databricks

The Databricks Lakehouse Platform

Modern data, analytics and AI workloads

Presenters

Jason Pohl

Principal Solutions Architect

Sean Owen

Principal Solutions Architect

The future is here

...it's just not evenly distributed

83%

of CEOs say AI is a strategic priority

MITSloan Management Review

85%

of big data projects fail

Gartner

$3.9T

in business value created by AI in 2022

Gartner

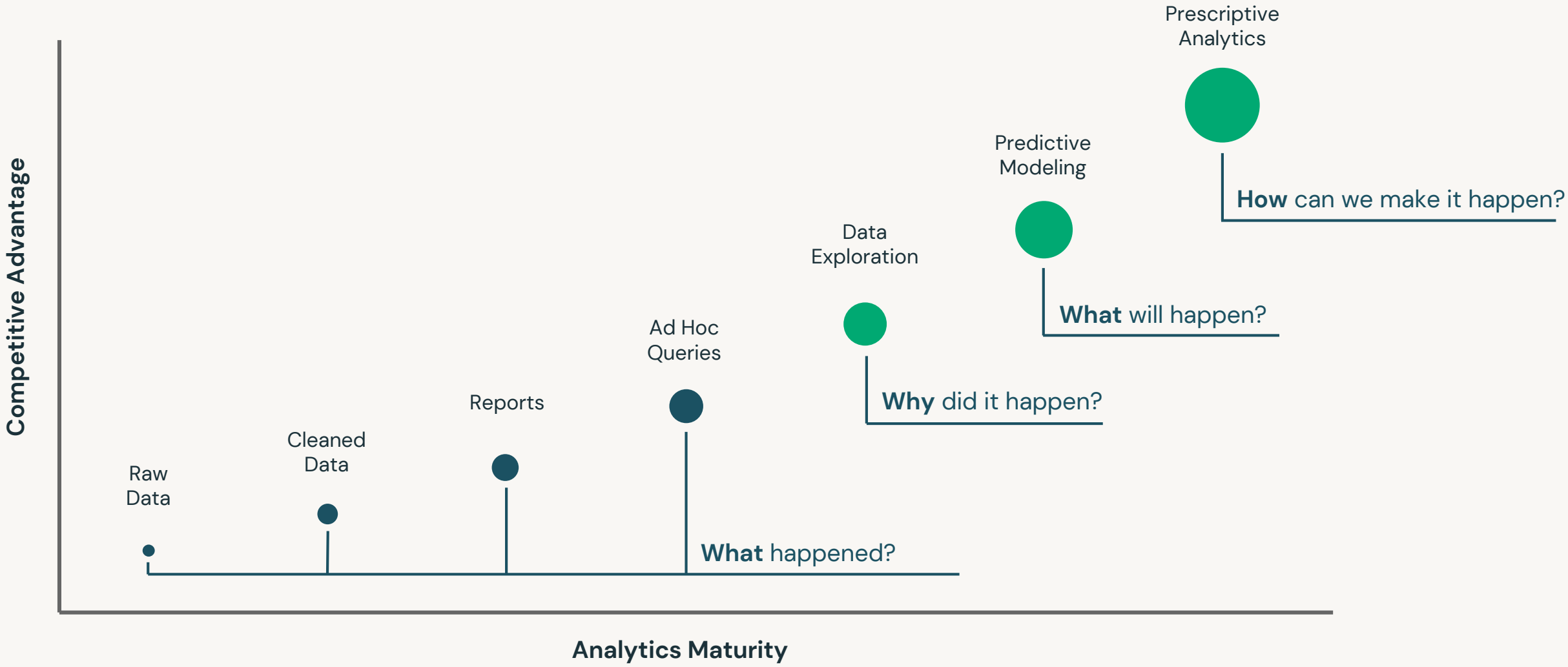
87%

of data science projects never make it into production

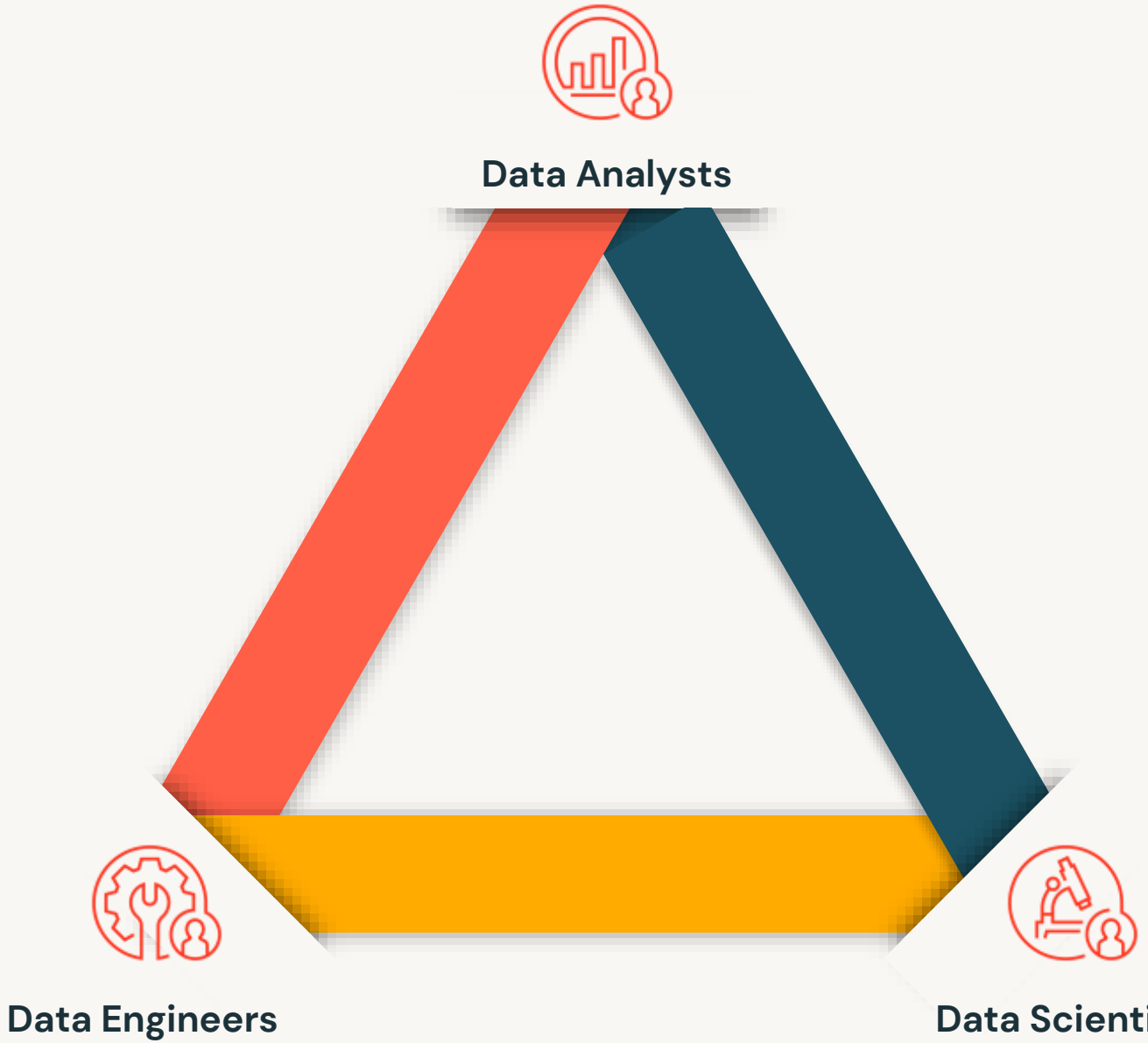
VB

The Data and AI Maturity Curve

From descriptive to prescriptive



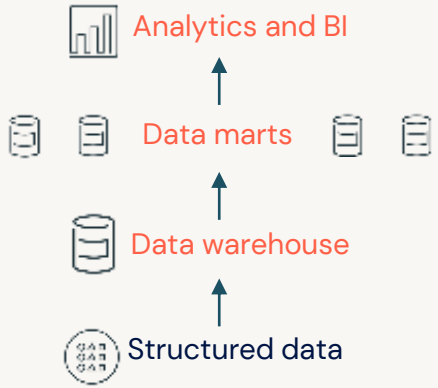
Modern Data Teams



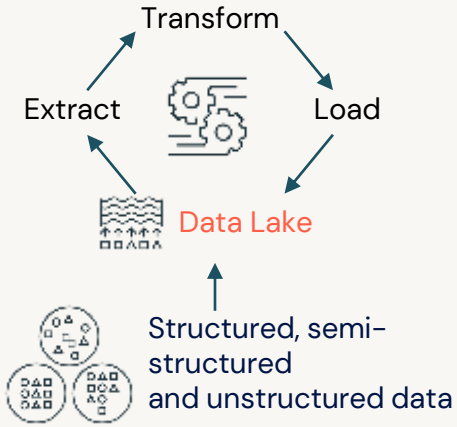
The vast majority of companies struggle to get this right

Most enterprises struggle with data

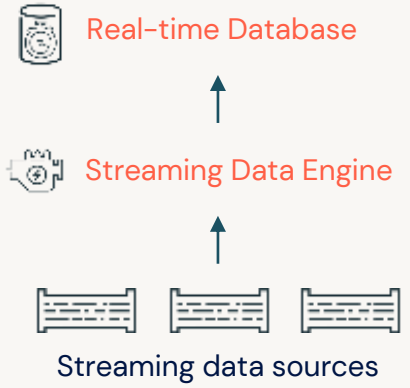
Data Warehousing



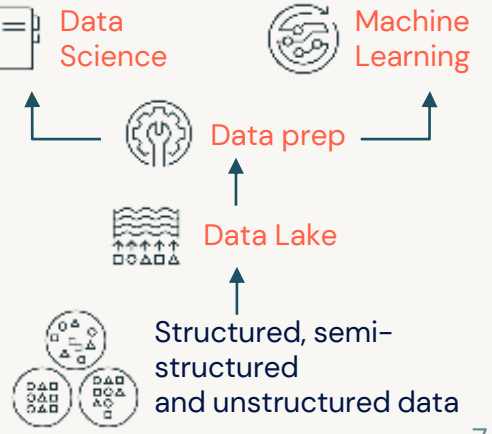
Data Engineering



Streaming



Data Science and ML

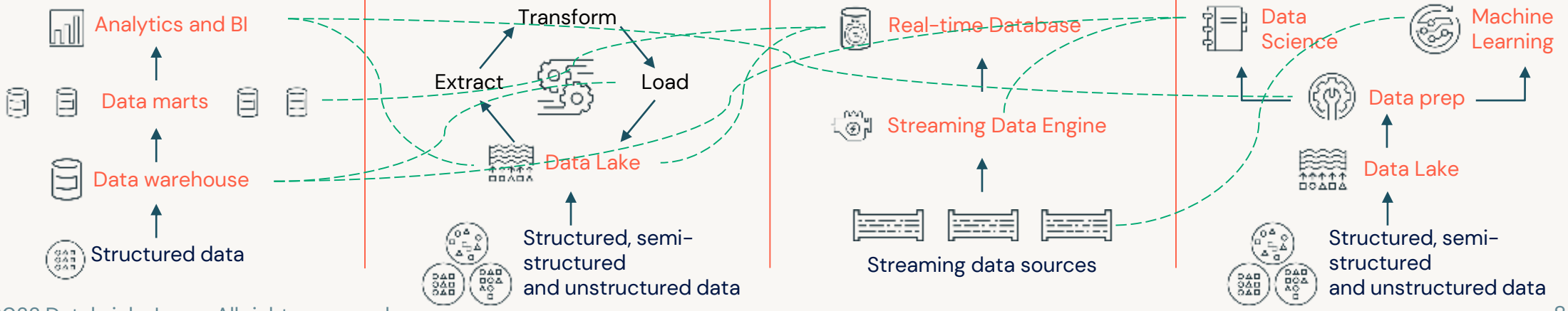


Siloed stacks increase data architecture complexity

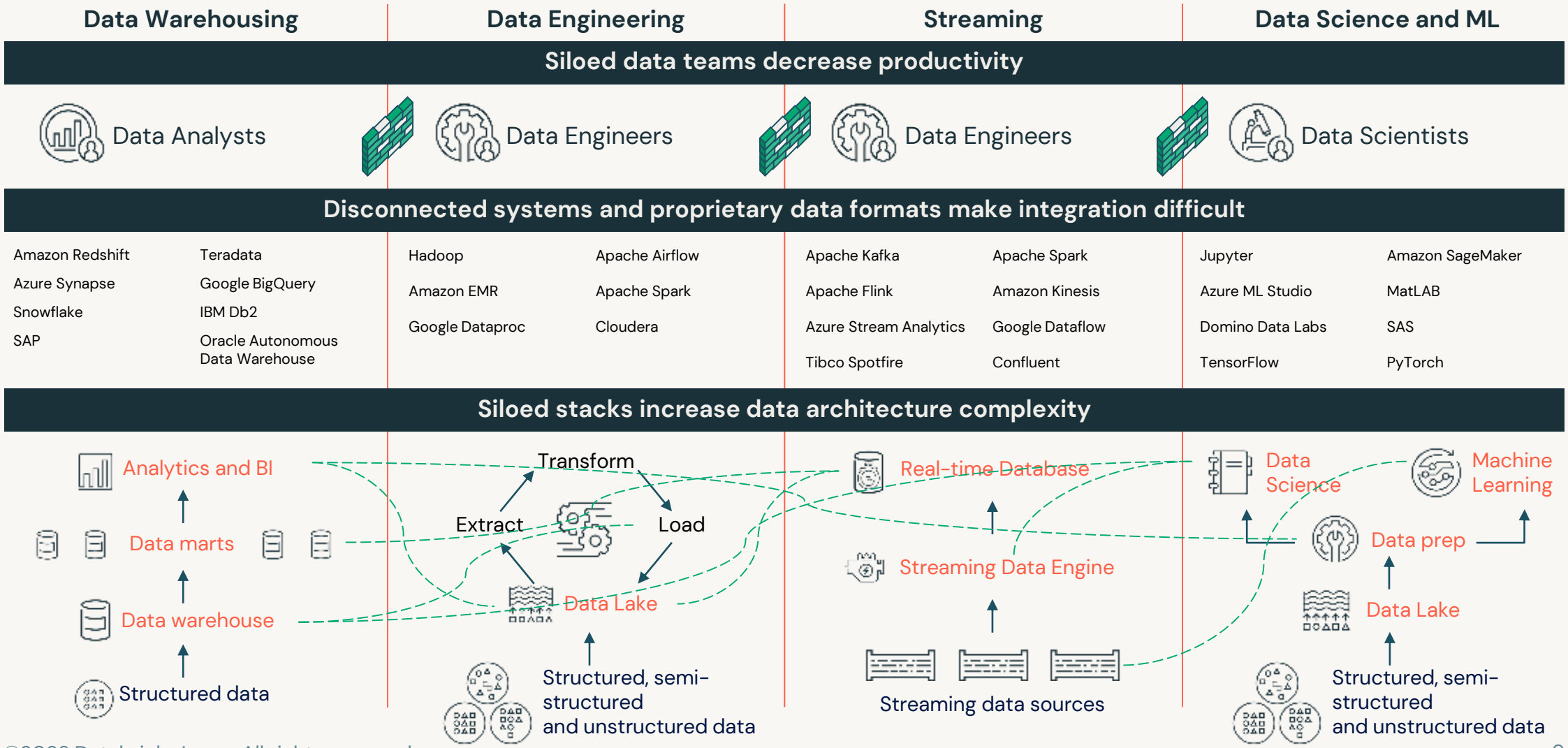
Most enterprises struggle with data

Data Warehousing		Data Engineering		Streaming		Data Science and ML	
Disconnected systems and proprietary data formats make integration difficult							
Amazon Redshift	Teradata	Hadoop	Apache Airflow	Apache Kafka	Apache Spark	Jupyter	Amazon SageMaker
Azure Synapse	Google BigQuery	Amazon EMR	Apache Spark	Apache Flink	Amazon Kinesis	Azure ML Studio	MatLAB
Snowflake	IBM Db2	Google Dataproc	Cloudera	Azure Stream Analytics	Google Dataflow	Domino Data Labs	SAS
SAP	Oracle Autonomous Data Warehouse			Tibco Spotfire	Confluent	TensorFlow	PyTorch

Siloed stacks increase data architecture complexity



Most enterprises struggle with data



Where does all this complexity come from?

data warehouses vs. data lakes

VS.

Data Warehouse

Data Lake

Warehouses and lakes create complexity

Two separate copies of the data

Warehouses	**Lakes**
Proprietary	Open

Incompatible interfaces

Warehouses	**Lakes**
SQL	Python

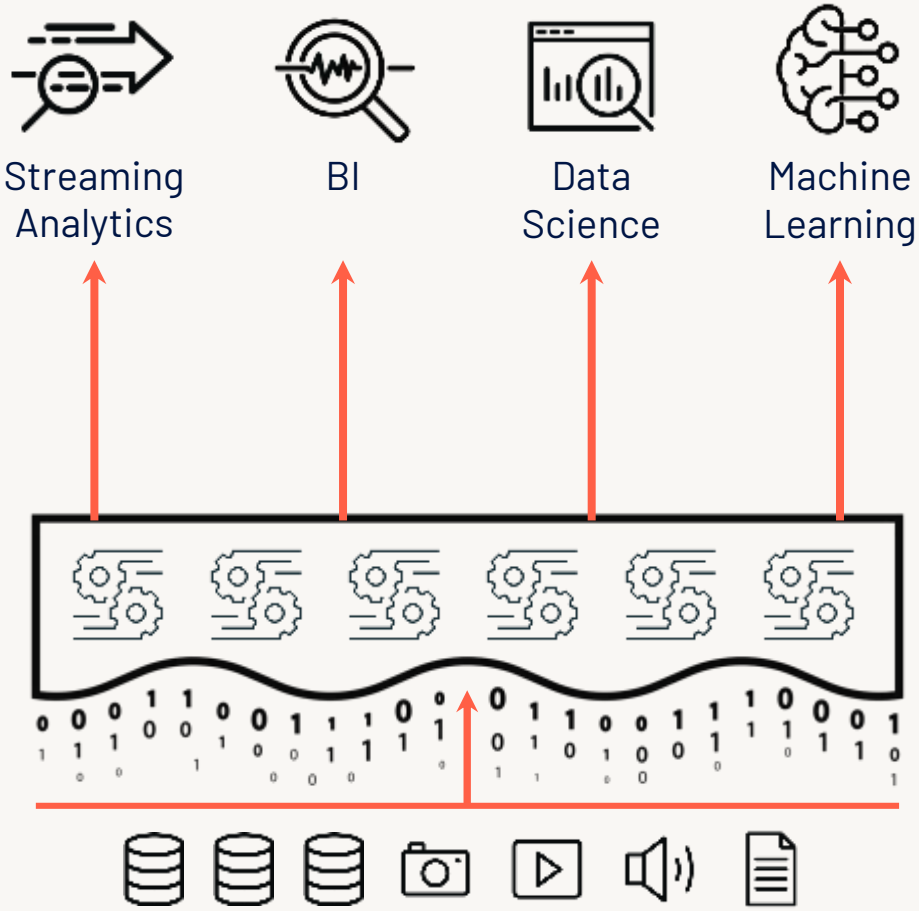
Incompatible security and governance models

Warehouses	**Lakes**
Tables	Files

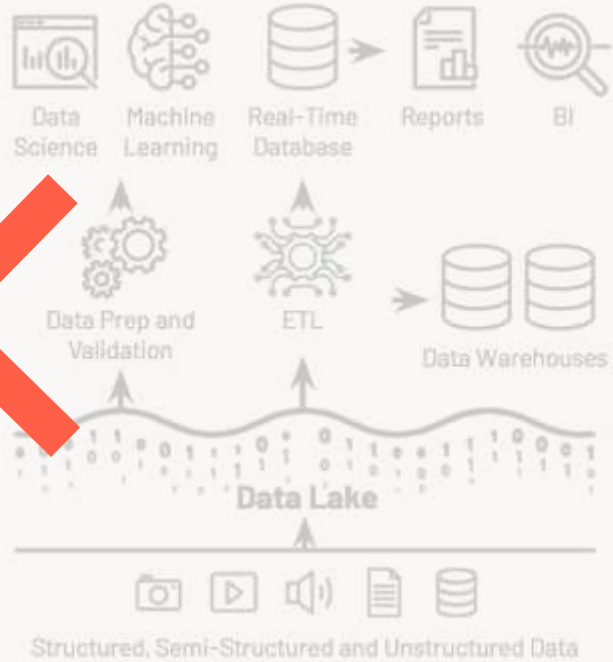
Data Lakehouse

One platform to unify all of
your data, analytics, and AI workloads

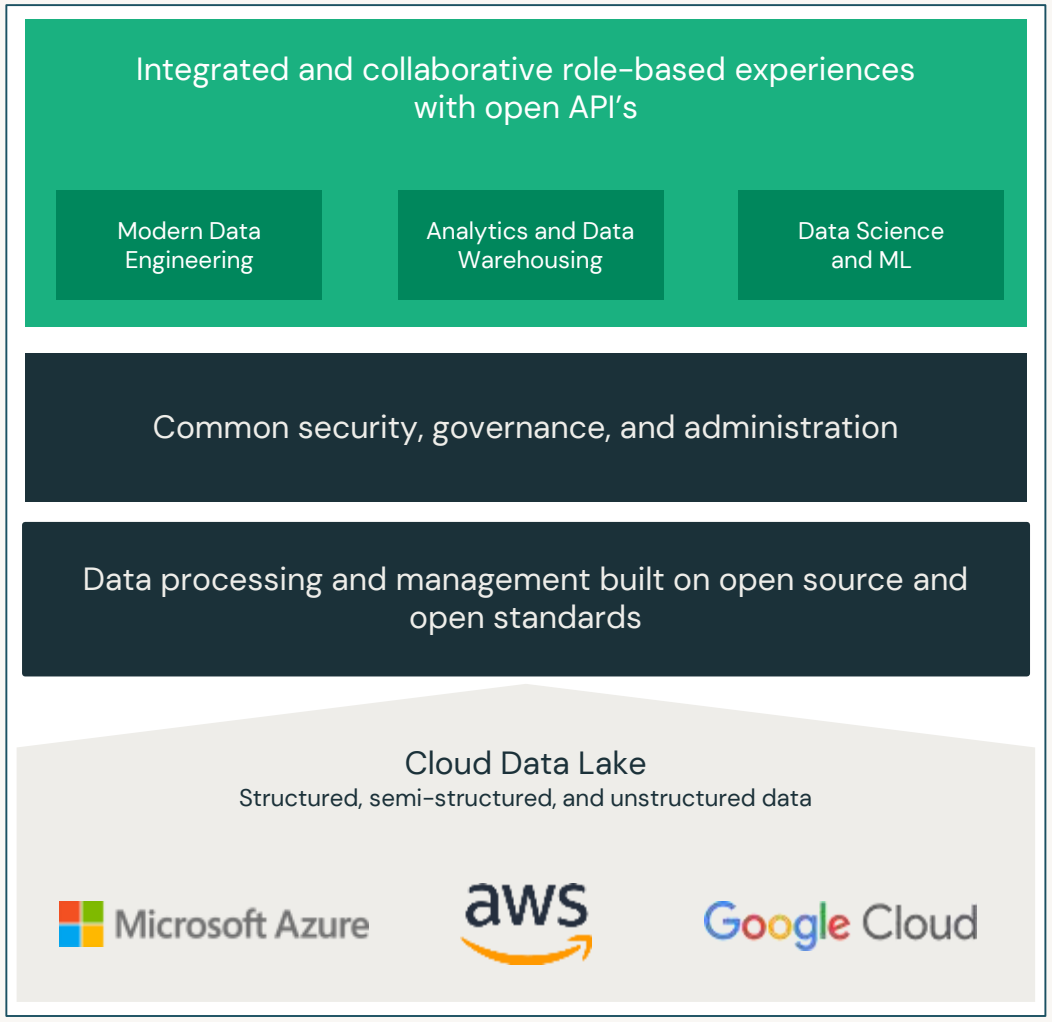
Data Warehouse



Data Lake



The data lakehouse offers a better path



Lake-first approach that builds upon where the freshest, most complete data resides

AI/ML from the ground up

High reliability and **performance**

Single approach to managing data

Support for all use cases on a single platform:

- Data engineering
- Data warehousing
- Real time streaming
- Data science and ML

Built on **open source** and open standards

Multi-cloud, work with your cloud of choice

Databricks

1. The Data Lakehouse Foundation
2. Modern Data Engineering on the Lakehouse
3. Analytics & Data Warehousing on the Lakehouse
4. Governance on the Lakehouse
5. Machine Learning on the Lakehouse

1 The Data Lakehouse Foundation

Data Lake

DELTA LAKE

An open approach to bringing data management and governance to data lakes

Better reliability with transactions

48x faster data processing with indexing

Data governance at scale with fine-grained access control lists

Data Warehouse

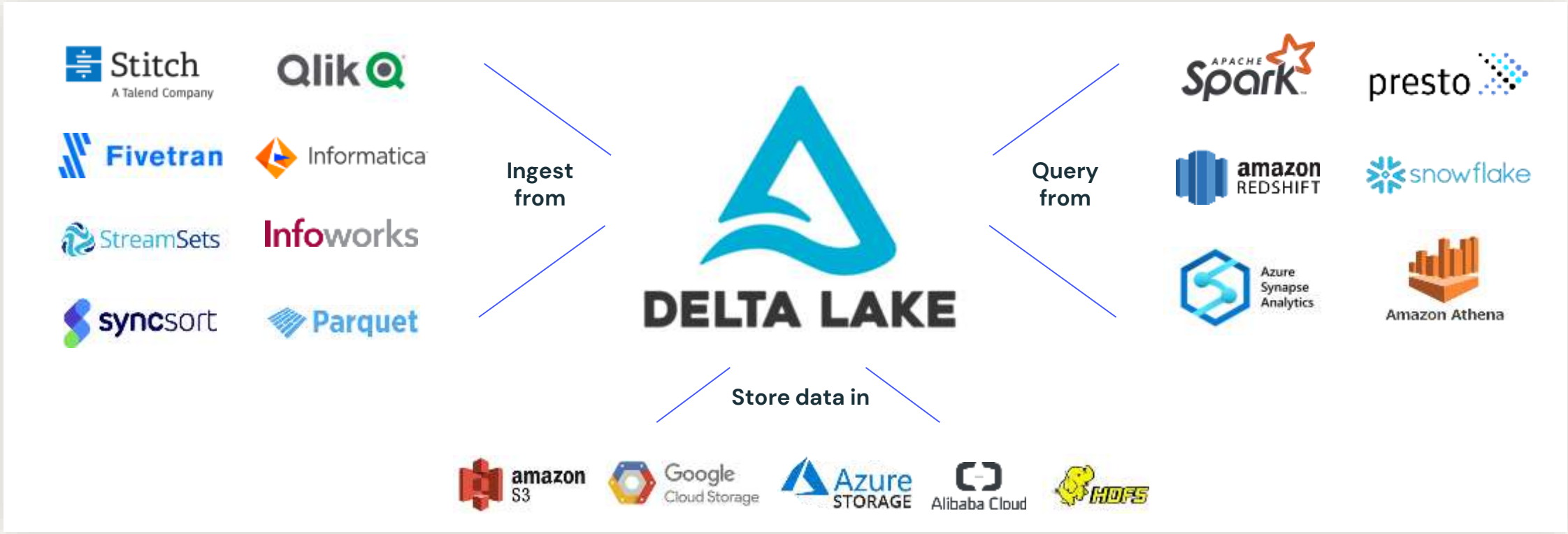
Delta Lake solves challenges with data lakes

RELIABILITY & QUALITY	→	ACID transactions
PERFORMANCE & LATENCY	→	Advanced indexing & caching
GOVERNANCE	→	Governance with Data Catalogs

Delta Lake adoption

Already >50% of Databricks workload

Broad industry support



What Delta Lake can do for you

Scale data insights throughout your organization with a simplified solution

Provide best price/performance

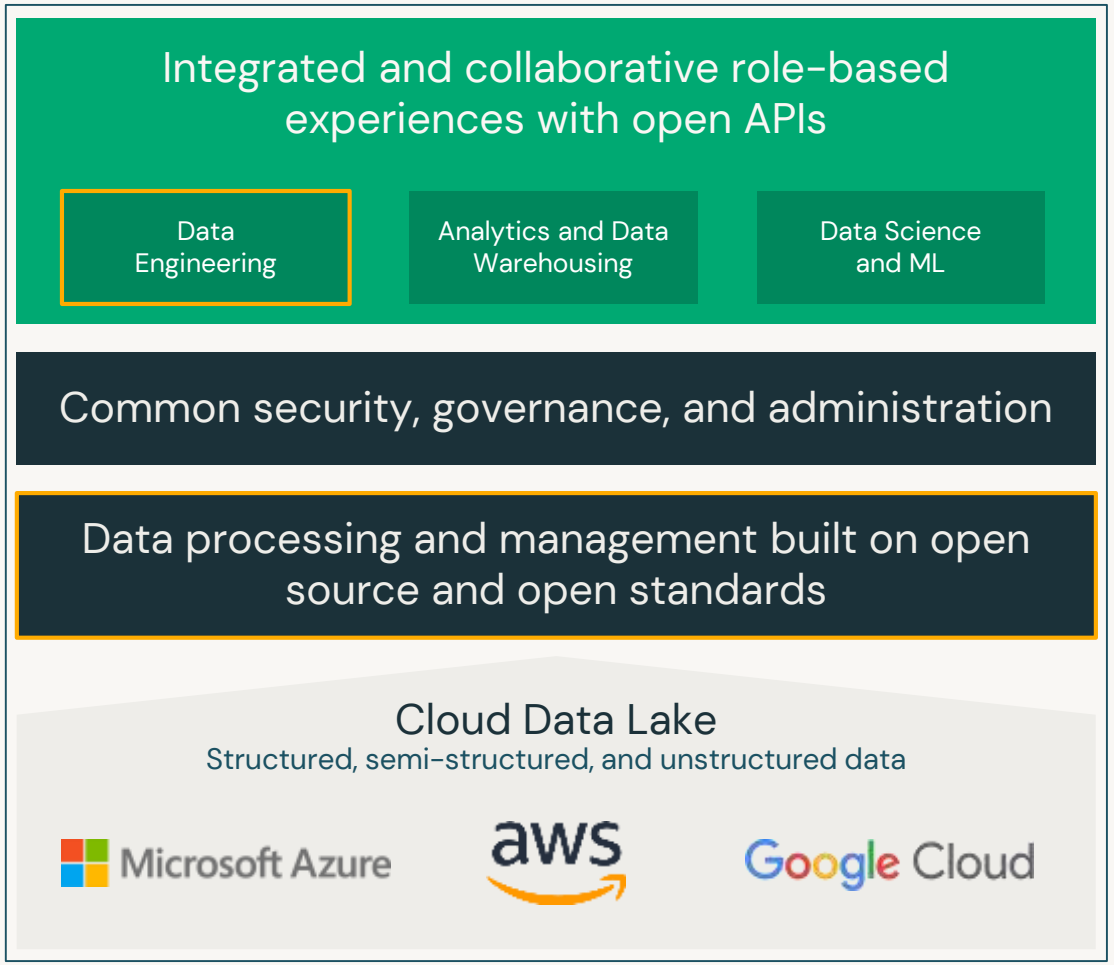
Enable a multi-cloud, secure infrastructure

The foundation of your lakehouse

Demo: Delta Lake

2 Modern Data Engineering on the Lakehouse

Modern data engineering on the lakehouse



Problems with Today's Architectures

Cheap to store all the data, but the 2-tier architecture is much more complex!

Data reliability suffers:

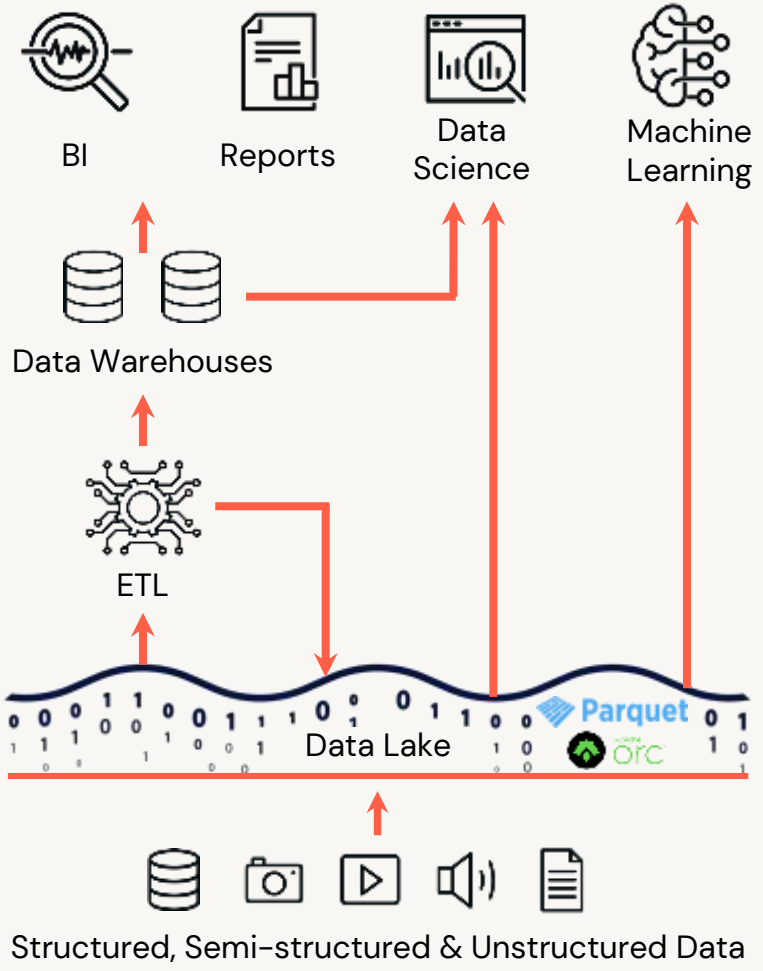
- Multiple storage systems with different semantics, SQL dialects, etc
- Extra ETL steps that can go wrong

Timeliness suffers:

- Extra ETL steps before data available in DW

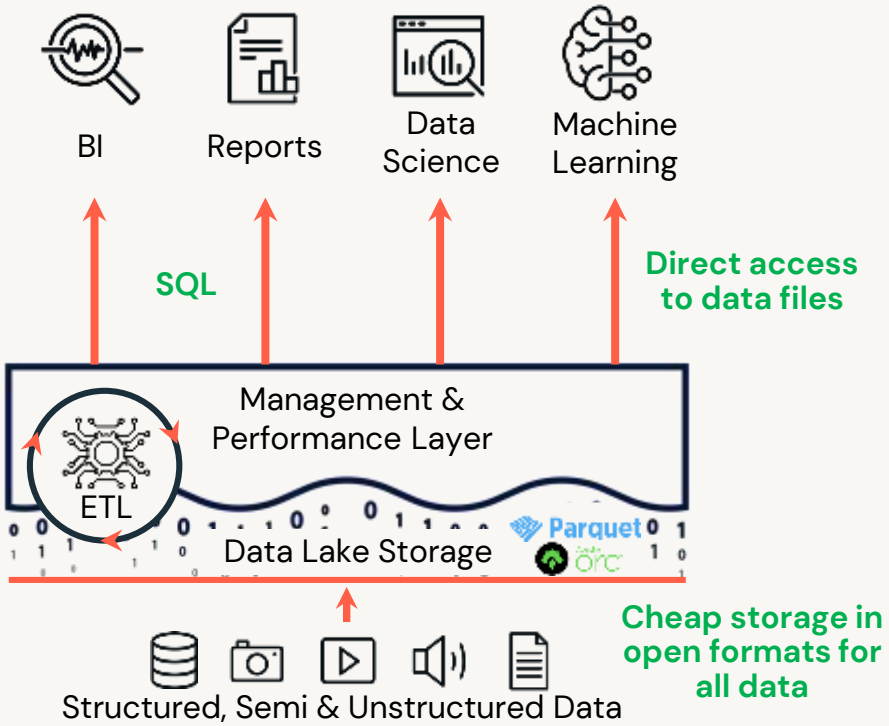
High cost:

- Continuous ETL, duplicated storage



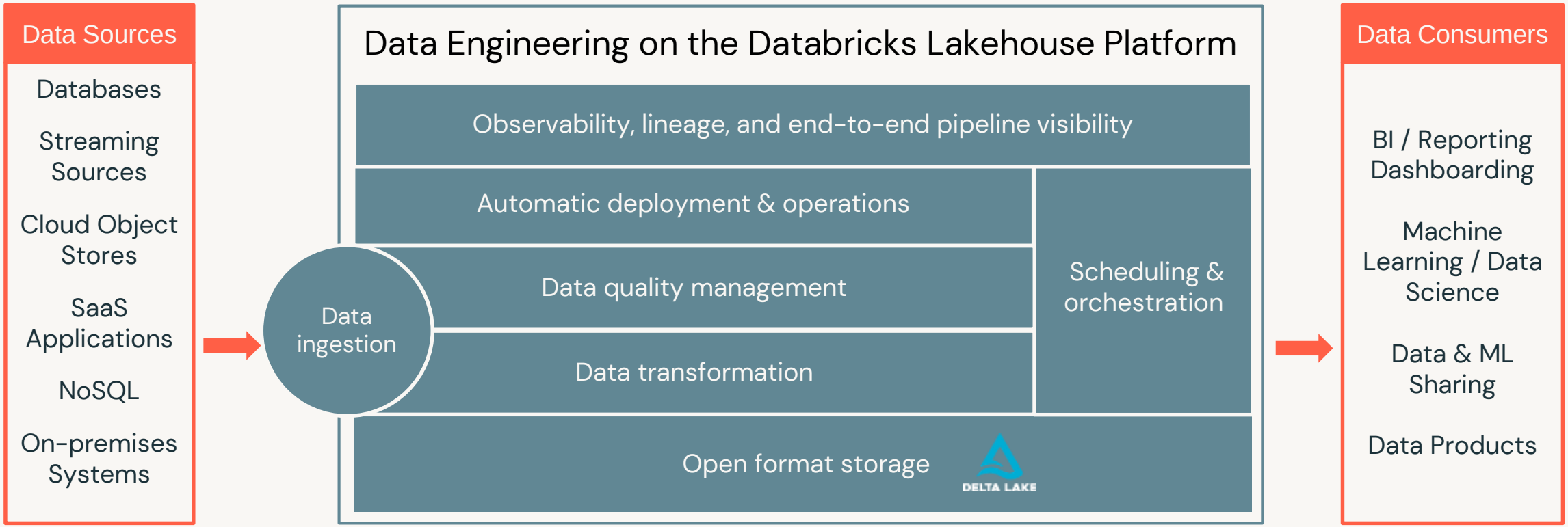
Lakehouse Systems

Implement data warehouse management and performance features on top of **directly-accessible data in open formats**



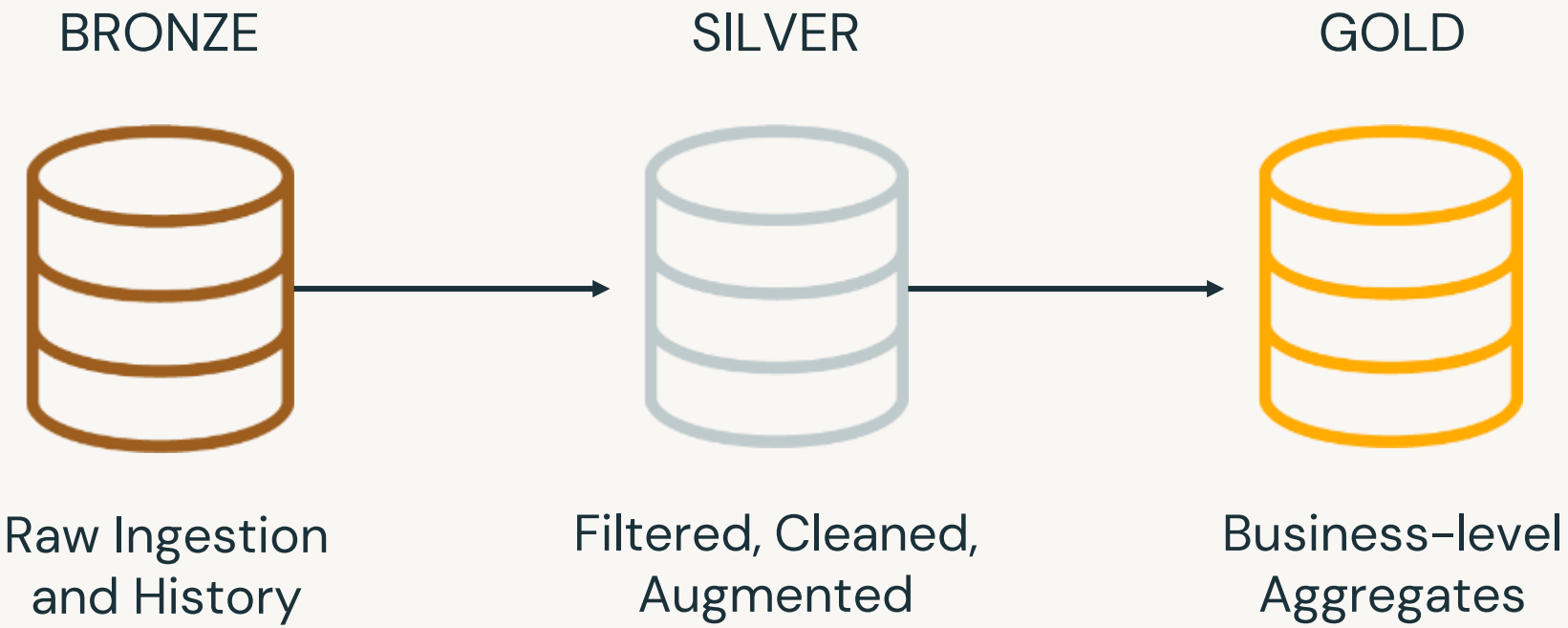
Can we get state-of-the-art performance & governance features with this design?

Modern data engineering on the lakehouse



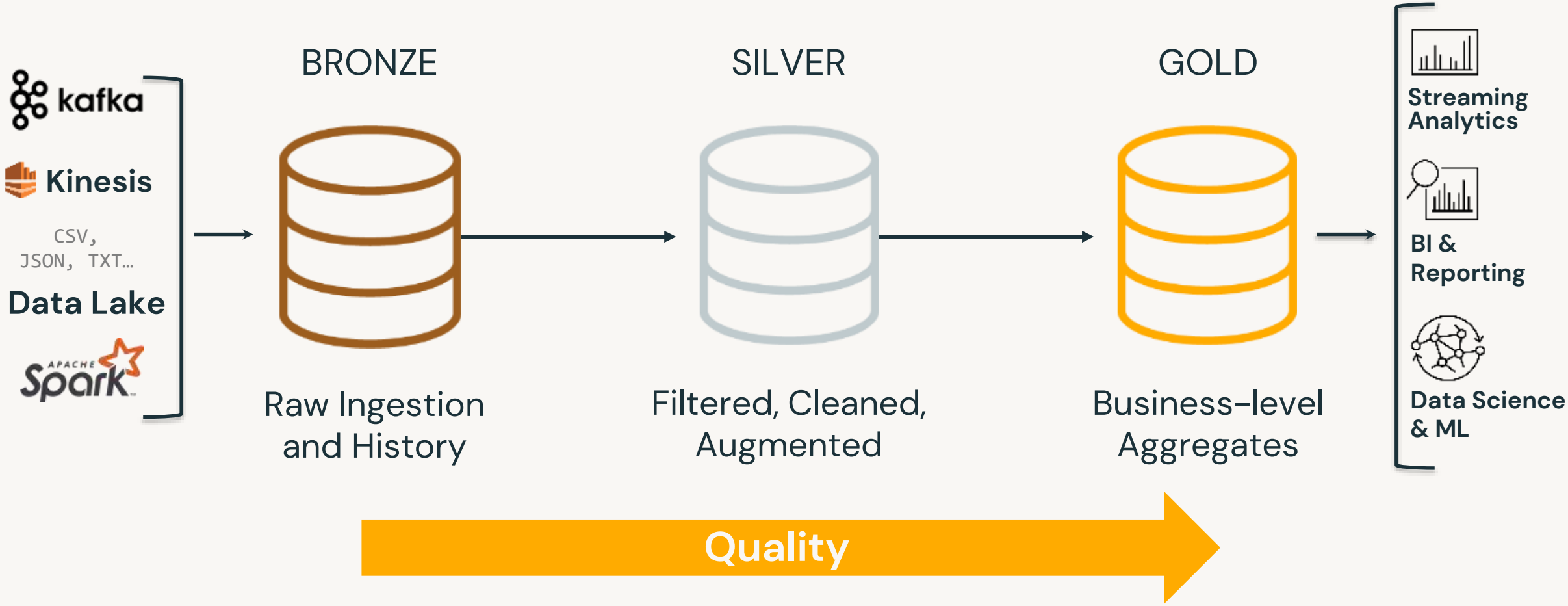
Building the foundation of a Lakehouse

Greatly improve the quality of your data for end users



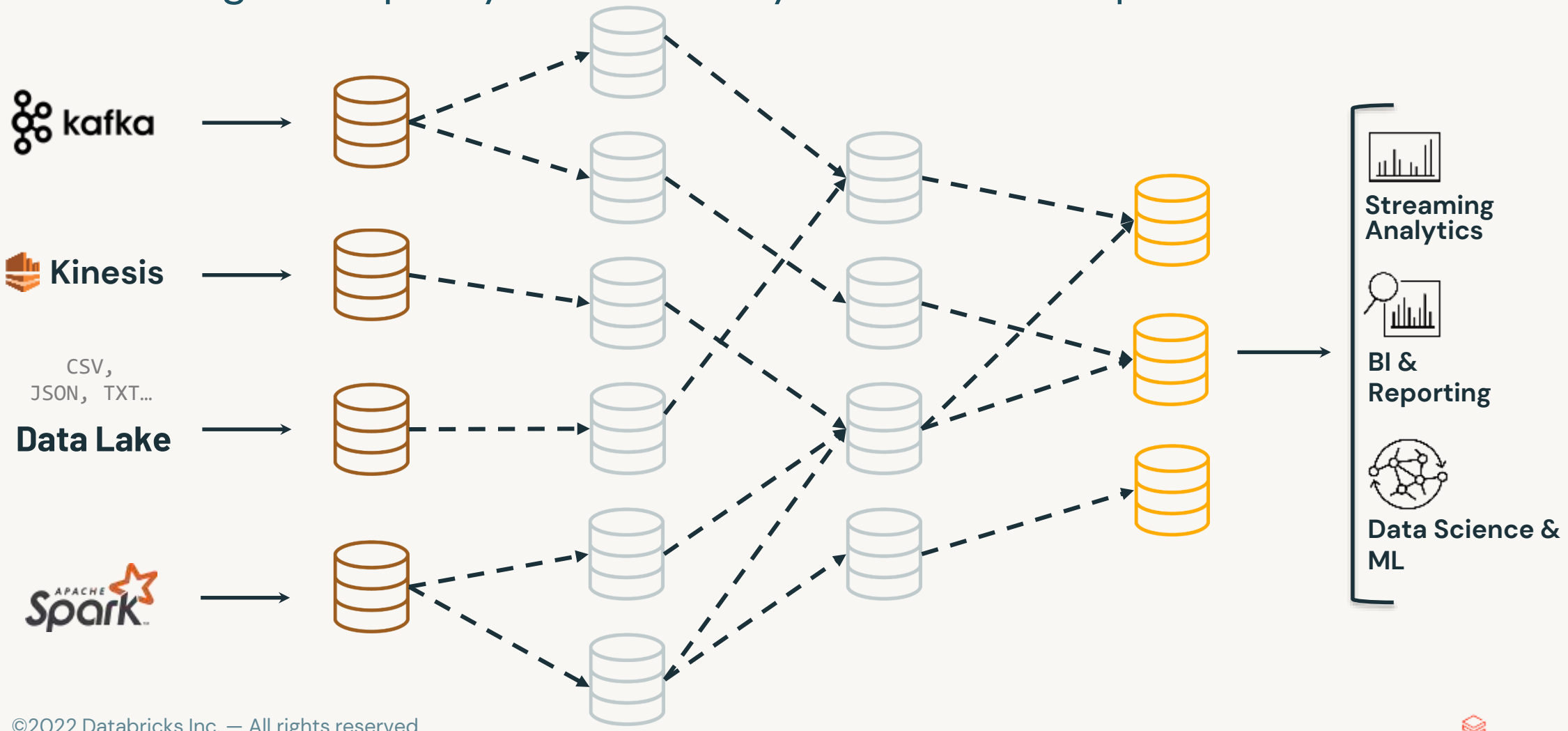
Building the foundation of a Lakehouse

Greatly improve the quality of your data for end users



But the reality is not so simple

Maintaining data quality and reliability at scale is complex and brittle



Large scale ETL is complex and brittle

Complex pipeline development

- Hard to build and maintain table **dependencies**
- Difficult to switch between **batch** and **stream** processing

Poor data quality

- Difficult to monitor & enforce **data quality**
- Impossible to trace data **lineage**

Difficult pipeline operations

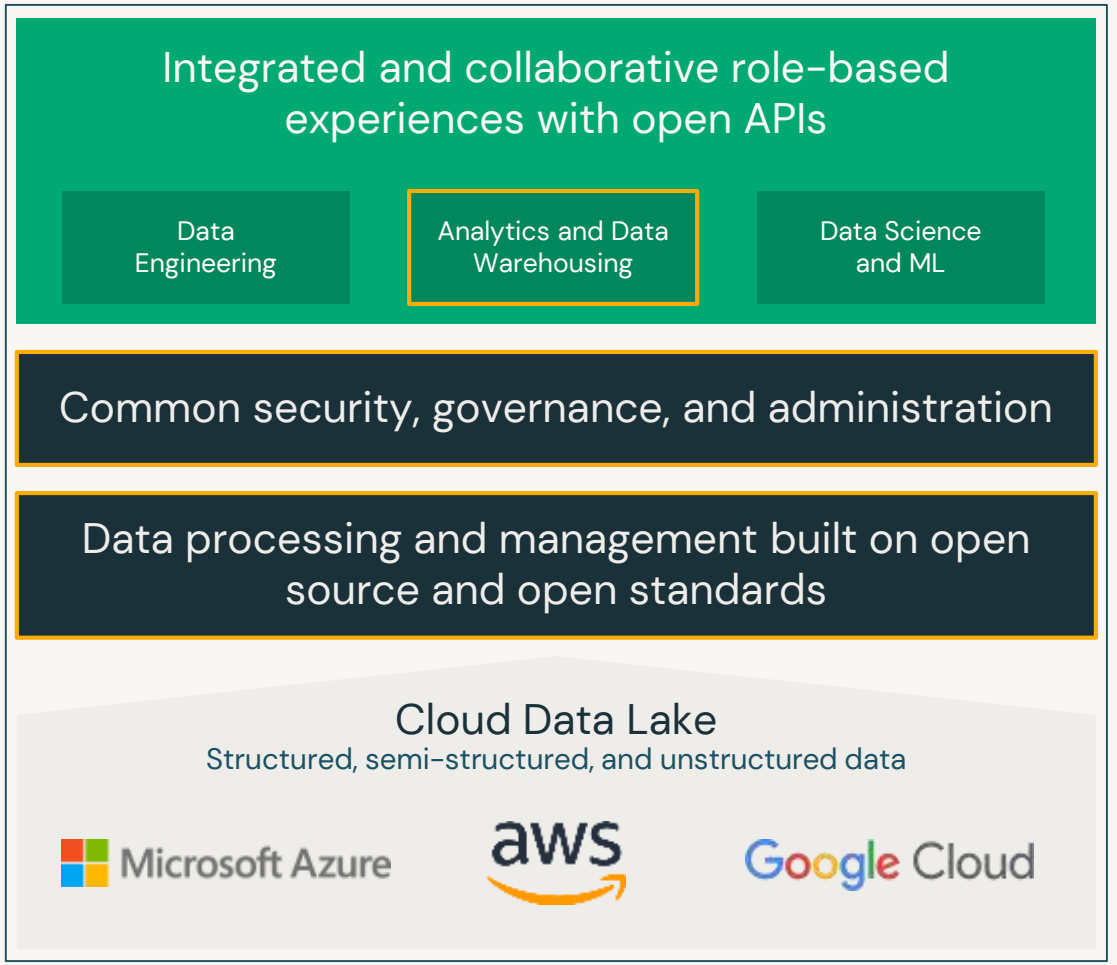
- Poor **observability** at granular, data level
- Error handling and **recovery** is laborious

Demo: Modern Data Engineering on the Lakehouse

3

Data Warehousing on the Lakehouse

Data Warehousing on the Lakehouse



BI & SQL workloads (DW) on Databricks

- Great performance and concurrency for BI and SQL workloads on Delta Lake
- Native SQL interface for analysts
- Support for BI tools to directly query your most recent data in Delta Lake
- Serverless

SQL Analytics

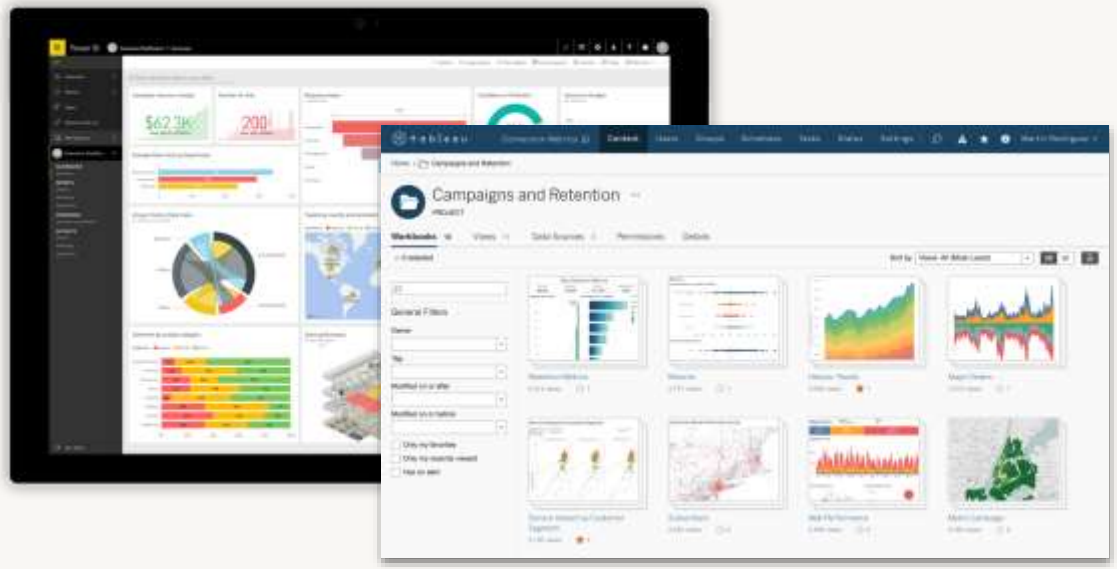
Analytics on the Lakehouse

World-Class Performance and Data Lake Economics

- Fast and predictable performance for all queries
- Simplified administration and fine-grained governance
- Analytics on the freshest data with your tools of choice

A platform for your tools of choice

Get critical business data in with one click integrations, and benefit from fast performance, low latency, and high user concurrency for **your existing BI tools**.



Ingest & ETL

dbt Fivetran

BI & Analytics

tableau Power BI

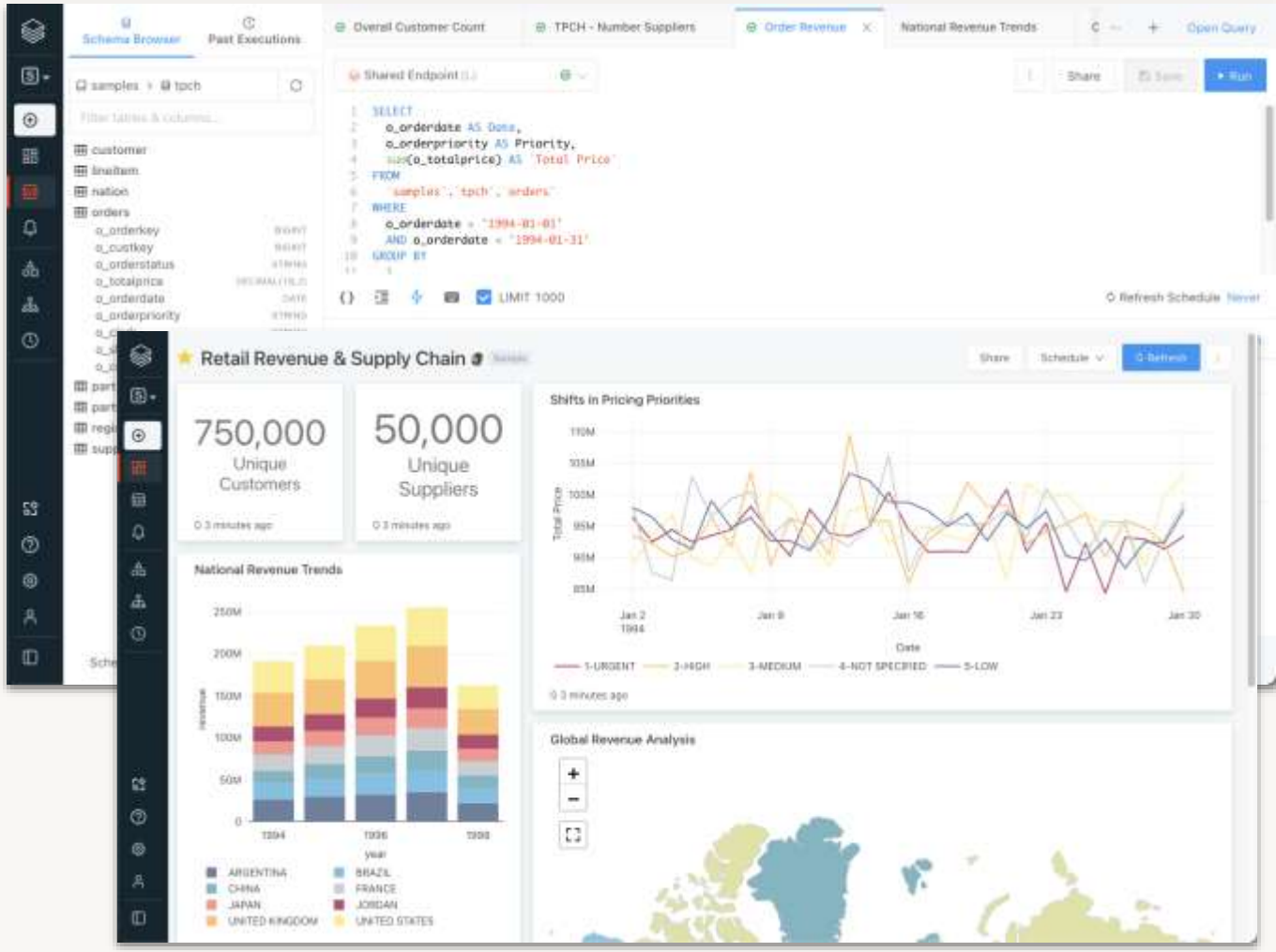
Qlik TIBCO Spotfire

MicroStrategy Looker

MODE ThoughtSpot

A first-class SQL development experience

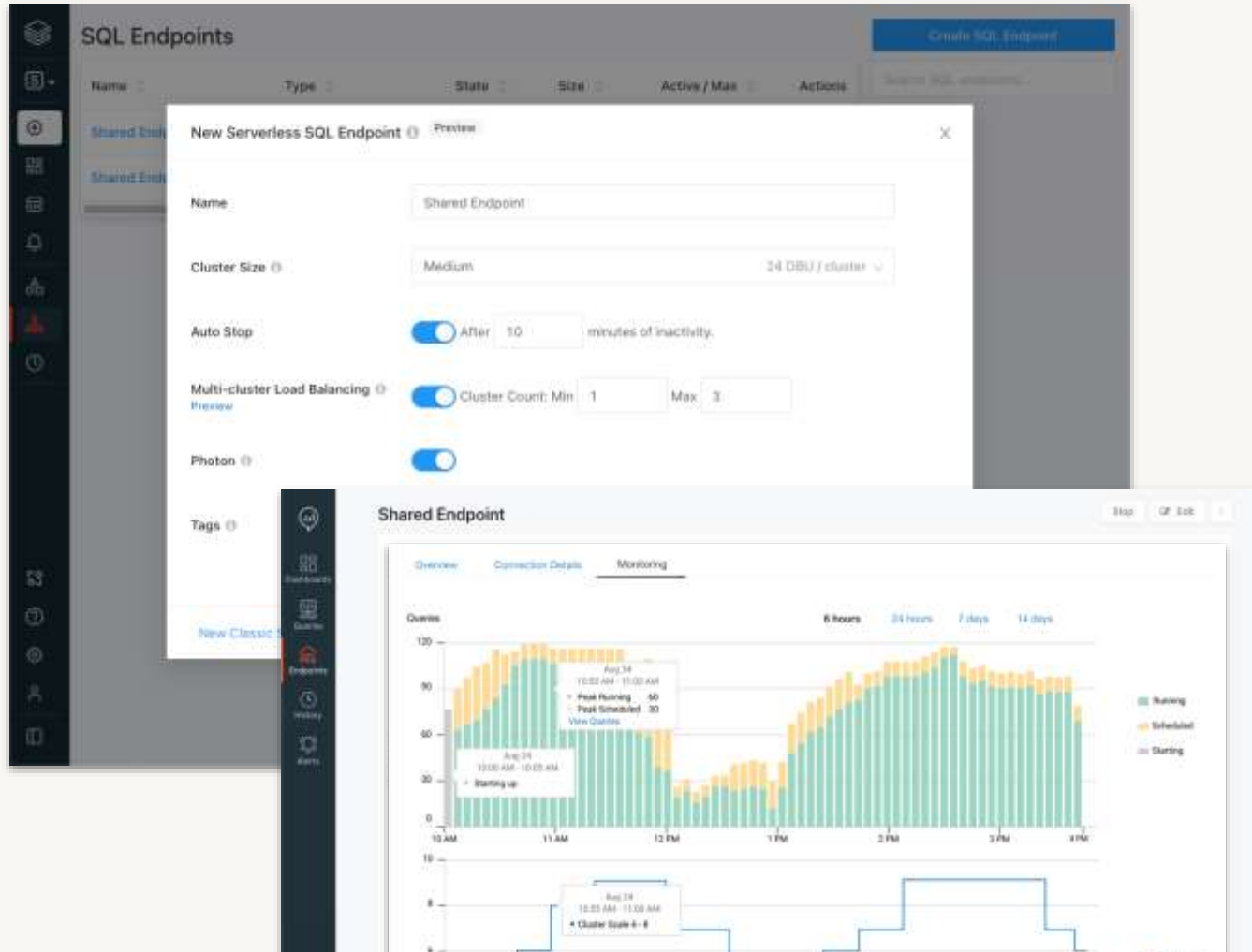
Query data lake data using familiar **ANSI SQL**, and find and share new insights faster with the built-in SQL query editor, alerts, visualizations, and interactive dashboards.



Simple administration and governance

Quickly setup instant, elastic SQL compute decoupled from storage. Databricks automatically determines instance types and configuration for the best price/performance.

Then, easily manage users, data, and resources with endpoint monitoring, query history, and fine-grained governance.

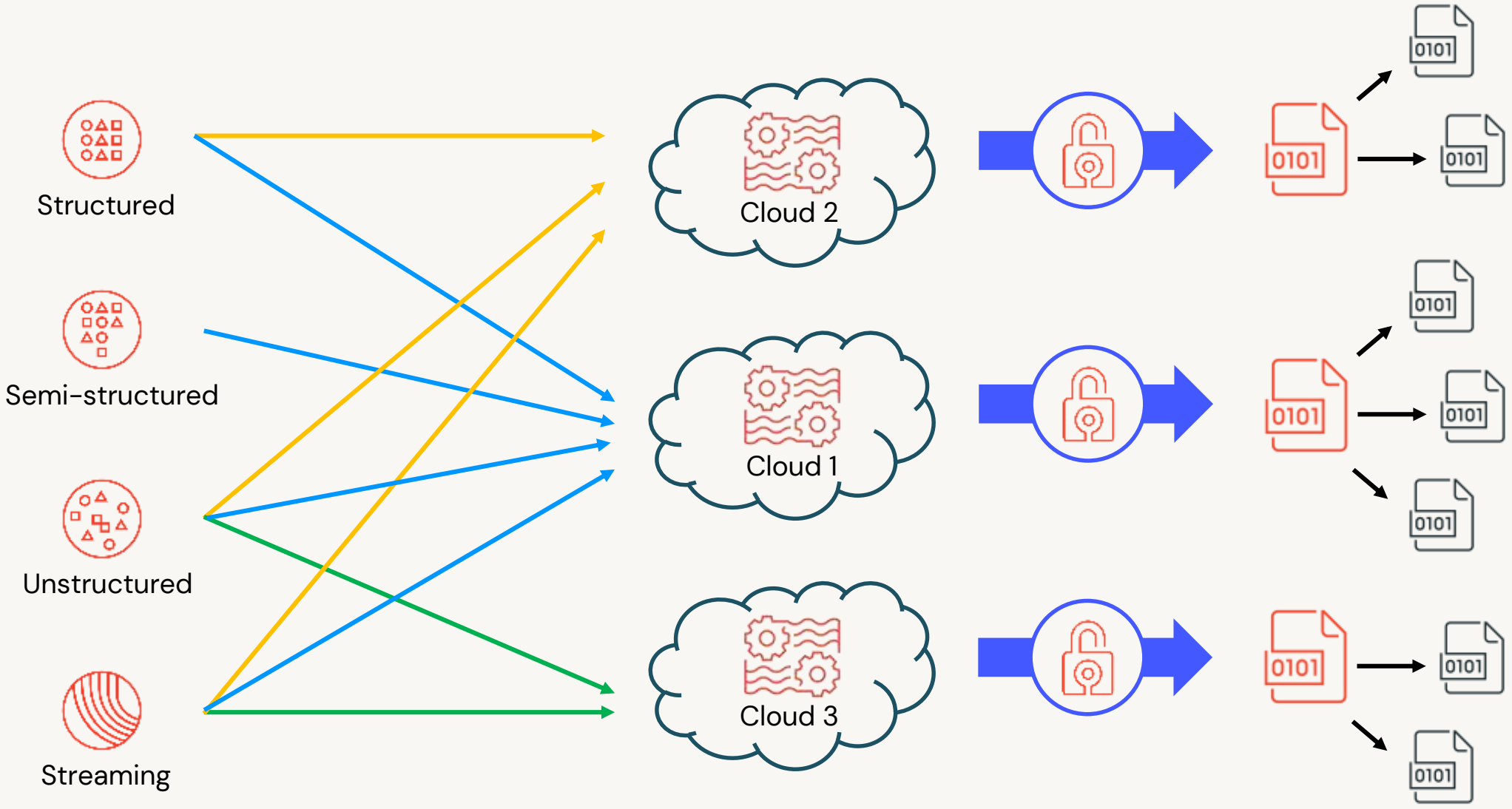


Demo: Data Warehousing on the Lakehouse

4 Data Governance on the Lakehouse

Governance requirements for data are quickly evolving

Governance is hard to enforce on data lakes



The problem is getting bigger

Enterprises need a way to share and govern a wide variety of data products

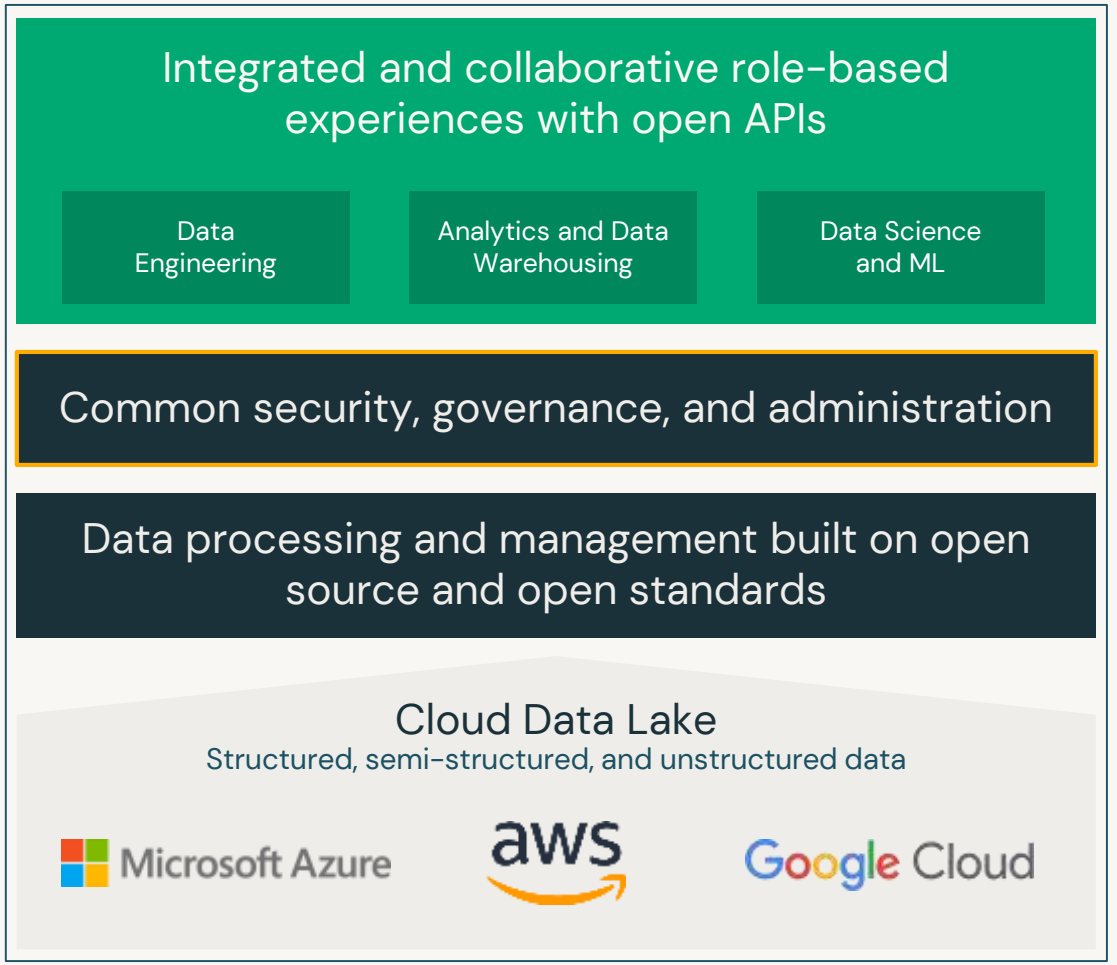
Files

Dashboards

Models

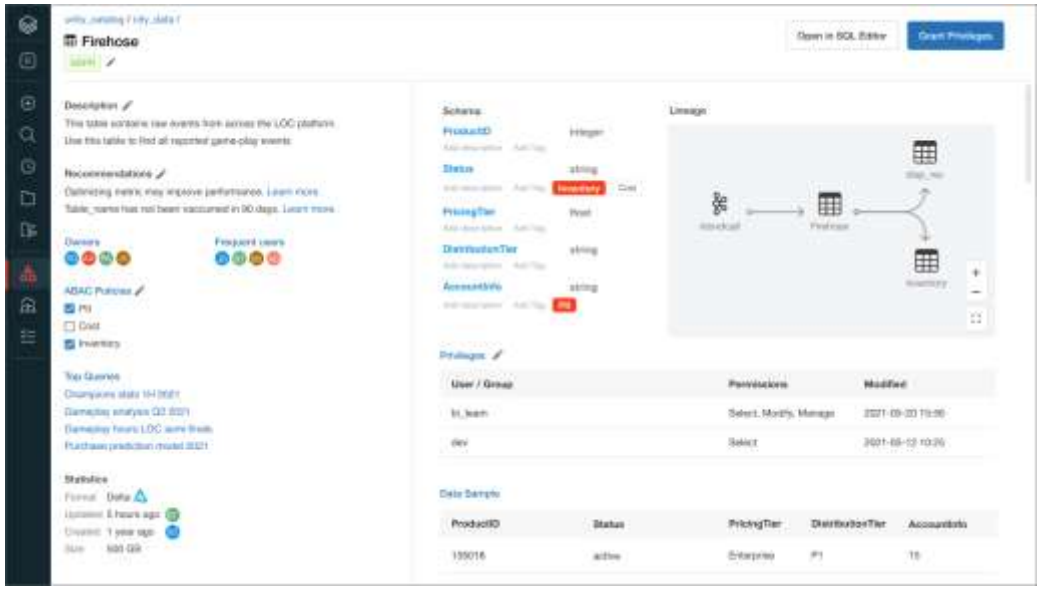
Tables

Data governance on the lakehouse



Unity Catalog for Lakehouse Governance

- **Centrally catalog, Search, and discover** data and AI assets
- Simplify governance with a **unified Cross-cloud governance model**
- **Easily integrate** with your existing Enterprise Data Catalogs
- **Securely share live data across platforms** with delta sharing



Data Sharing is Critical in the Digital Economy

Data Sharing is Critical in the Digital Economy

To **fully realize the value locked in data**, enterprises must be able to **securely exchange data** with trusted customers, partners, and suppliers

Current options are not fit for purpose

Homegrown Solutions
Built on APIs, direct connectivity, or SFTP

- Not scalable
- Complex
- Brittle

Commercial Solutions
Vendor provided technology

- Expensive
- Locked in
- Inflexible

The open approach to sharing

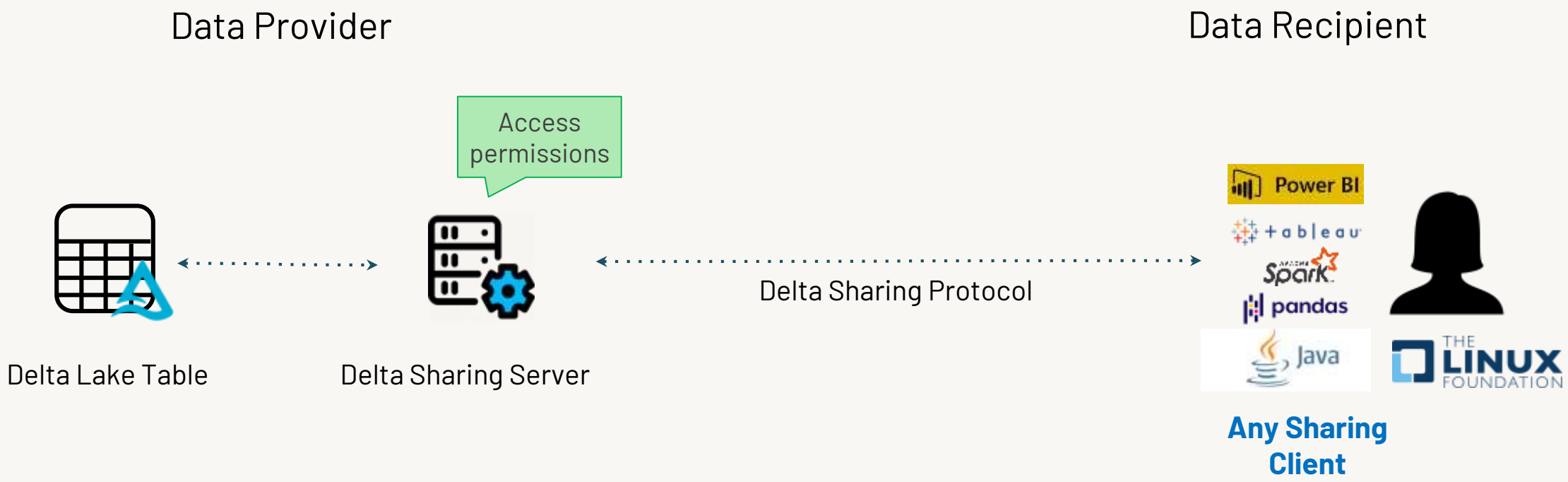
Fully open, without proprietary lock-in using any computing platform

Simple to share data with other organizations

Easily managed privacy, security, and compliance

DELTA SHARING

Permissible Access via Open Source Protocol



5

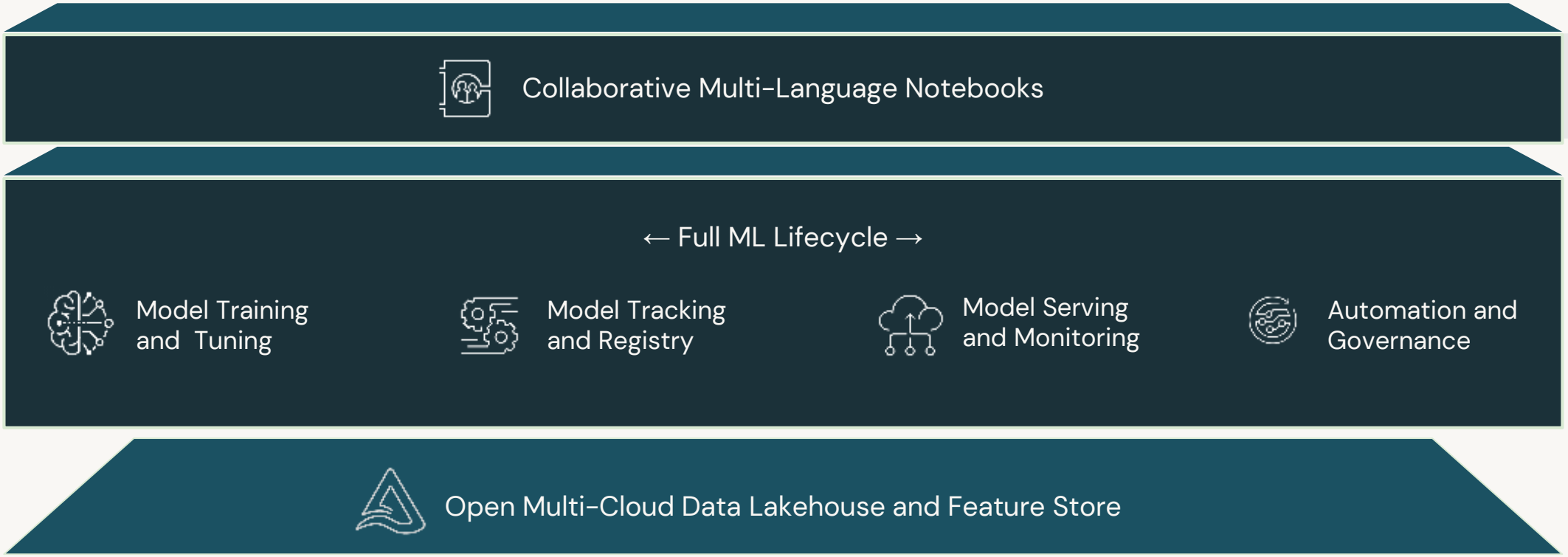
Machine Learning on the Lakehouse

Sean Owen
Principal Solution Architect @ Databricks

Machine Learning on the Lakehouse

Data Science and Machine Learning

A data-native and collaborative solution for the full ML lifecycle



Three Data Users

Business Intelligence

- SQL and BI tools
- Prepare and run reports
- Summarize data
- Visualize data
- (Sometimes) Big Data
- Data Warehouse data store

Data Science

- R, SAS, some Python
- Statistical analysis
- Explain data
- Visualize data
- Often small data sets
- Database, data warehouse data store; local files

Machine Learning

- Python
- Deep learning and specialized GPU hardware
- Create predictive models
- Deploy models to prod
- Often big data sets
- Unstructured data in files

Data Warehouse vs Data Lake

Which appeals to each user?

Data Warehouse

Business Intelligence

Data Science

Machine Learning

Data Lake

What ML Needs from a Lakehouse

How Is ML Different?

- Operates on **unstructured** data like text and images
- Can require learning from **massive data** sets, not just analysis of a sample
- Uses **open source** tooling to manipulate data as “DataFrames” rather than with SQL
- Outputs are **models** rather than data or reports
- Sometimes needs **special hardware**

What Does ML Need from a Lakehouse?

Your subtitle here

Access to Unstructured Data

- Images, text, audio, custom formats
- Libraries understand files, not tables
- Must *scale* to petabytes

Open Source Libraries

- OSS dominates ML tooling (Tensorflow, scikit-learn, xgboost, R, etc)
- Must be able to apply these in Python, R

Specialized Hardware, Distributed Compute

- Scalability of algorithms
- GPUs, for deep learning
- Cloud elasticity to manage that cost!

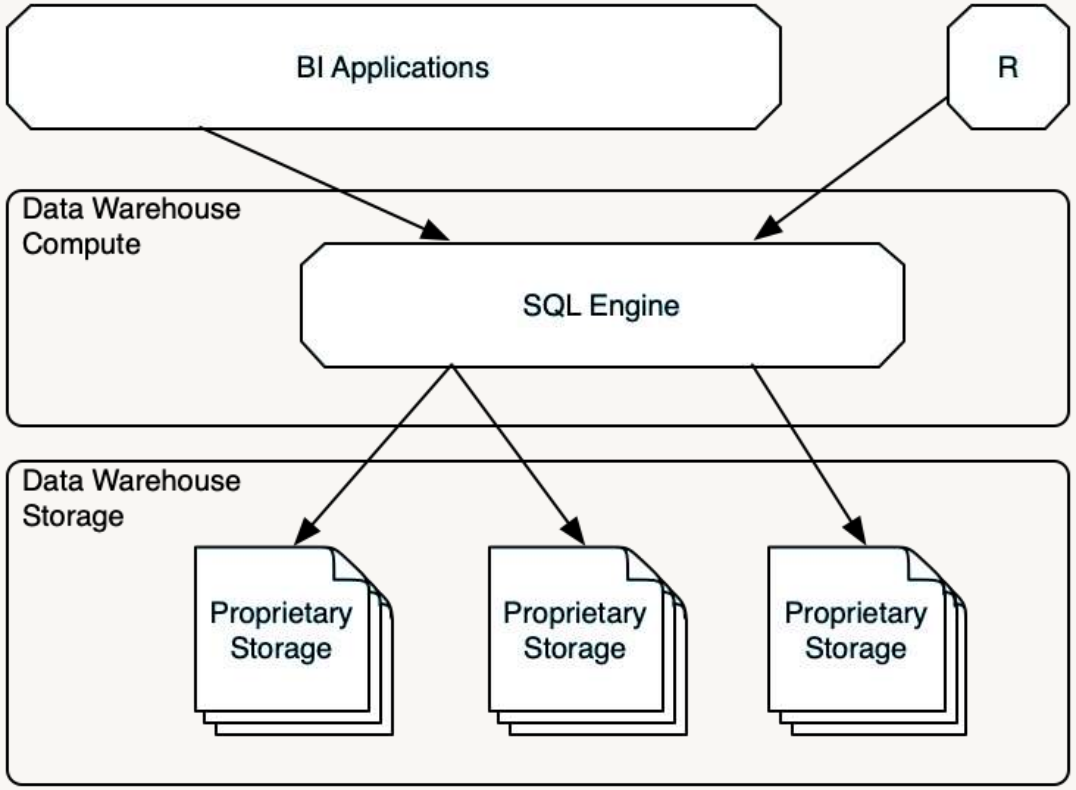
Model Lifecycle Management

- Outputs are model artifacts
- Artifact lineage
- Productionization of model

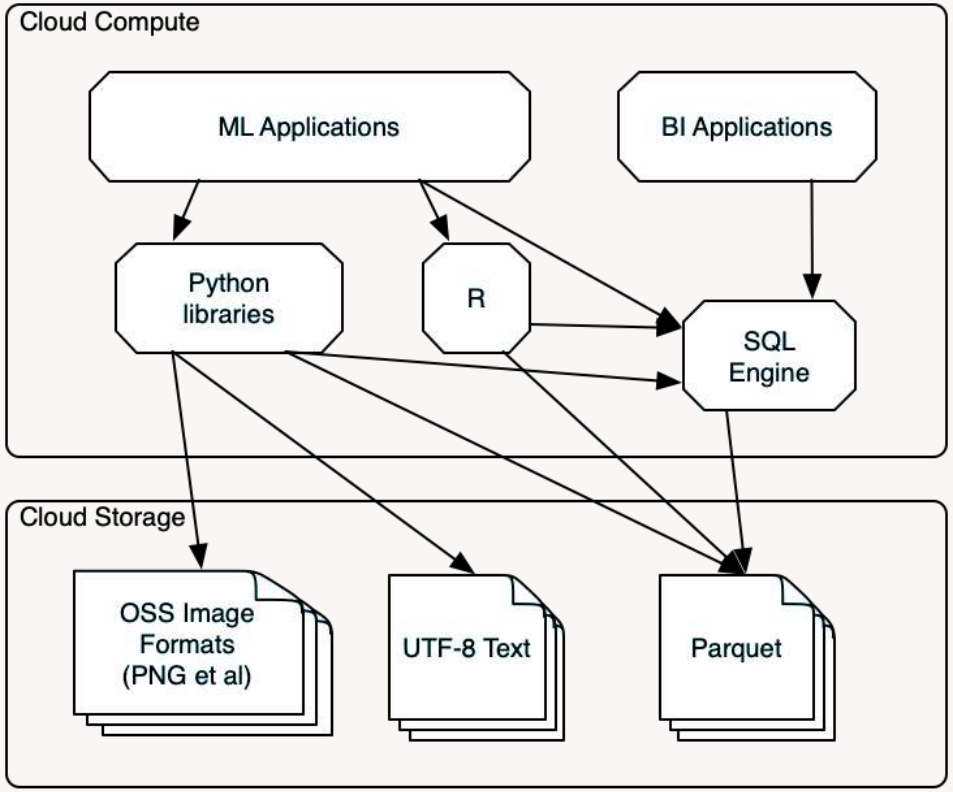
Architecture Sketches

Your subtitle here

Data Warehouse



Data Lakehouse



MLOps

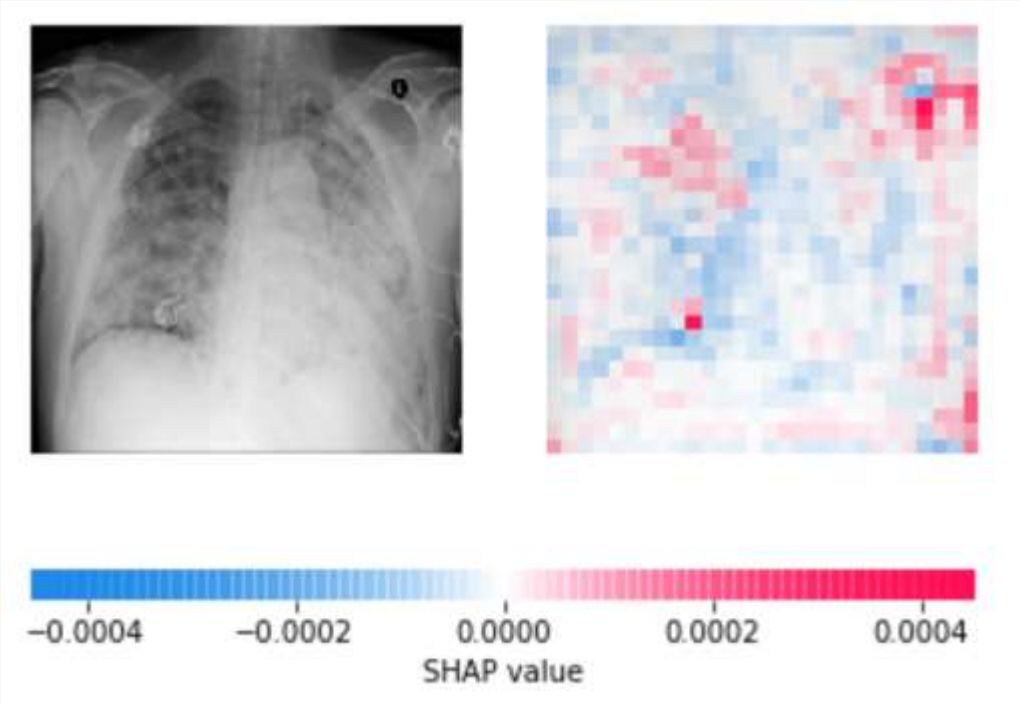
MLOps and the Lakehouse

- Applying open tools in-place to data in the lakehouse is a win for *training*
- Applying them for *operating* models is important too!
- "Models are data too"
- Need to apply models to *data*
- **MLFlow** for MLOps on the lakehouse
  - Track and manage model data, lineage, inputs
  - Deploy models as lakehouse "services"

Example: Chest X-Ray Classification

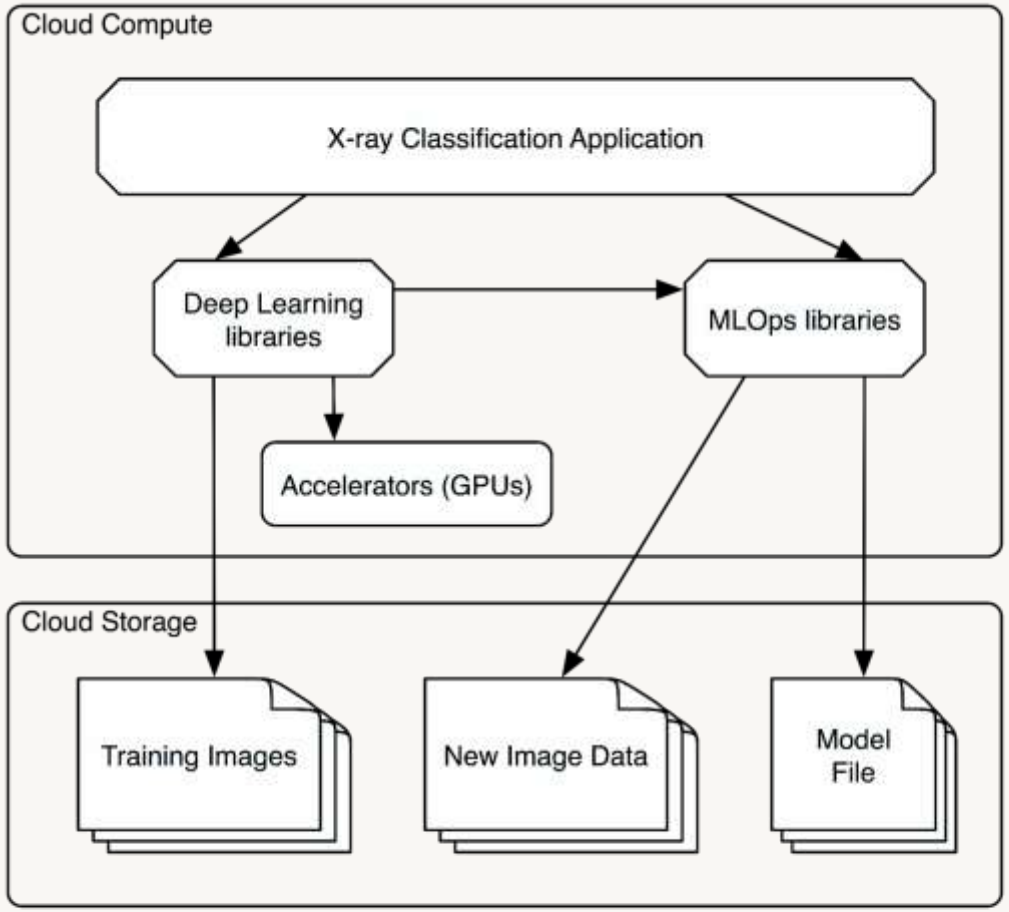
Classifying Chest X-rays

- 45,000 X-ray images
  - About 50GB
  - Includes correct doctor diagnosis
  - From National Institute of Health
- Relatively easy deep learning problem
  - If you have access to the data
  - If you have deep learning open source software
  - If you have GPU hardware
- *(Don't diagnose at home!)*



https://databricks.com/p/webinar/operationalizing-machine-learning-at-scale

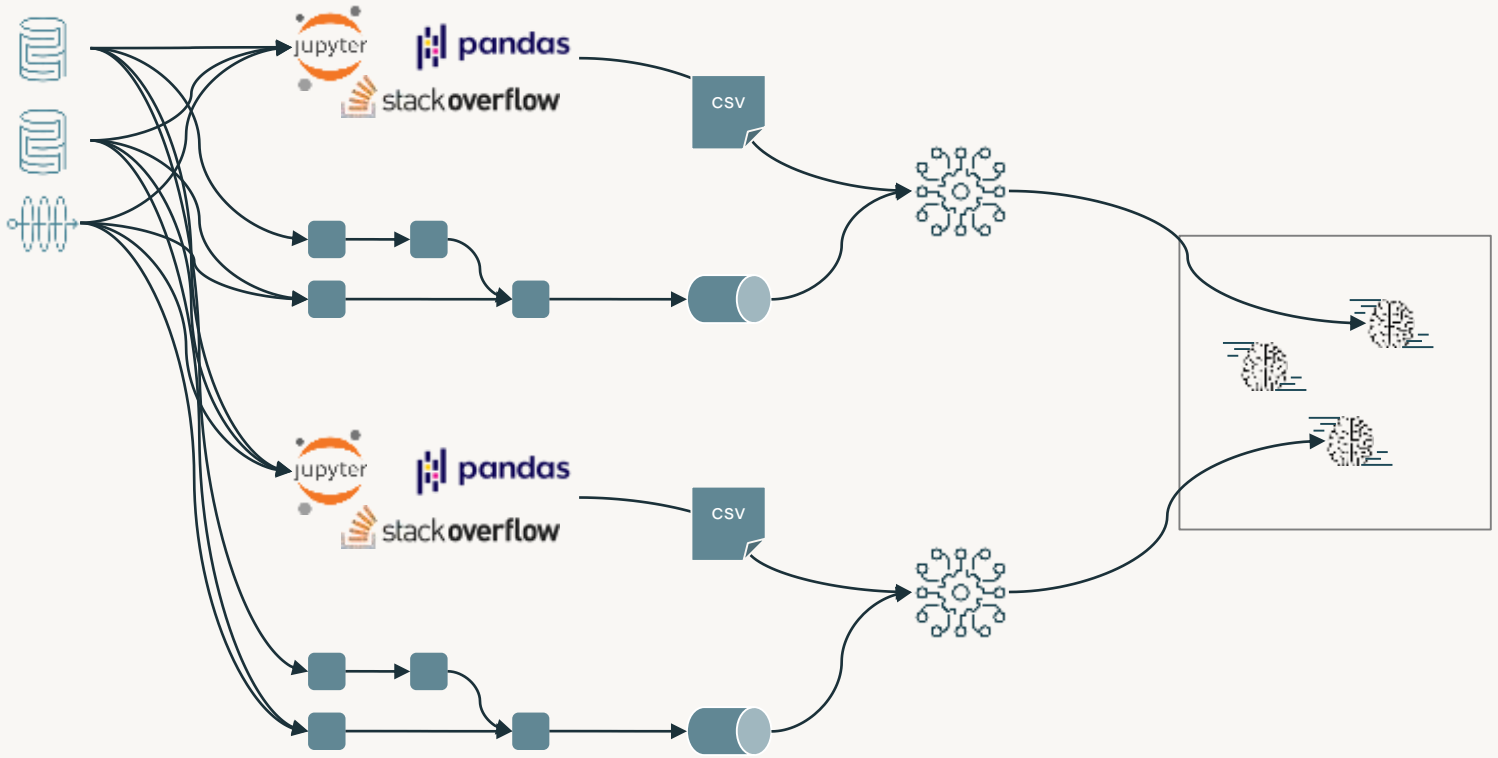
Architecture



Feature Stores

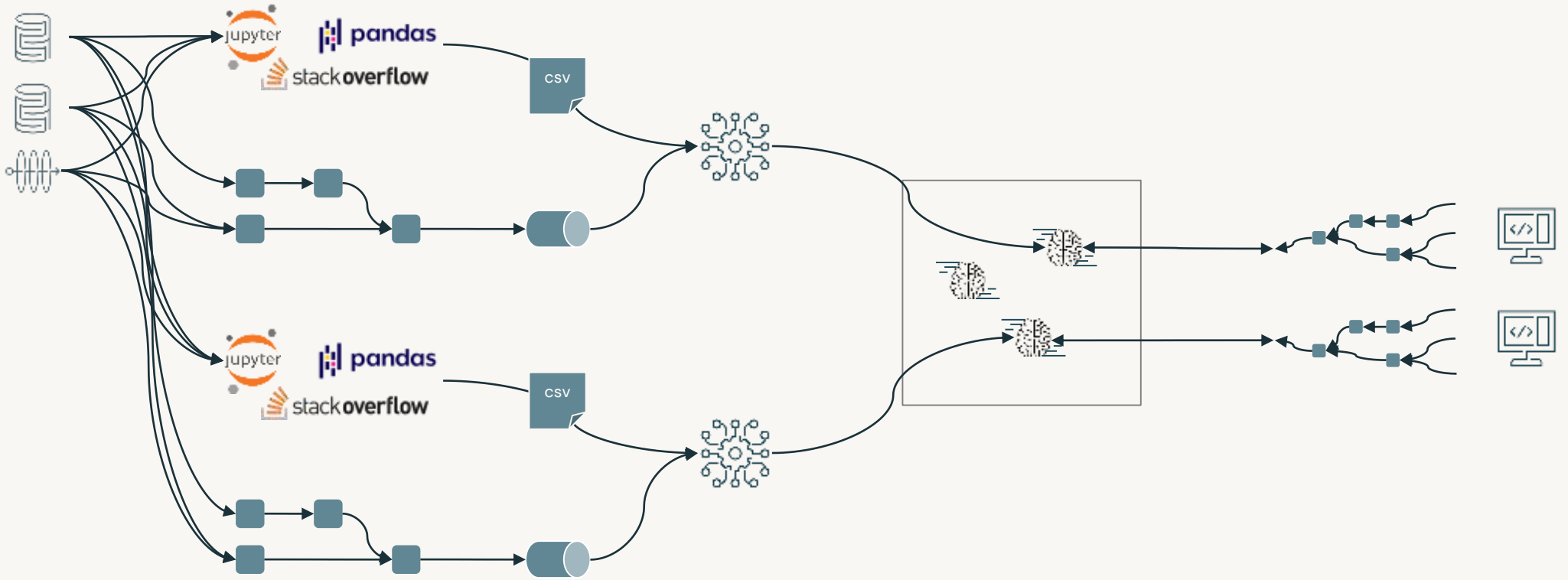
A day (or 6 months) in the life of an ML model

Raw Data	Featurization Joins, Aggregates, Transforms, etc.	Training	Serving



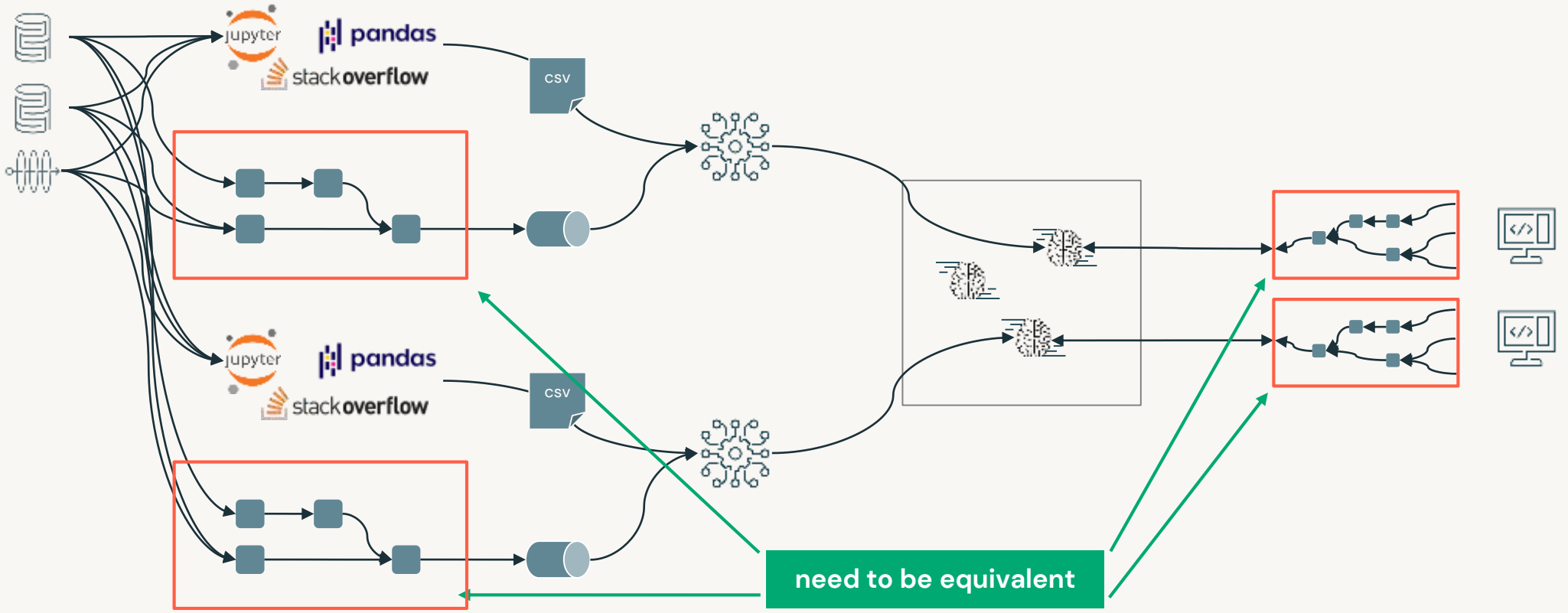
A day (or 6 months) in the life of an ML model

Raw Data	Featurization Joins, Aggregates, Transforms, etc.	Training	Serving	Client



A day (or 6 months) in the life of an ML model

Raw Data	Featurization Joins, Aggregates, Transforms, etc.	Training	Serving	Client



Feature Stores for Model Inputs

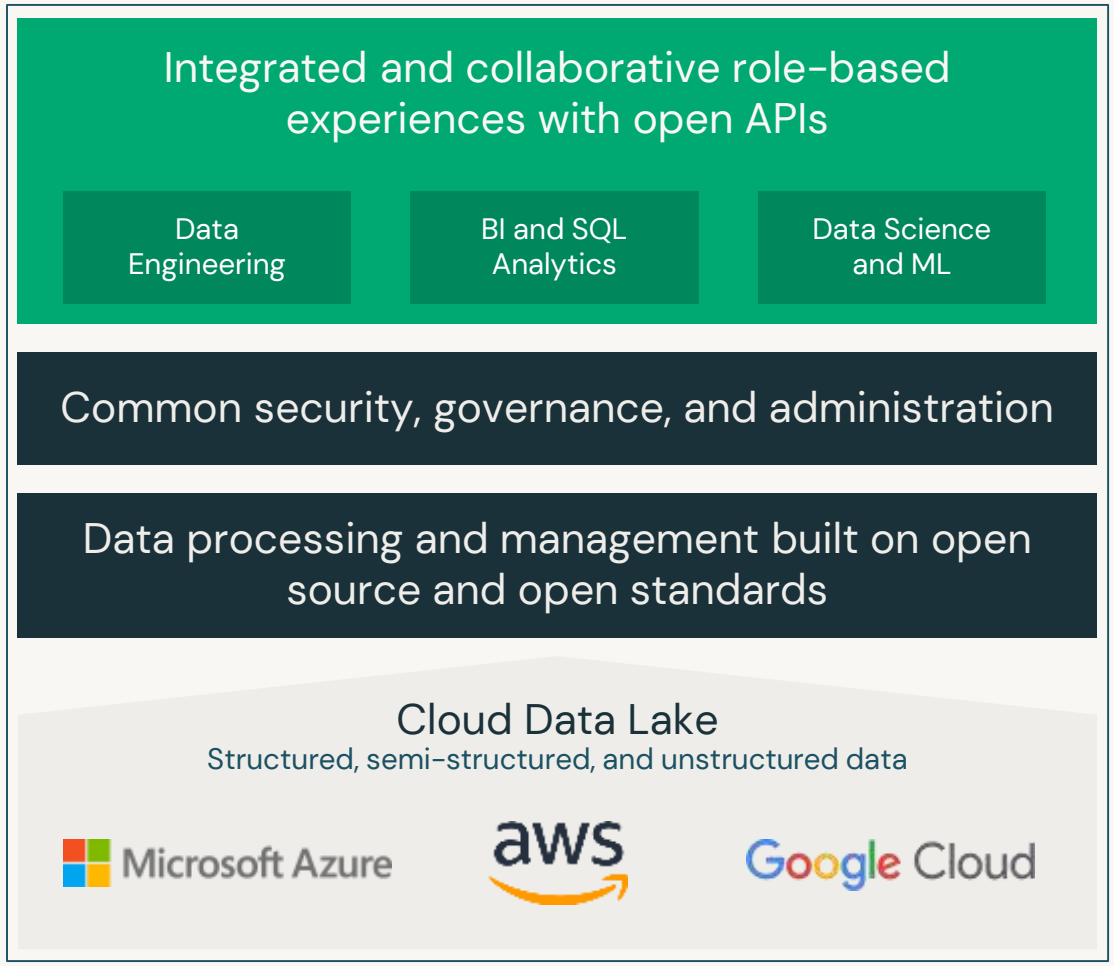
- Tables are OK for managing model input
  - Input often structured
  - Well understood, easy to access
- … but not quite enough
  - Upstream lineage: how were features computed?
  - Downstream lineage: where is the feature used?
  - Model caller has to read, feed inputs
  - How to do (also) access in real time?

Example: Telco Churn Classification

Bonus: Auto ML

Learn More

The data lakehouse offers a better path



Lake-first approach that builds upon where the freshest, most complete data resides

AI/ML from the ground up

High reliability and **performance**

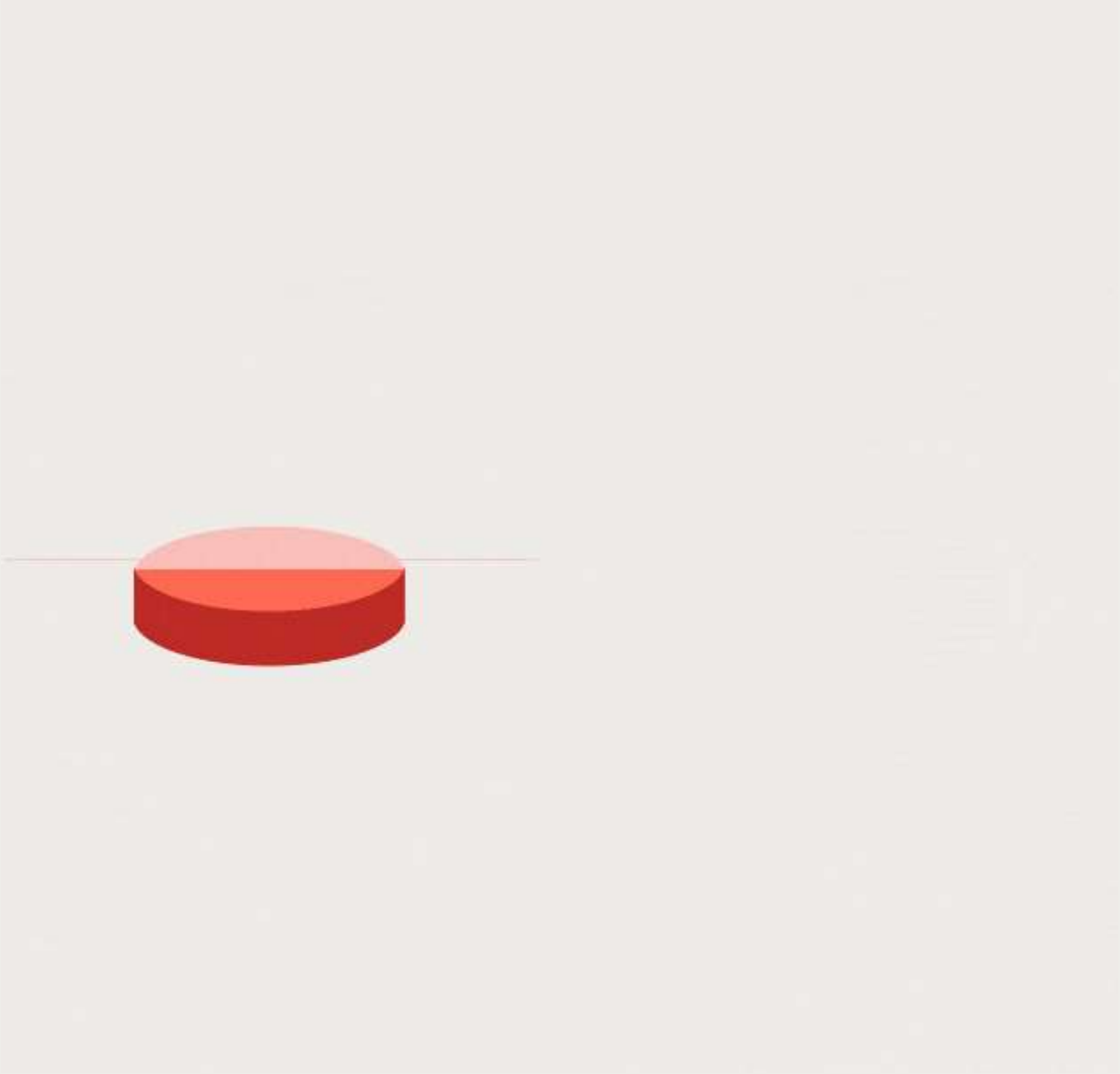
Single approach to managing data

Support for all use cases on a single platform:

- Data engineering
- Data warehousing
- Real time streaming
- Data science and ML

Built on **open source** and open standards

Multi-cloud, work with your cloud of choice



Thank you