Transforming Data into Gold: A Live Demonstration of Microsoft Fabric Lakehouse

Fabric February, Oslo, 6/2-2025

twoday

Thank you to our Fabric February Friends!

twoday

bouvet

sopra steria

DATAmasterminds

ADVANCING ANALYTICS

Evidi

Profisee
Master Data Management

Tabular Editor

KURANT

Fraktal

CluedIn

Dufrain
THE DATA COMPANY

Agenda

Transforming Data into Gold

Fabric is Fantastic – but not as easy as Microsoft marketing says

Starting-point Fabric Data Lakehouse Architecture

Kickstart the Fabric Data Lakehouse Setup

AquaShack Data Lakehouse Accelerator

Layers in the Data Lakehouse

twoday

Just Blindbæk

- Principal Architect at twoday
  - Pre-sales and marketing
  - Internal practice development
  - Academy: External training
- Microsoft Data Platform MVP
- Found and organizer of
  - Danish Microsoft BI Community (MsBIP.dk)
  - Power BI UG Denmark (PowerBI.dk)
  - Power BI Next Step and Data Platform Next Step

linkedin.com/in/blindbaek/

twoday

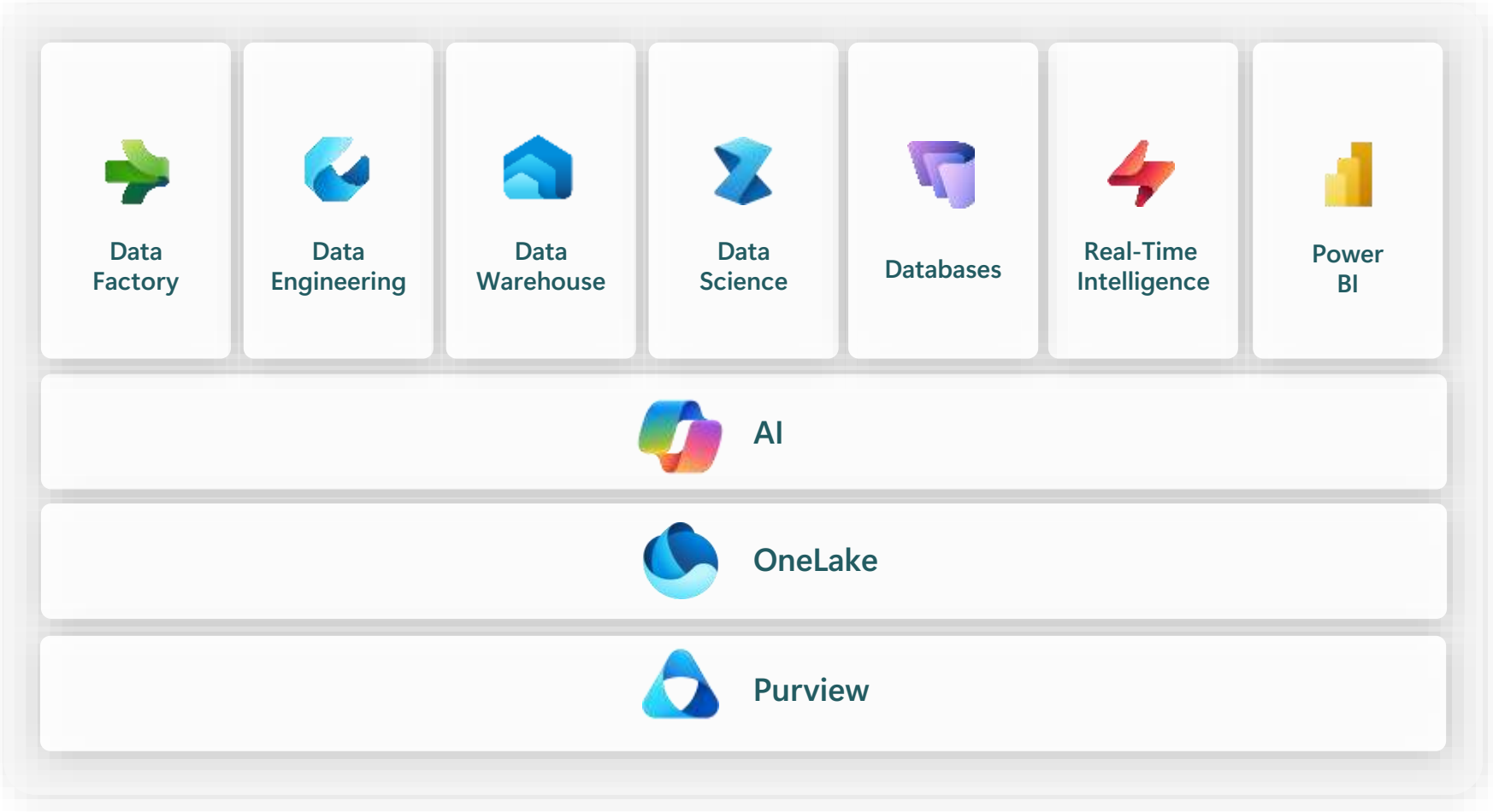
(Christian) Henrik Reich

- Principal Architect @ twoday Data & AI (Technology And Architecture)
- Redi-School Teacher. Teaching Microsoft AI Technologies to immigrant women. Starting 5th semester.
- Ex-Pluralsight assessment tests author
- Øredev program committee member since 2019 (Data track) (Malmø)
- Agile Global Summit 2025 program committee member (Tallinn)
- Ex-Pass Summit program committee member 2023 (Seattle)
- Holding 25 active microsoft certificates
- Been a developer in many niches

linkedin.com/in/christian-henrik-reich-6346492a
medium.com/@christianhenrikreich

twoday

Fabric: End-to-end unified analytical platform



Software as a Service

Seven workloads

- Low-code/no-code
- Pro developer / engineers

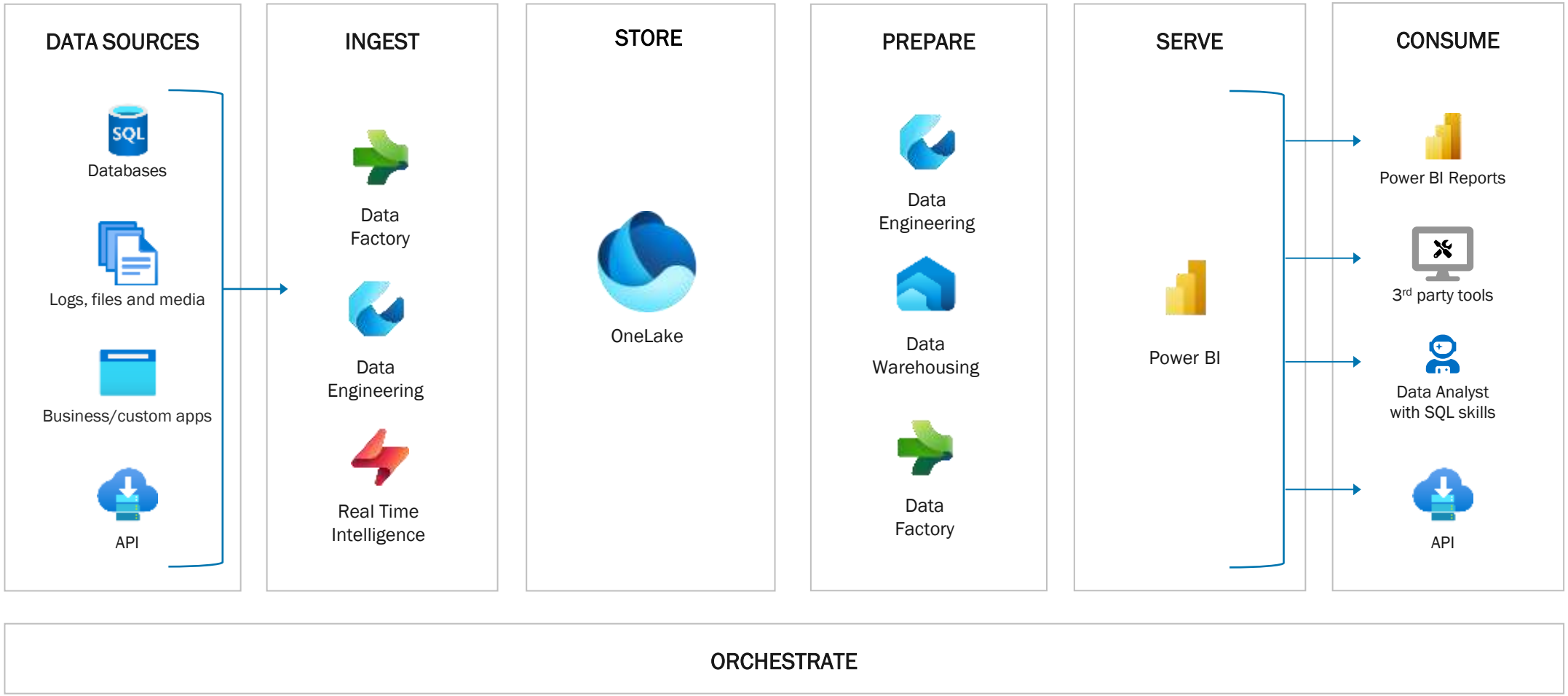
- 30+ items

OneLake & Delta format

- Separation of storage and compute

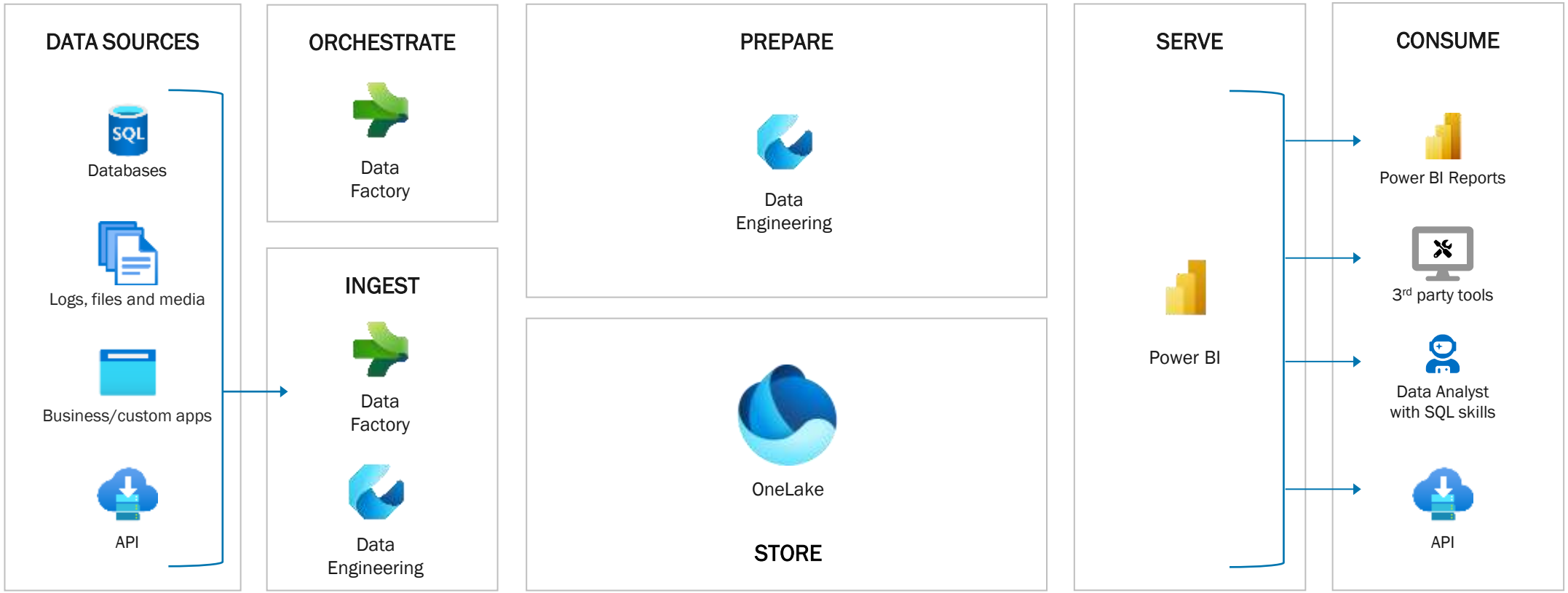
twoday

Architecture view of Fabric workloads



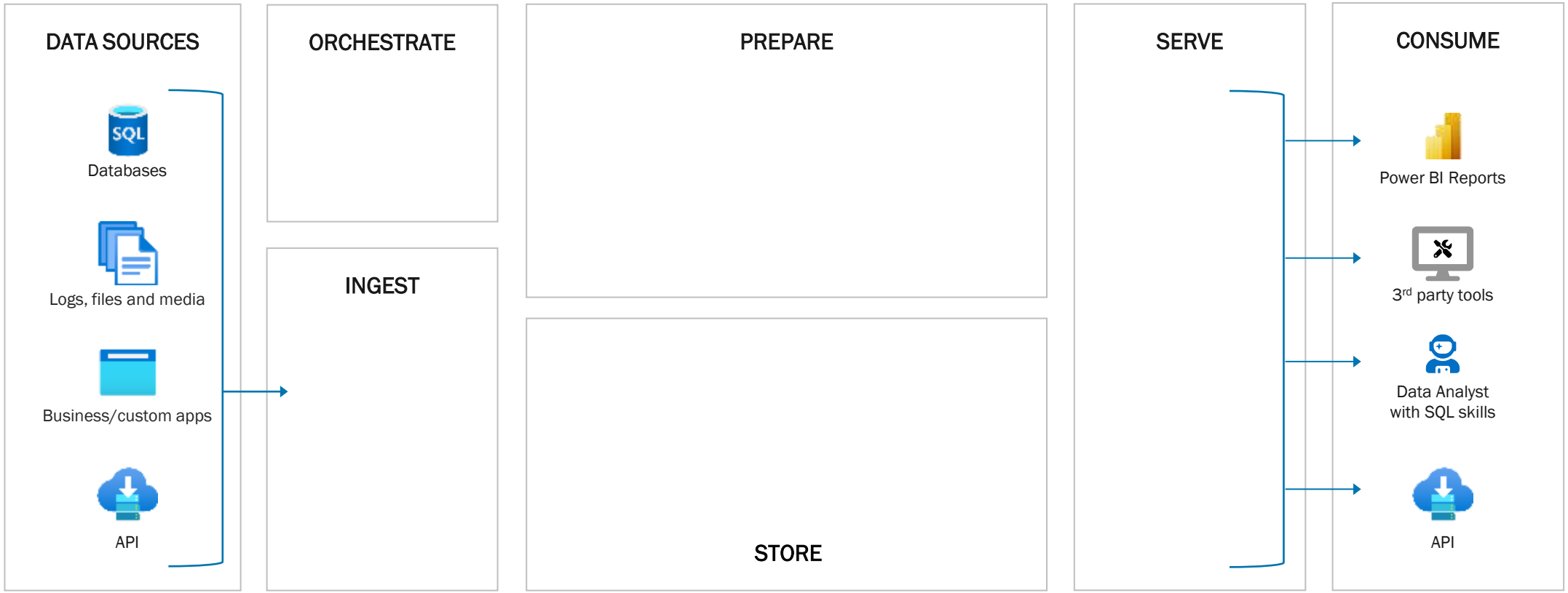
twoday

Lakehouse view of Fabric workloads



twoday

Dataflow in a Data Lakehouse in Fabric

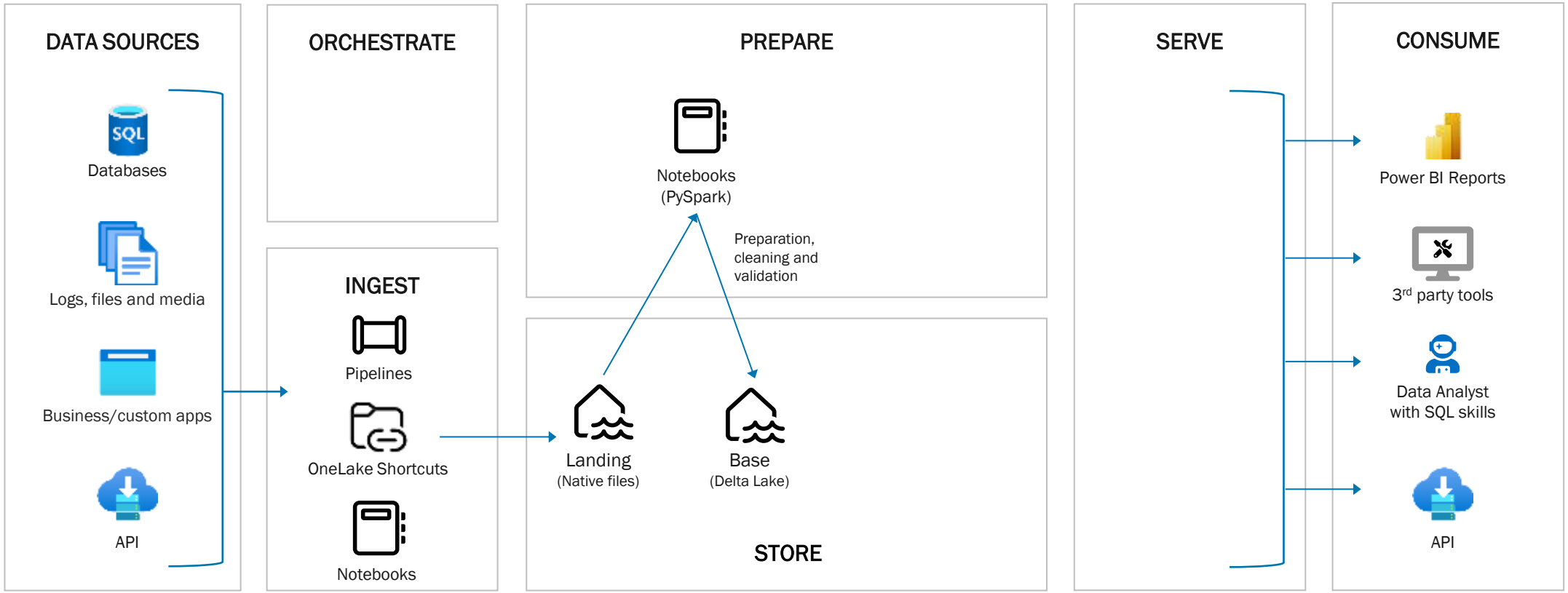


twoday

Dataflow in a Data Lakehouse in Fabric

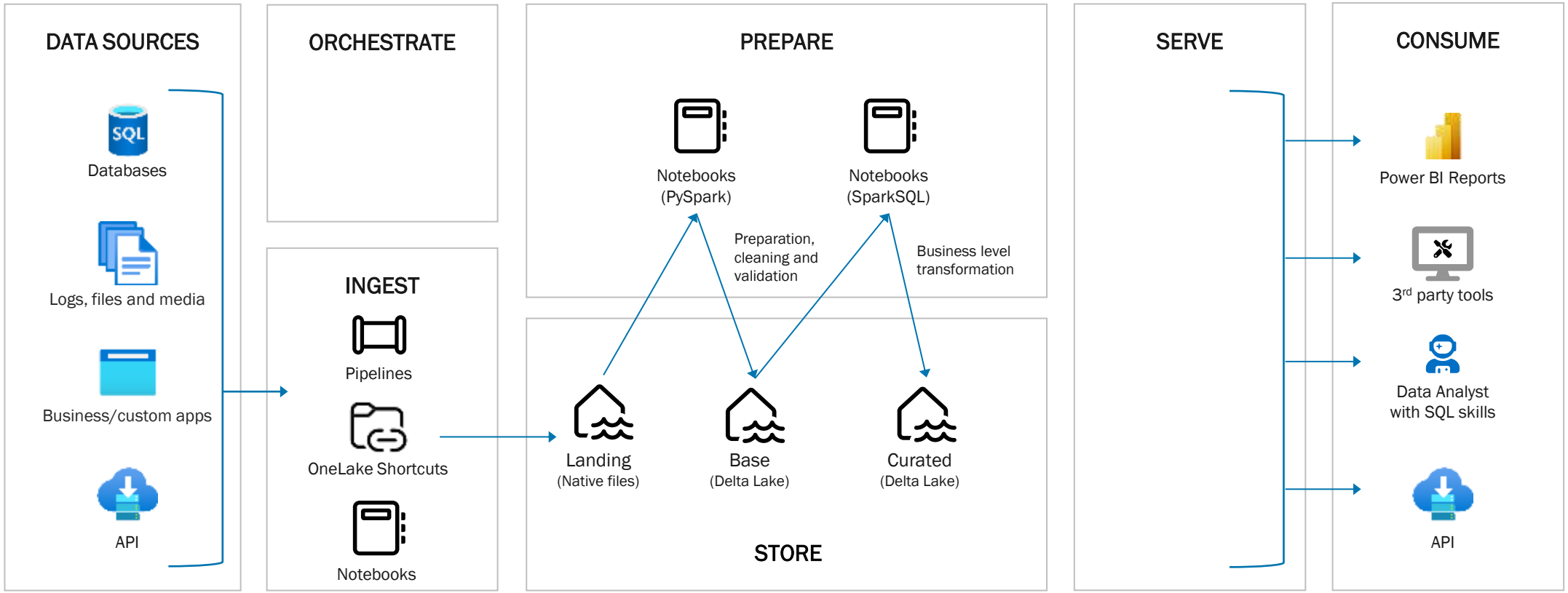
twoday

Dataflow in a Data Lakehouse in Fabric



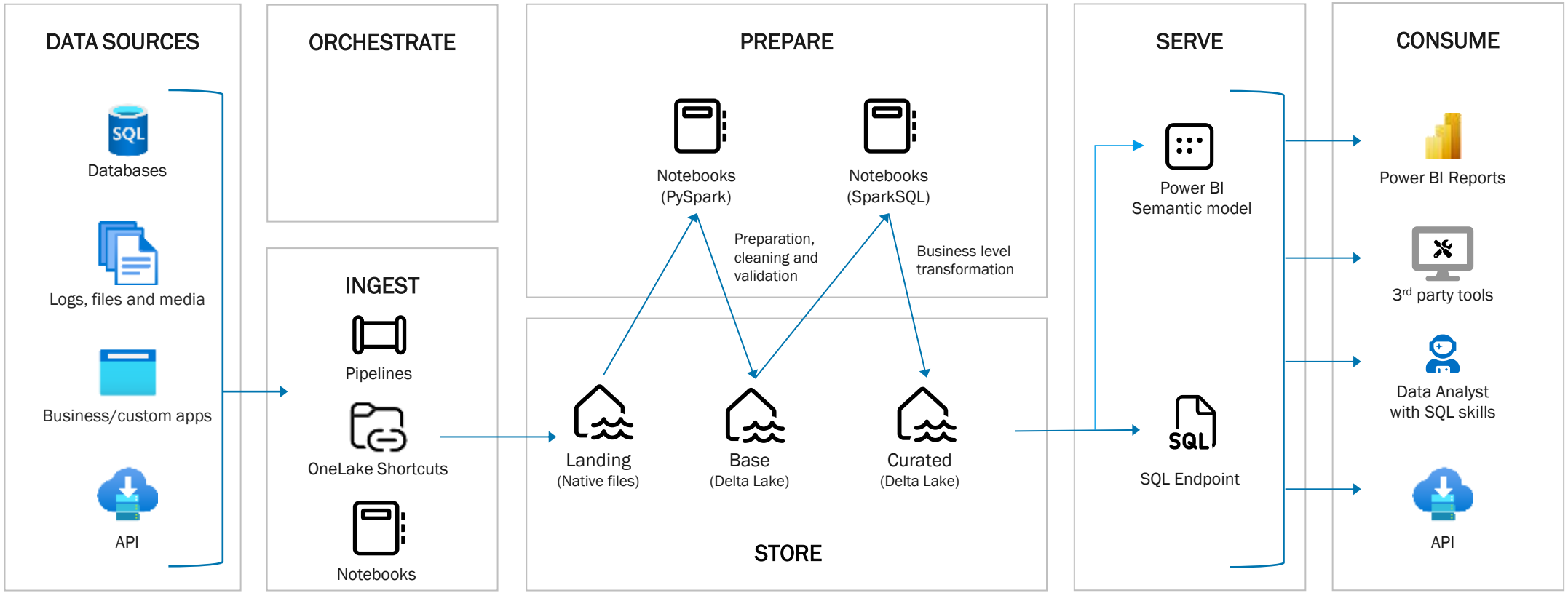
twoday

Dataflow in a Data Lakehouse in Fabric



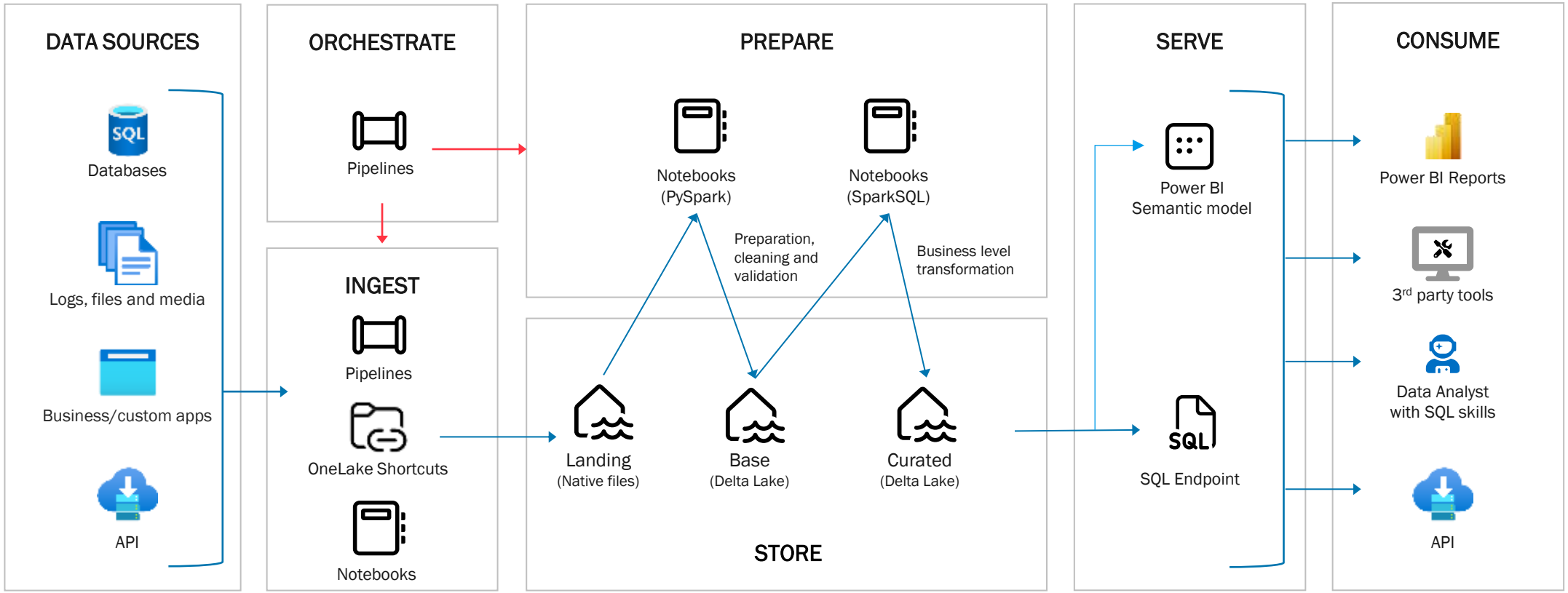
twoday

Dataflow in a Data Lakehouse in Fabric



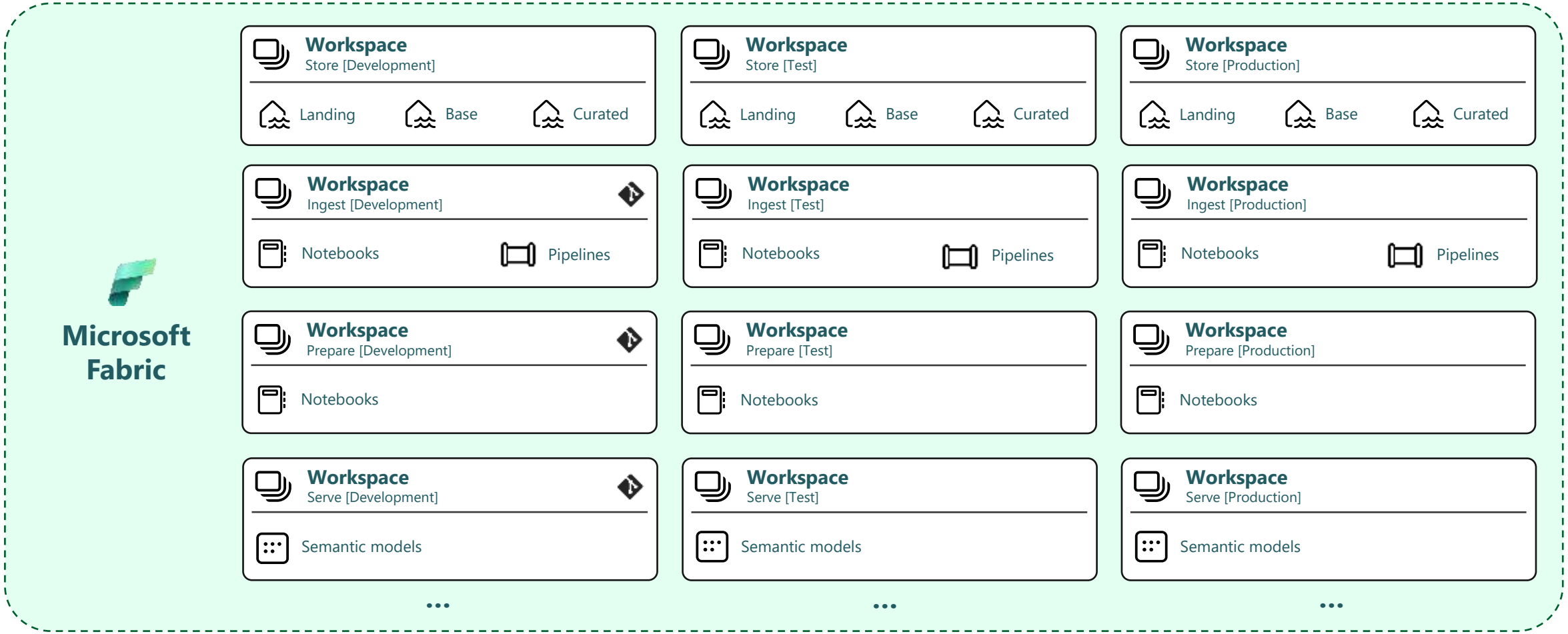
twoday

Dataflow in a Data Lakehouse in Fabric



twoday

Platform Infrastructure with Fabric workspaces



twoday

Setup the Fabric Lakehouse structure in 1 minute!

Initialization script & recipe

Using Service Principal (SPN)
Create workspaces
Assign workspace permissions
Assign capacities to workspaces
Create Fabric items (lakehouses)
Create managed private endpoints

Using User principal (UPN)*
Connects workspaces to git
Initializes Git connection
Updates workspaces from Git

Fabric REST APIs

Core APIs

Workspaces
Items
Long Running Operations
Managed private endpoints
Git*

Visit Peer's blog post "Automating Fabric: Kickstart your Fabric Setup with Python and Fabric REST APIs" on https://peerinsights.hashnode.dev

twoday

Demo time

twoday

AquaShack: Data Lakehouse accelerator

- Pico-example of a meta-data driven lakehouse for Microsoft. Little brother to AquaVilla
- Data Lakehouse with three layers, where data is moved between.
  - Landing: Holds the data in original format if possible. Relation sources are stored in parquet.
  - Base: Aligned clean data, all stored in Delta tables.
  - Curated: Data for serving, business logic is applied, star schemas are defined etc.
- The notebooks builds upon functions defined in **AquaShack_functions**.
- We are reading metadata from JSON, it could have been from a database or YAML file as well. The important part is it ends in a common structure.

twoday

Layers in the Data Lakehouse

LANDING

- One to one with source
- Native format for files (json, csv, xml etc)
- Parquet for columnar sources
- Data is immutable
- Store duplicate copies
- No schema validation
- Full history / “Stored forever”

BASE

- Cleansed
- Deduplicated
- Quality enforced
- Error handled
- Renamed attributes
- Exploded/projected data
- Null values handled
- Functional history (SCD1/SCD2)
- Keys may be applied

CURATED

- Business level data
- Consolidation logic
- Aggregated
- Structured in facts and dimensions
- Ready to serve for consumers

twoday

Metadata driven from Landing to Base

- The **AquaShack_Functions** notebook contains general functions used for operations like reading DataFrames, writing DataFrames, and utilities for manipulating DataFrames. To use the functions, include the notebook in your Spark session with the %run command. Once included, you can reference any function within the notebook, such as read_meta_from_json, write_to_delta_overwrite, and others.

- Data movement between the Landing and Base layers is managed by the LandingToBase notebook. This notebook reads metadata from a designated json-file and processes the source path (folder) specified within it.

- Metadata is maintained in the meta_data.json file. This metadata defines the rules for reading data from the Landing layer and writing it to the Base layer.

```
[
    {
        "source": "Files/External/Sample data/Sales/Customers/Customers.csv",
        "format": "csv",
        "destination": "sales_Customers",
        "projected_columns": ["CustomerID", "FirstName", "LastName"]
    },
    {
        "source": "Files/External/Sample data/Sales/Products/Products.parquet",
        "format": "parquet",
        "destination": "sales_Products",
        "projected_columns": []
    },
    {
        "source": "Files/External/Sample data/Sales/Transactions/",
        "format": "csv",
        "destination": "sales_Transactions",
        "projected_columns": []
    }
]
```

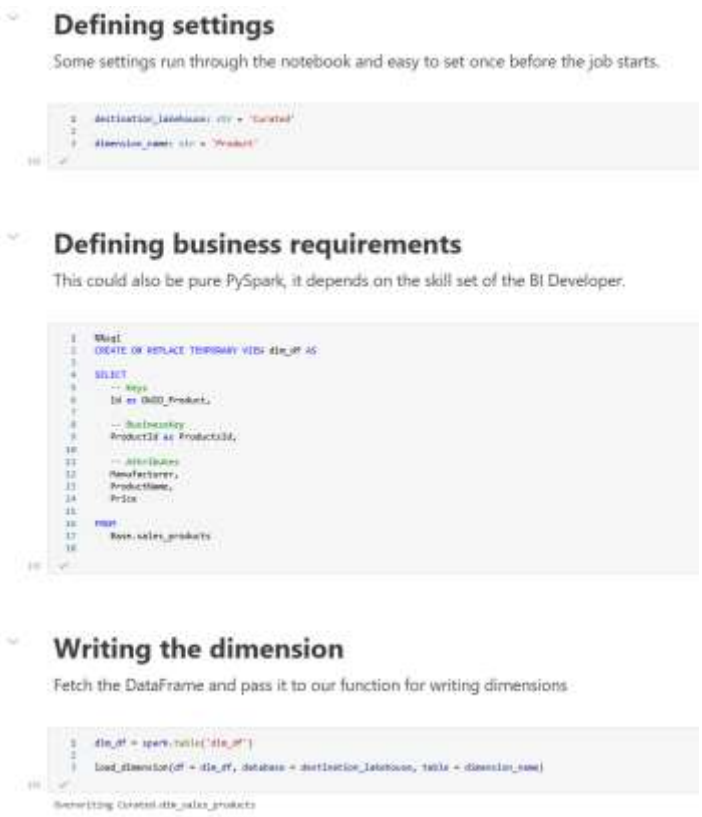
twoday

Demo time

twoday

Base to Curated

- The Curated layer stores data in a dimensional model with entities like dimensions, facts, and bridge tables.
- Entity-specific Spark notebooks handle data creation, named as:
  - Load_Dimension_[dimension_name]
  - Load_Fact_[fact_name]
- Each notebook follows a standard structure:
  - Parameters & Settings.
  - Business Logic: Written in SparkSQL.
  - Generic Load Execution: Functions like load_dimension or load_fact.
- All notebooks are tied to the Base Lakehouse item for centralized management.



twoday

Demo time

twoday

Getting started with AquaShack

- **GitHub:** https://github.com/ChristianHenrikReich/AquaShack
- The Setup.ipynb notebook holds all to get started and sets up the AquaShack lakehouse example. The only required action is to run this code in a notebook within the target Fabric Workspace.

twoday

Demo time

twoday

AquaVilla: Fabric Lakehouse Accelerator

Infrastructure

Publisher app and scripts

- Azure infrastructure
- Fabric env. setup

Solution

Metadata-driven ingest, transform and orchestration

- Data Pipelines
- Notebooks
- Onelake as storage
- Matadata-driven development

CI/CD

Pipelines and scripts

- Build and release
- Handling features

Documentation, learning and guidelines

twoday

Selected features of AquaVilla

- **Automated setup:** Full setup of infrastructure, workspaces and items
- **Metadata:** Integration, processing and orchestration with metadata
- **Ingestion**: Batch ingest from multiple sources using delta and full.
- **Common transformations:** Deduplication, flattening of complex data structures, data conversion, translations and more.
- **Update strategies:** Append, overwrite, upsert type 1 & 2 etc.
- **Data warehousing:** Functions specifically for dimensional modelling
- **Samples:** Sample implementations for all functionality and layers
- **Orchestration:** Through Pipelines and Notebooks
- **Deployment:** Full enterprise CI/CD using Azure DevOps Pipelines

Learn more about AquaVilla: https://www.youtube.com/watch?v=0fvLLYW5yW4

twoday

Time for questions!

twoday

twoday

Share **your thoughts** and help our speakers!

fabfeb.app/feedback