Microsoft

Azure Synapse Analytics

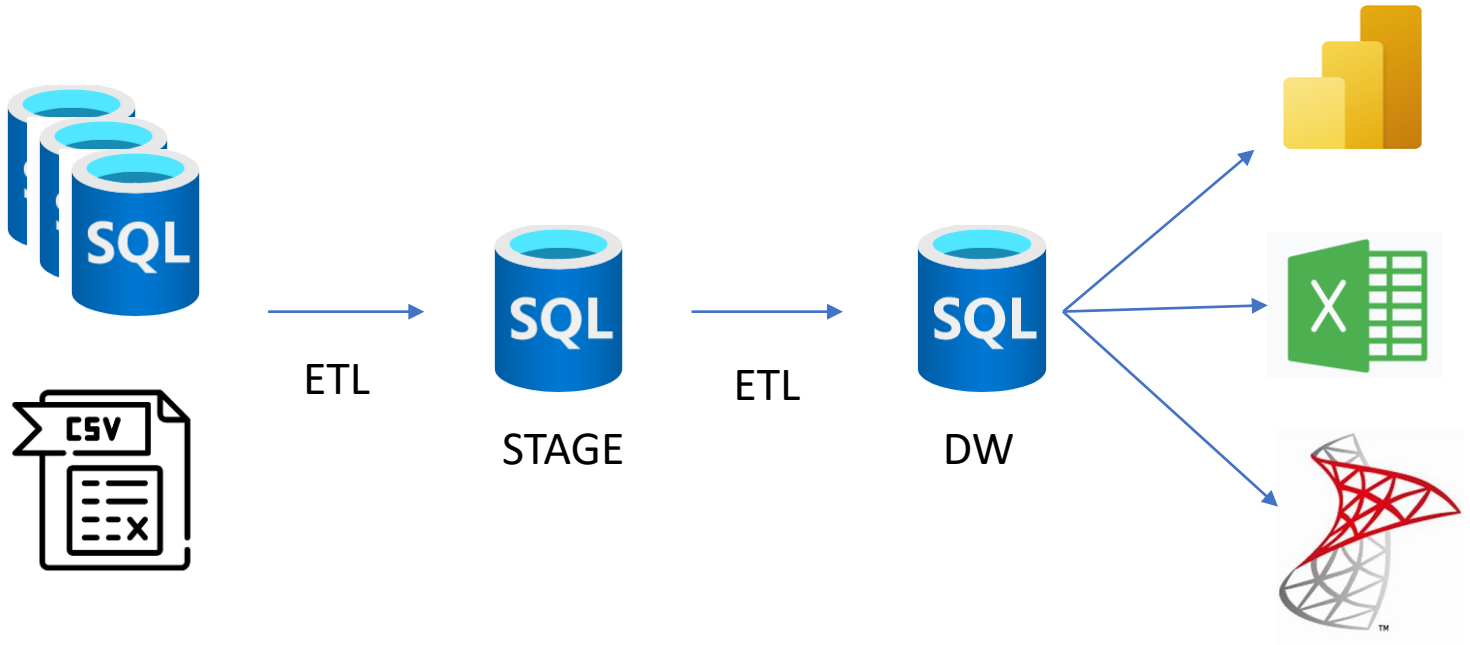
What's a Lakehouse?

Luke Newport

Technical Architect – Data & AI

Microsoft Technology Center, St. Louis

Legacy On Premises Data Warehouse

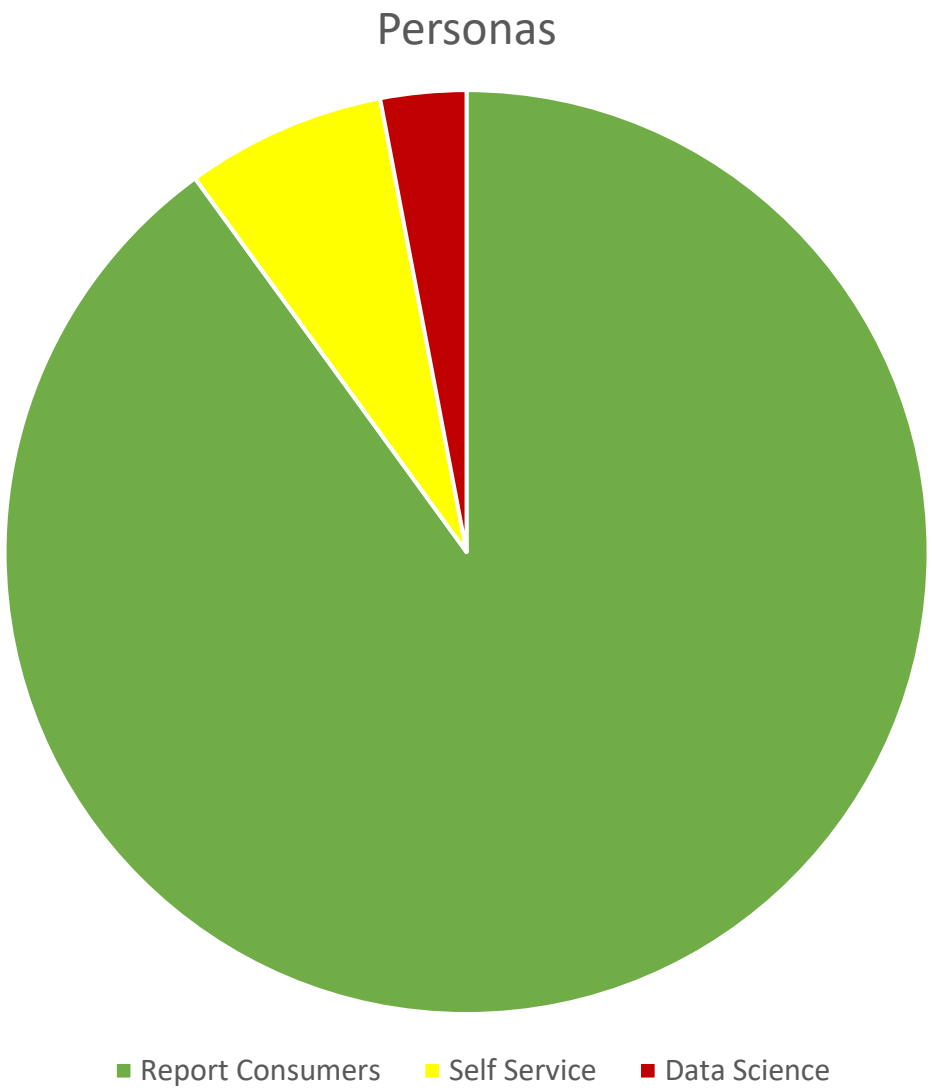


Challenges:

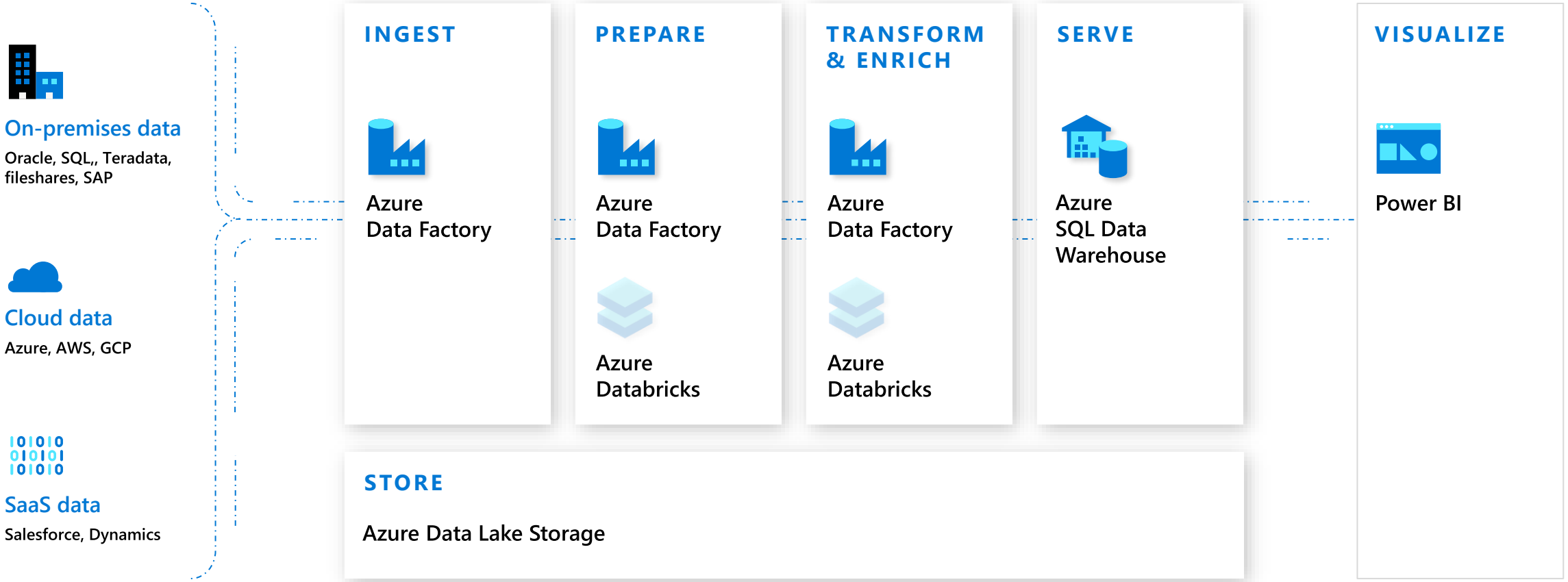
- Scalable – to a point.
- Relational, ideal for:
  - Tabular
  - Columnar
- Not ideal for:
  - JSON
  - Clickstream
  - Unstructured
- Change is time consuming, because of schema on write.
- Difficult to explore (Data Science) and fold back in.

Data Consumers

- Inverse Relationship – Size of Data vs Count of Consumers
- Report Consumers : Published Reports
- Self – Service : Reports + Mash up with DW+ODS
- Data Science : We want it all.
- Traditional DW struggles to serve all three due to volume and velocity.

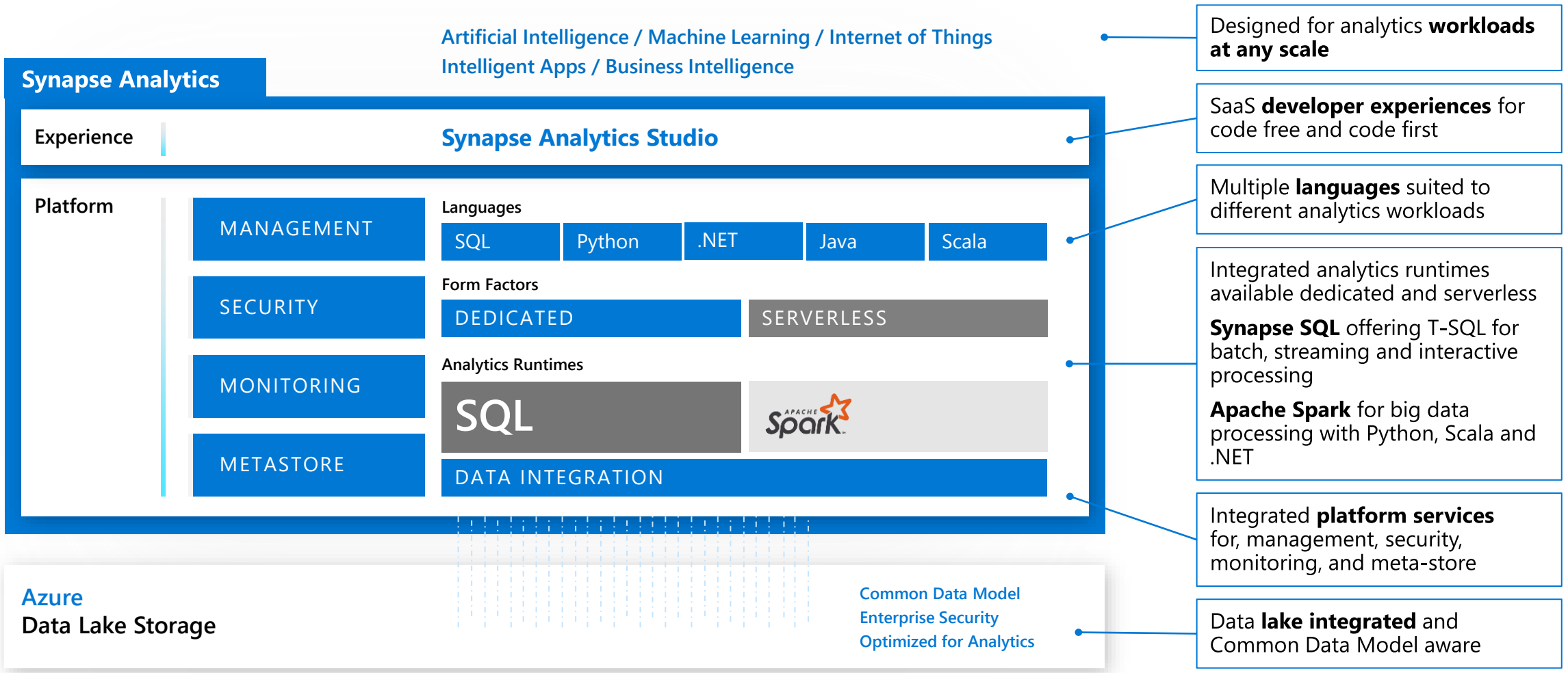


Modern Data Warehouse (old architecture)

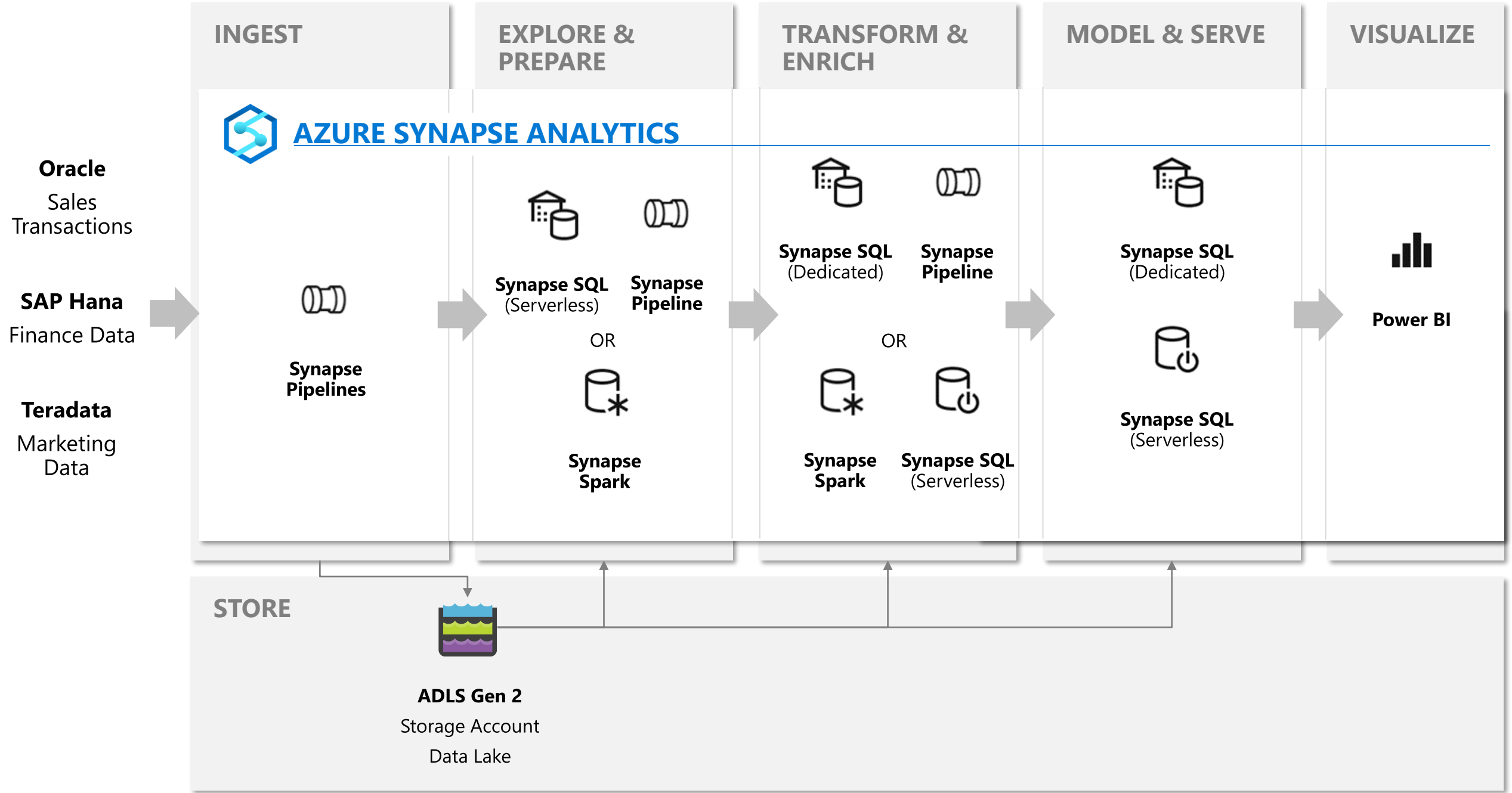


Azure Synapse Analytics – *Data Lakehouse*

Azure Synapse Analytics



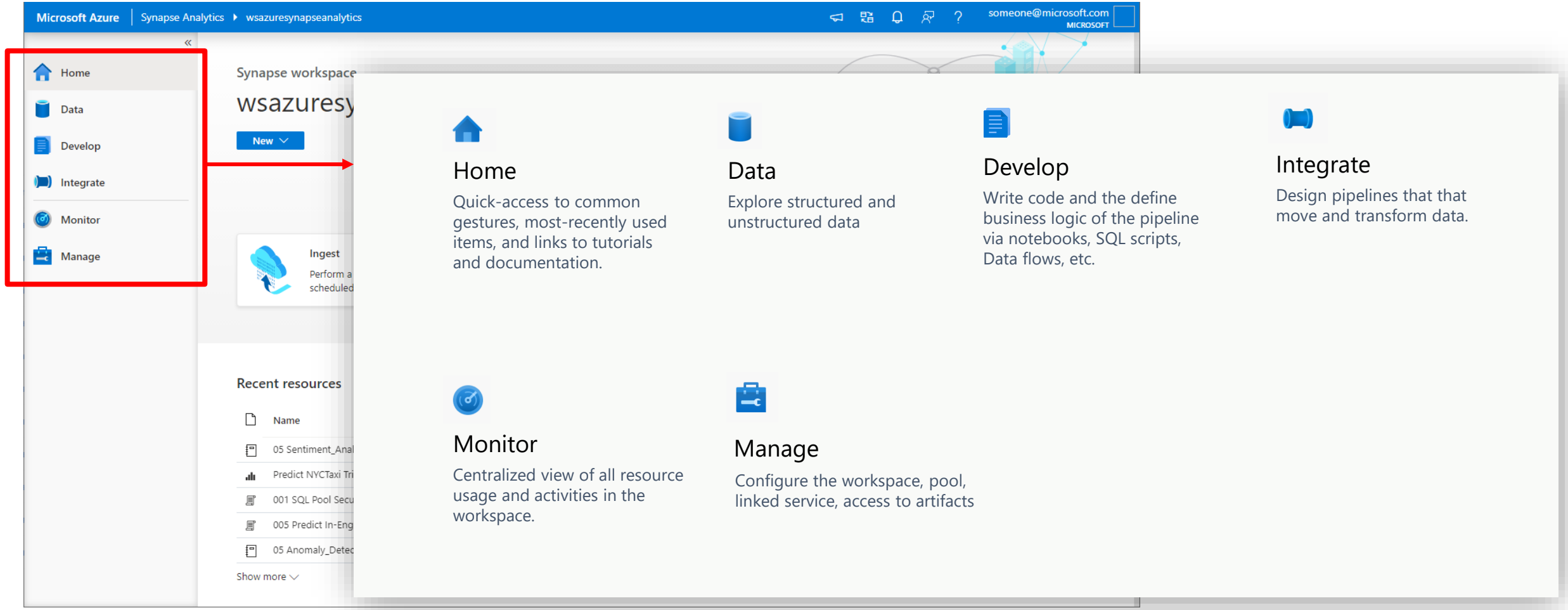
Modern Data Warehouse (new architecture)



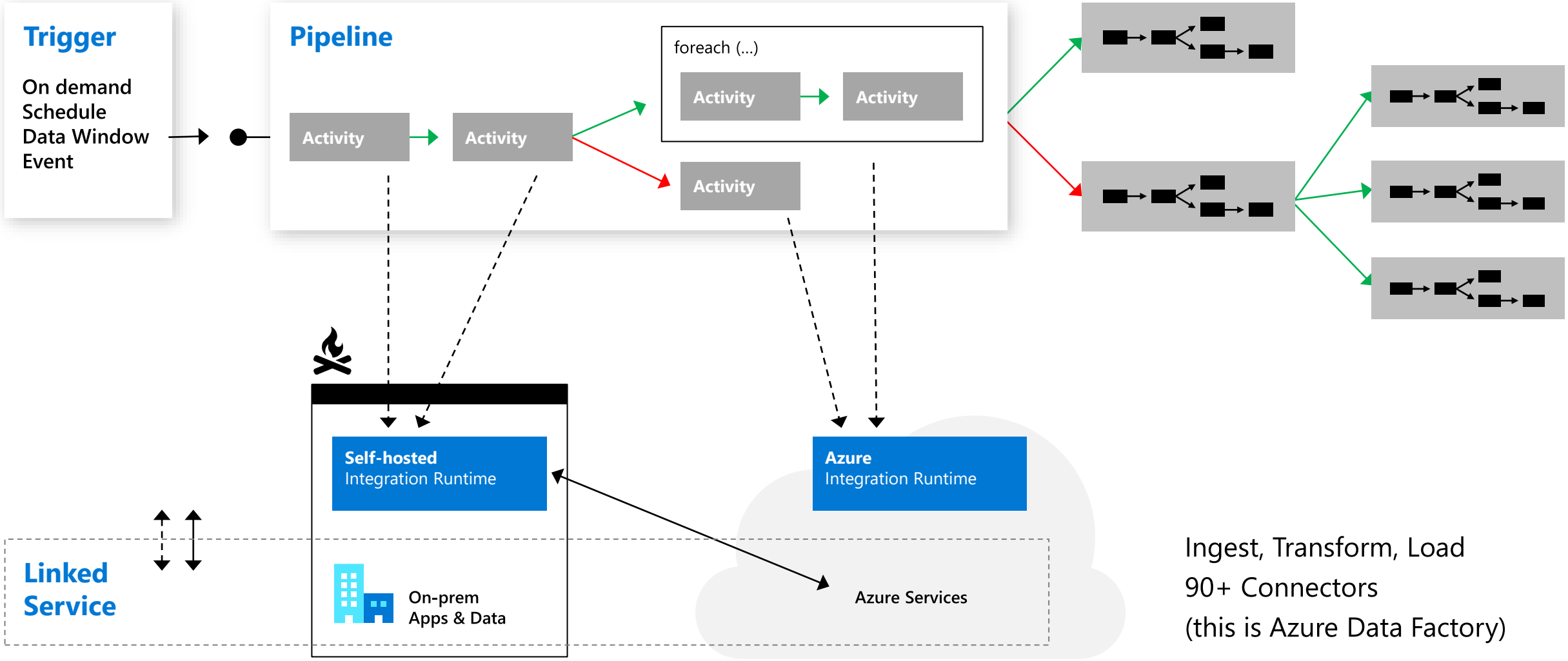
Synapse Studio

Synapse Studio divided into **Activity hubs**.

These organize the tasks needed for building analytics solutions.



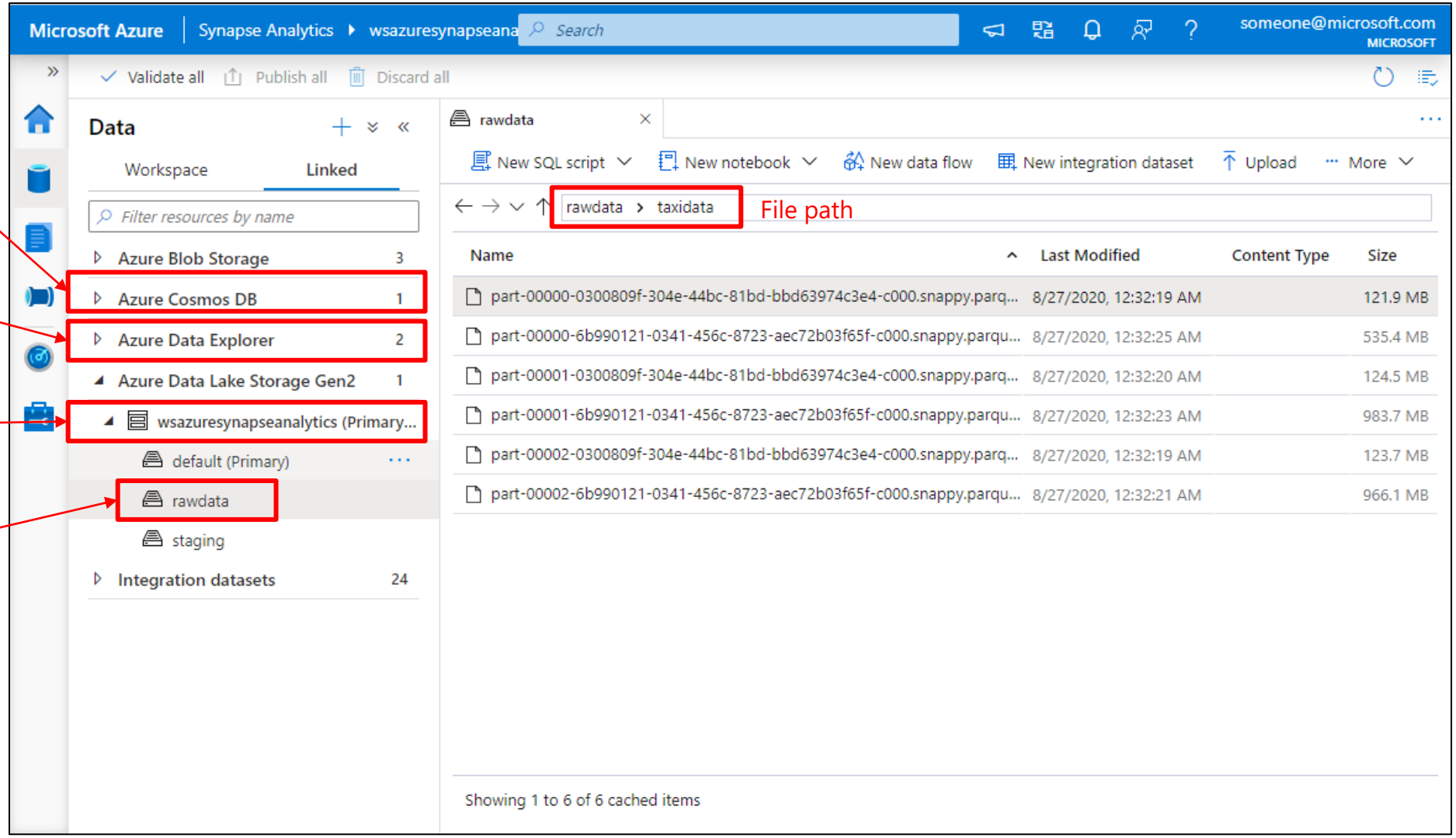
Synapse Integrate



Ingest, Transform, Load
90+ Connectors
(this is Azure Data Factory)

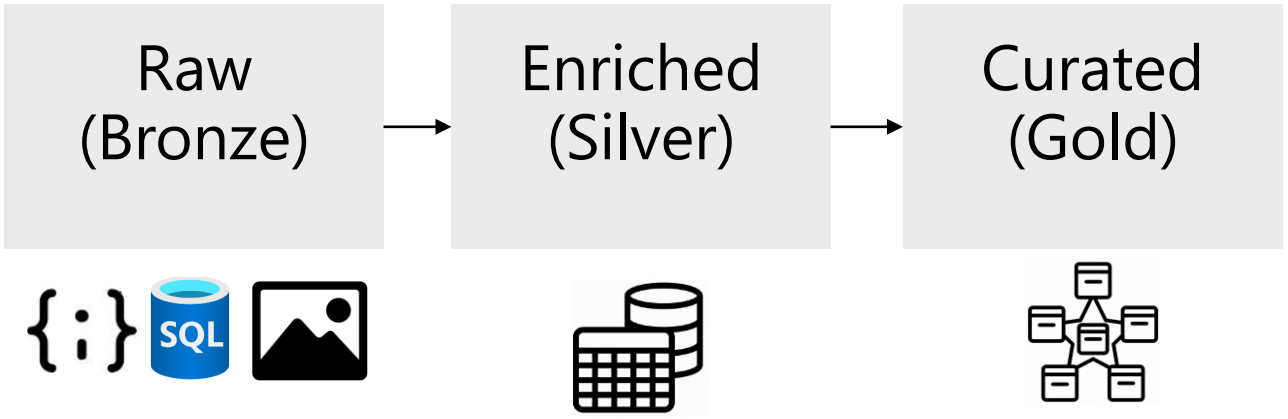
Synapse Data Hub – Linked Storage

Browse Azure Data Lake Storage Gen2 accounts – filesystems, Azure Data Explorer – clusters, Azure Cosmos DB -containers



Synapse – Azure Data Lake Gen2

- ADLSg2 – polyglot storage
- Posix ACLs
- HDFS endpoint on blob
- AAD Integration
- /…/…/…/*.parquet (delta_log)
- Storage tiering by policy (



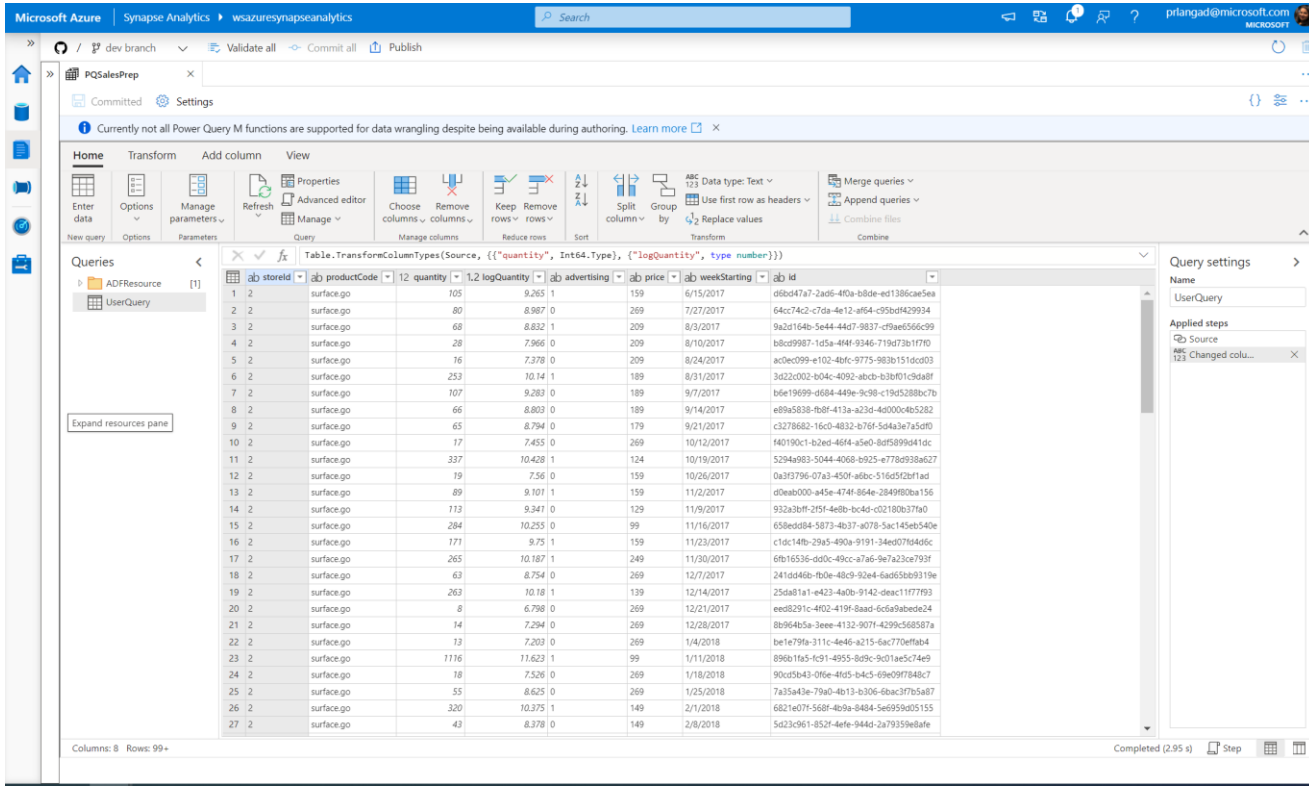
Synapse - Prep & Transform Data – Low Code

Data Flows

Code free data transformation @scale

Power Query

Code free data preparation @scale



Azure Synapse x Apache Spark – Code First

- **Apache Spark 3.2 derivation**
  - Linux Foundation Delta Lake 1.1 support
  - .Net Core 3.0 support
  - Python 3.8 + Anacondas support
- **Tightly coupled to other Azure Synapse services**
  - Integrated security and sign on
  - Integrated Metadata
  - Integrated and simplified provisioning
  - Integrated UX including nteract based notebooks
  - Fast load of Synapse SQL (provisioned) pools
- **Core scenarios**
  - Data Prep/Data Engineering/ETL
  - Machine Learning via Spark ML and Azure ML integration
  - Extensible through library management
- **Efficient resource utilization**
  - Fast Start
  - Auto scale (up and down)
  - Auto pause
  - Min cluster size of 3 nodes
- **Multi Language Support**
  - .Net (C#), PySpark, Scala, Spark SQL, Java

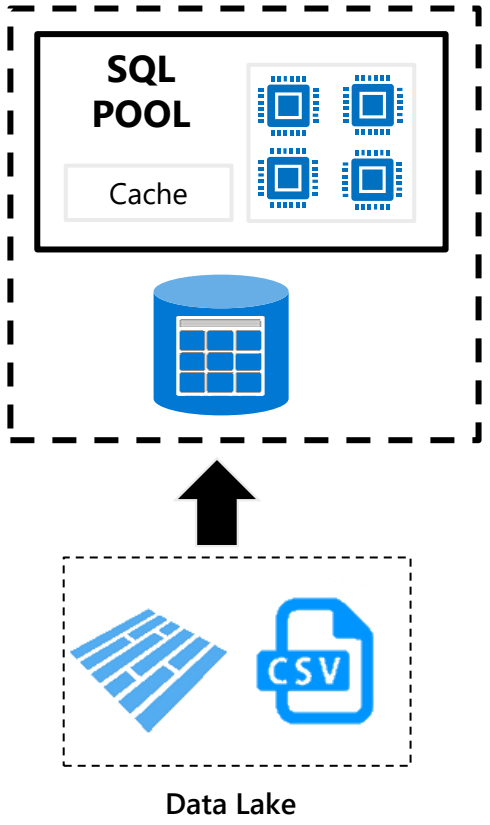
Synapse Dedicated SQL Pools

Advanced storage system

- MPP – evolution of Azure SQL DW
- ColumnStore Indexes
- Table partitions
- Distributed tables
- Isolation modes
- Materialized Views
- Nonclustered Indexes
- Result-set caching

Complete SQL object model

- Tables
- Views
- Stored procedures
- Functions
- **It's SQL Server at scale**



Serverless SQL Pools

Overview

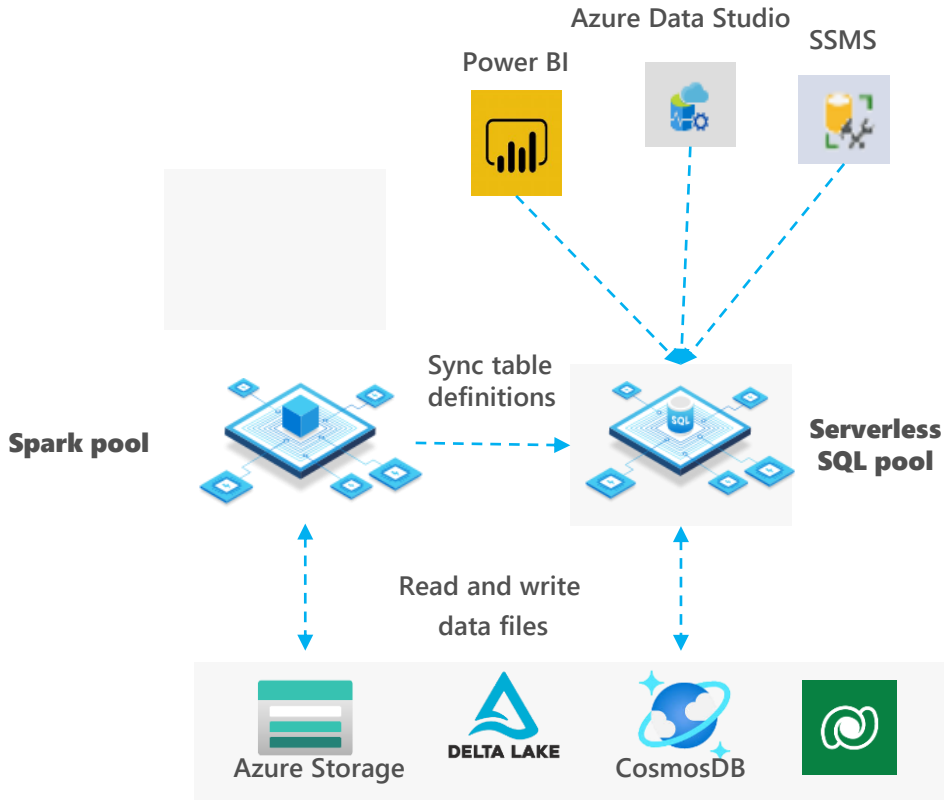
- An interactive query service that enables you to use standard T-SQL queries over Data Lake, CosmosDB, Dataverse.

Benefits

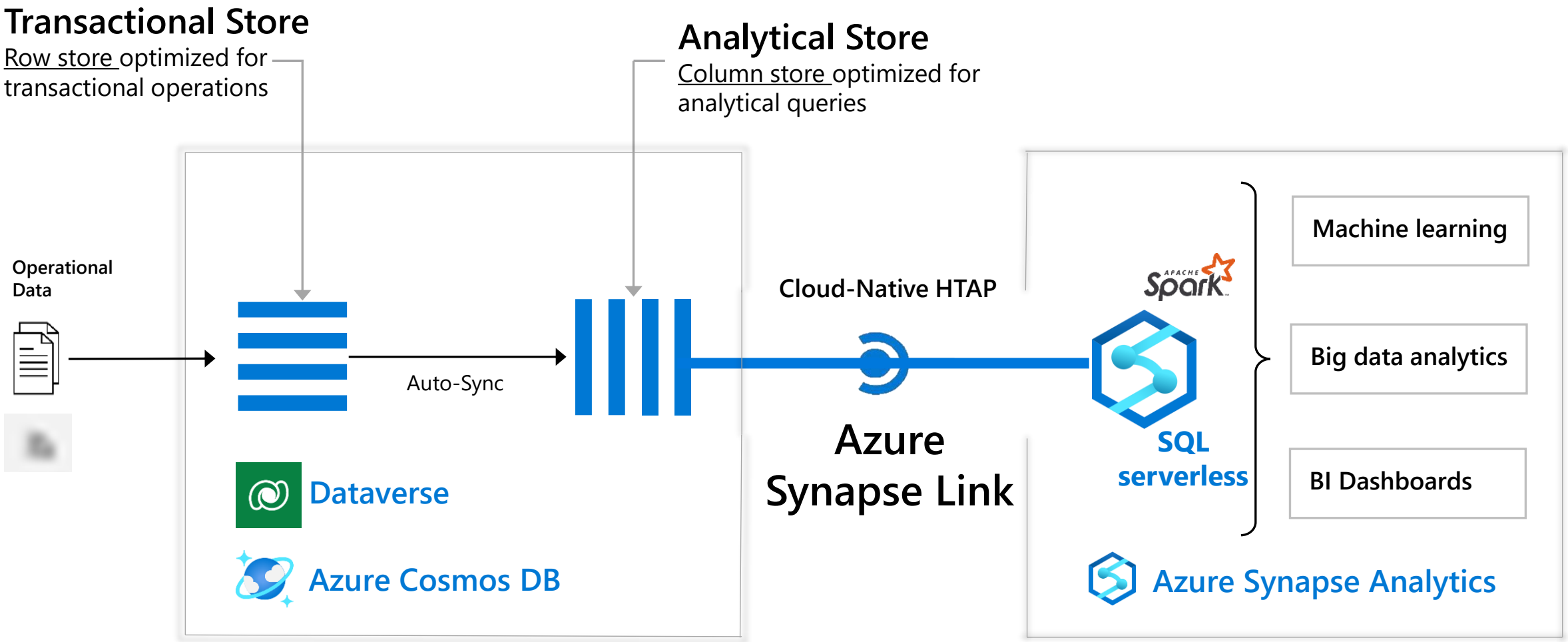
- Use T-SQL language
- Supports any tool or library that uses T-SQL to query data
- Automatically synchronize tables from Spark pool
- Queries multiple storages (Lake, CosmosDB)

Serverless experience

- Auto Scale & Manage
- Pay-per-use model
- Fast (roughly DWU 2000 per workspace)
- Automatic schema inference



Synapse Link – near real-time analytics



Demos!

A walk through Synapse ingestion, exploration, transformation, serving and more!

Repo: github.com/lnewport (pdf of deck will be here)

The Big Picture