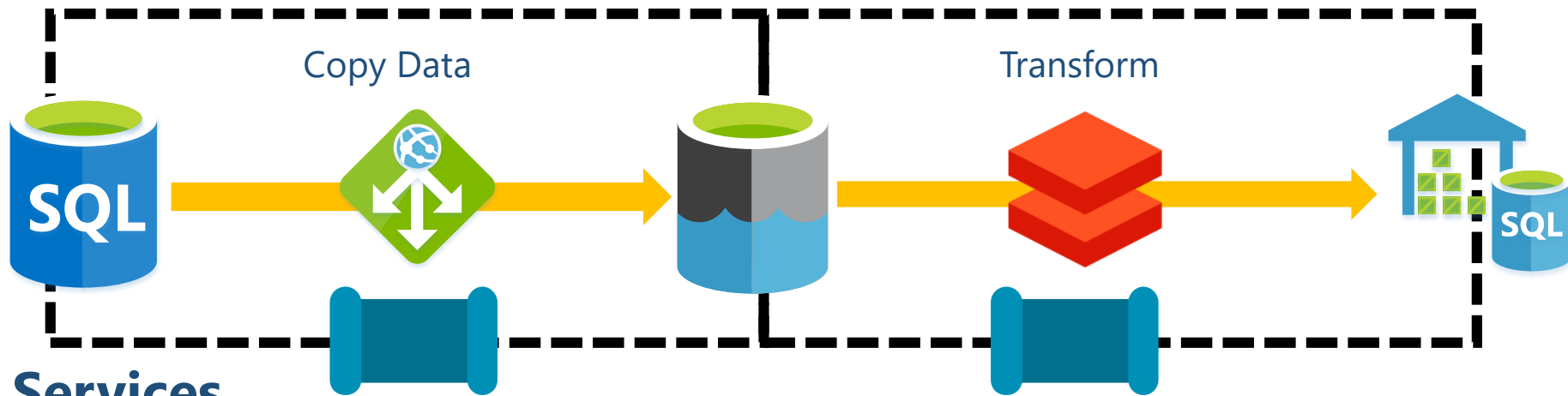
5 Triggers



- Manual via UI
- **Tumbling Windows** - AKA Time Slices
- Scheduled
- Blob File Events
- Logic App Calls



# Data Factory Components



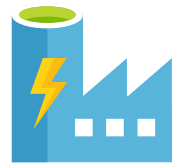
1 Linked Services

2 Data Sets

3 Activities

4 Pipelines

5 Triggers



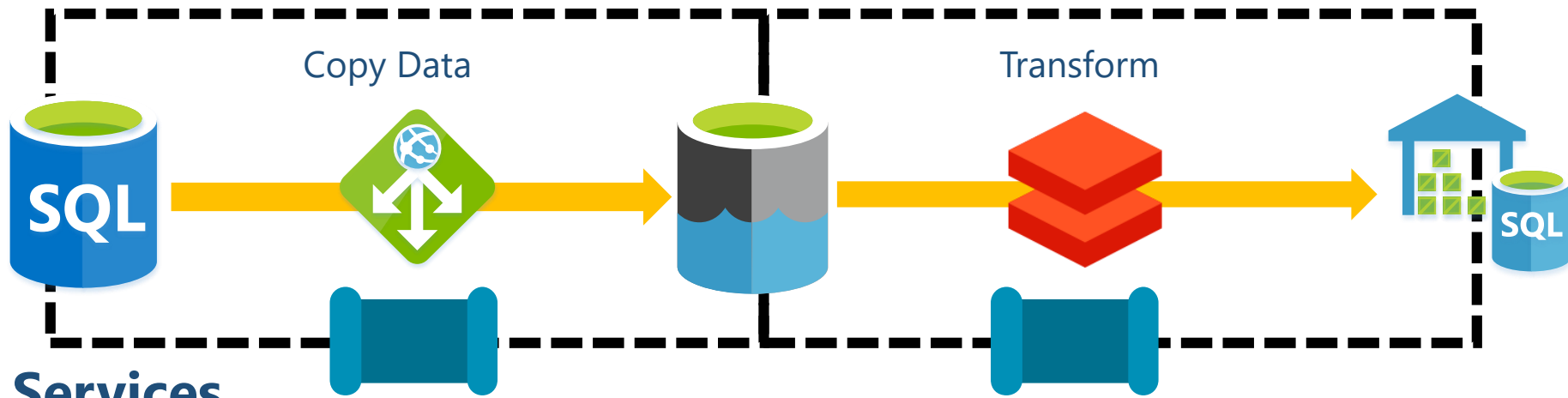
- Manual via UI
- Tumbling Windows
- **Scheduled**
- Blob File Events
- Logic App Calls



- Every 1 minute.
- UTC



# Data Factory Components



1

Linked Services

2

Data Sets

3

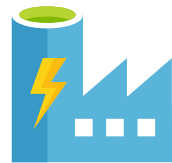
Activities

4

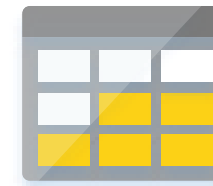
Pipelines

5

Triggers

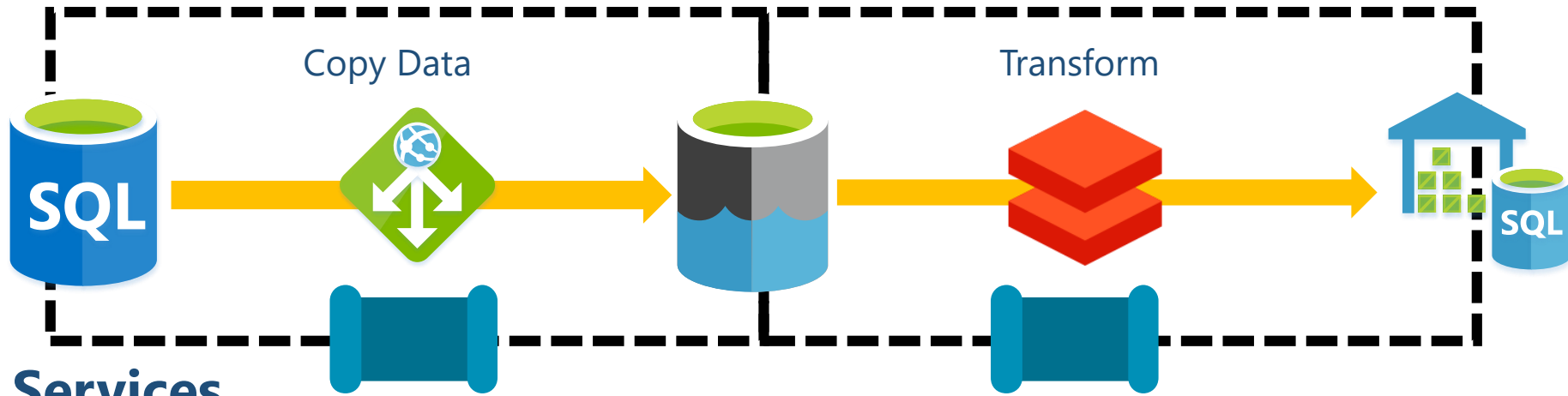


- Manual via UI
- Tumbling Windows
- Scheduled
- **Blob File Events**
- Logic App Calls

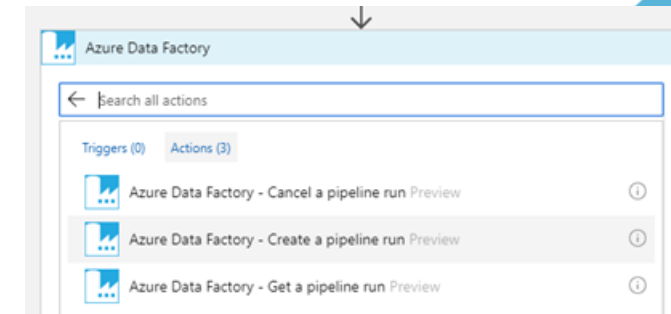
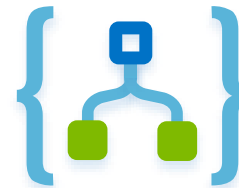


- {Path} Created
- {Path} Deleted

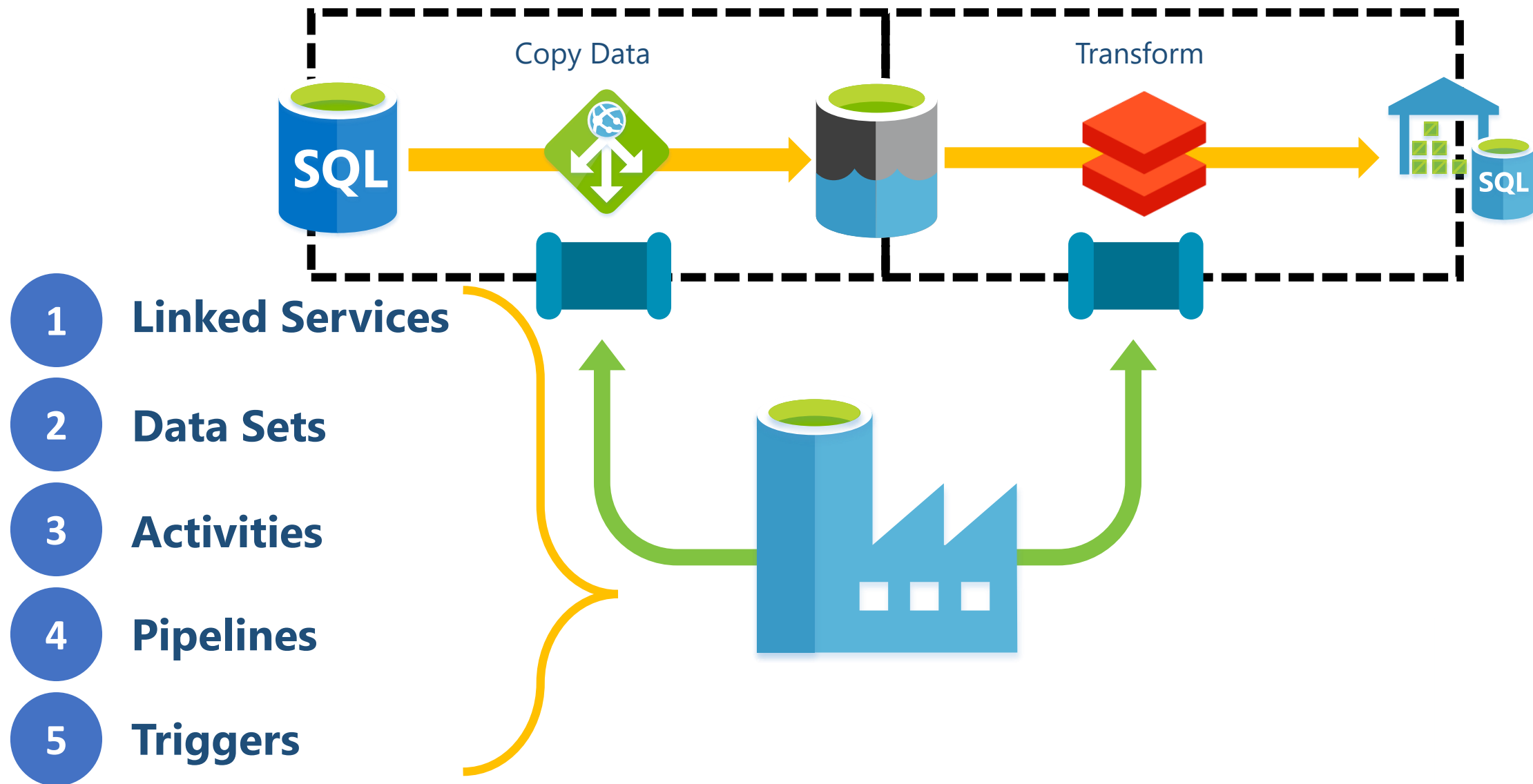
# Data Factory Components



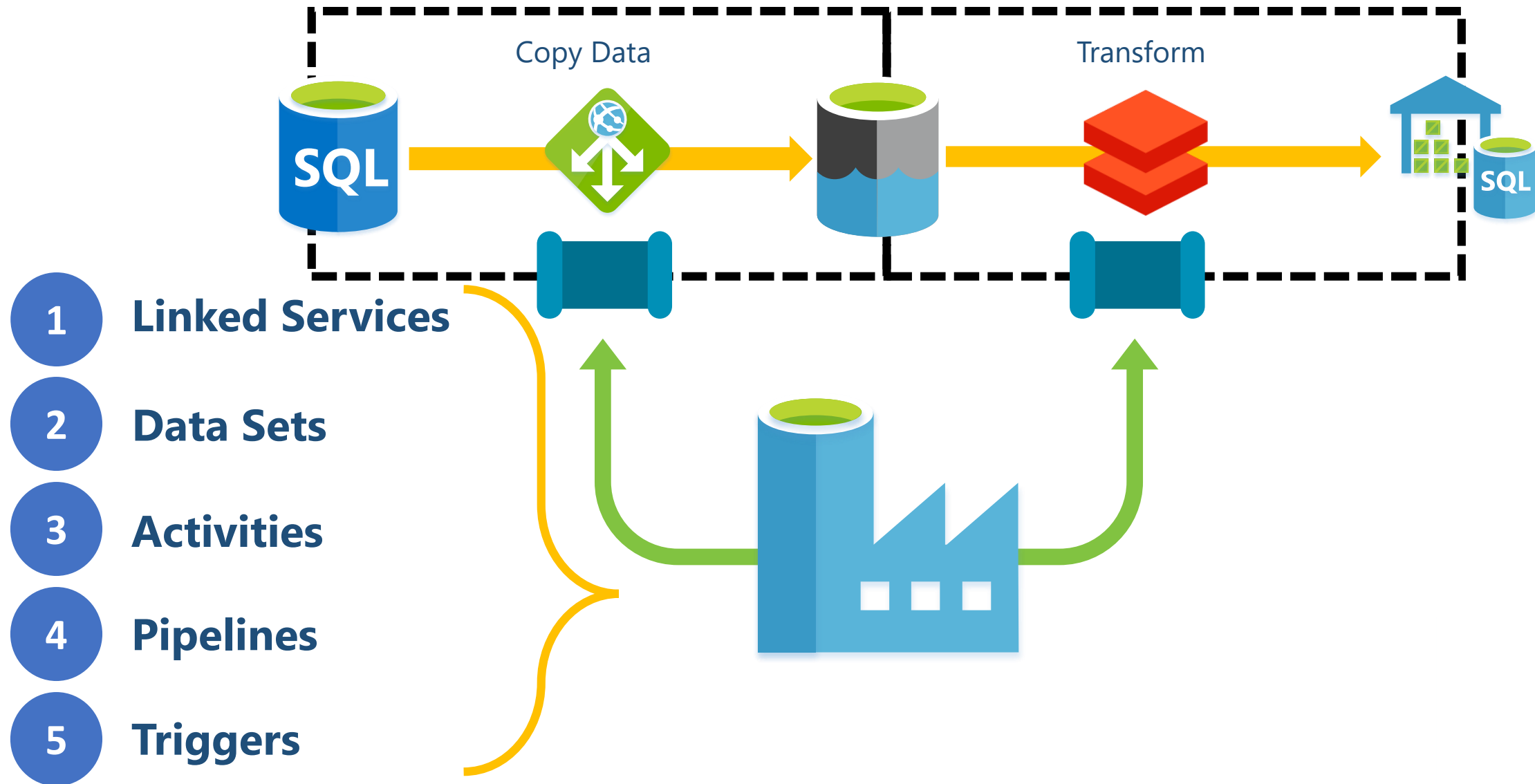
- Manual via UI
- Tumbling Windows
- Scheduled
- Blob File Events
- **Logic App Calls**



# Data Factory Components



# Data Factory Control Flow Components



# Integration Runtimes



1

**Azure**  
Integration Runtime

Movement Hours

Activity Orchestration







2

**SSIS**  
Integration Runtime

SSIS Package Execution







3

**Self Hosted**  
Integration Runtime

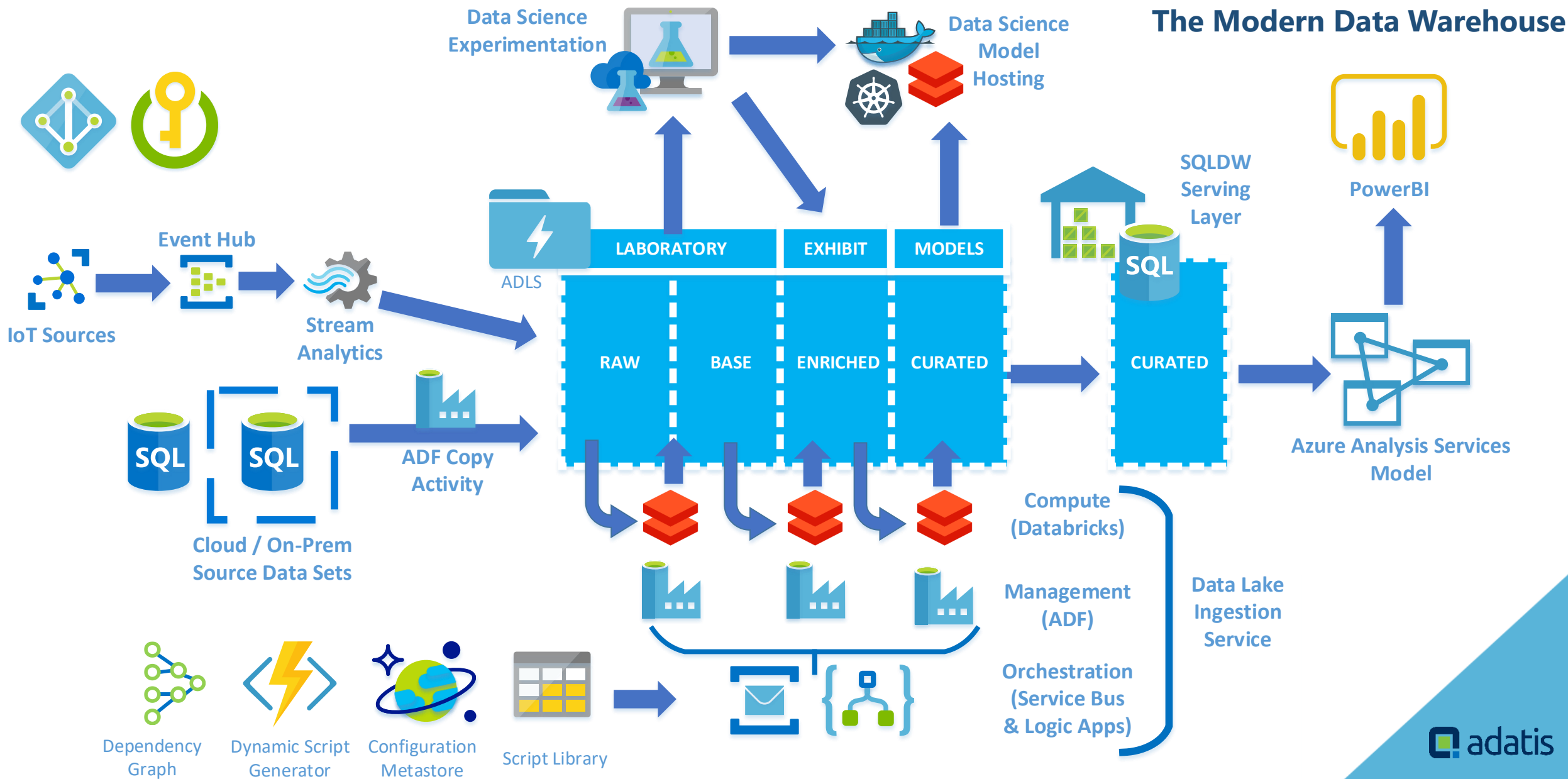
Gateway Access

Activity Orchestration





# Why use Azure Data Factory?



# Data Factory What & Why - Recap

**1 Linked Services**

**2 Data Sets**

**3 Activities**

**4 Pipelines**

**5 Triggers**

**1**

**Azure**

Integration Runtime

**2**

**SSIS**

Integration Runtime

**3**

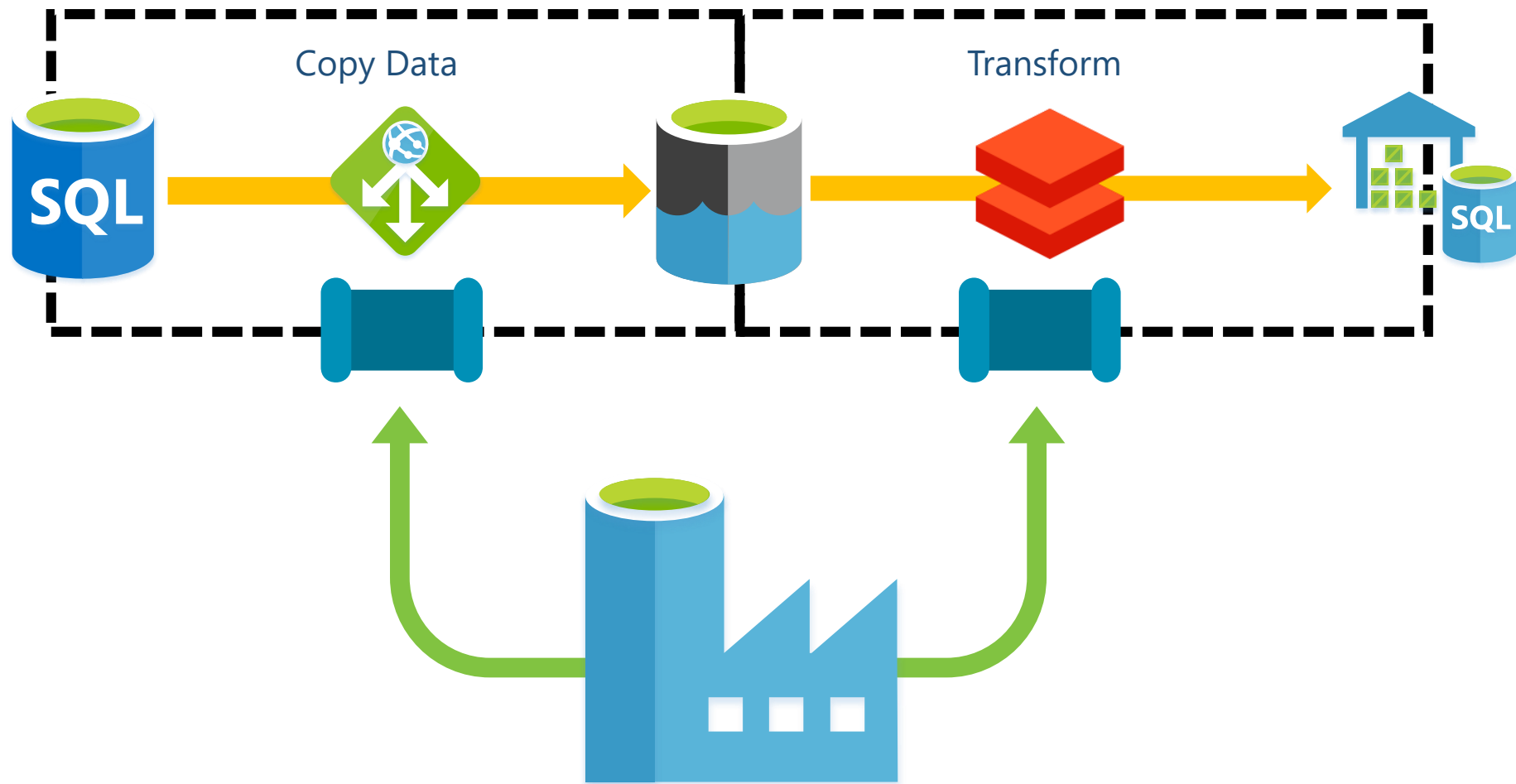
**Self Hosted**

Integration Runtime

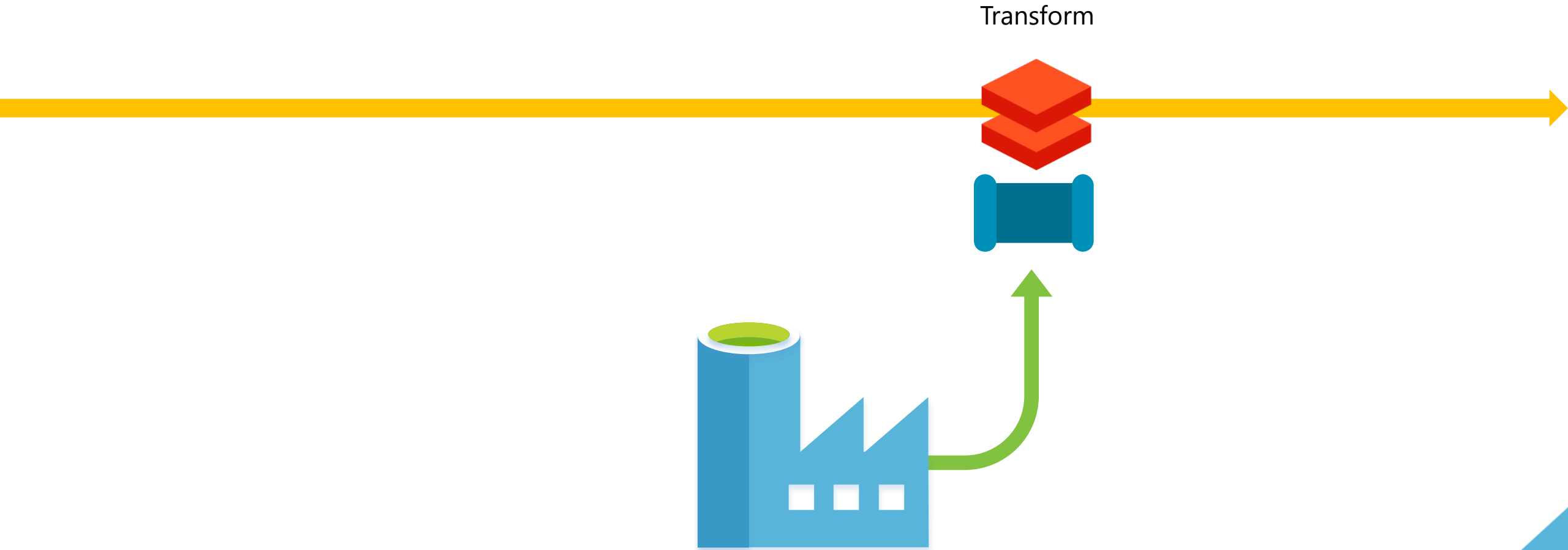
# Data Transformation in zure



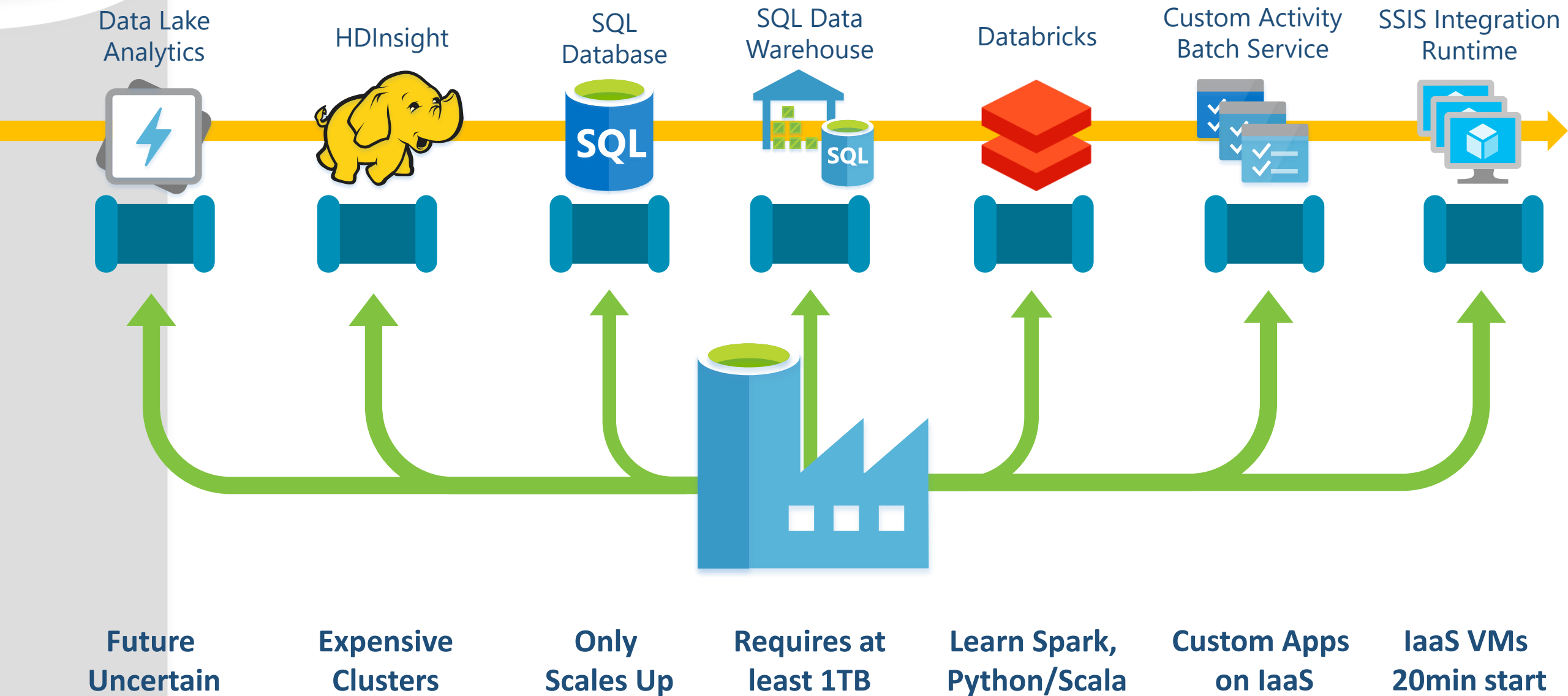
# Data Factory Control Flow Components



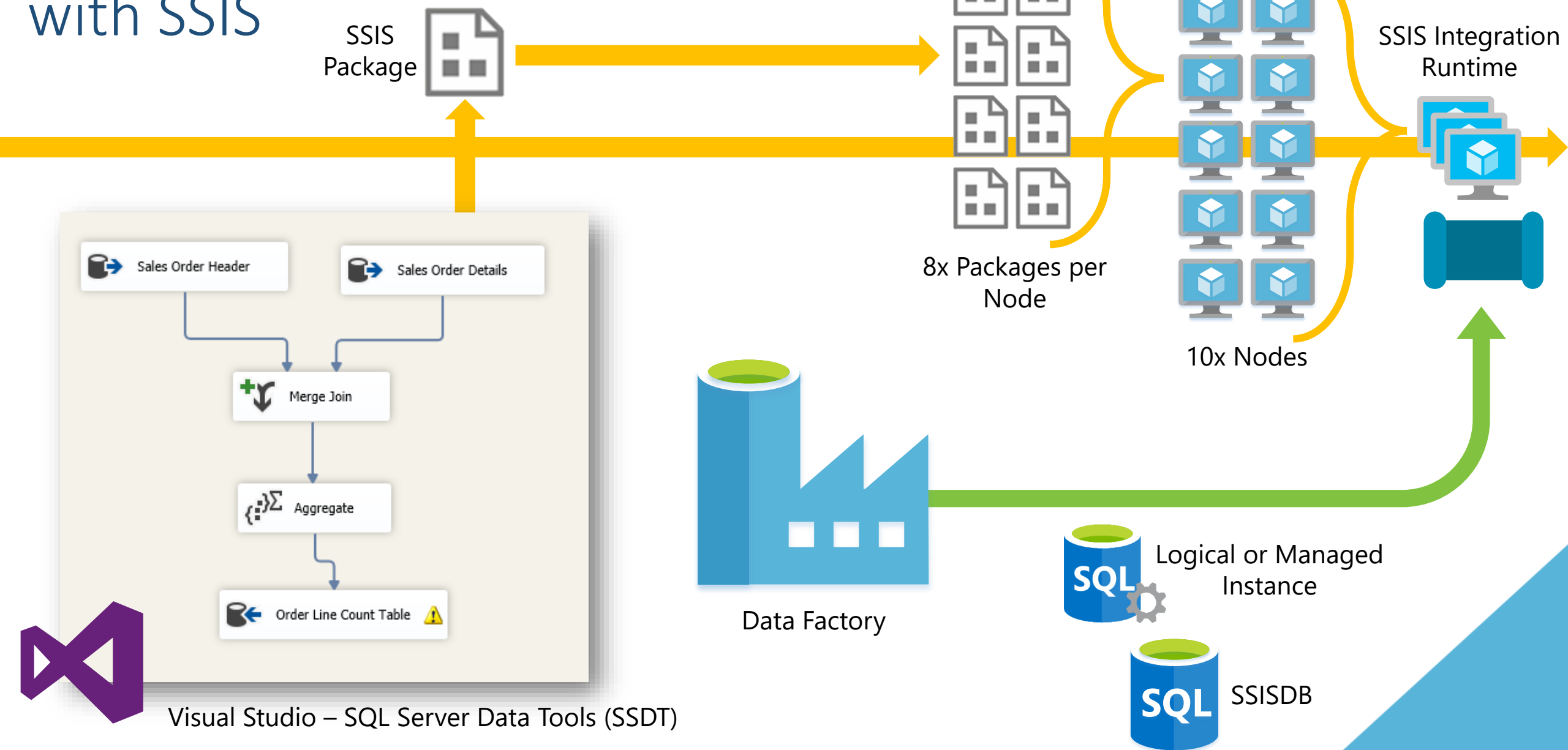
# Data Transformation in zure



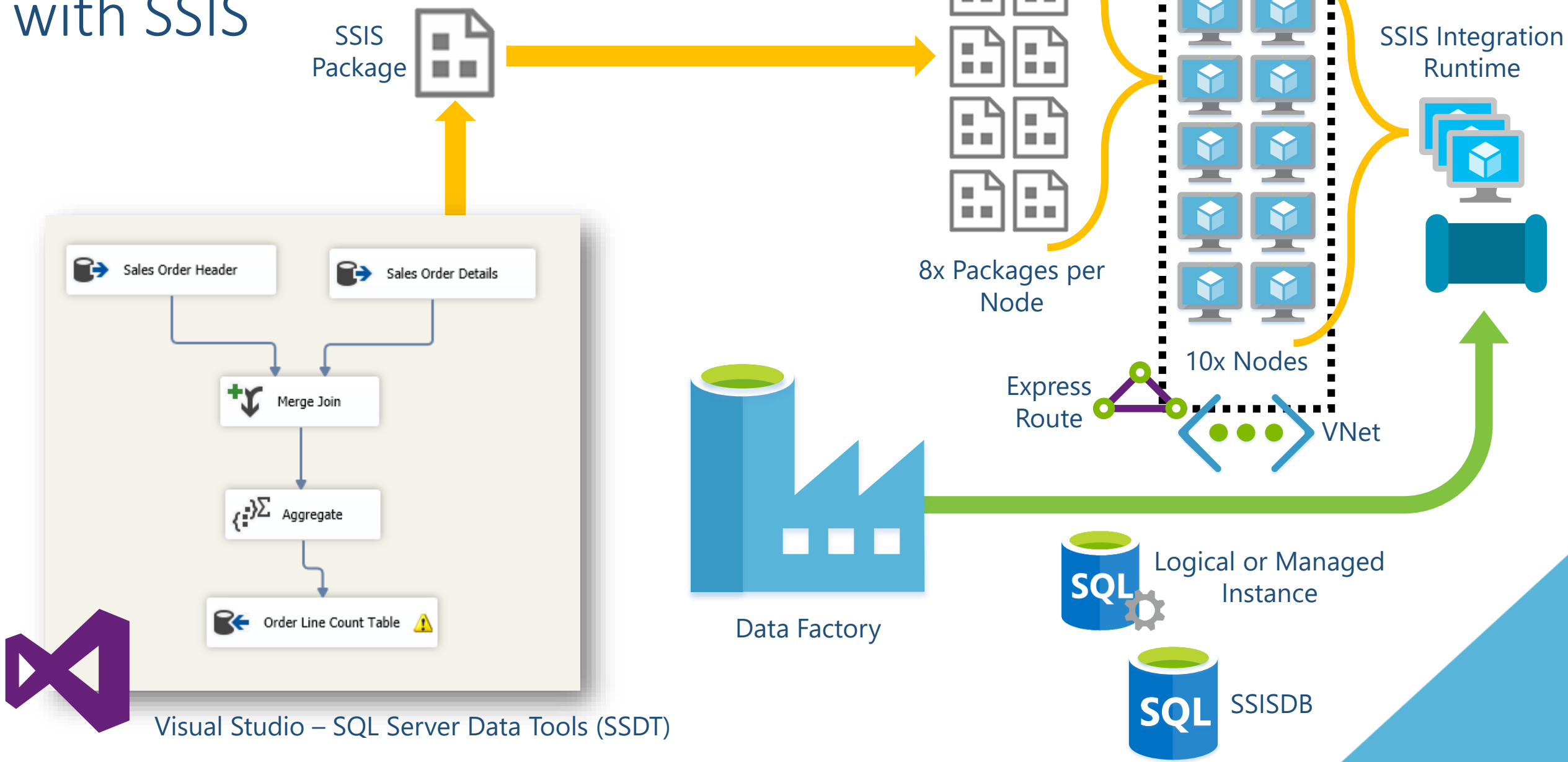
# Data Transformation in Azure



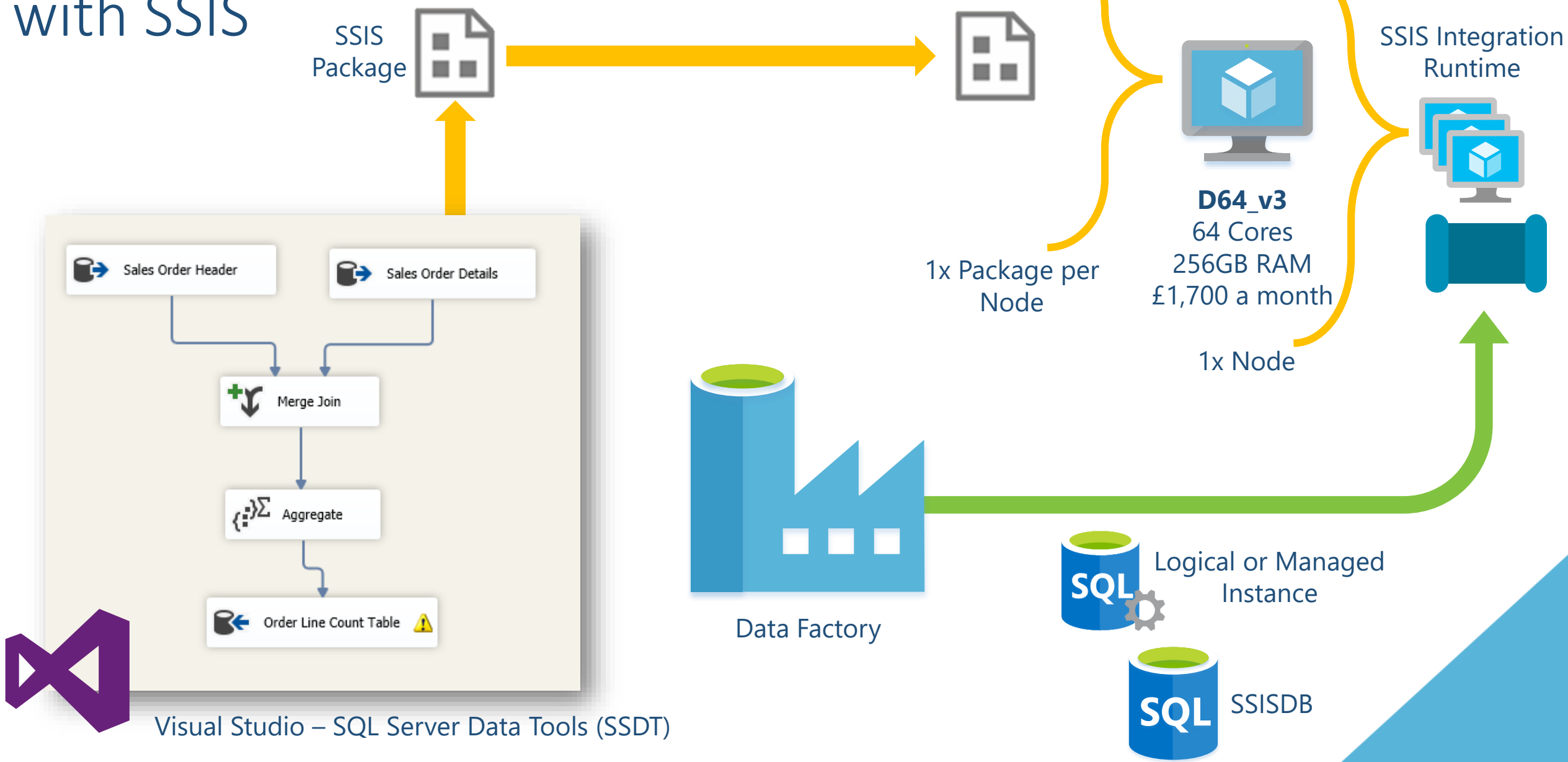
# Data Transformation in zure with SSIS



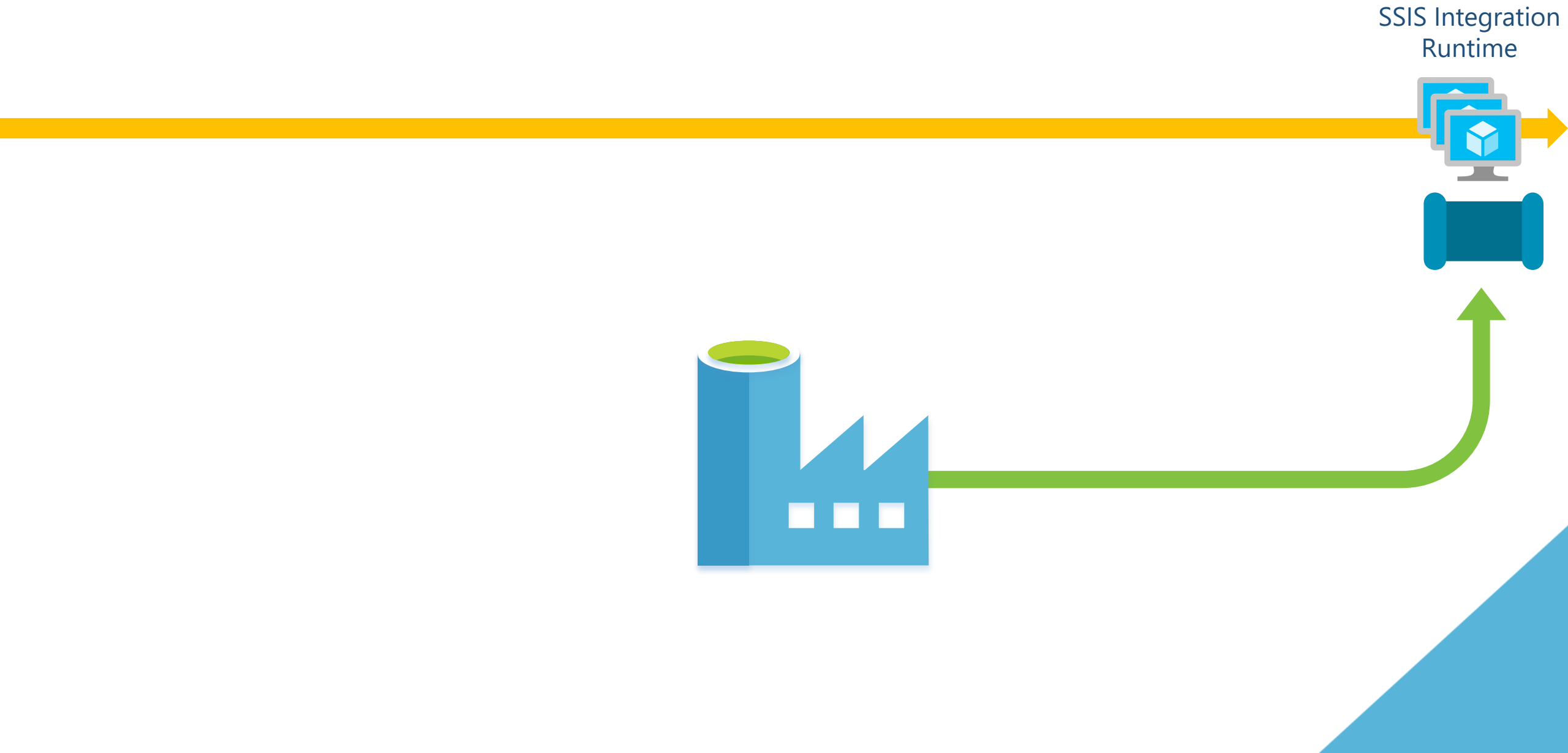
# Data Transformation in zure with SSIS



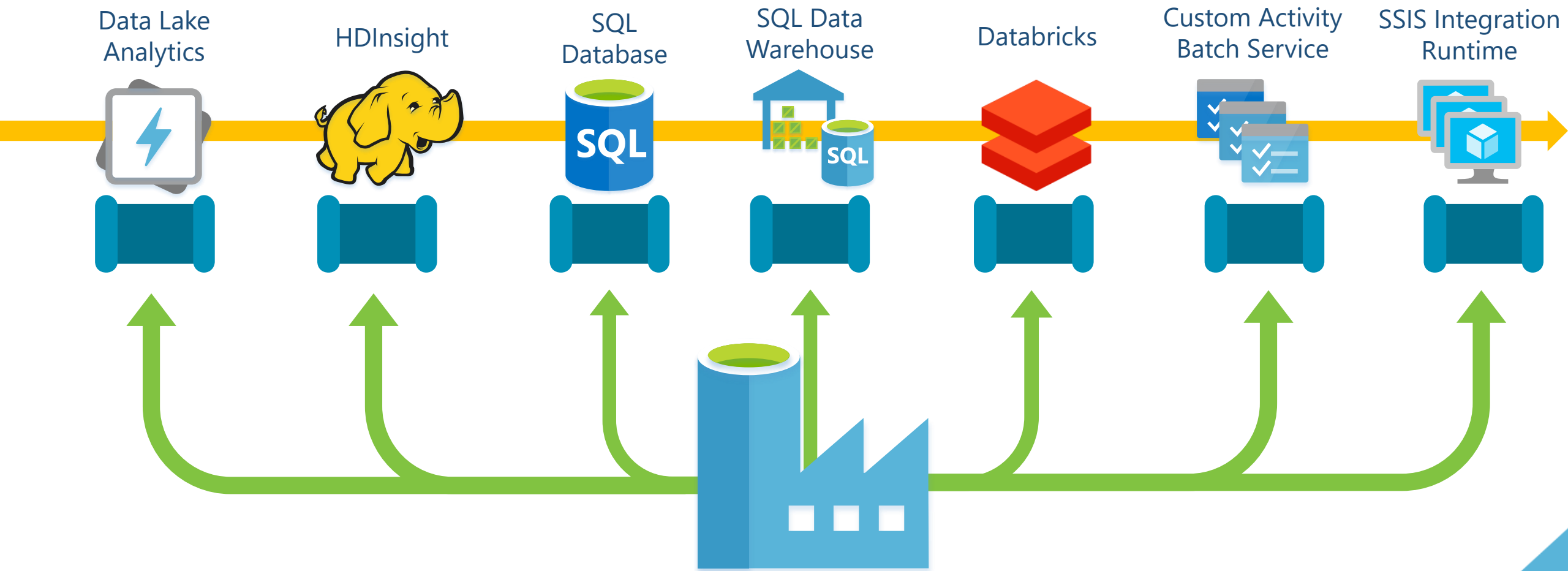
# Data Transformation in zure with SSIS



# Data Transformation in zure

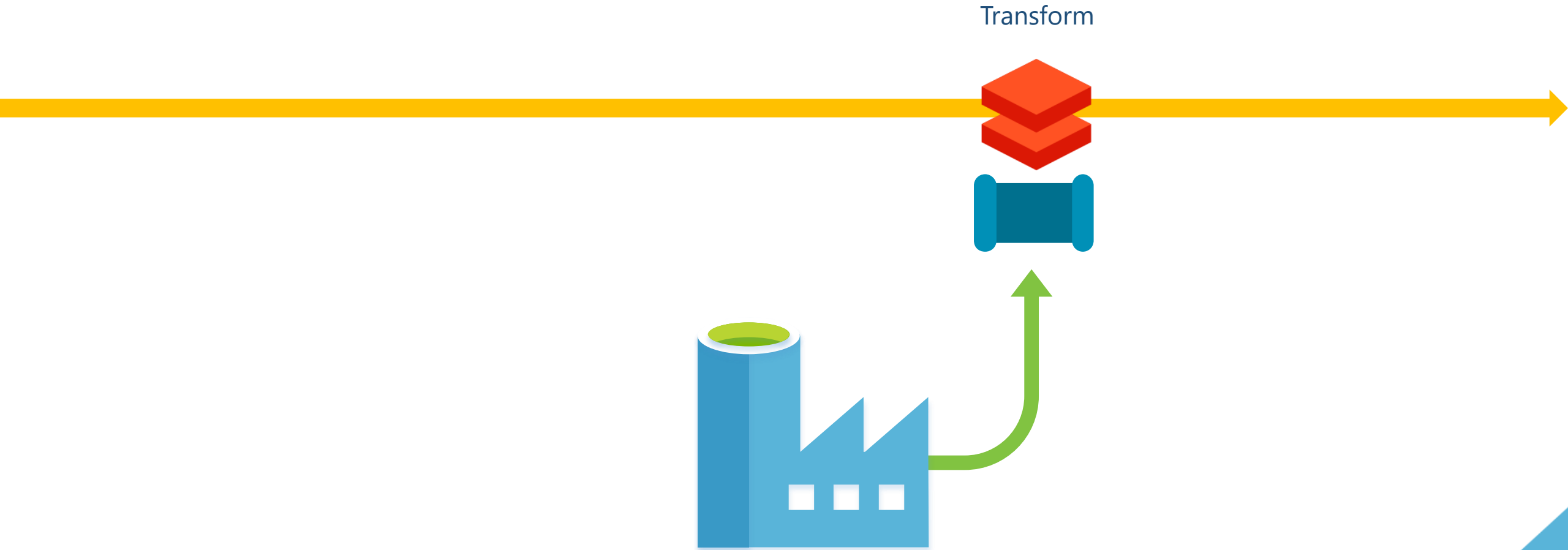


# Data Transformation in zure

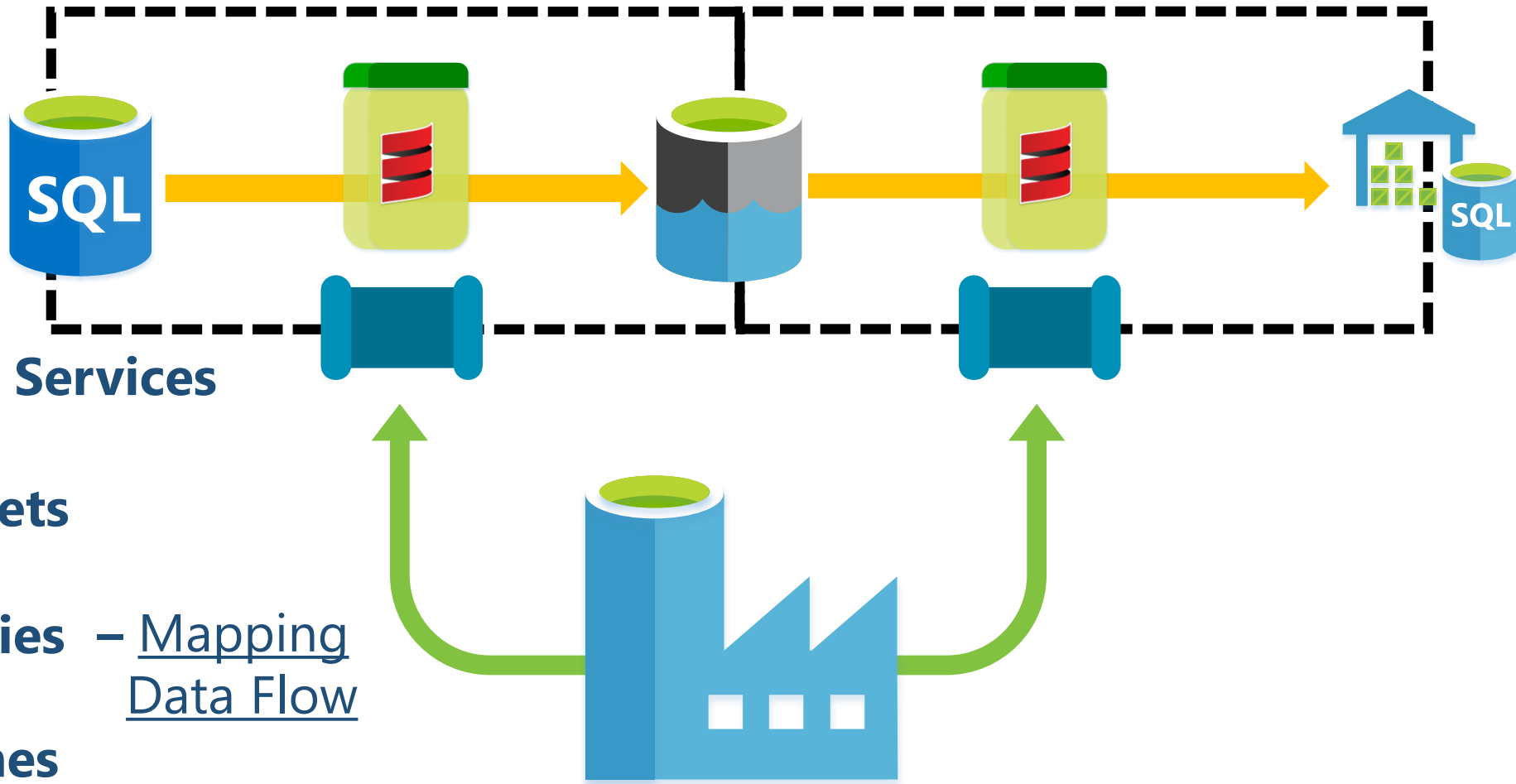




# Data Transformation in zure



# Data Factory Control Flow Components



1

**Linked Services**

2

**Data Sets**

3

**Activities** – Mapping Data Flow

4

**Pipelines**

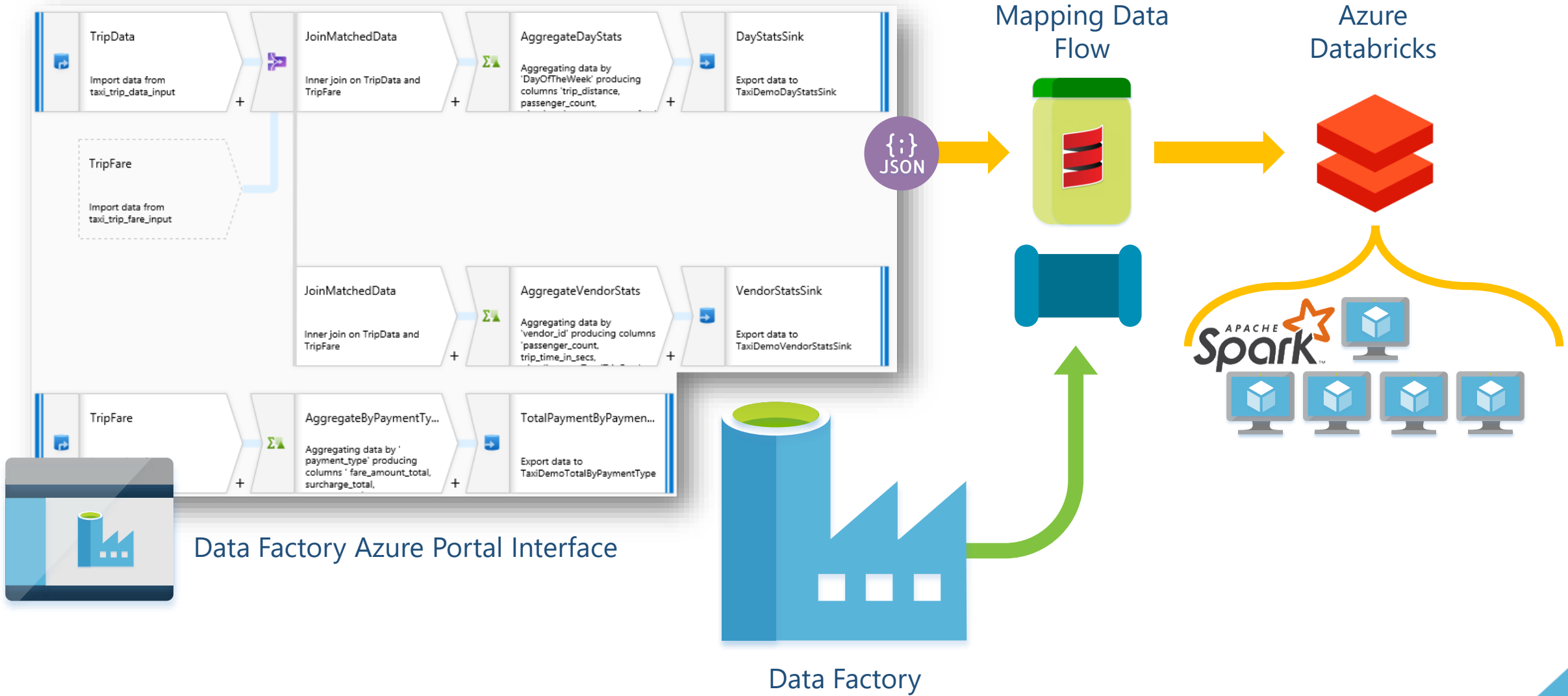
5

**Triggers**

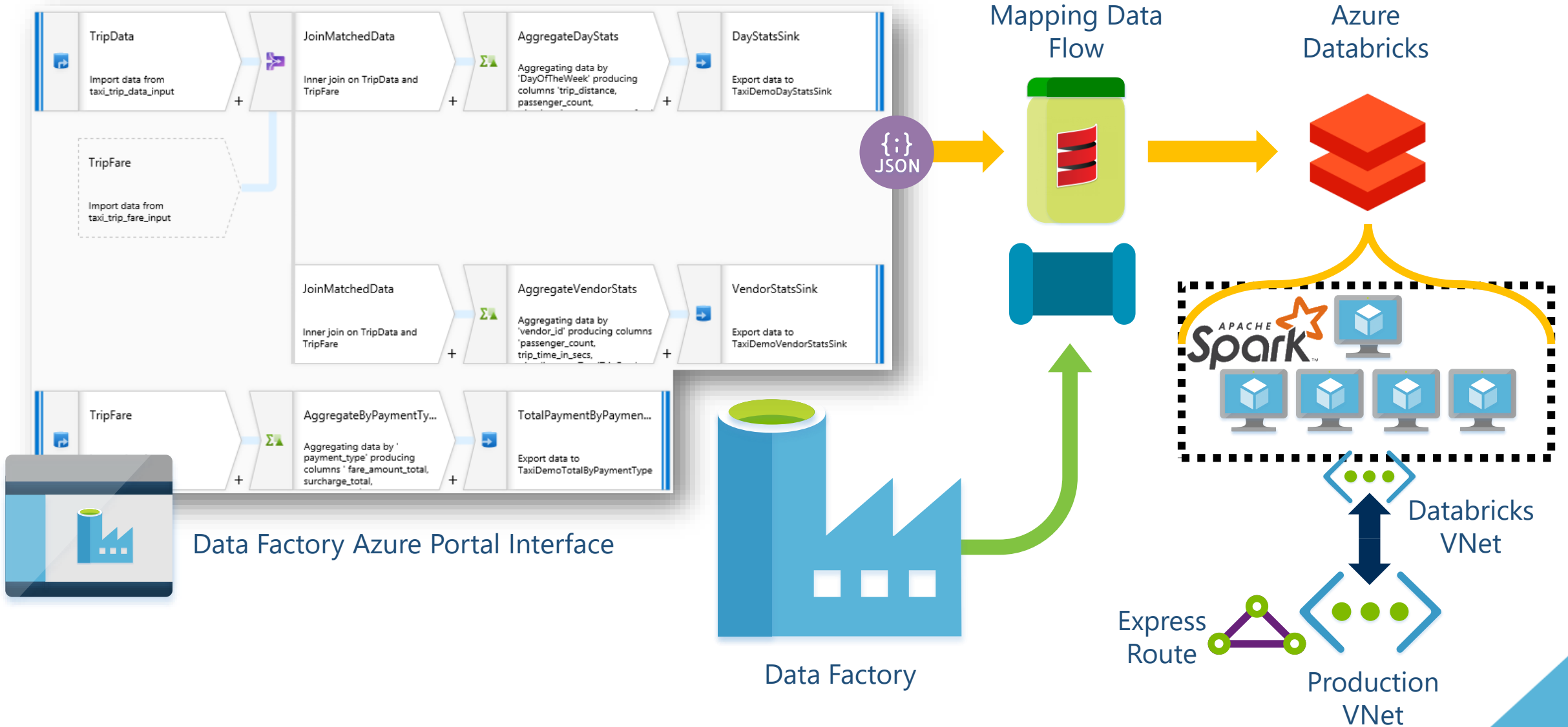
# Mapping Data Flows

A large, solid blue triangle is positioned in the bottom right corner of the slide, pointing towards the top right.

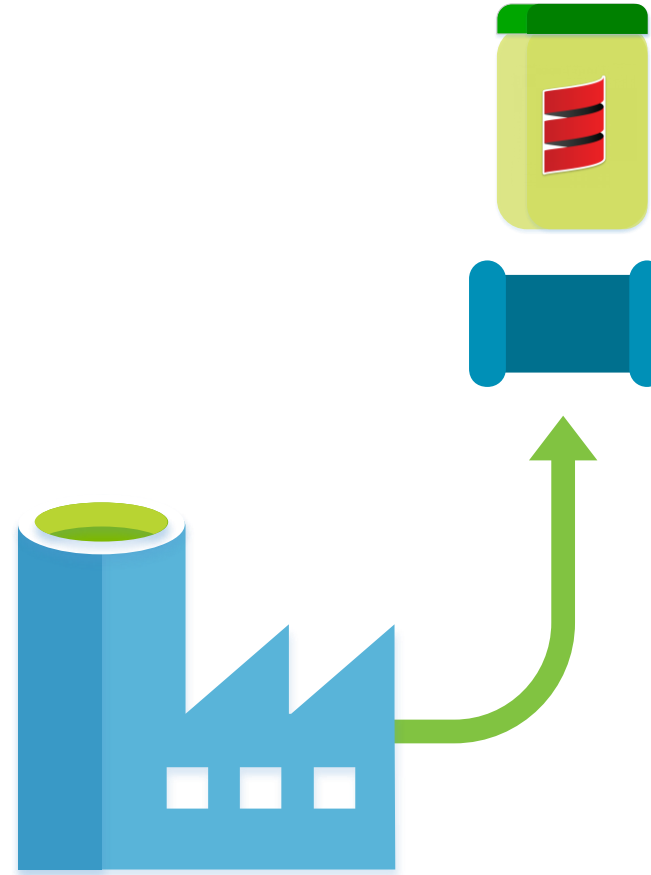
# What is a Mapping Data Flow?



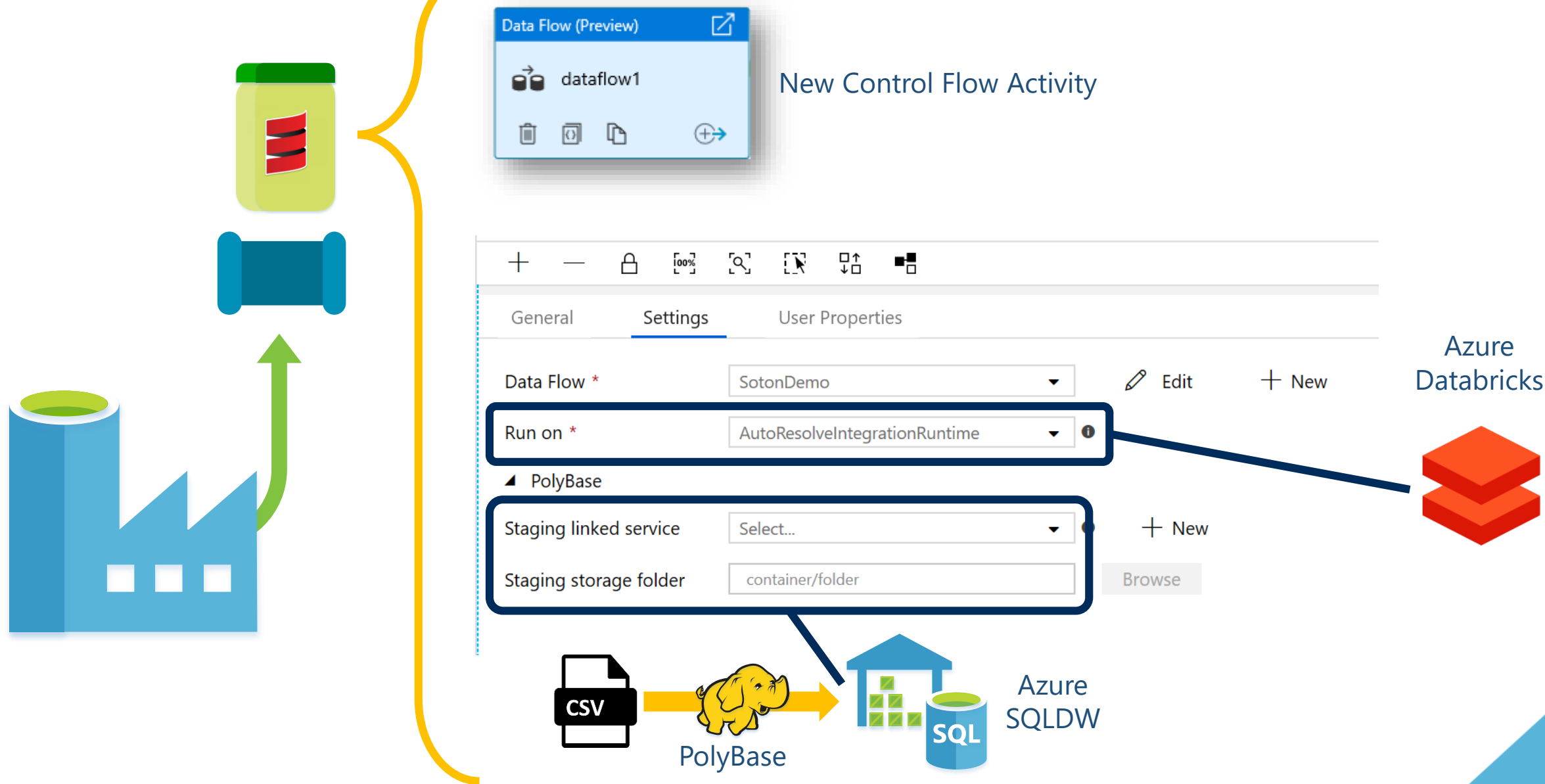
# What is a Mapping Data Flow?



# Mapping Data Flows



# Mapping Data Flows – Settings & Concepts



# Integration Runtimes



1

**Azure**  
Integration Runtime

Movement Hours



Activity  
Orchestration







2

**SSIS**  
Integration Runtime

SSIS Package  
Execution







3

**Self Hosted**  
Integration Runtime

Gateway Access

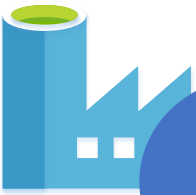
Activity  
Orchestration







# Integration Runtimes – Mapping Data Flow Cluster



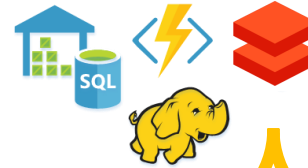
1

## Azure Integration Runtime

Movement Hours



Activity  
Orchestration



Flexible Region



### ▲ Data Flow run time

Compute Type \*

General Purpose

Core count \*

4 (4 Driver Cores)

Time to live (in minutes)

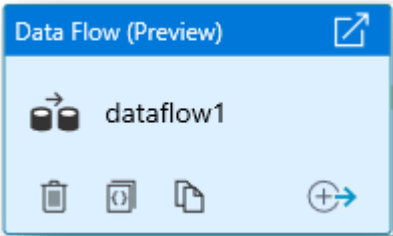
Time to live feature is coming soon

- General Purpose
- Memory Optimised
- Compute Optimised

=



# Mapping Data Flows – Settings & Concepts



New Control Flow Activity

General Settings User Properties

Data Flow \* SotonDemo Edit + New

Run on \* AutoResolveIntegrationRuntime ⓘ

PolyBase

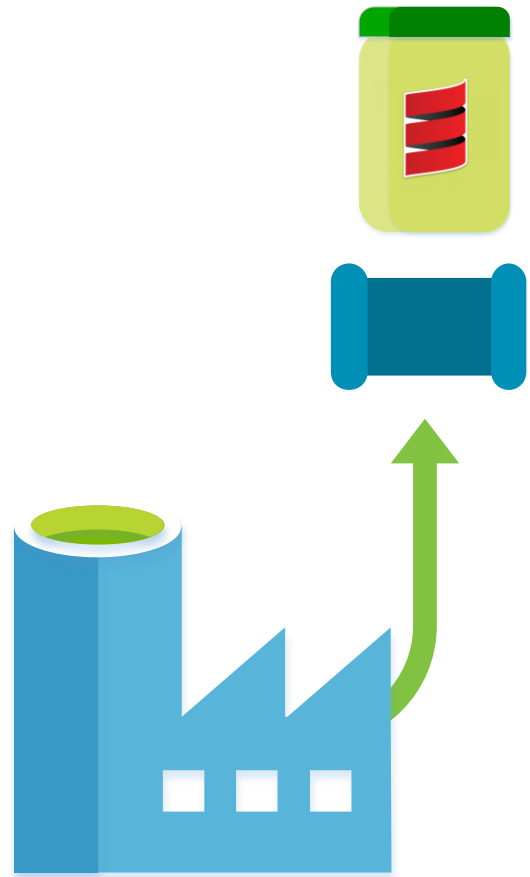
Staging linked service Select... ⓘ + New

Staging storage folder container/folder Browse

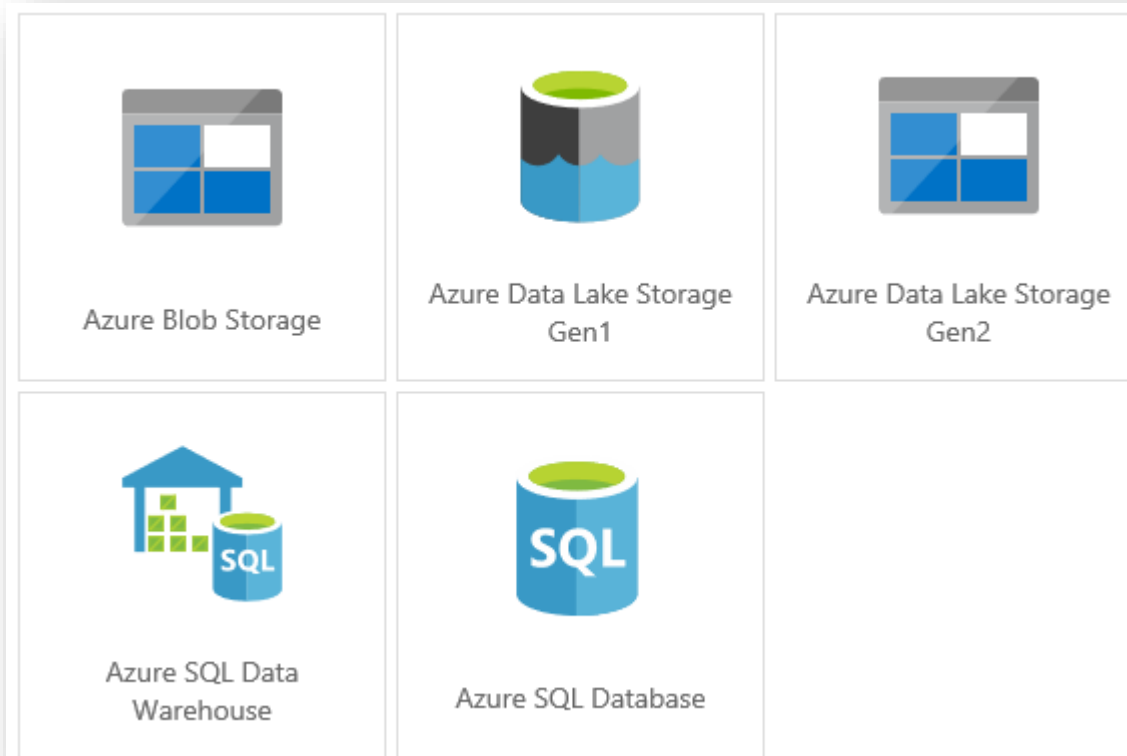
Azure Databricks



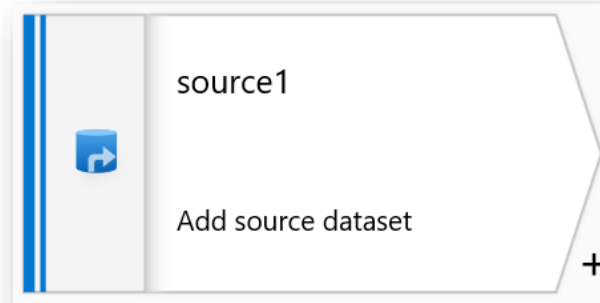
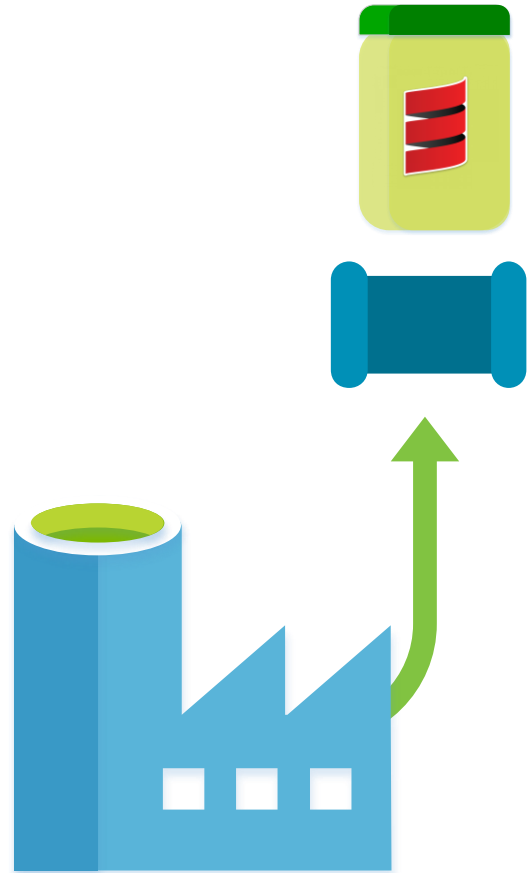
# Mapping Data Flows – Settings & Concepts



Currently Available:



# Mapping Data Flows – Settings & Concepts



Source Settings

Output stream name \*

Source dataset \*  GenericSQLTable  Edit  New

Options

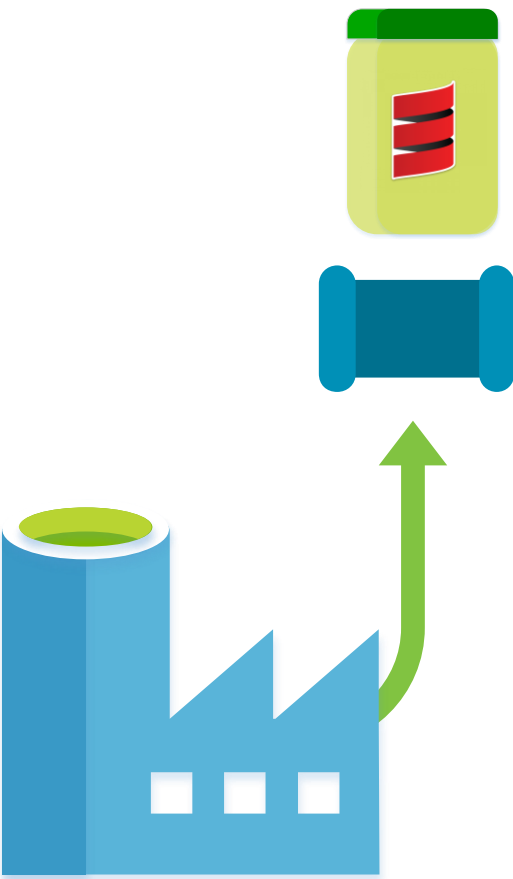
☒ Allow schema drift


☒ Validate schema

Sampling \* ☒ Enable ☐ Disable

Rows limit

# Mapping Data Flows – Settings & Concepts



sink1

Add sink dataset

SinkSettingsMappingOptimizeInspectData Preview

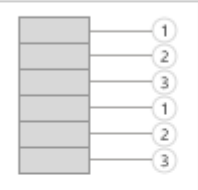
Partition option \*

☐ Use current partitioning

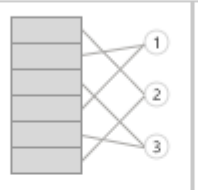
☐ Single partition

☒ Set Partitioning

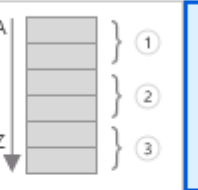
Partition type \*



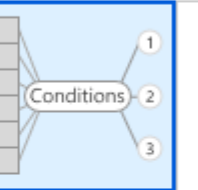
Round Robin



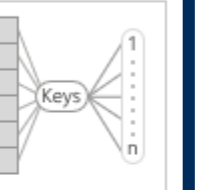
Hash



Dynamic Range



Fixed Range



Key

Number of partitions \*

2

Condition to partition \*

Condition

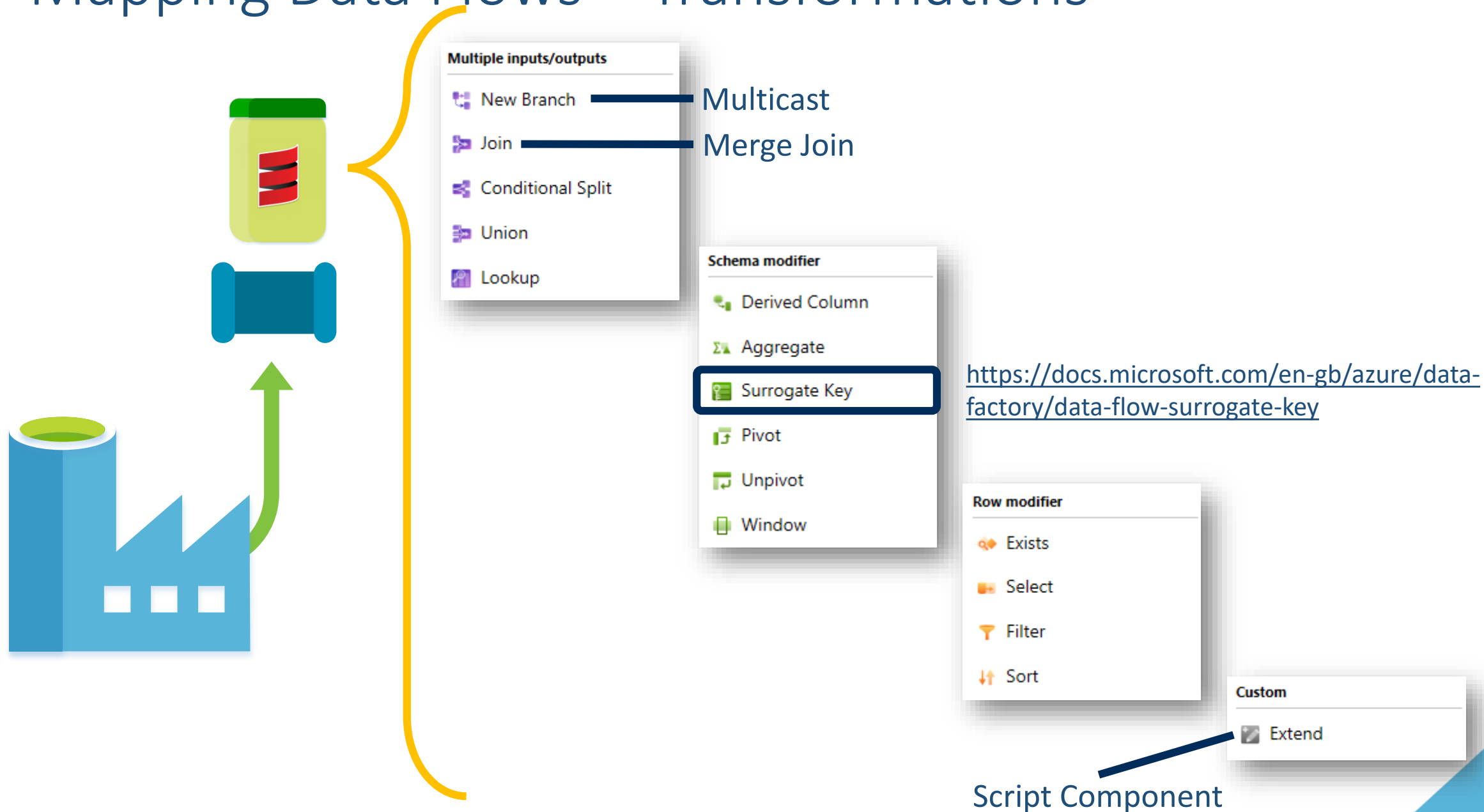
Enter expression...

ANY

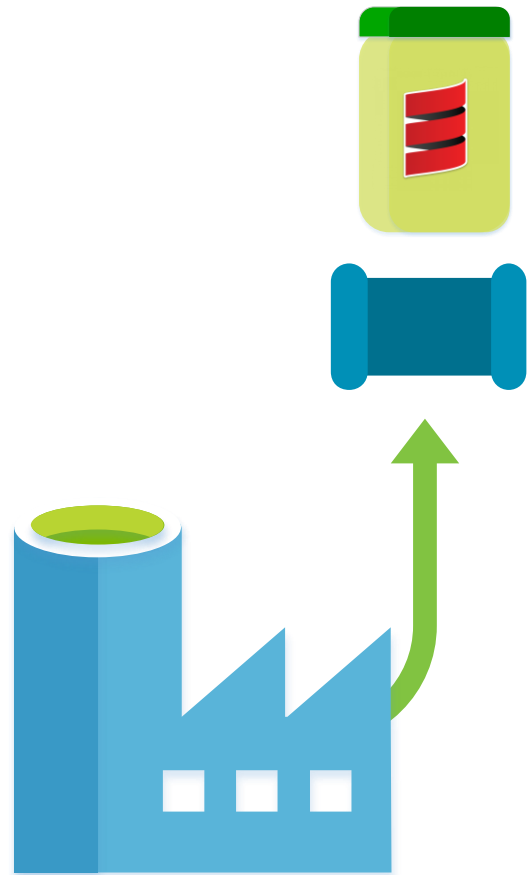
+

🗑️

# Mapping Data Flows – Transformations



# Mapping Data Flows – Expression Builder



### Visual Expression Builder

Currently working on: year

Filter...

All String Math Date Logical Input

abc md5(ANY expression)

123 nextSequence()

abc **regexExtract(abc string, abc regex to find, ANY match group 1-based index)**

✖ regexMatch(abc string, abc regex to match)

abc right(abc string to subset, ANY number of characters)

✖

Extract a matching substring for a given regex pattern. The last parameter identifies the match group and is defaulted to 1 if omitted. Use `<regex>` (back quote) to match a string without escaping

Examples

1. regexExtract('Cost is between 600 and 800 dollars', '(\d+) and (\d+)', 2) -> '800'
2. regexExtract('Cost is between 600 and 800 dollars', '(\d+) and (\d+)', 2) -> '800'

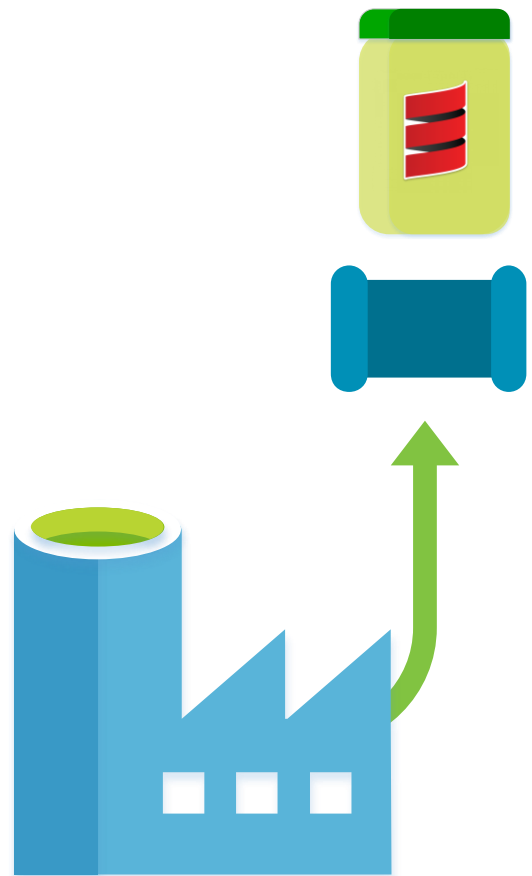
+ - \* / ||

**Data preview**

⚠ Please turn on the debug mode and wait until cluster is ready to preview data...

Output: year 123	title abc
-	-

# Mapping Data Flows – Debug Mode



ADWAnalysis X ADWAnalysisB... X

☒ Debug

Saved

✓ Validate

Source Settings

Cluster ● ForDataFlow

OrderHeader

Columns:  
22 total

OrderDetails

Import data from  
ADWSalesOrderDetail

Join1

Inner join on OrderHeader and  
OrderDetails

Aggregate1

Aggregating data by  
'SalesOrderNumber'  
producing columns  
'OrderLineCount'

sink1

Export data to  
ADWOrderLineCountTable

Add Source

Source Settings


Define schema

Optimize

Inspect

Data Preview

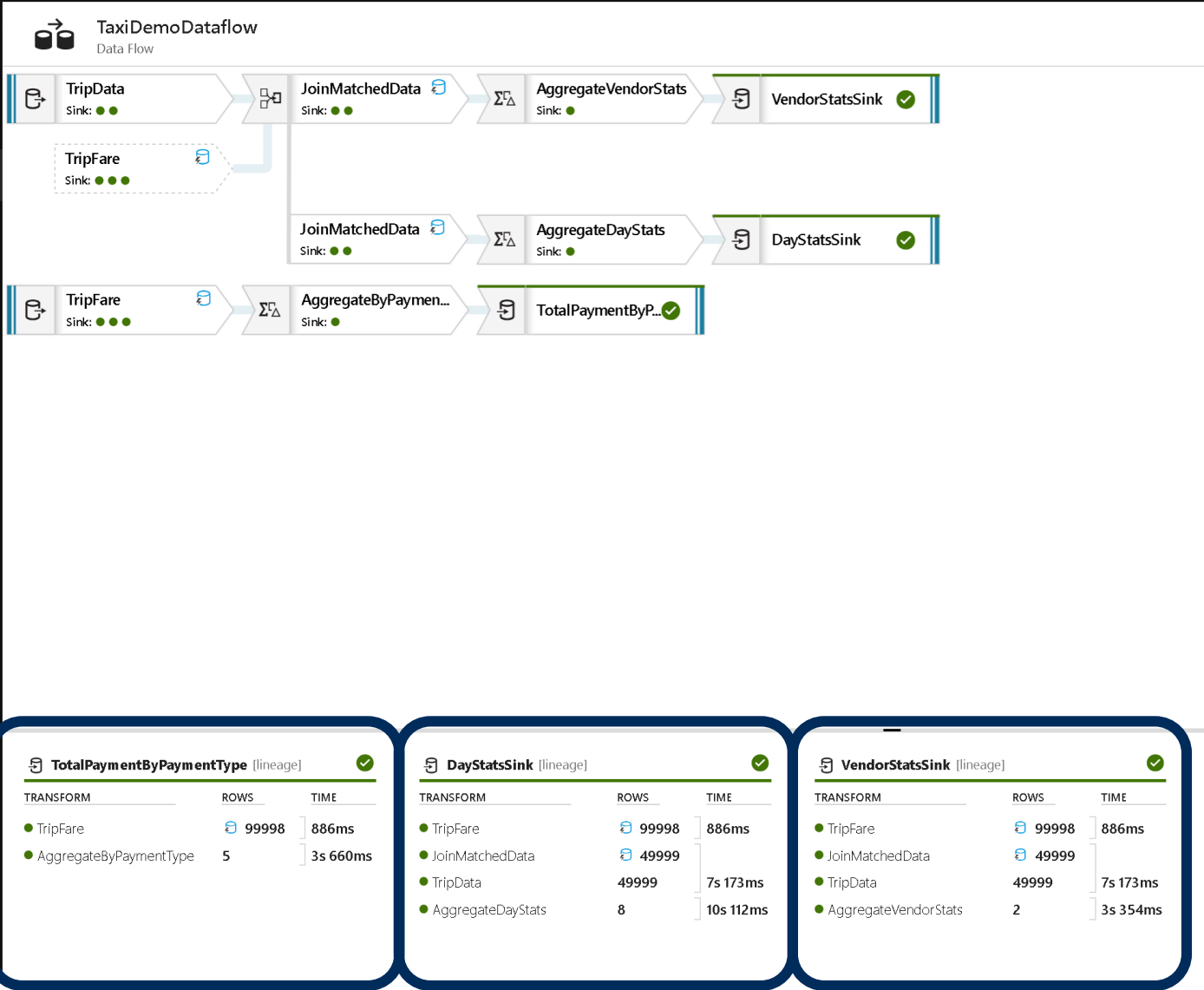
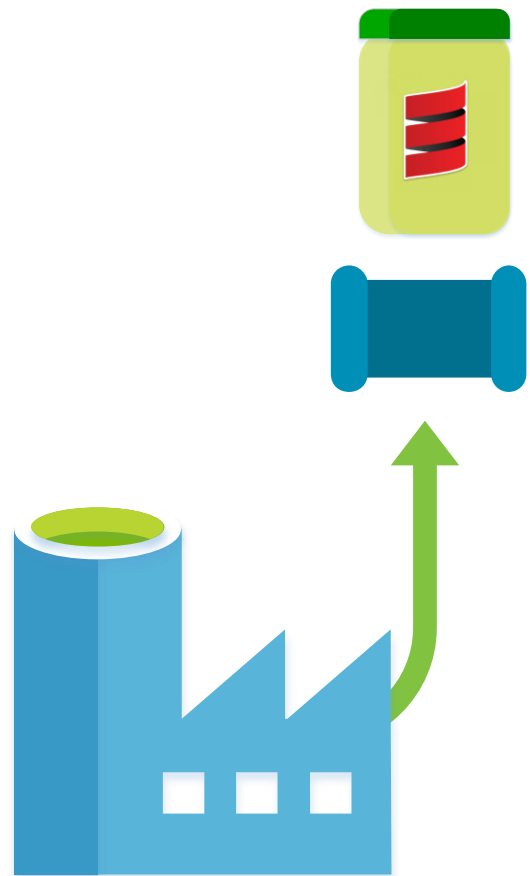
	Updated*	New*	Unchanged	Total
Number of rows	N/A	N/A	N/A	32
SalesOrderID 123	RevisionNumber abc	OrderDate 🕒	DueDate 🕒	ShipDate
71774	2	06/01/2008 00:06:00	06/13/2008 00:06:00	06/08/2008
71776	2	06/01/2008 00:06:00	06/13/2008 00:06:00	06/08/2008



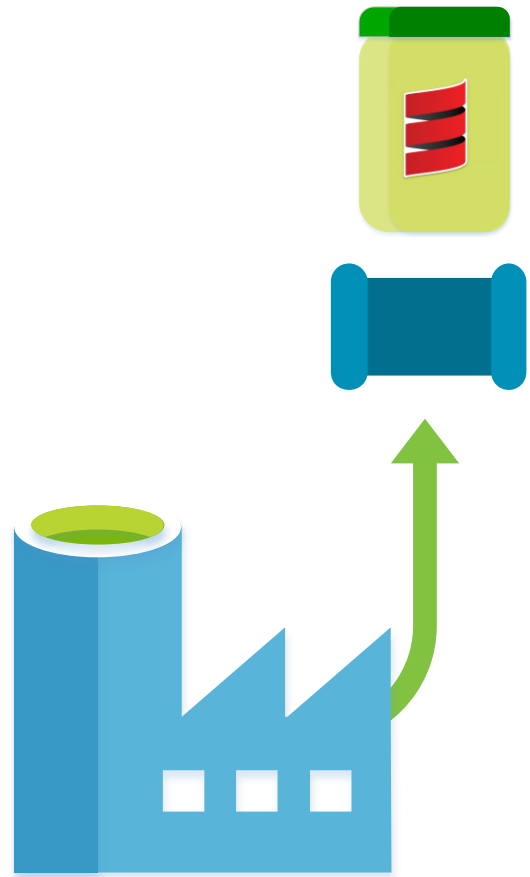
Only gives you a General Purpose cluster



# Mapping Data Flows – Monitoring



# Mapping Data Flows



1

## Activity

<https://docs.microsoft.com/en-gb/azure/data-factory/concepts-data-flow-overview>

2

## Source & Sink

<https://docs.microsoft.com/en-gb/azure/data-factory/concepts-data-flow-schema-drift>

3

## Transformations

<https://docs.microsoft.com/en-gb/azure/data-factory/data-flow-aggregate>

4

## Expression Builder

<https://docs.microsoft.com/en-gb/azure/data-factory/data-flow-expression-functions>

5

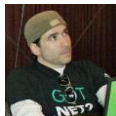
## Debug Mode

<https://docs.microsoft.com/en-gb/azure/data-factory/concepts-data-flow-debug-mode>

6

## Monitoring

<https://docs.microsoft.com/en-gb/azure/data-factory/concepts-data-flow-monitoring>







## Mark Kromer

<https://github.com/kromerm/adfdataflowdocs>

# Demo

**Start clusters!**

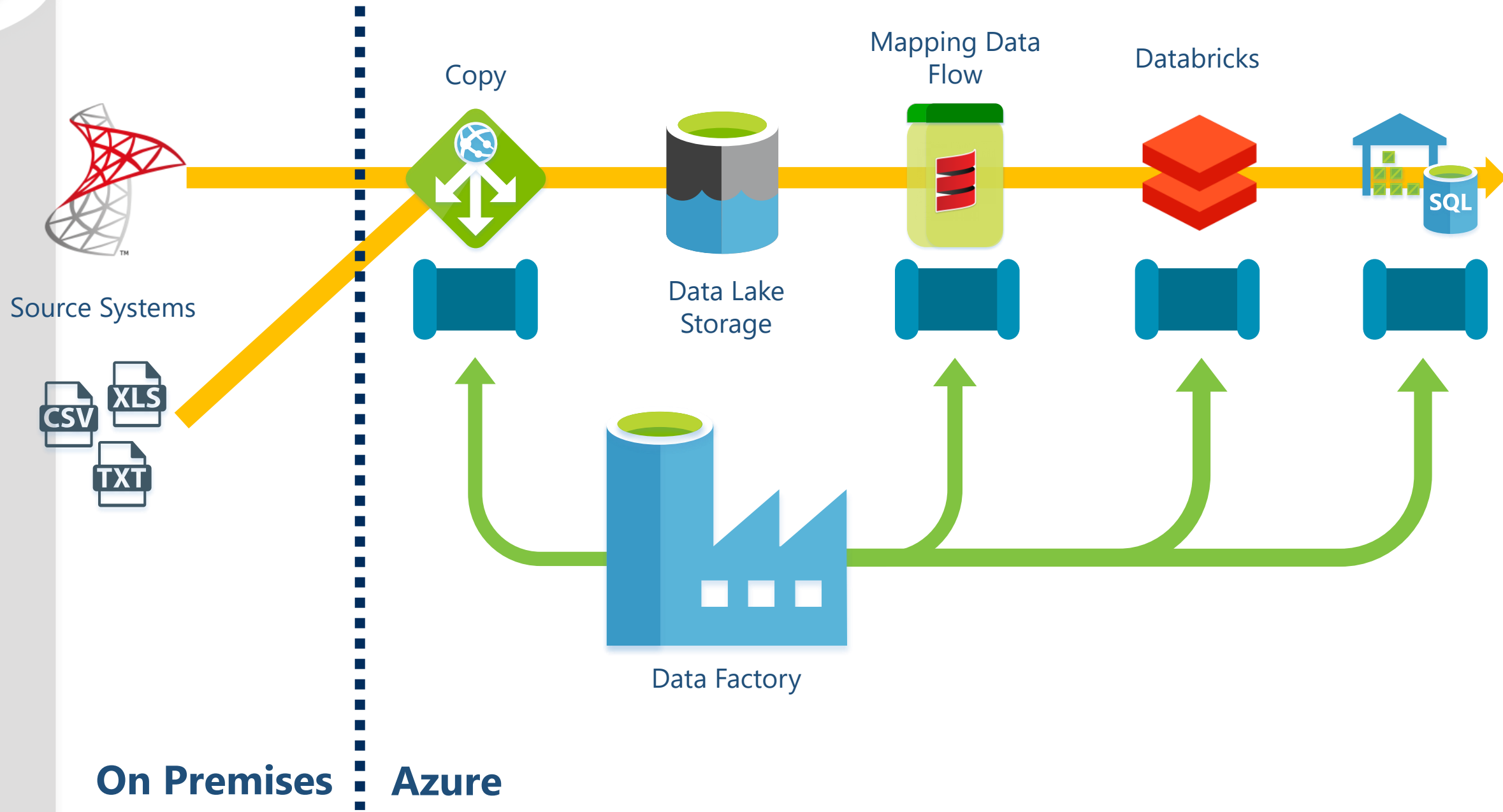
# Demo Summary

Transformation Method		Graphical UI	Scales Out	Scales Up	Cloud Native Tech
	T-SQL (SQLDB)	✗	✗	✓	✗
	SSIS	✓	✗	✓	✗
	Scala (Databricks)	✗	✓	✓	✓
	Mapping Data Flow	✓	✓	✓	✓

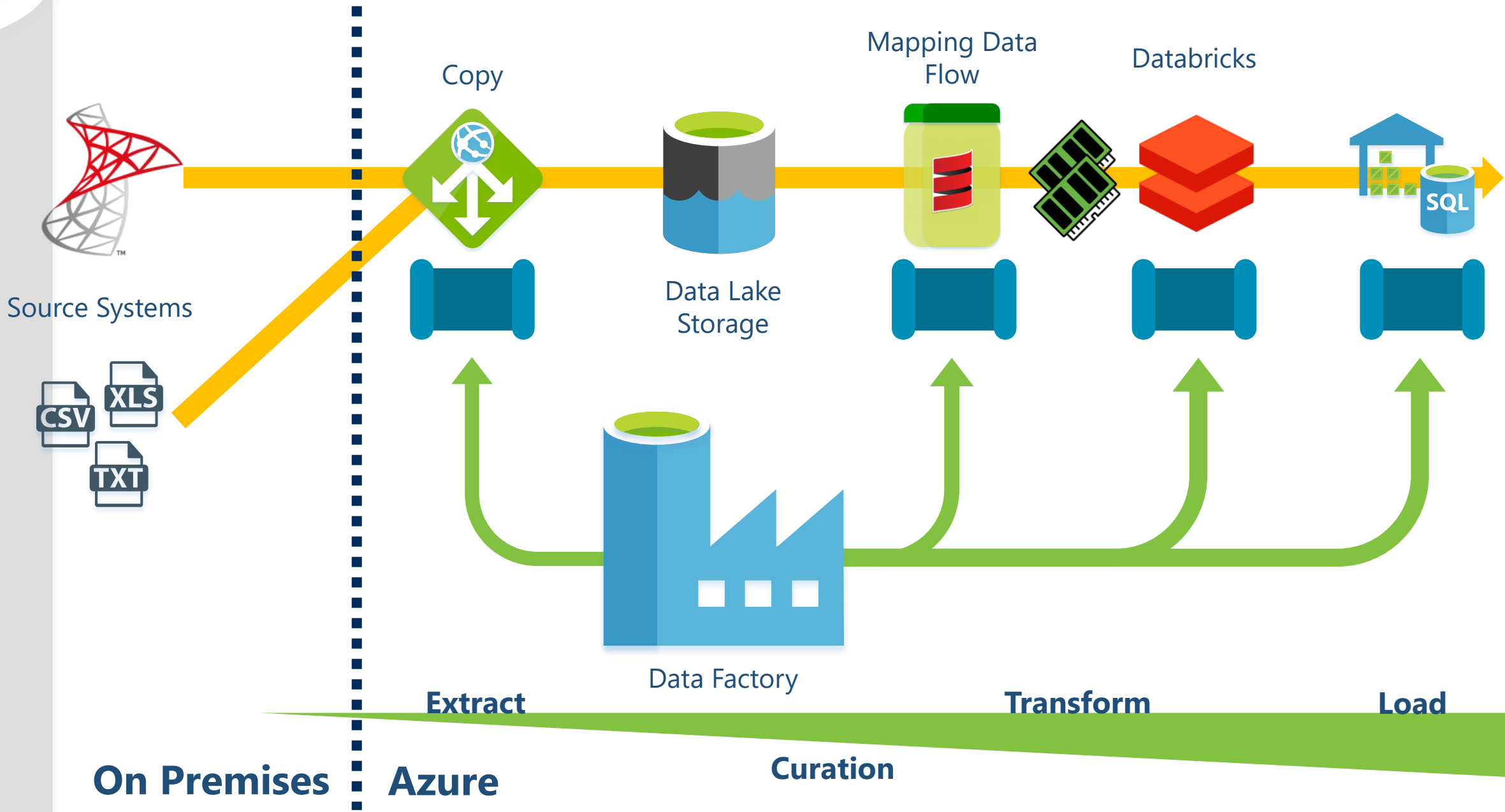
# Design Patterns

A large, solid blue triangle is positioned in the bottom right corner of the slide, pointing towards the top right.

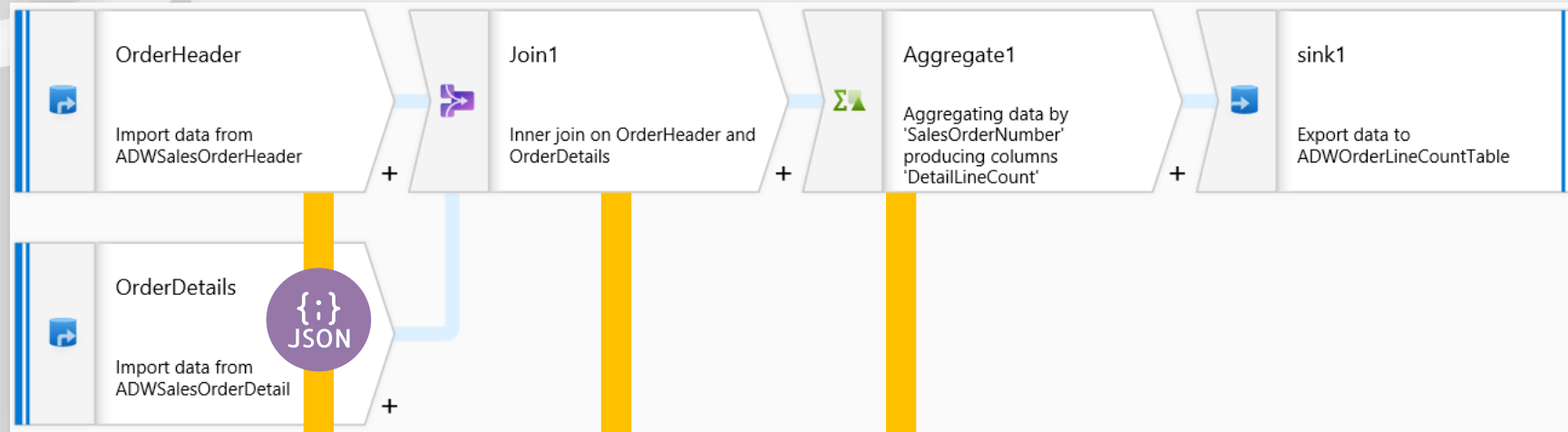
# Mapping Data Flow Future Design Patterns ???



# Mapping Data Flow Future Design Patterns ???

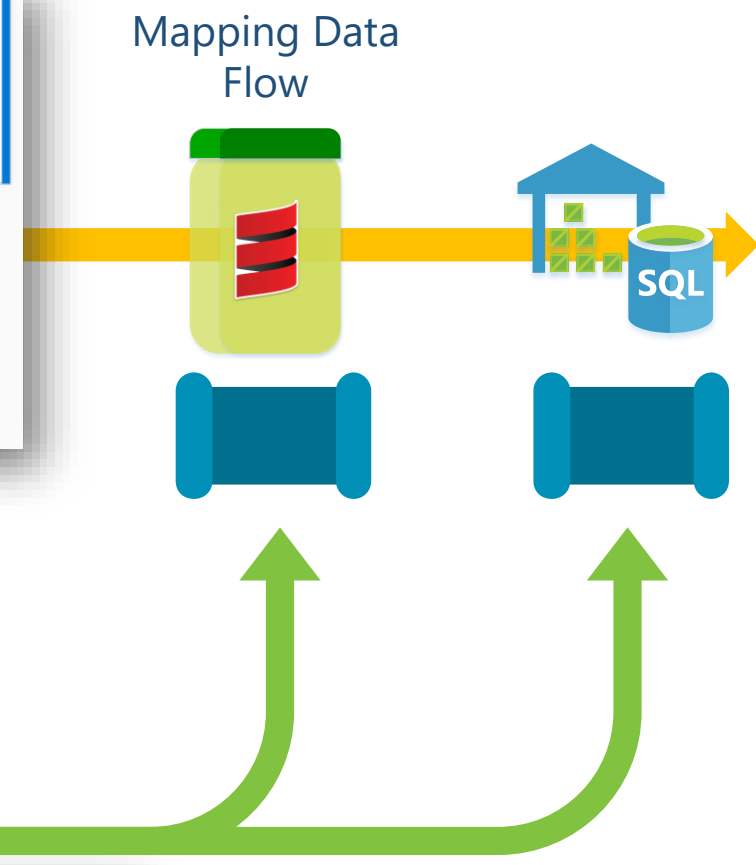


# Mapping Data Flow Future Design Patterns ???



```
"fileName": {  
  "value": "@dataset().FileName",  
  "type": "Expression"  
},  
"folderPath": {  
  "value": "@dataset().SourceDIR",  
  "type": "Expression"  
}
```

```
"transformations": [  
  {  
    "name": "Join1",  
    "script": "OrderHeader, OrderDetail join(OrderHeader@SalesOrderID == OrderDetail@SalesOrderID, \n\tjoinType:'inner', \n\tbroadcast: 'none') ~> Join1"  
  },  
  {  
    "name": "Aggregate1",  
    "script": "Join1 aggregate(groupBy(SalesOrderNumber), \n\tDetailLineCount = count(SalesOrderDetailID)) ~> Aggregate1"  
  }  
]
```

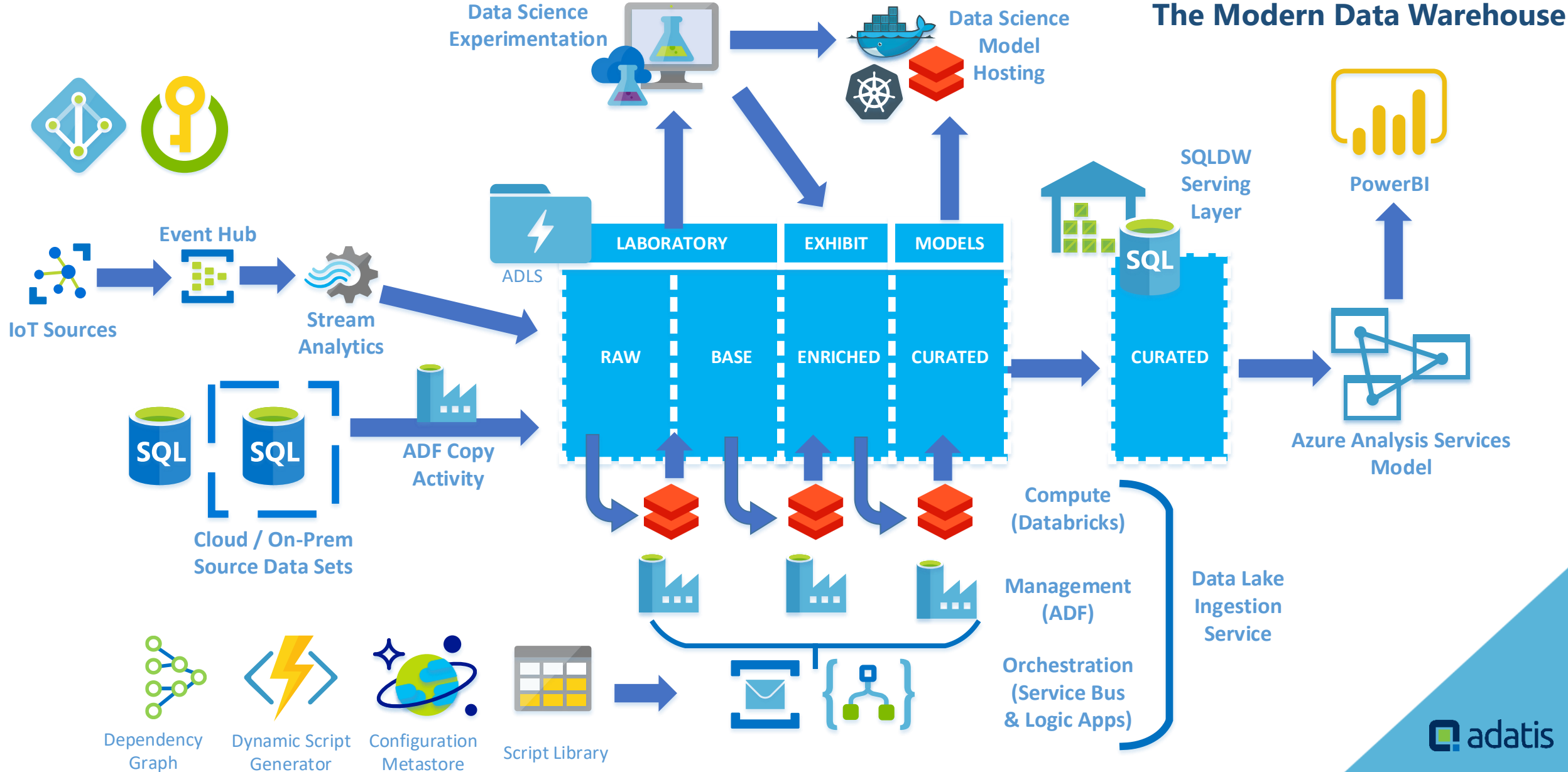




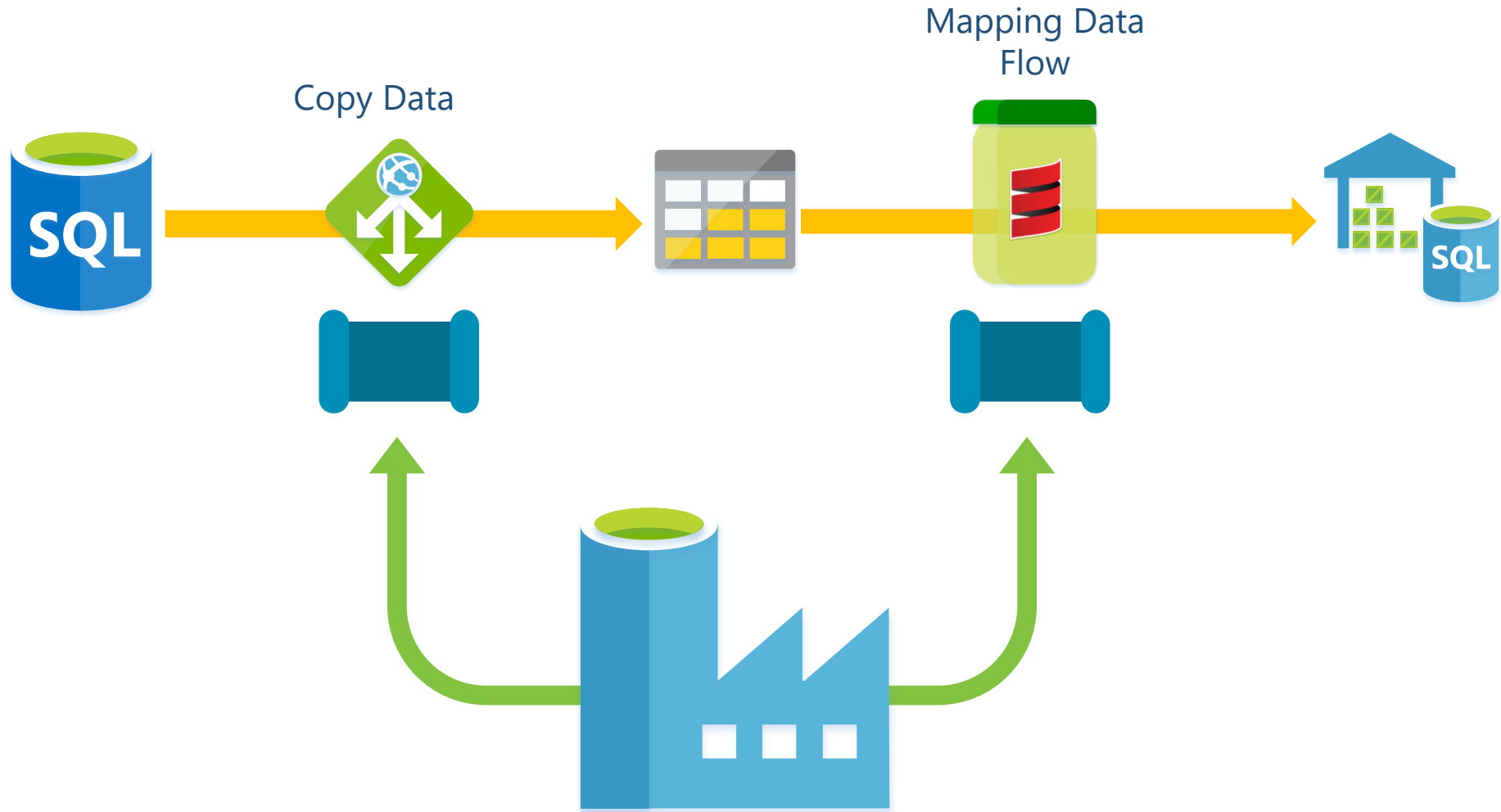
Conclusions

A large, solid blue triangle is positioned in the bottom right corner of the slide, pointing towards the top right.

# Why use Azure Data Factory?



# What is Azure Data Factory?



Orchestrator of our solution Control Flow operations.

Orchestrator of our solution Data Flow transformations.

... using cloud native technology in  Azure and now with a user interface for both.

# Thanks for Listening

## Paul Andrew

 @MrPaulAndrew



**Email:** [paul@mrpaulandrew.com](mailto:paul@mrpaulandrew.com)

**Blog:** [mrpaulandrew.com](http://mrpaulandrew.com)

**GitHub:** [github.com/mrpaulandrew](https://github.com/mrpaulandrew)

