databricks

Data Engineering with Databricks

Course Objectives

- Leverage the Databricks Lakehouse Platform to perform core responsibilities for data pipeline development
- Use SQL and Python to write production data pipelines to extract, transform, and load data into tables and views in the lakehouse
- Simplify data ingestion and incremental change propagation using Databricks-native features and syntax
- Orchestrate production pipelines to deliver fresh results for ad-hoc analytics and dashboarding

Course Agenda

- Module 1: Databricks Workspace and Services
- Module 2: Delta Lake
- Module 3: Relational Entities on Databricks
- Module 4: ETL With Spark SQL
- Module 5: OPTIONAL Python for Spark SQL
- Module 6: Incremental Data Processing
- Module 7: Multi-Hop Architecture
- Module 8: Delta Live Tables
- Module 9: Task Orchestration with Jobs
- Module 10: Running a DBSQL Query
- Module 11: Managing Permissions
- Module 12: Productionalizing Dashboards and Queries in DBSQL

Who Am I

Please introduce yourself and put it into zoom chat

Bo Frank Zhang

Sydney, Australia

Senior Technical Instructor

@Databricks

Spark, Data Warehouse

LinkedIn: /in/bo-frank-cloud-man

Who Am I

Bo Frank Zhang

Sydney, Australia

Senior Technical Instructor

@Databricks

Spark, Data Warehouse

LinkedIn: /in/bo-frank-cloud-man

The Databricks Lakehouse Platform

Using the Databricks Lakehouse Platform

Module 1 Learning Objectives

- Describe the components of the Databricks Lakehouse
- Complete basic code development tasks using services of the Databricks Data Science and Engineering Workspace
- Perform common table operations using Delta Lake in the Lakehouse

Using the Databricks Lakehouse Platform

Module 1 Agenda

- Introduction to the Databricks Lakehouse Platform
- Introduction to the Databricks Workspace and Services
  - Using clusters, files, notebooks, and repos
  - Lab & Breaks - Get Started with Databricks
- Introduction to Delta Lake
  - Manipulating and optimizing data in Delta tables
  - Labs & Breaks - Manipulating Delta tables & Delta Lake Versioning

Lakehouse

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

Customers

7000+

across the globe

Original creators of:

Apache Spark™ · Delta Lake · mlflow™

Supporting enterprises in every industry

Healthcare & Life Sciences

Humana, gsk, AMGEN, OPTUM, NHS, Biogen, REGENERON, AstraZeneca

Manufacturing & Automotive

Schneider Electric, Rolls-Royce, DAIMLER, COLAS, ZEISS, thyssenkrupp, BECHTEL

Media & Entertainment

CONDÉ NAST, RIOT GAMES, COMCAST, SHOWTIME, viacom, PUBLICIS GROUPE, zynga

Financial Services

HSBC, BNP PARIBAS, Nasdaq, Nationwide, S&P Global, CREDIT SUISSE, ABN·AMRO

Public Sector

cfpb, CMS, U.S. Citizenship and Immigration Services, LOS ANGELES FIRE DEPARTMENT, New York Power Authority

Retail & CPG

sam's club, CVSHealth, Foot Locker, 7 ELEVEN, Henkel, MARS, H&M

Energy & Utilities

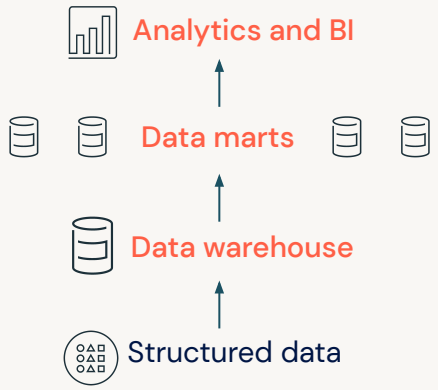
TOTAL, devon, ExxonMobil, aggreko, Quby

Digital Native

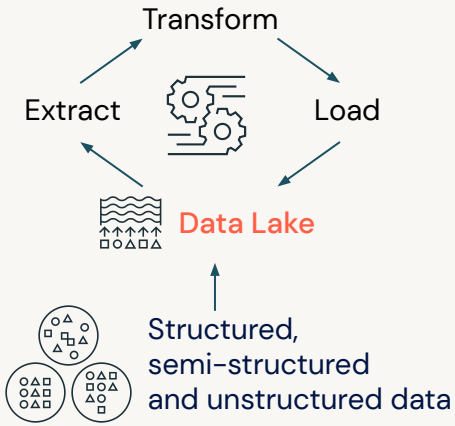
gousto, DOLLAR SHAVE CLUB, Grab, zalando, BUTCHER BOX, wehkamp, SCRIBD

Most enterprises struggle with data

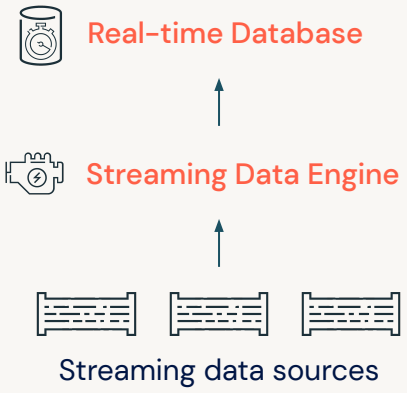
Data Warehousing



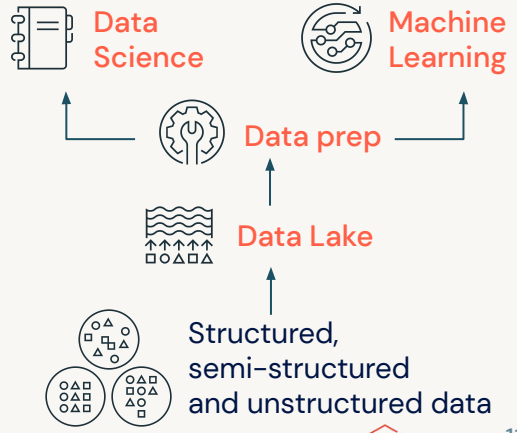
Data Engineering



Streaming



Data Science and ML



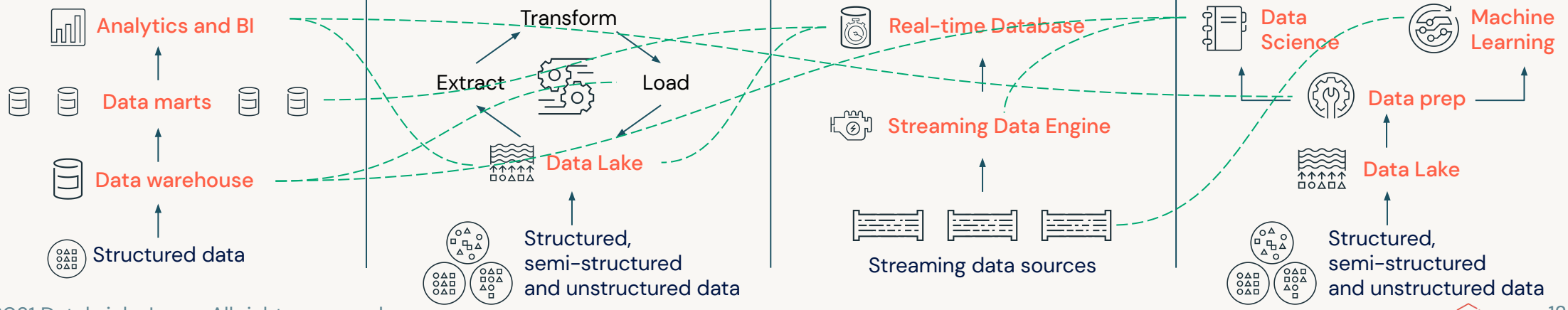
Siloed stacks increase data architecture complexity

Most enterprises struggle with data

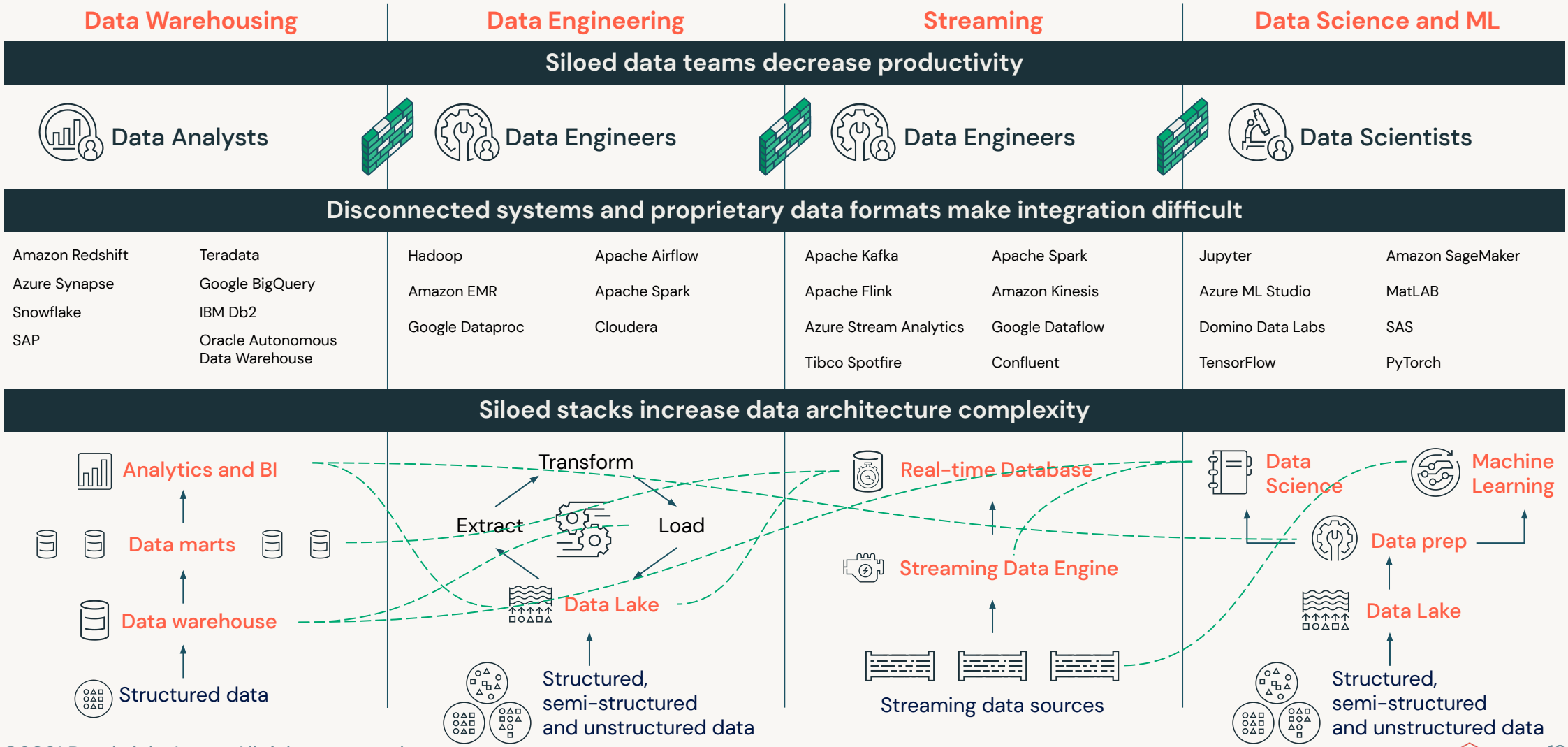
Data Warehousing		Data Engineering		Streaming		Data Science and ML	
Amazon Redshift	Teradata	Hadoop	Apache Airflow	Apache Kafka	Apache Spark	Jupyter	Amazon SageMaker
Azure Synapse	Google BigQuery	Amazon EMR	Apache Spark	Apache Flink	Amazon Kinesis	Azure ML Studio	MatLAB
Snowflake	IBM Db2	Google Dataproc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

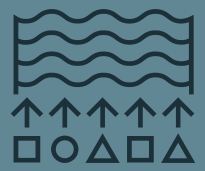
Disconnected systems and proprietary data formats make integration difficult

Siloed stacks increase data architecture complexity



Most enterprises struggle with data



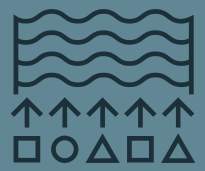


Data Lake

Lakehouse

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

Data Warehouse



Data Lake

DELTA LAKE

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

Better reliability with transactions

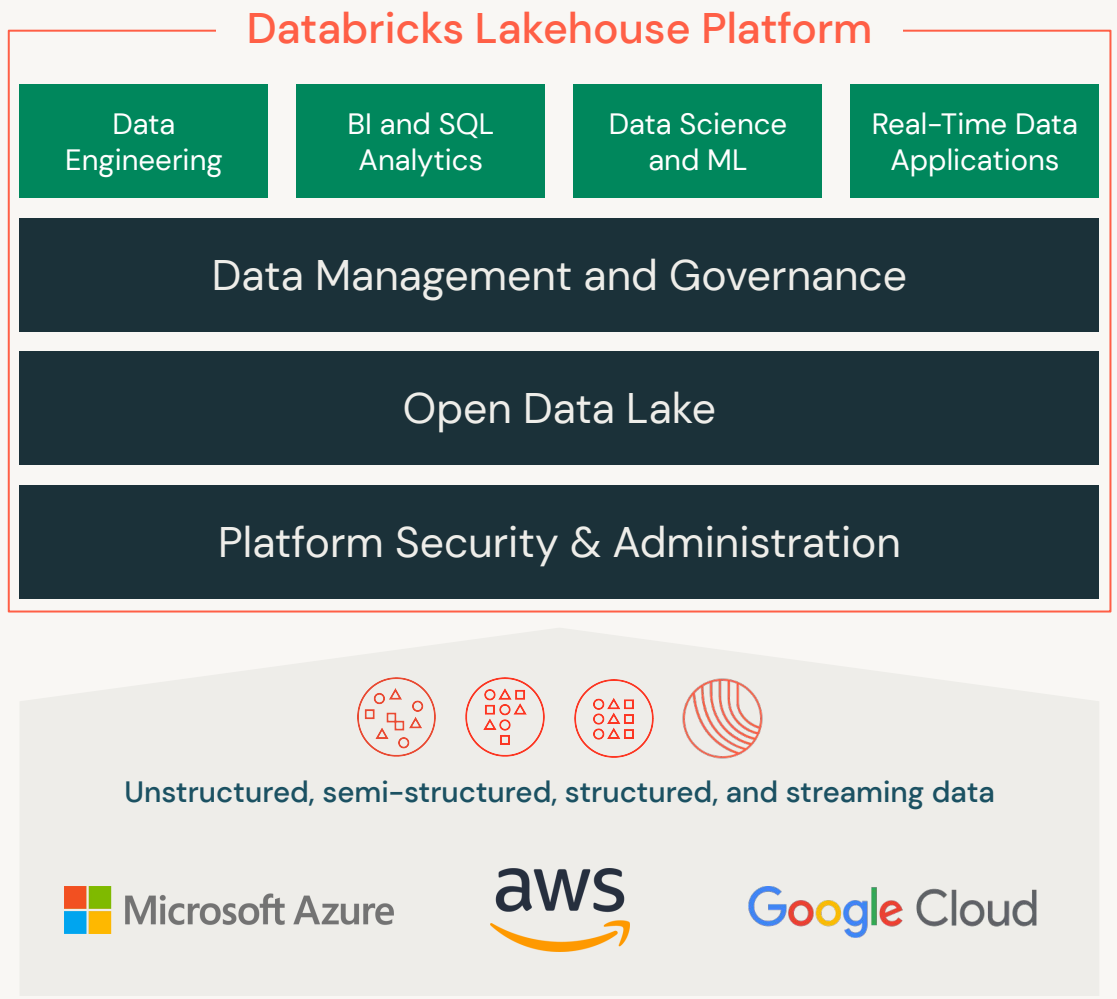
48x faster data processing with indexing

Data governance at scale with fine-grained access control lists

Data Warehouse

The Databricks Lakehouse Platform

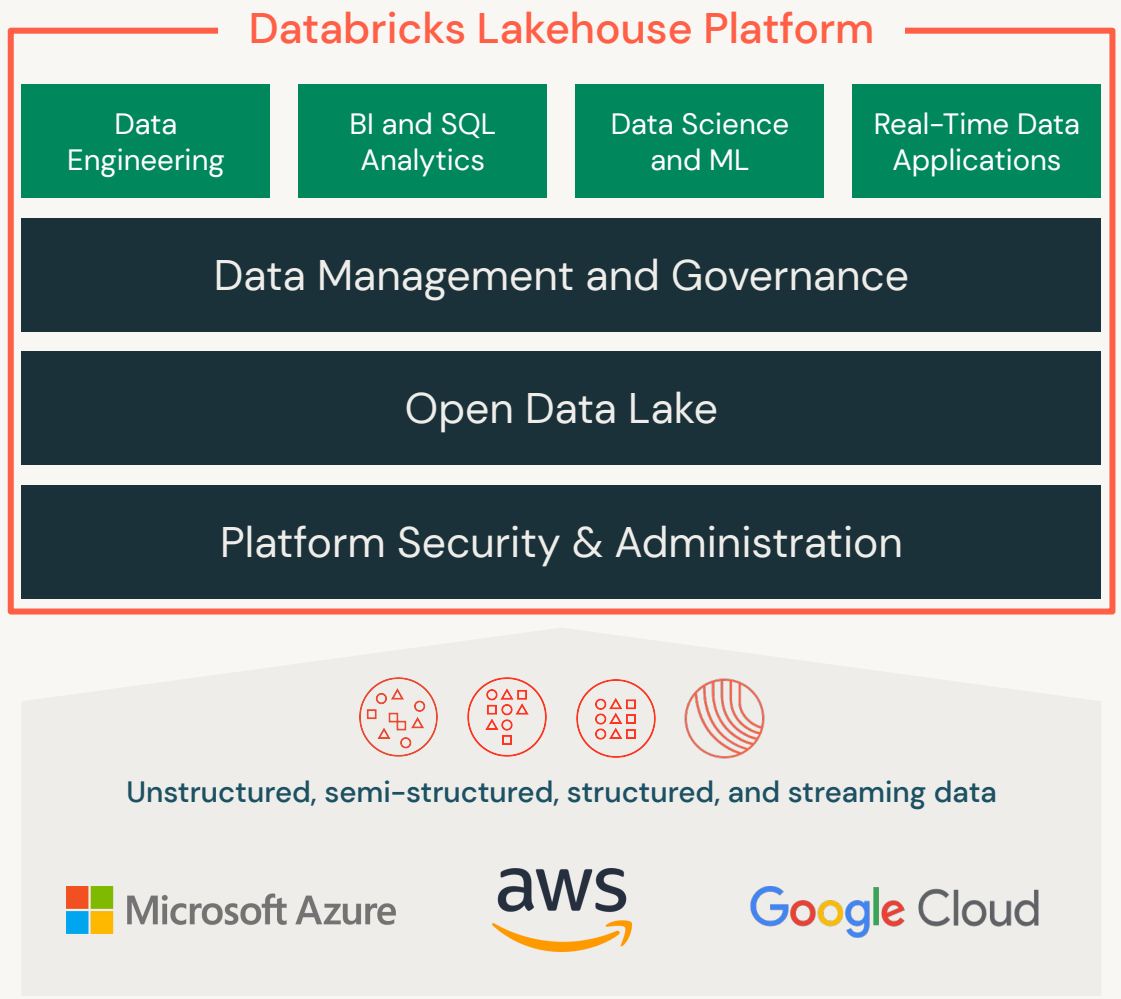
- Simple
- Open
- Collaborative



The Databricks Lakehouse Platform

Simple

Unify your data, analytics, and AI on one common platform for all data use cases



The Databricks Lakehouse Platform

Open

Unify your data ecosystem with open source standards and formats.

Built on the innovation of some of the most successful open source data projects in the world

30 Million+

Monthly downloads

The Databricks Lakehouse Platform

Open

Unify your data ecosystem with open source standards and formats.

450+

Partners across the data landscape

Visual ETL & Data Ingestion

Azure Data Factory, Fivetran, Confluent, Informatica, talend, Qlik, StreamSets, precisely, Infoworks, WANdisco, Trifacta, dbt, mongoDB., Matillion

Business Intelligence

tableau, Power BI, Azure Synapse, Qlik, plotly, looker, Google BigQuery, ThoughtSpot, TIBCO Spotfire, Amazon Redshift

Machine Learning

Amazon SageMaker, Azure Machine Learning, MathWorks, Google AI Platform, RStudio, H2O.ai

databricks
Lakehouse Platform
Microsoft Azure, aws, Google Cloud

Data Providers

dun & bradstreet, SAFEGRAPH, experian., yipitDATA, arm INSIGHT

Centralized Governance

AWS Glue, Informatica, Alation, PRIVACERA, IMMUTA, collibra

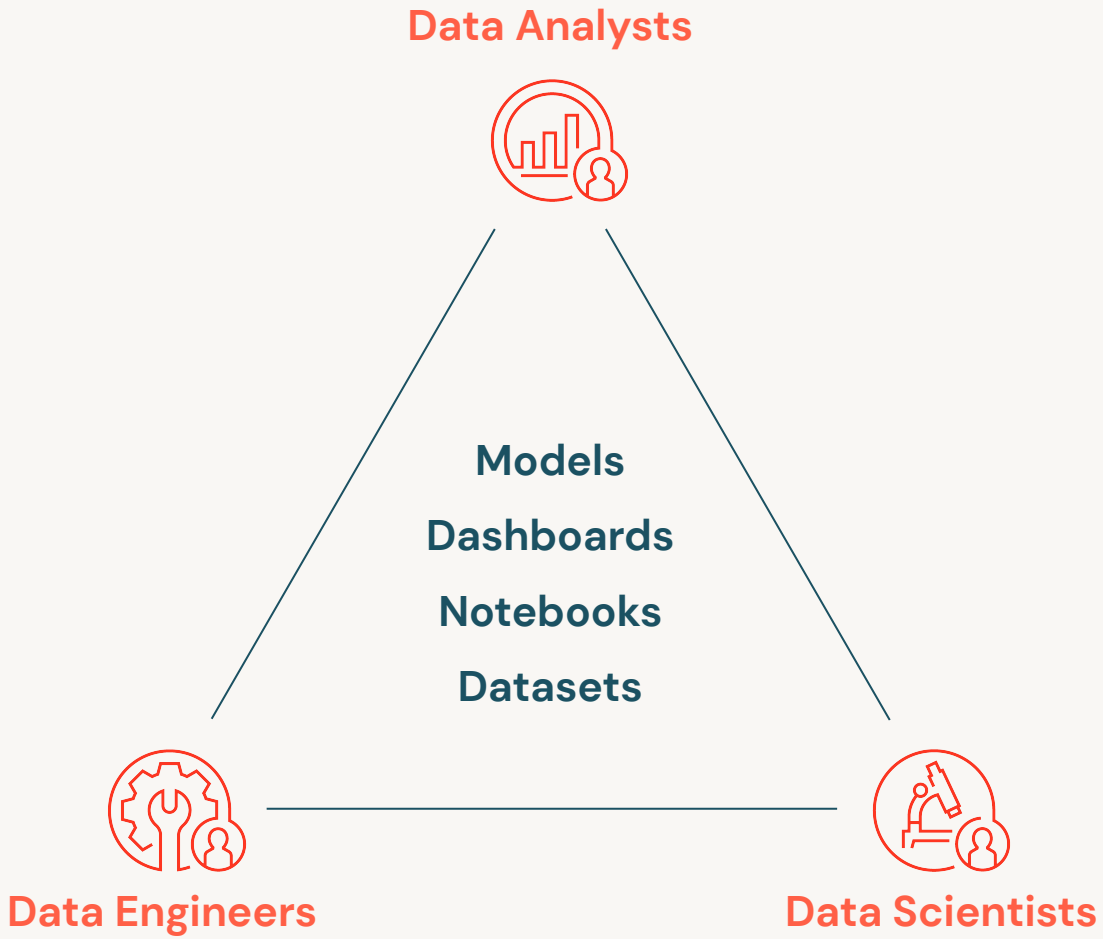
Top Consulting & SI Partners

Insight, NEUDESIC, PARIVEDA, accenture, avanade, wipro, Infosys, Tech Mahindra, GO DATA DRIVEN, ZEAL, AGILISIUM, BLUEGRANITE, Hitachi Solutions, ONICA, Capgemini, Cognizant, Deloitte., ELASTACLOUD, OPENVALUE, b.telligent, kumulus, BlueShift, deep_dive, Mindtree, slalom, TATA CONSULTANCY SERVICES, JUST ANALYTICS, Intellify, Version

The Databricks Lakehouse Platform

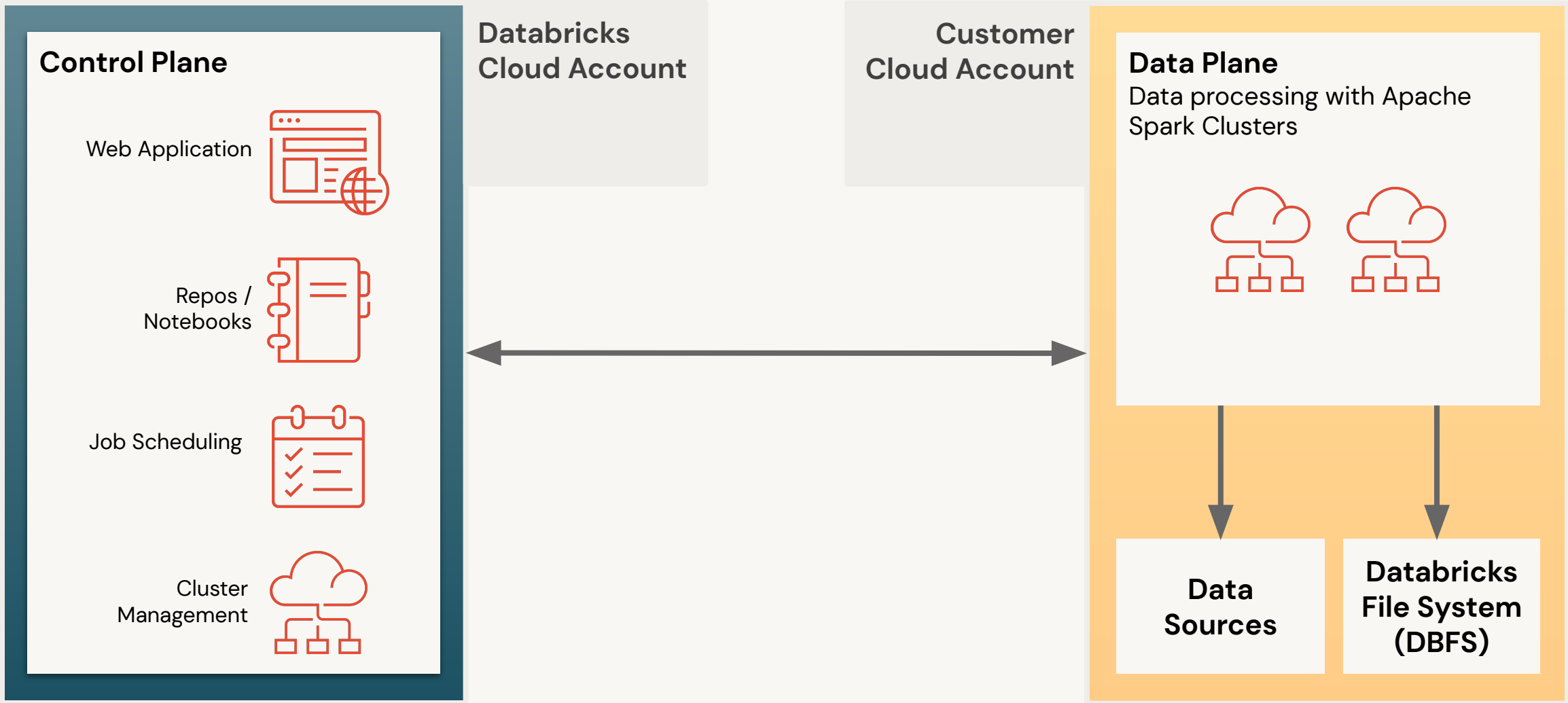
Collaborative

Unify your data teams to collaborate across the entire data and AI workflow



Databricks Architecture and Services

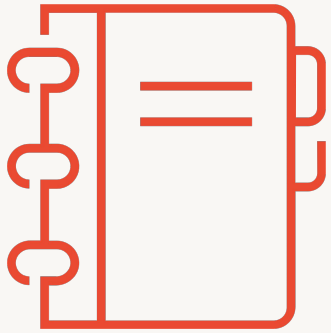
Databricks Architecture



Databricks Services

Control Plane in Databricks
Manage customer accounts, datasets, and clusters

Databricks Web Application

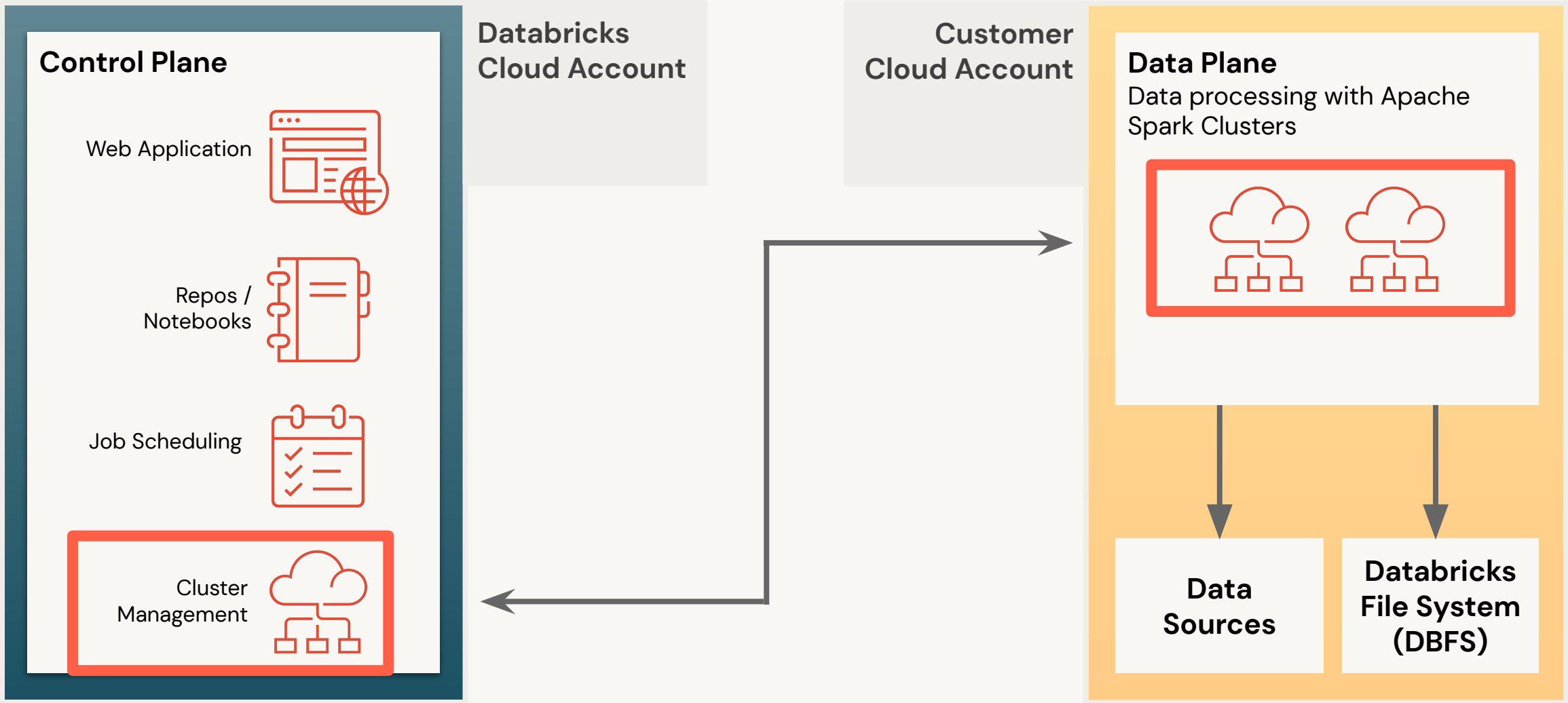


Repos / Notebooks

Jobs

Cluster Management

Clusters



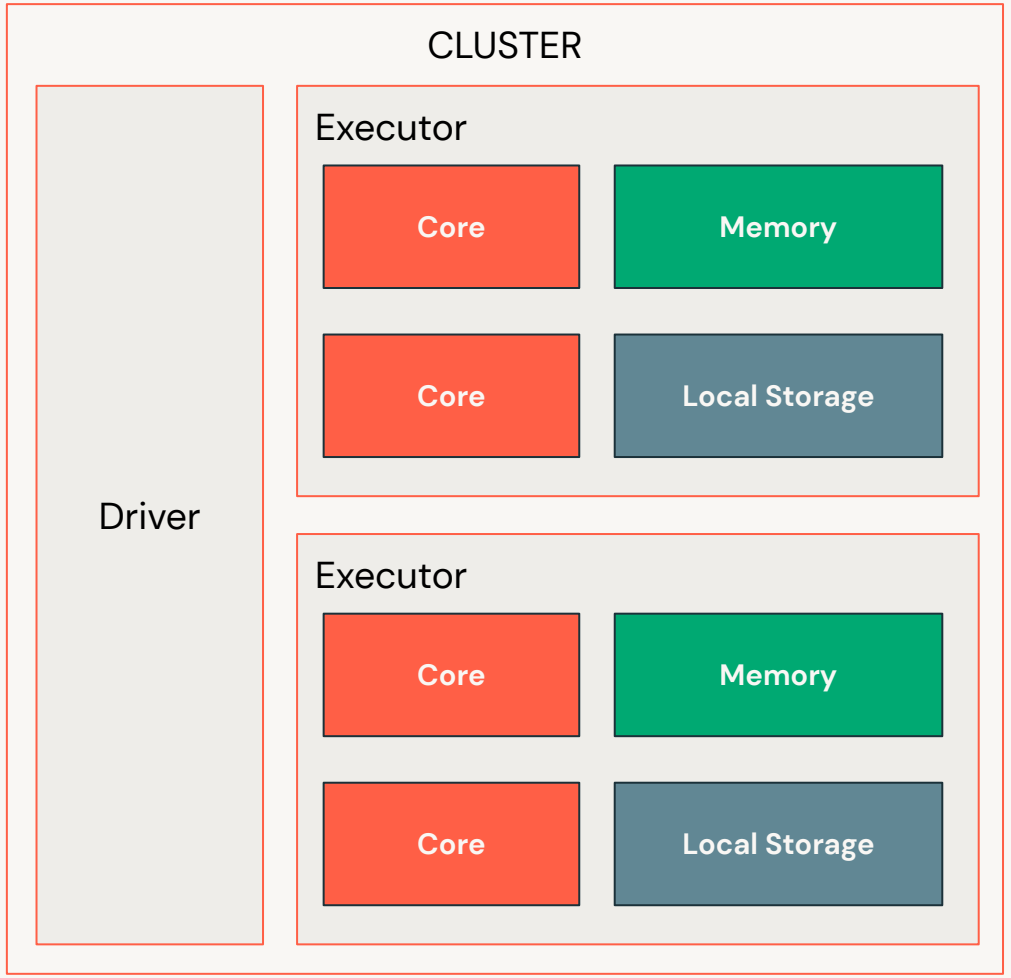
Clusters

Overview

Clusters are made up of one or more virtual machine (VM) instances

Driver coordinates activities of executors

Executors run tasks composing a Spark job



Clusters

Types

All-purpose Clusters

Analyze data collaboratively using interactive notebooks

Create clusters from the Workspace or API

Retains up to 70 clusters for up to 30 days.

Job Clusters

Run automated jobs

The Databricks job scheduler creates job clusters when running jobs.

Retains up to 30 clusters.

Getting Access to workspace

Creating a Workspace Guide

Step 1: Registration

Step 2: Launching the workspace

Step 3: Accessing the workspace

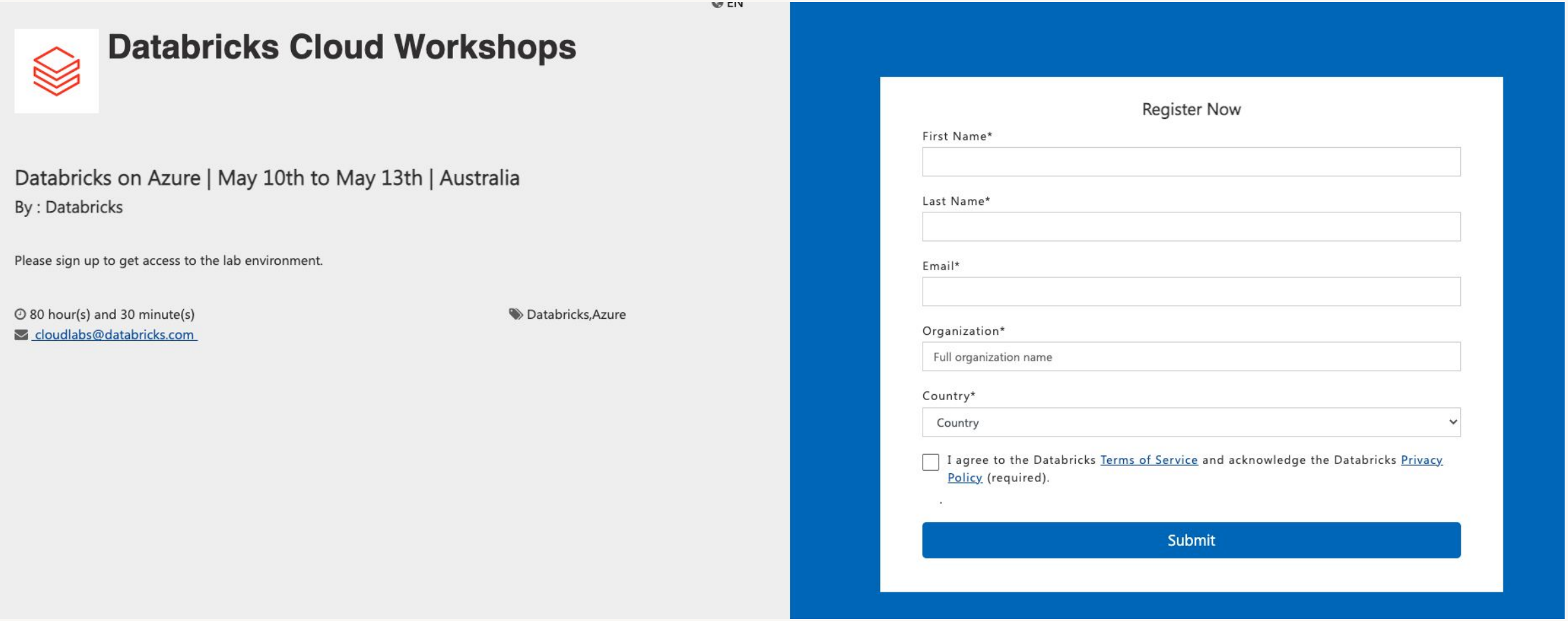
Step 4–5: Signing in

Arrive in your Databricks workspace

Getting back into the workspace

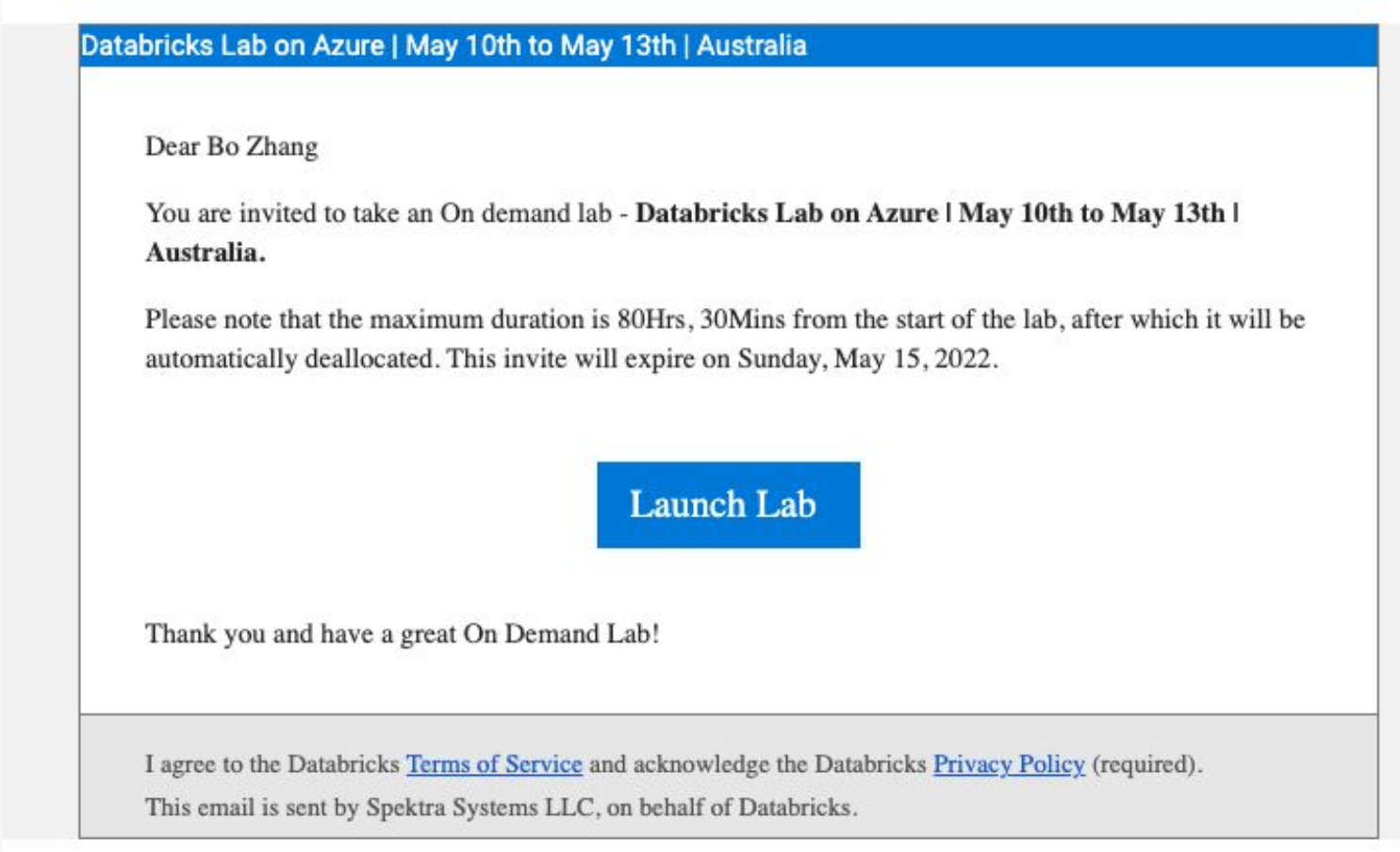
Access Databricks Workspace

Registration email From CLoudLabs



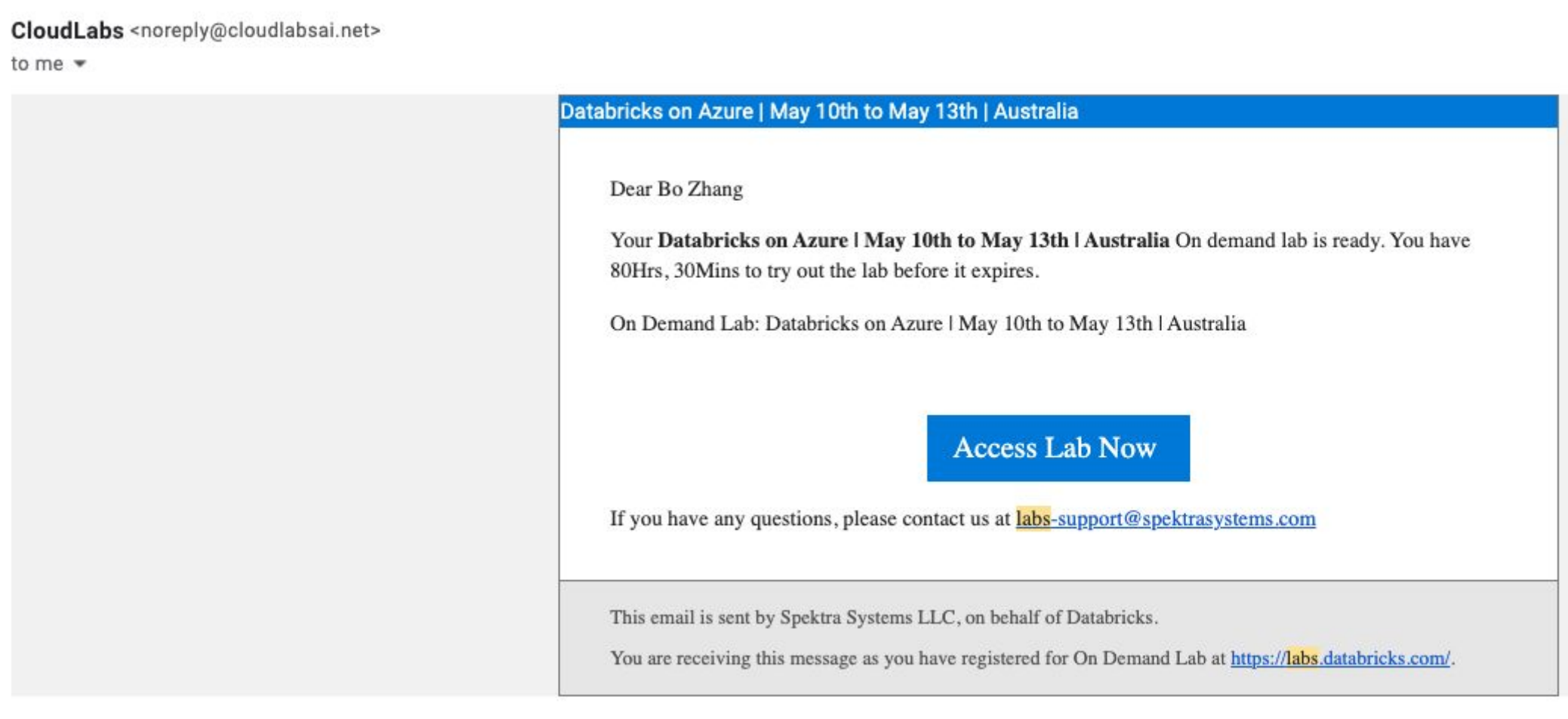
Access Databricks Workspace

Launch Lab



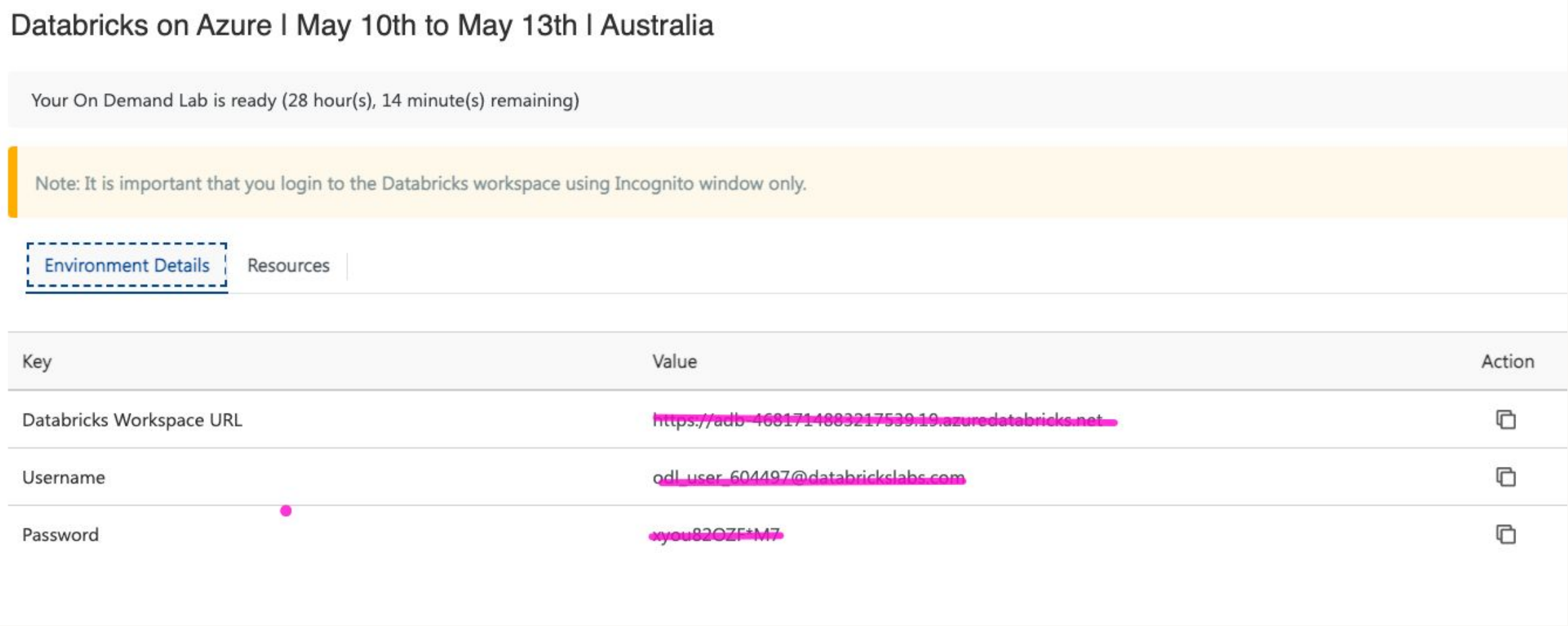
Access Databricks Workspace

Access Lab Now



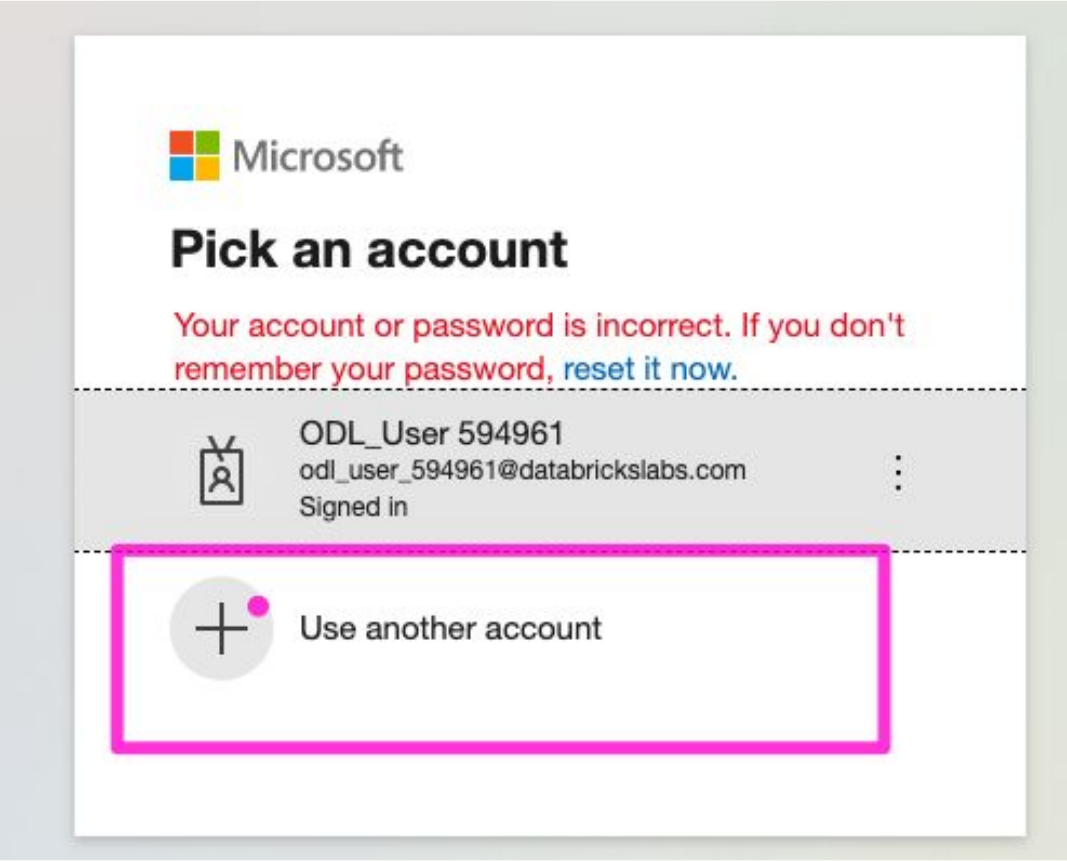
Access Databricks Workspace

Find Environment Details via CLoudLabs



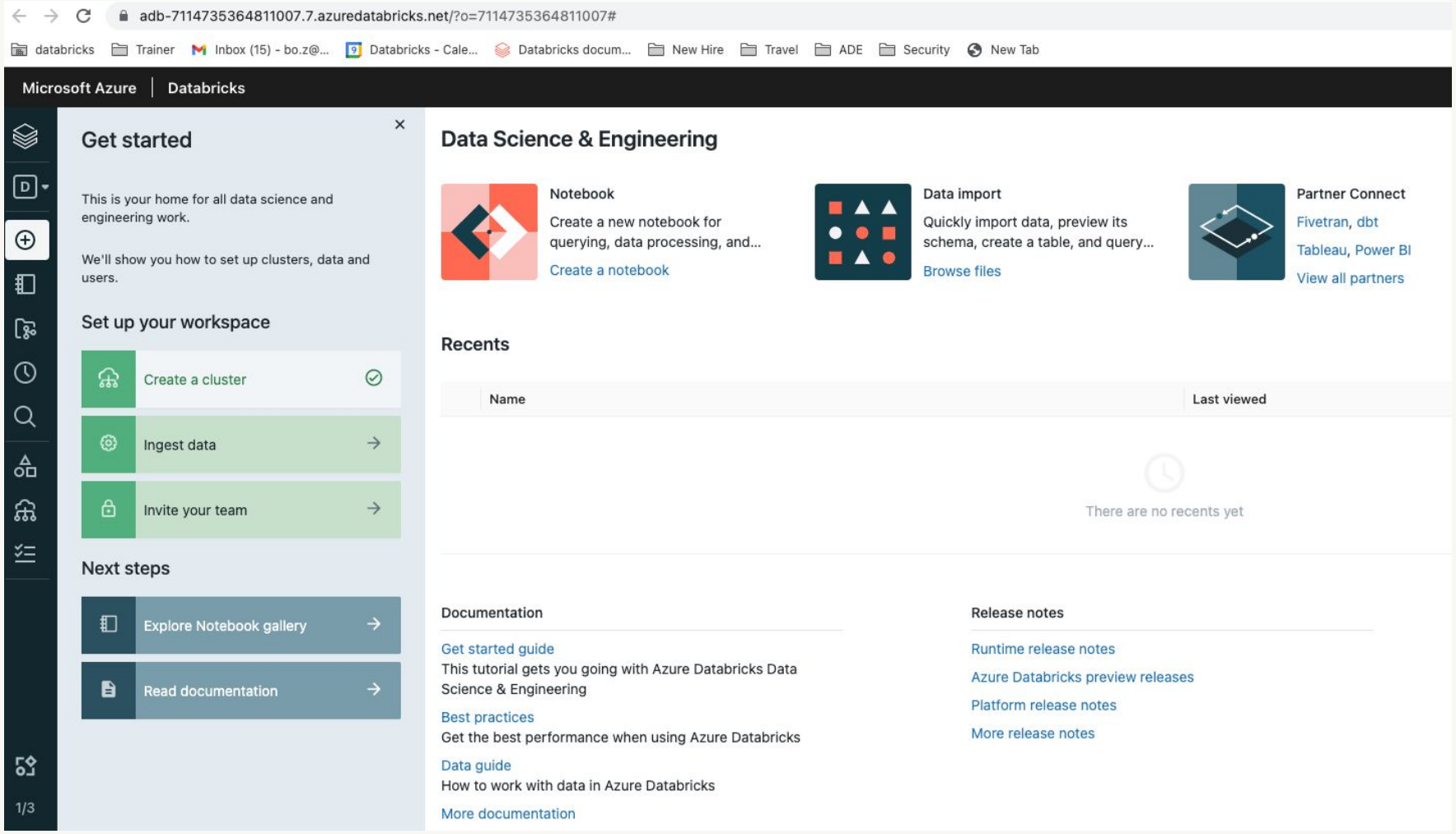
Access Databricks Workspace

via CLoudLabs



Access Databricks Workspace

via CLoudLabs



Git Versioning with Databricks Repos

Databricks Repos

Overview

Git Versioning

Native integration with Github, Gitlab, Bitbucket and Azure Devops

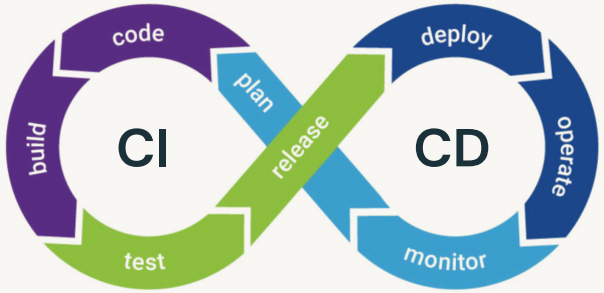
UI-based workflows

Azure DevOps GitHub GitLab Bitbucket

CI/CD Integration

API surface to integrate with automation

Simplifies the dev/staging/prod multi-workspace story



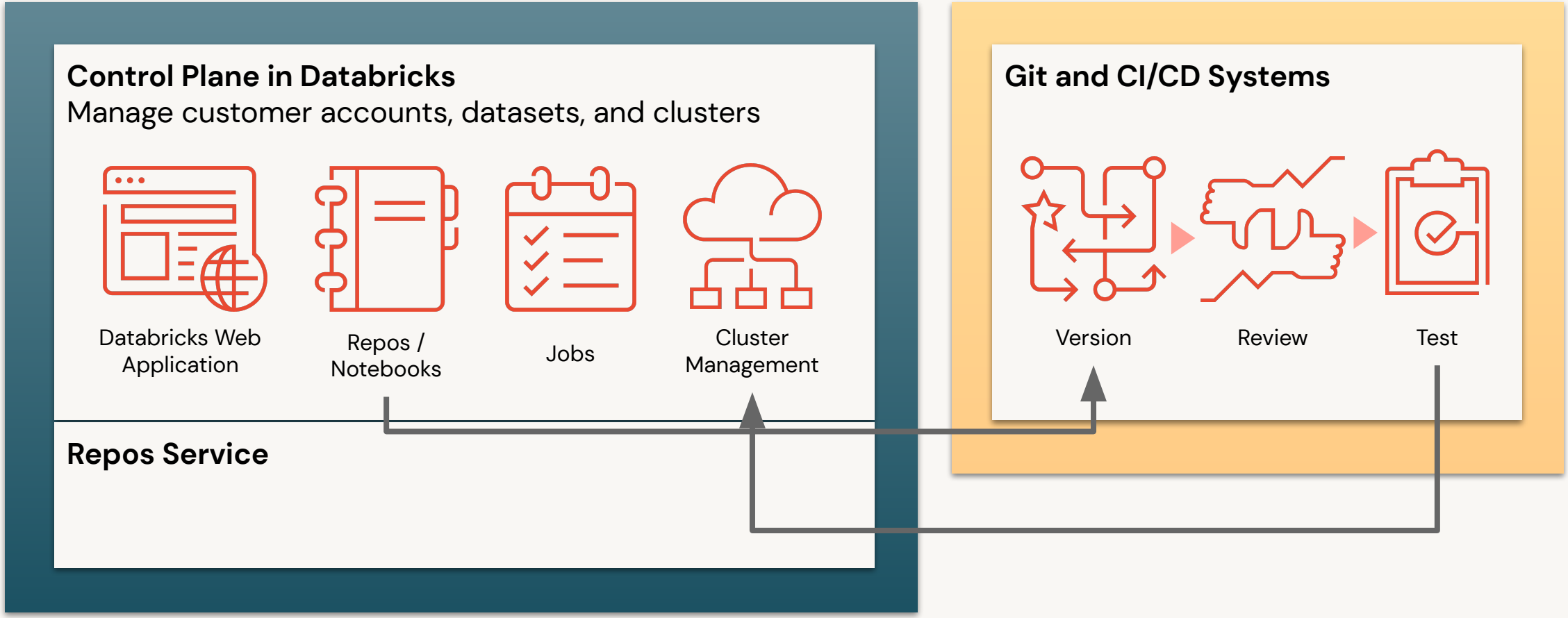
Enterprise ready

Allow lists to avoid exfiltration

Secret detection to avoid leaking keys

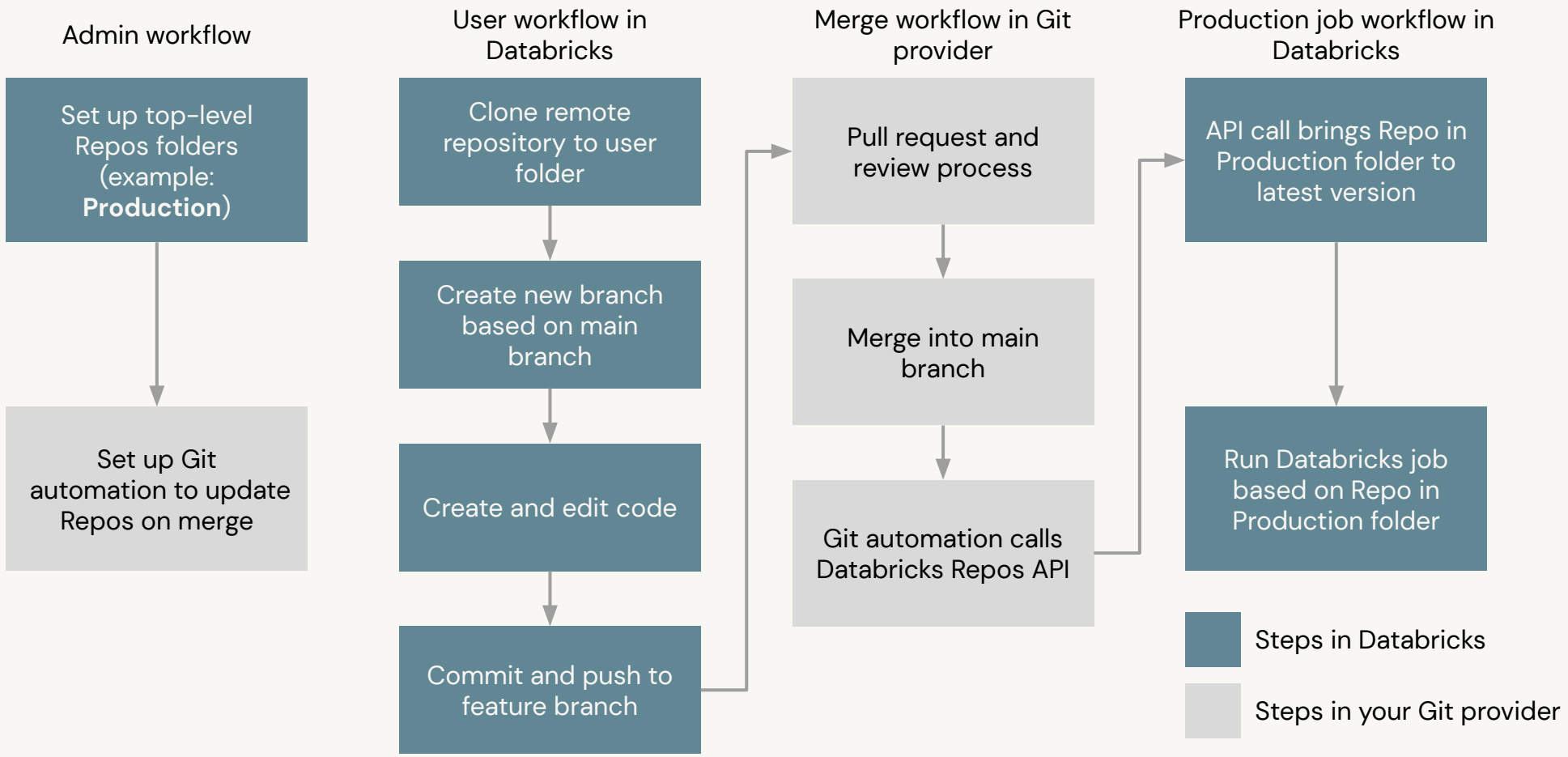
Databricks Repos

CI/CD Integration



Databricks Repos

Best practices for CI/CD workflows



databricks

What is Delta Lake?

Delta Lake is an open-source project that enables building a data lakehouse on top of existing storage systems

Delta Lake Is Not...

- Proprietary technology
- Storage format
- Storage medium
- Database service or data warehouse

Delta Lake Is...

- Open source
- Builds upon standard data formats
- Optimized for cloud object storage
- Built for scalable metadata handling

Delta Lake brings ACID to object storage

- **A**tomicity
- **C**onsistency
- **I**solation
- **D**urability

Problems solved by ACID

1. Hard to append data
2. Modification of existing data difficult
3. Jobs failing mid way
4. Real-time operations hard
5. Costly to keep historical data versions

Delta Lake is the default for all tables created in Databricks

databricks

Welcome to Day 2

ETL with Spark SQL and Python

ETL With Spark SQL and Python

Module 2 Learning Objectives

- Leverage Spark SQL DDL to create and manipulate relational entities on Databricks
- Use Spark SQL to extract, transform, and load data to support production workloads and analytics in the Lakehouse
- Leverage Python for advanced code functionality needed in production applications

ETL With Spark SQL and Python

Module 2 Agenda

- Working with Relational Entities on Databricks
  - Managing databases, tables, and views
  - Lab 3.3L - Databases, Tables, Views
- ETL with Spark SQL
  - Extracting data from external sources, loading and updating data in the lakehouse, and common transformations
  - Lab 4.5L & Lab 4.9L - Data Cleaning and Reshaping Data
  - (Optional)Just Enough Python for Spark SQL
    - Building extensible functions with Python-wrapped SQL

databricks

Incremental Data and Delta Live Tables

Incremental Data and Delta Live Tables

Module 3 Learning Objectives

- Incrementally process data to power analytic insights with Spark Structured Streaming and Auto Loader
- Propagate new data through multiple tables in the data lakehouse
- Leverage Delta Live Tables to simplify productionalizing SQL data pipelines with Databricks

Incremental Data and Delta Live Tables

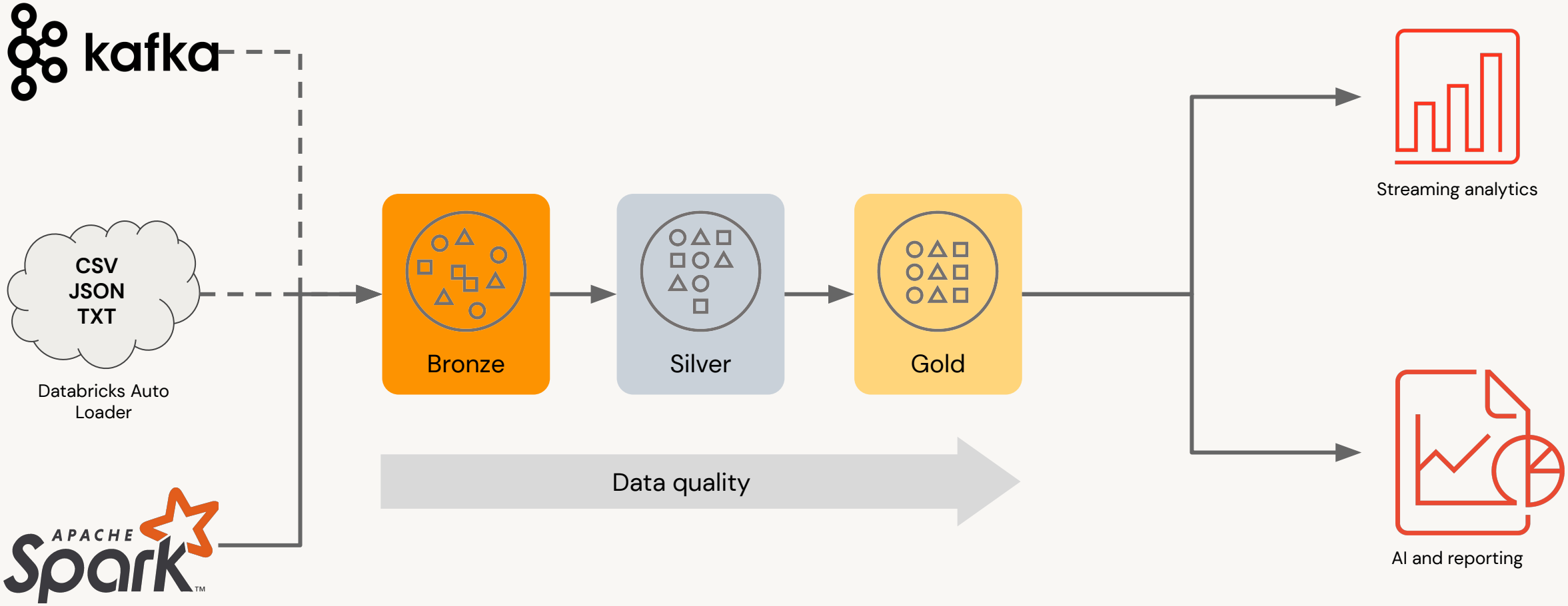
Module 3 Agenda

- Incremental Data Processing with Structured Streaming and Auto Loader
  - Processing and aggregating data incrementally in near real time
- Multi-hop in the Lakehouse
  - Propagating changes through a series of tables to drive production systems
- Using Delta Live Tables
  - Simplifying deployment of production pipelines and infrastructure using SQL

databricks

Multi-hop Architecture

Multi-Hop in the Lakehouse



Multi-Hop in the Lakehouse

Bronze Layer

Typically just a raw copy of ingested data

Replaces traditional data lake

Provides efficient storage and querying of full, unprocessed history of data

Multi-Hop in the Lakehouse

Silver Layer

Reduces data storage complexity, latency, and redundancy

Optimizes ETL throughput and analytic query performance

Preserves grain of original data (without aggregations)

Eliminates duplicate records

Production schema enforced

Data quality checks, corrupt data quarantined

Multi-Hop in the Lakehouse

Gold Layer

Powers ML applications, reporting, dashboards, ad hoc analytics

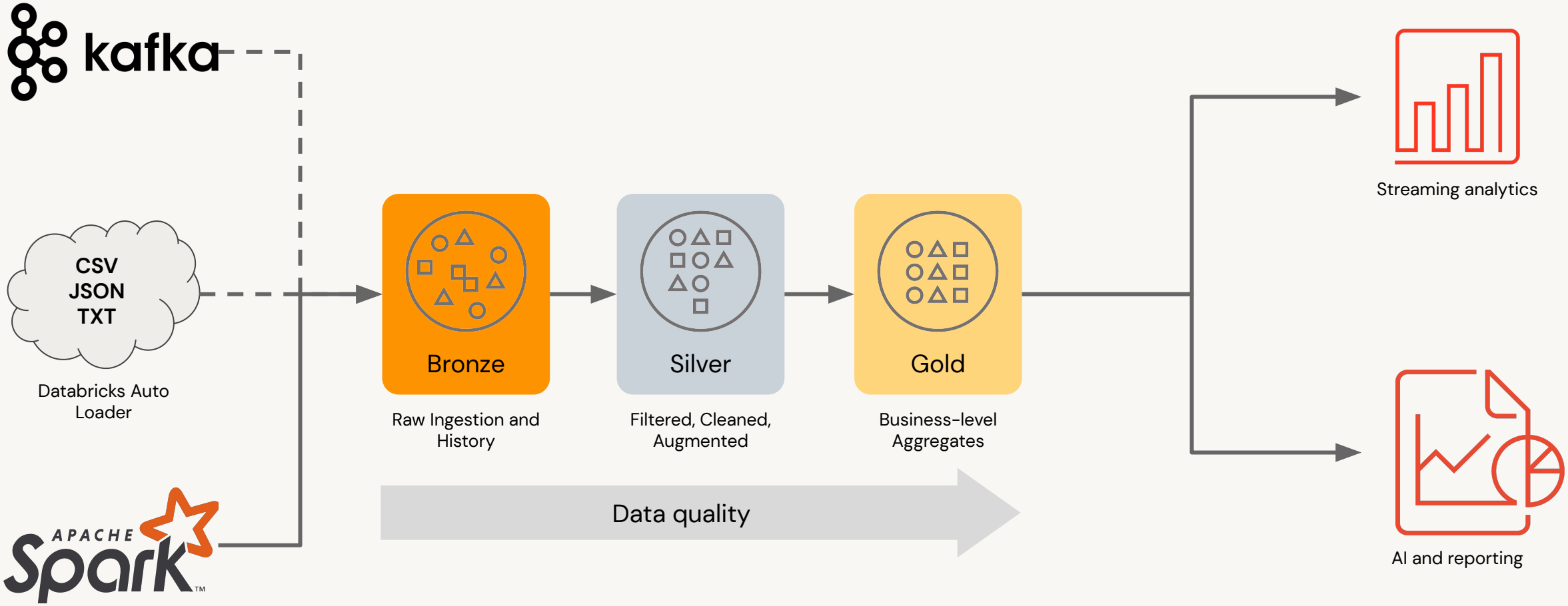
Refined views of data, typically with aggregations

Reduces strain on production systems

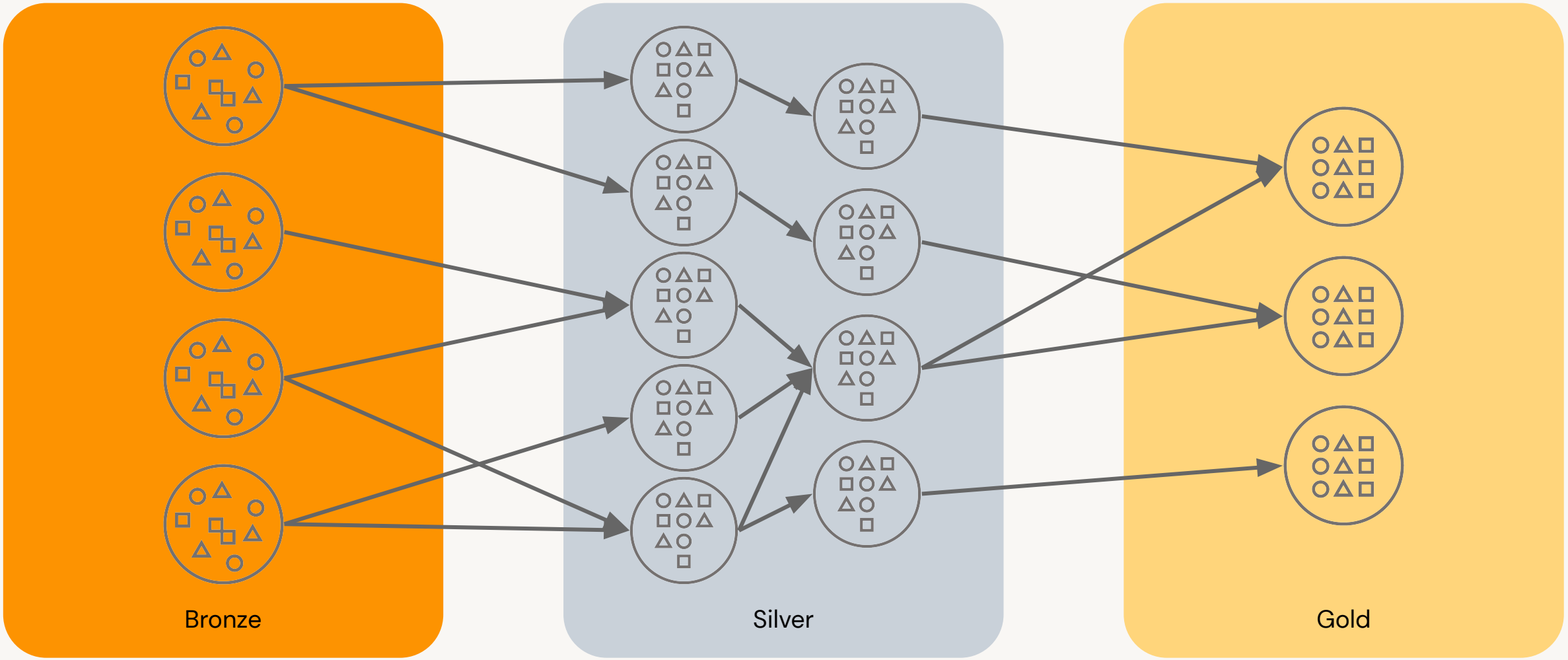
Optimizes query performance for business-critical data

Introducing Delta Live Tables

Multi-Hop in the Lakehouse



The Reality is Not so Simple



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

Data quality and governance

Difficult to monitor and enforce **data quality**

Impossible to trace data **lineage**

Difficult pipeline operations

Poor **observability** at granular, data level

Error handling and **recovery** is laborious

Introducing Delta Live Tables

Make reliable ETL easy on Delta Lake

Operate with agility

Declarative tools to build batch and streaming data pipelines

Trust your data

DLT has built-in declarative quality controls

Declare quality expectations and actions to take

Scale with reliability

Easily scale infrastructure alongside your data

databricks

Managing Data Access and Production Pipelines

Managing Data Access and Production Pipelines

Module 4 Learning Objectives

- Orchestrate tasks with Databricks Jobs
- Use Databricks SQL for on-demand queries
- Configure Databricks Access Control Lists to provide groups with secure access to production and development databases
- Configure and schedule dashboards and alerts to reflect updates to production data pipelines

Managing Data Access and Production Pipelines

Module 4 Agenda

- Task Orchestration with Databricks Jobs
  - Scheduling notebooks and DLT pipelines with dependencies
- Running Your First Databricks SQL Query
  - Navigating, configuring, and executing queries in Databricks SQL
- Managing Permissions in the Lakehouse
  - Configuring permissions for databases, tables, and views in the data lakehouse
- Productionalizing Dashboards and Queries in DBSQL
  - Scheduling queries, dashboards, and alerts for end-to-end analytic pipelines

Introducing Unity Catalog

Data Governance Overview

Four key functional areas

Data Access Control

Control who has access to which data

Data Access Audit

Capture and record all access to data

Data Lineage

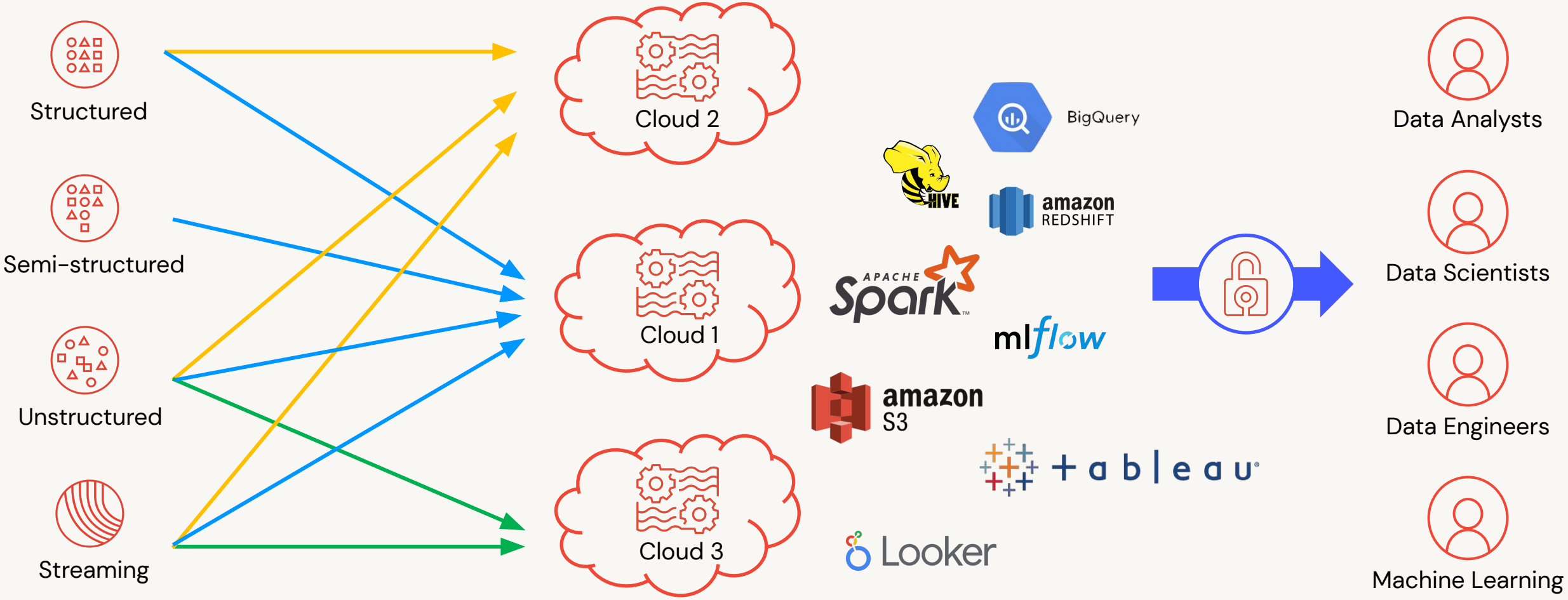
Capture upstream sources and downstream consumers

Data Discovery

Ability to search for and discover authorized assets

Data Governance Overview

Challenges



Databricks Unity Catalog

Overview

Unify governance across clouds

Fine-grained governance for data lakes across clouds - based on open standard ANSI SQL.

Unify data and AI assets

Centrally share, audit, secure and manage all data types with one simple interface.

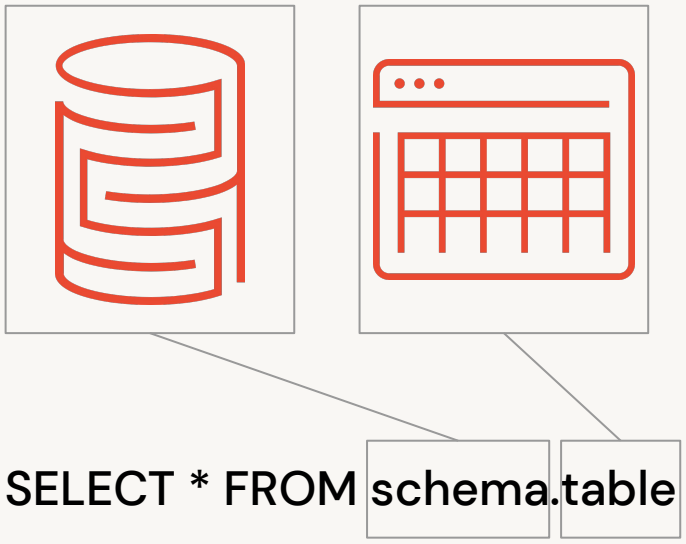
Unify existing catalogs

Works in concert with existing data, storage, and catalogs - no hard migration required.

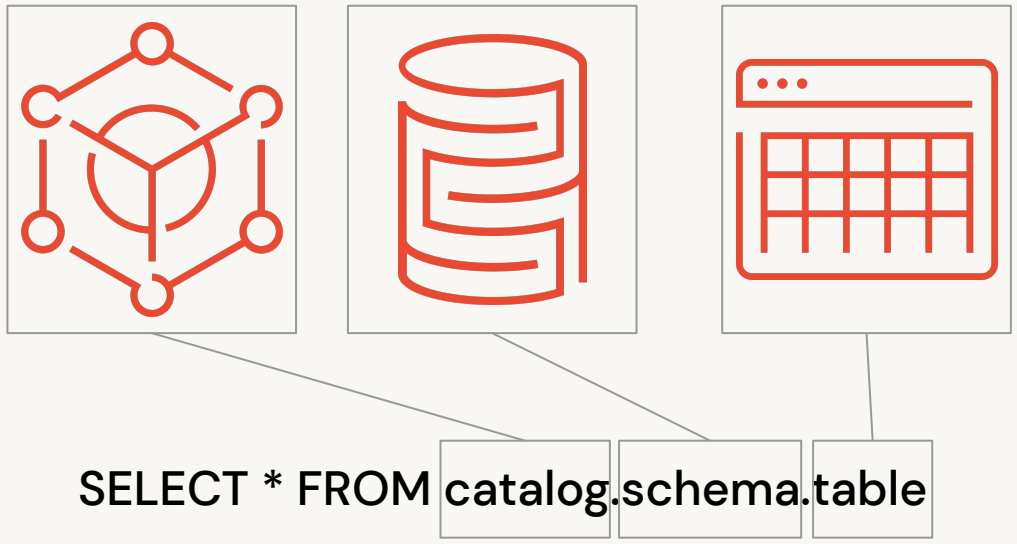
Databricks Unity Catalog

Three-layer namespace

Traditional two-layer namespace



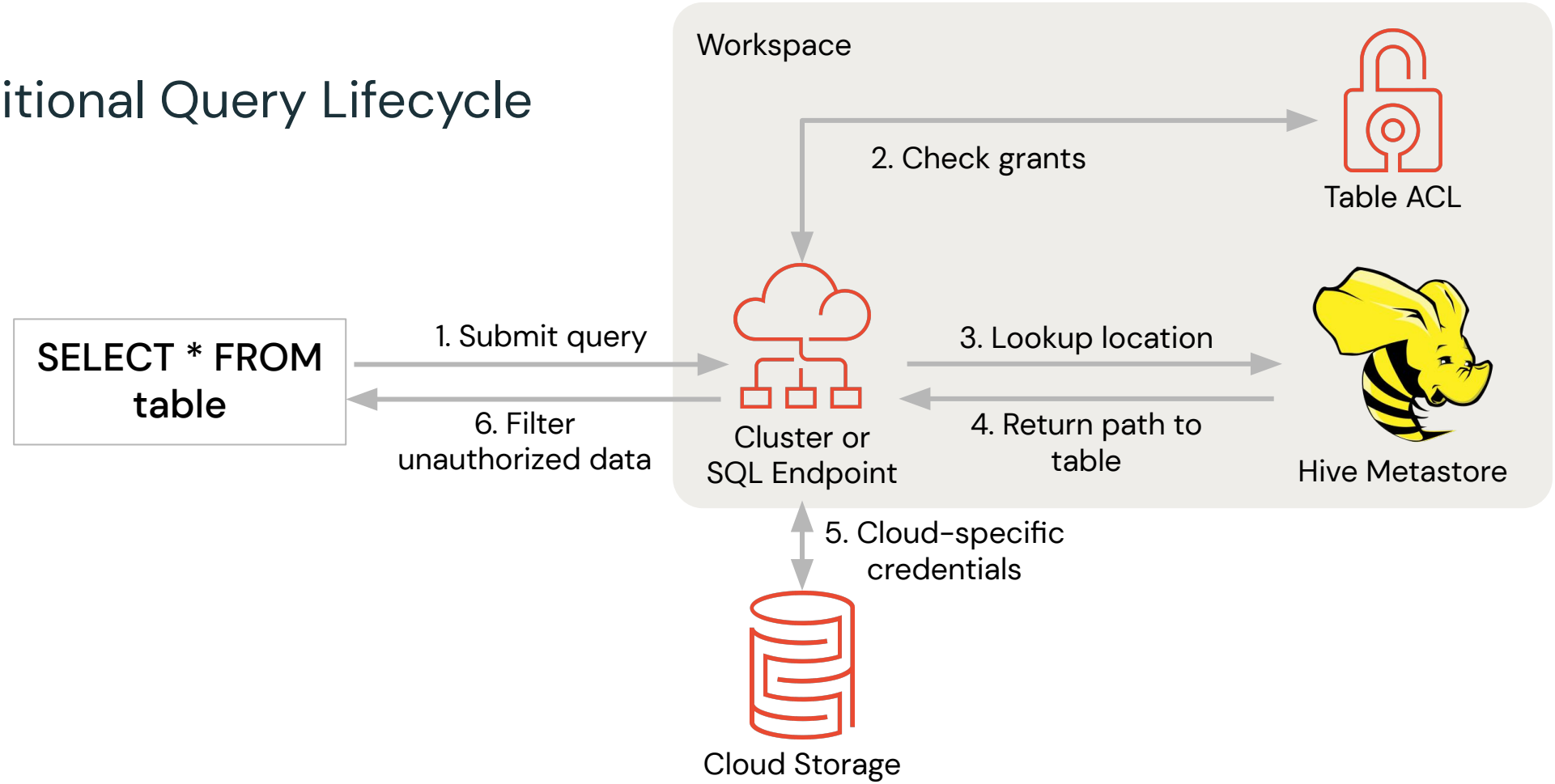
Three-layer namespace with Unity Catalog



Databricks Unity Catalog

Security Model

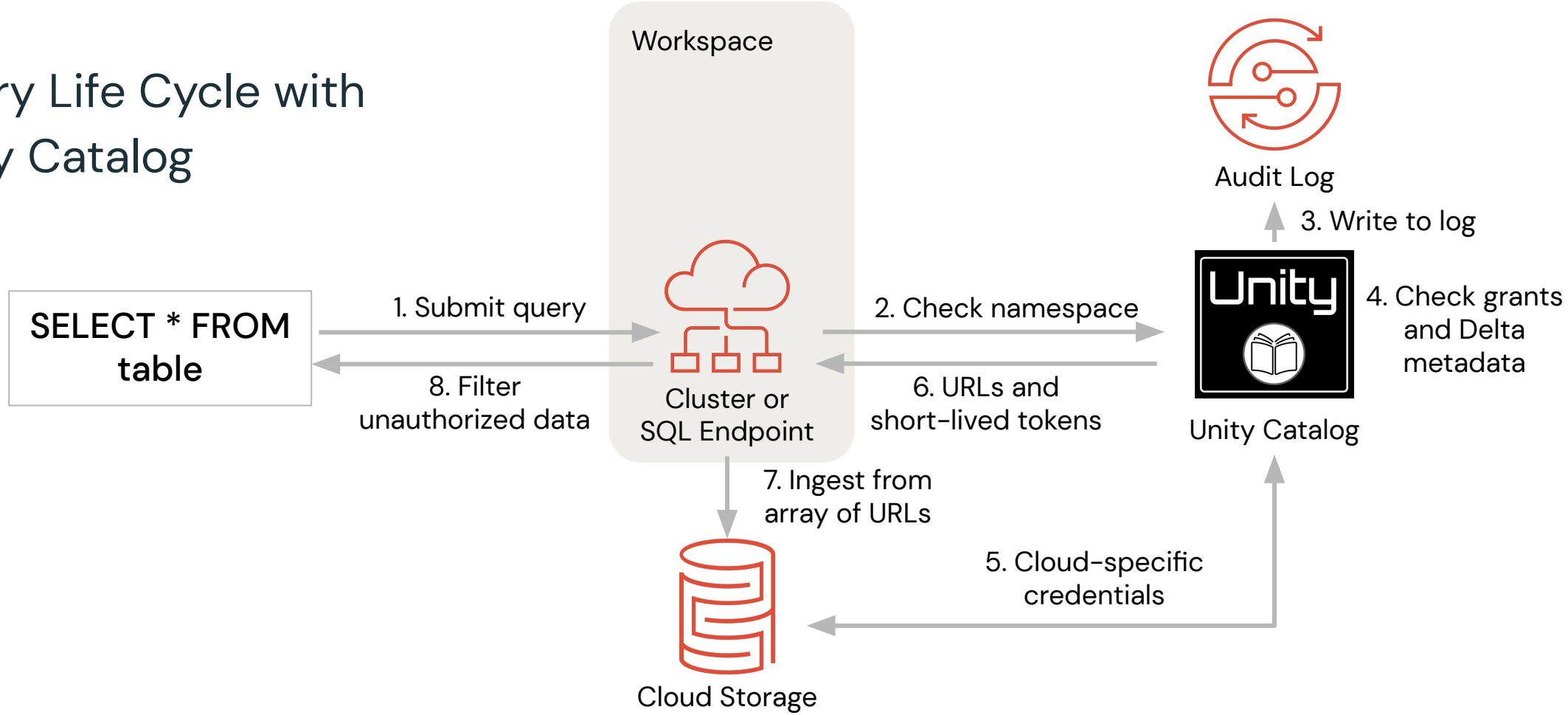
Traditional Query Lifecycle



Databricks Unity Catalog

Security Model

Query Life Cycle with Unity Catalog



Course Recap

Course Objectives

- Leverage the Databricks Lakehouse Platform to perform core responsibilities for data pipeline development
- Use SQL and Python to write production data pipelines to extract, transform, and load data into tables and views in the lakehouse
- Simplify data ingestion and incremental change propagation using Databricks-native features and syntax
- Orchestrate production pipelines to deliver fresh results for ad-hoc analytics and dashboarding