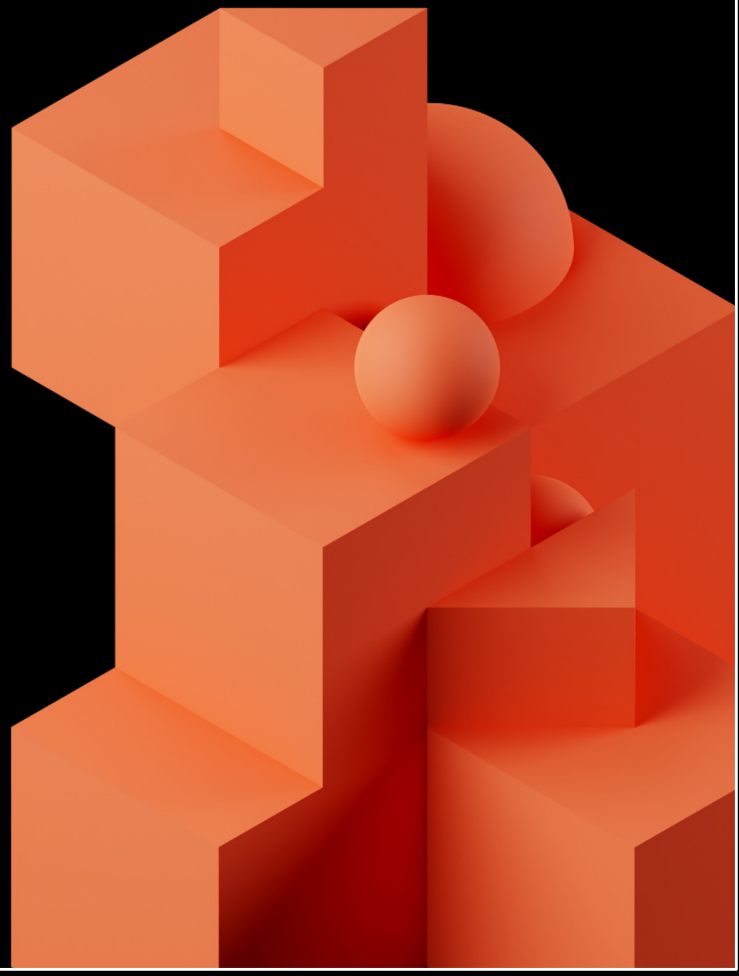
DATA+AI SUMMIT
BY databricks

Distributing Data Governance: How Unity Catalog Allows For A Collaborative Approach

Gilad Asulin

Pulkit Chadha

June 27th 2023



Your Presenters

Gilad Asulin

Sr. Big Data Team Lead @ Akamai

@giladasulin

Pulkit Chadha

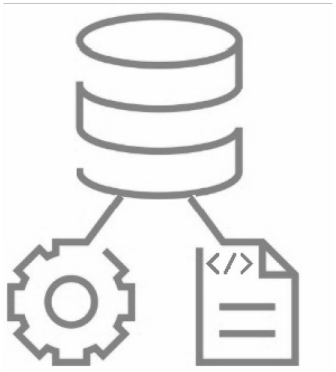
Sr. Solutions Architect @ Databricks

@pulkitchadha

What You will Learn Today?

Why **Unity Catalog is essential** for Databricks users?

How can you **start your journey** to use Unity Catalog?



What **utilities** can be built using Unity Catalog's **metadata**?

@giladasulin, @pulkitchadha

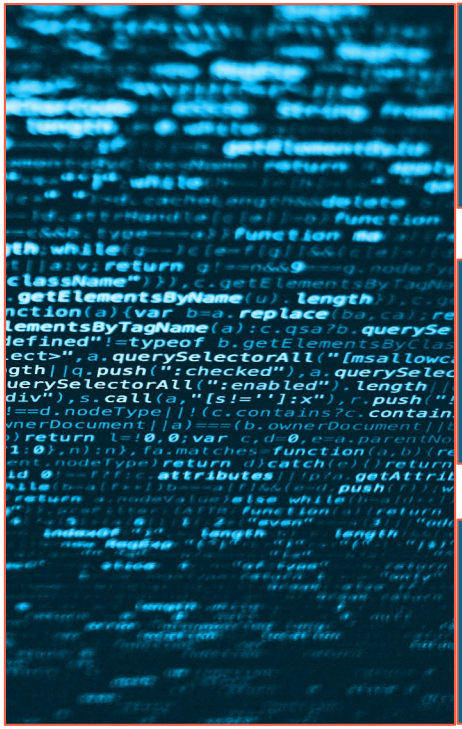
Akamai Technologies Inc

The world's most distributed platform for cloud computing, security, and content delivery

Everywhere you do business, and **anywhere** customers come online, Akamai is closer, with more capacity and integrated security, acceleration, and computing capability than anyone.

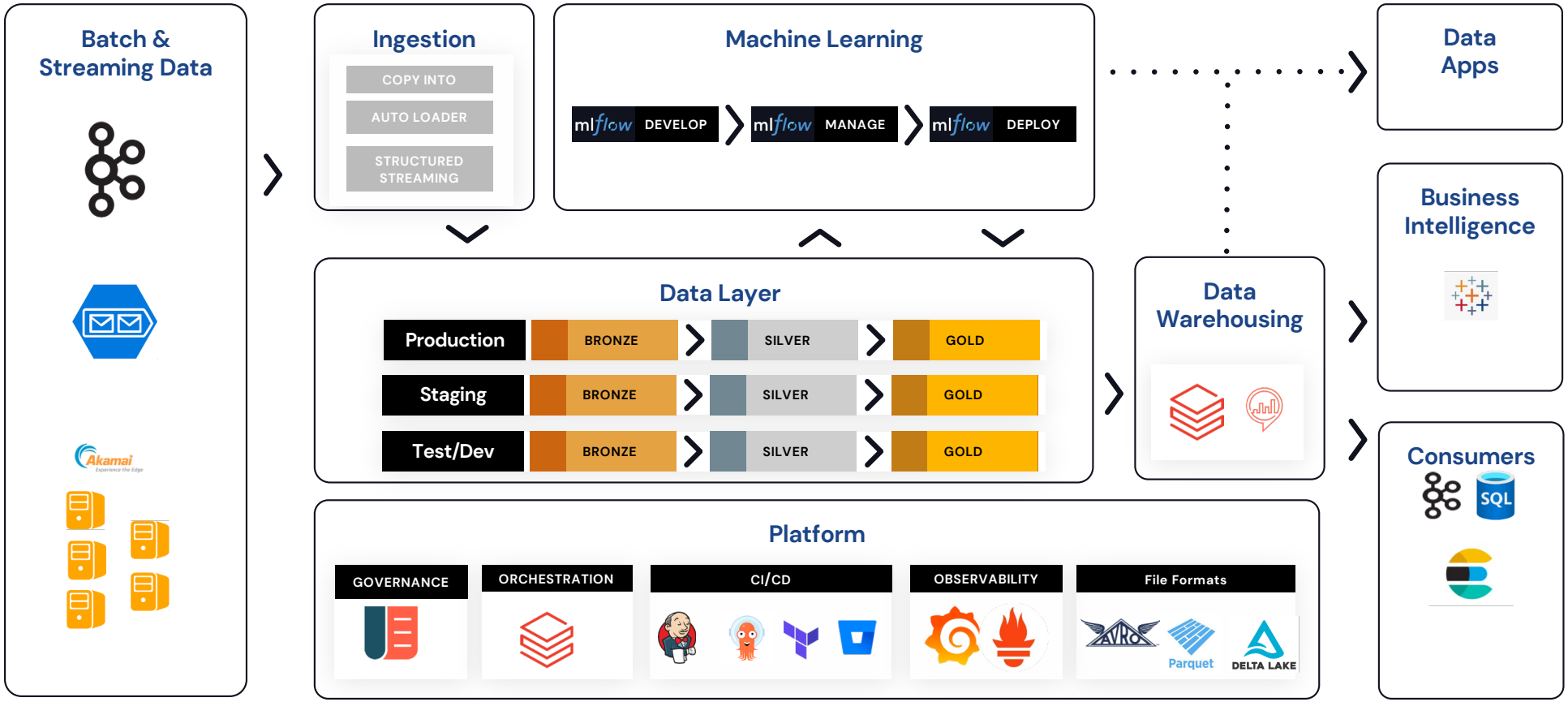
@giladasulin, @pulkitchadha

Akamai Application Security (AppSec)



>650TB	Analyzed Per Day
120B	Suspicious Events Per Day
~7,000	Customers

Akamai AppSec Lakehouse



@giladasulin, @pulkitchadha

Akamai

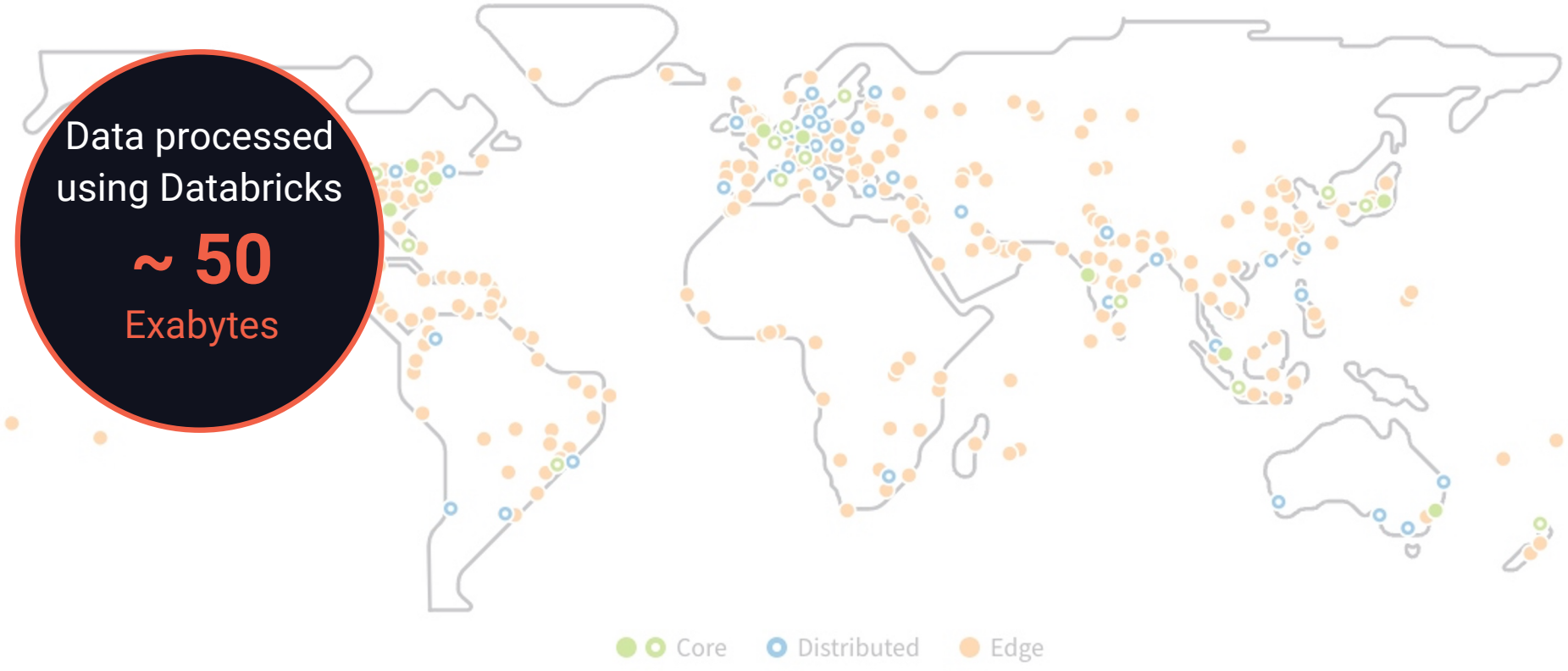
Databricks at Akamai

In Numbers

Data processed using Databricks

~ 50

Exabytes



Databricks at Akamai

In Numbers

Data processed using Databricks

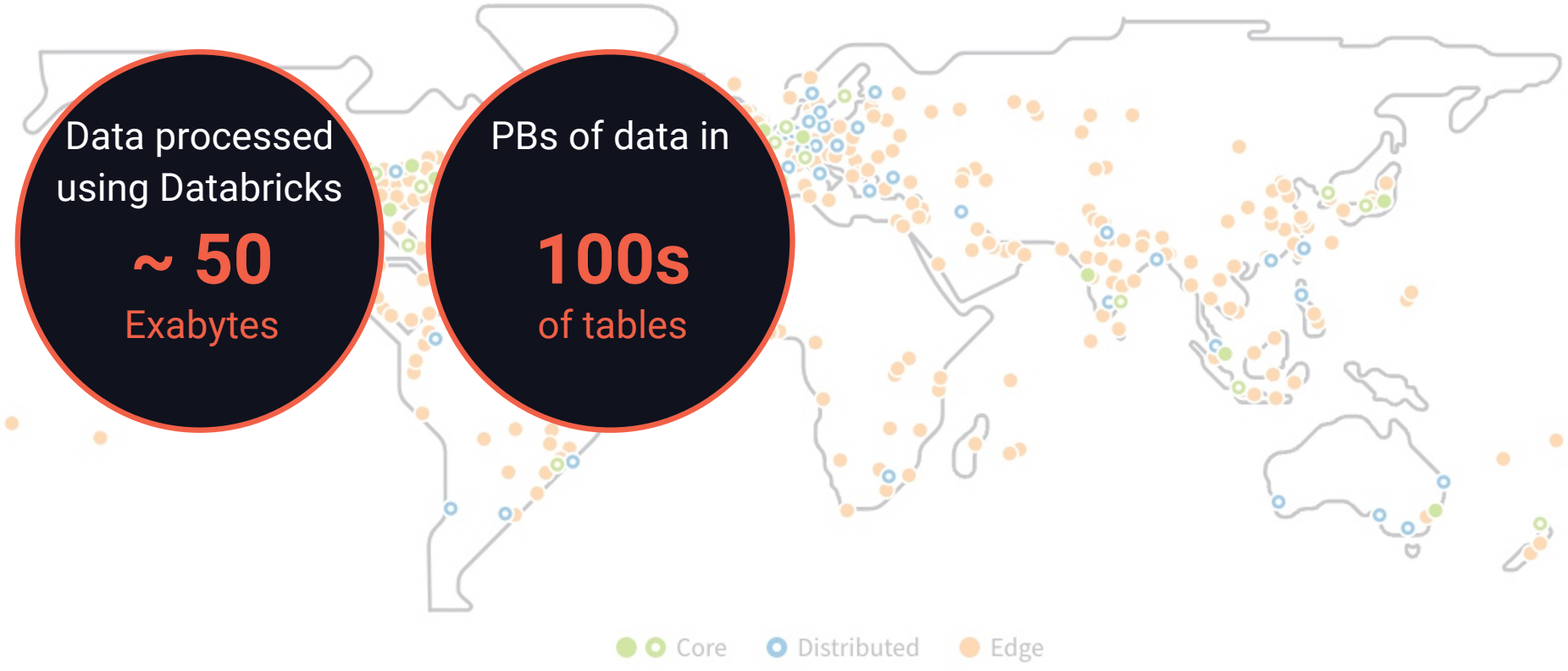
~ 50

Exabytes

PBs of data in

100s

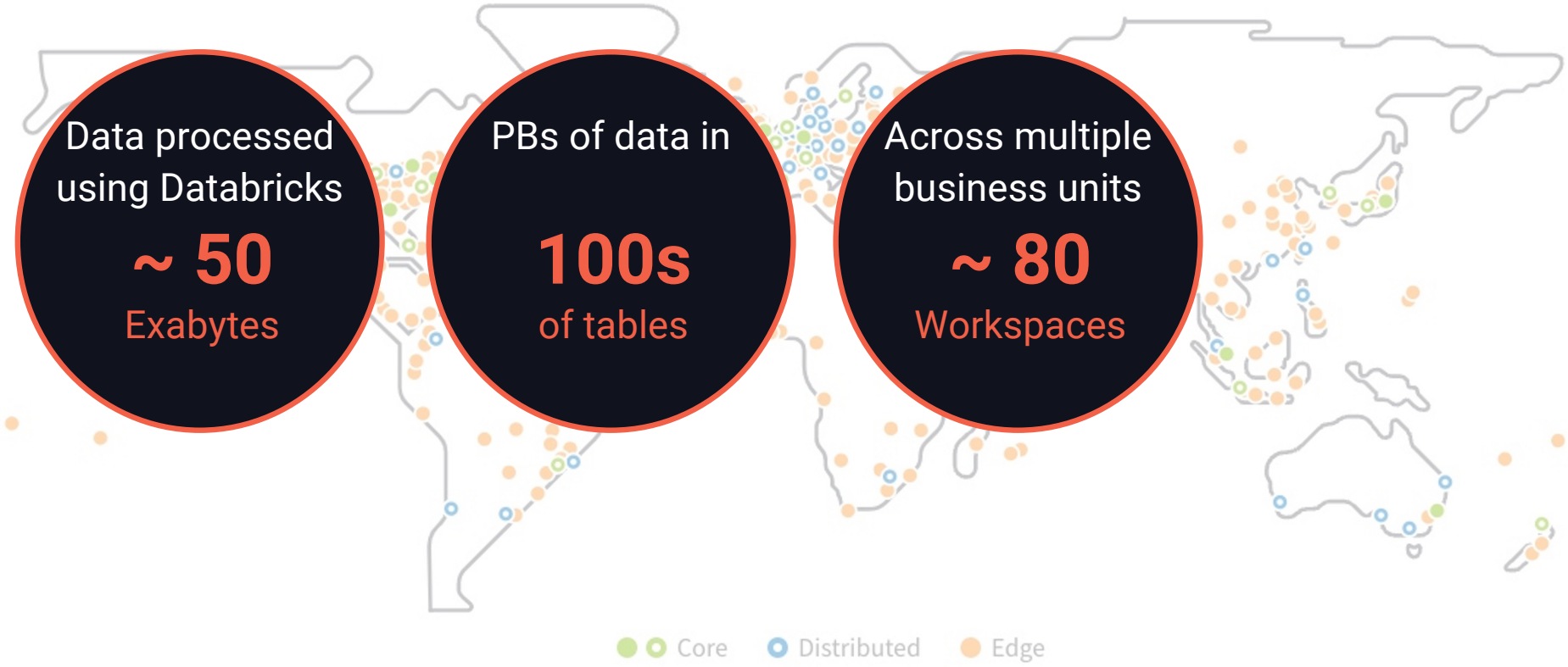
of tables



@giladasulin, @pulkitchadha

Databricks at Akamai

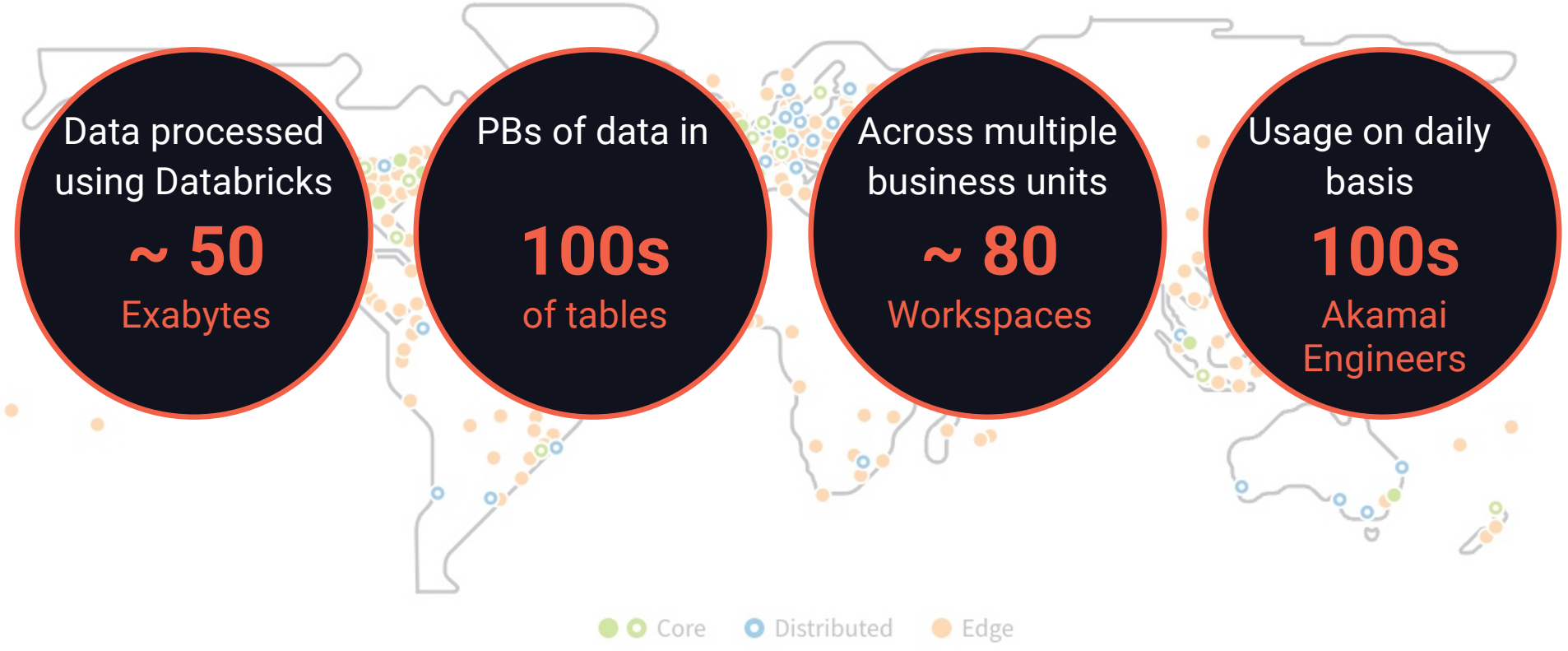
In Numbers



@giladasulin, @pulkitchadha

Databricks at Akamai

In Numbers



Why Do We Need Unity Catalog?

What Unity Catalog Helps With?

Our Challenges Using Databricks Prior to UC

What Unity Catalog Helps With?

Data Sharing

Behavior: Can't share managed tables across workspaces

Our Challenges Using Databricks Prior to UC

What Unity Catalog Helps With?

Data Sharing

Behavior: Can't share managed tables across workspaces

Optional Approach:
Create external tables

- A lot of boilerplate code
- Mount points are unsecure & unreliable

Our Challenges Using Databricks Prior to UC

What Unity Catalog Helps With?

Data Sharing

Behavior: Can't share managed tables across workspaces

Optional Approach: Create external tables

- A lot of boilerplate code
- Mount points are unsecure & unreliable

Least Privileges

Behavior: There is no option to restrict users in workspace level

Our Challenges Using Databricks Prior to UC

What Unity Catalog Helps With?

Data Sharing

Behavior: Can't share managed tables across workspaces

Optional Approach: Create external tables

- A lot of boilerplate code
- Mount points are unsecure & unreliable

Least Privileges

Behavior: There is no option to restrict users in workspace level

Optional Approach: Create separate workspaces

- Leads to difficulty in data sharing
- Causes duplicate work between workspaces

Our Challenges Using Databricks Prior to UC

What Unity Catalog Helps With?

Data Sharing	Least Privileges	User Governance
Behavior: Can't share managed tables across workspaces	**Behavior:** There is no option to restrict users in workspace level	**Behavior:** Standalone user management system (w/o using AAD)
Optional Approach: Create external tables • A lot of boilerplate code • Mount points are unsecure & unreliable	**Optional Approach**: Create separate workspaces • Leads to difficulty in data sharing • Causes duplicate work between workspaces	

@giladasulin, @pulkitchadha

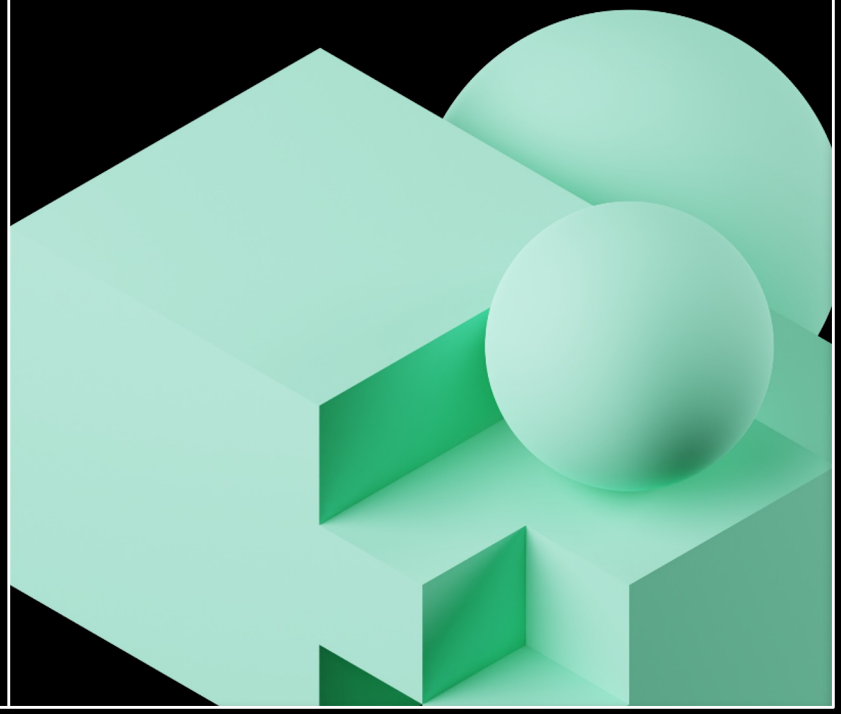
Our Challenges Using Databricks Prior to UC

What Unity Catalog Helps With?

Data Sharing	Least Privileges	User Governance
Behavior: Can't share managed tables across workspaces	**Behavior:** There is no option to restrict users in workspace level	**Behavior:** Standalone user management system (w/o using AAD)
Optional Approach: Create external tables • A lot of boilerplate code • Mount points are unsecure & unreliable	**Optional Approach**: Create separate workspaces • Leads to difficulty in data sharing • Causes duplicate work between workspaces	• Duplicate user management for each workspace • Unable to govern user access control

@giladasulin, @pulkitchadha

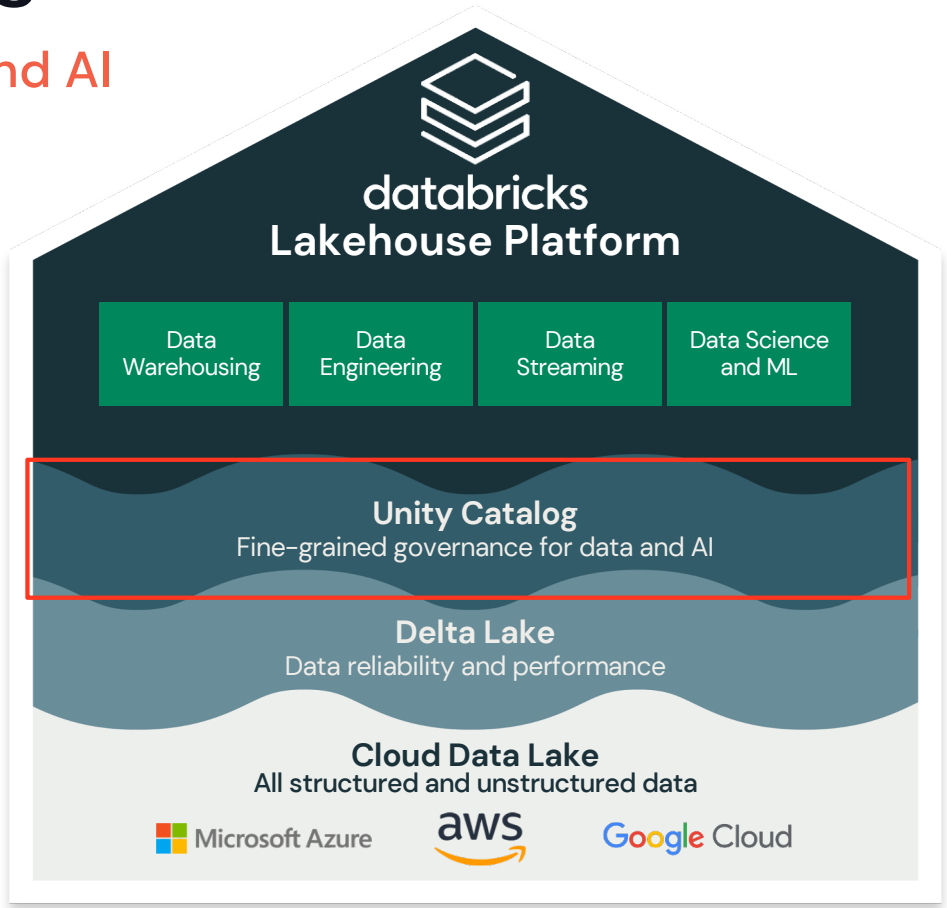
Introduction to Unity Catalog



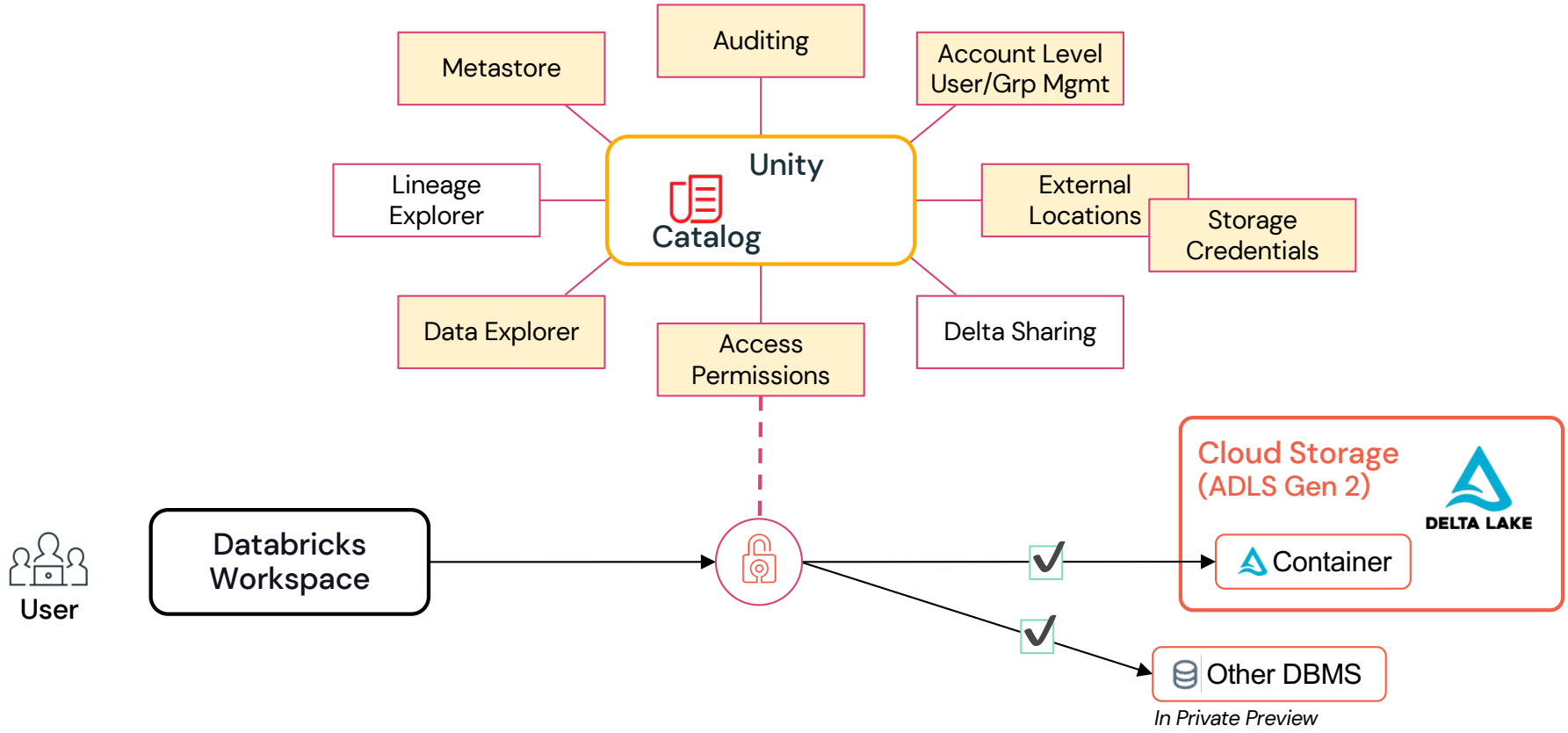
Databricks Unity Catalog

Unified governance for data, analytics and AI

- Unified visibility into data and AI
- Single permission model for data and AI
- AI-driven monitoring and reporting
- Open data sharing



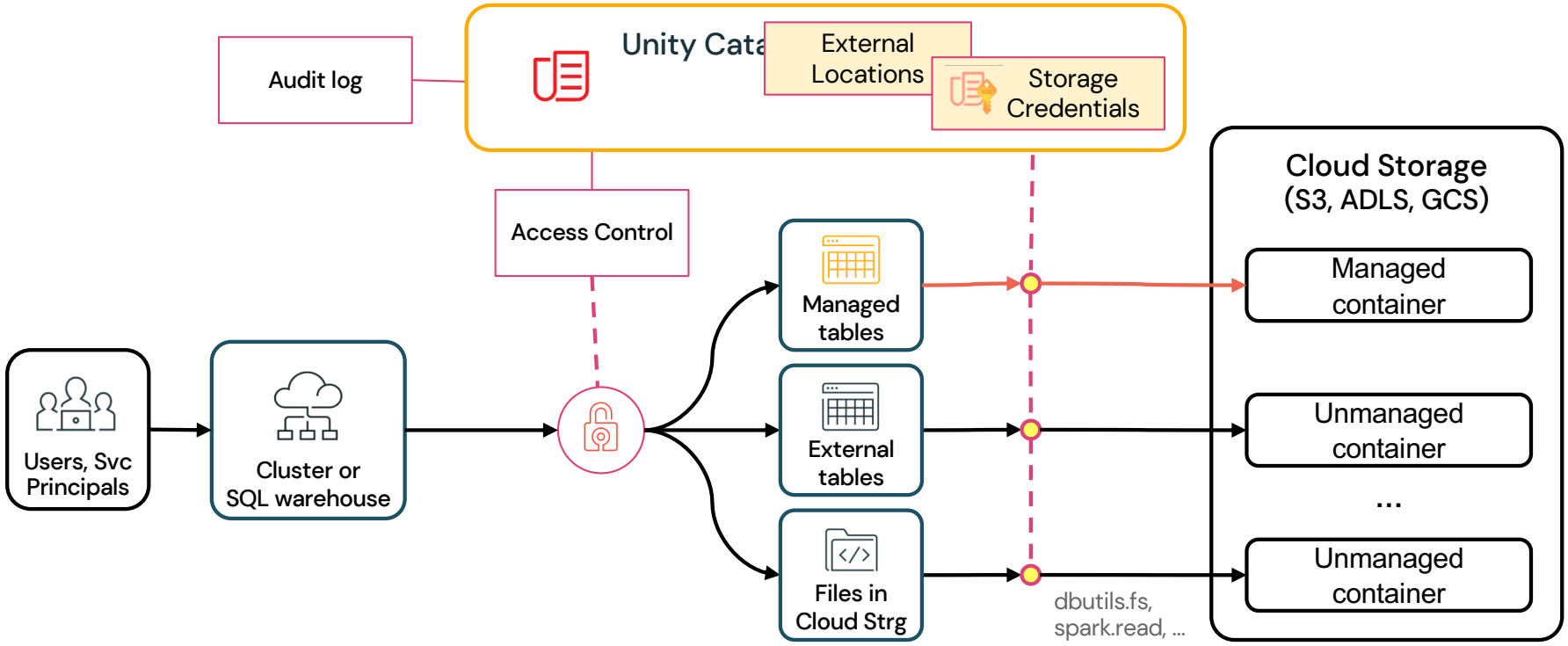
Unity Catalog -Components



@giladasulin, @pulkitchadha

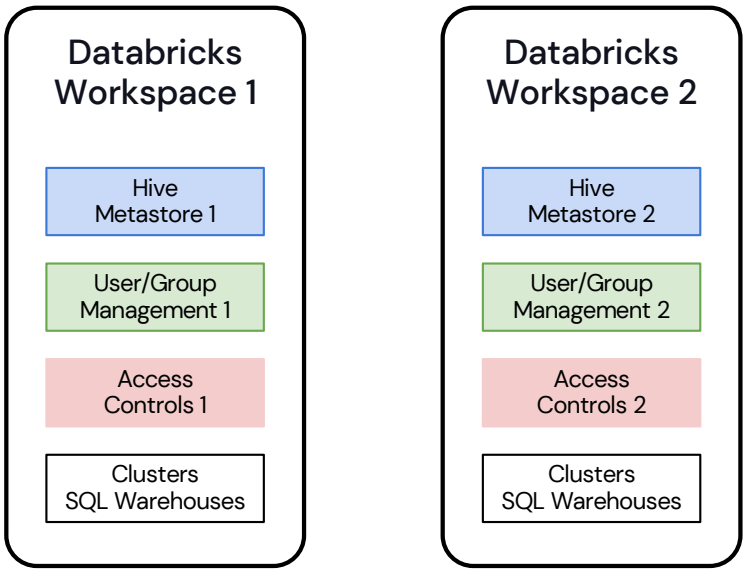
Unity Catalog External Locations

Improves data governance and simplifies data access administration

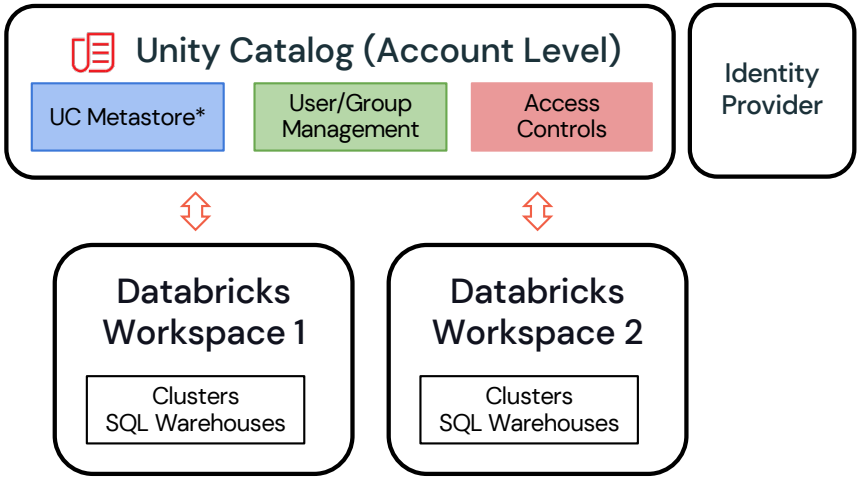


Centralized Metadata, Identity, and Access Management

Without Unity Catalog



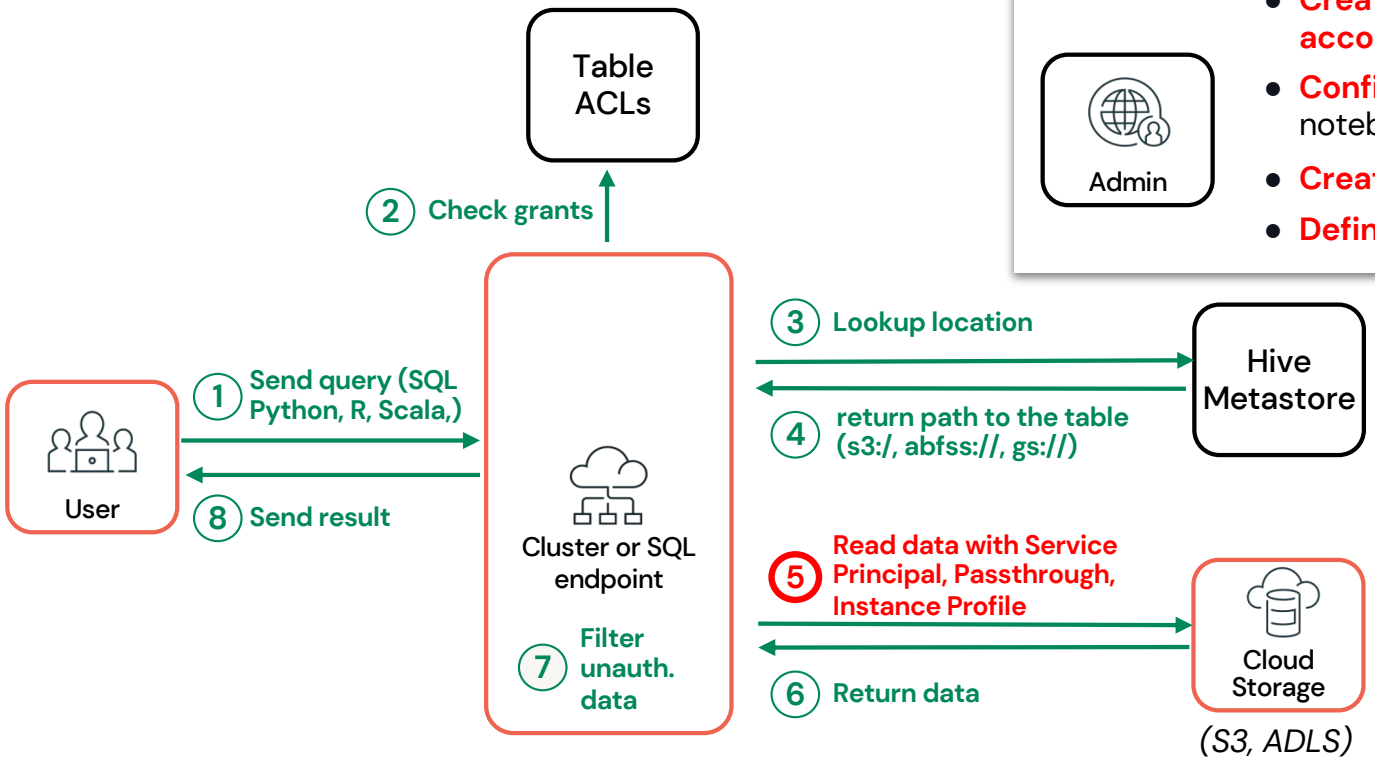
With Unity Catalog



*** Important: only one UC metastore per region!**

Life of a query *without* Unity Catalog

Difficult to administer; overprovisioned access



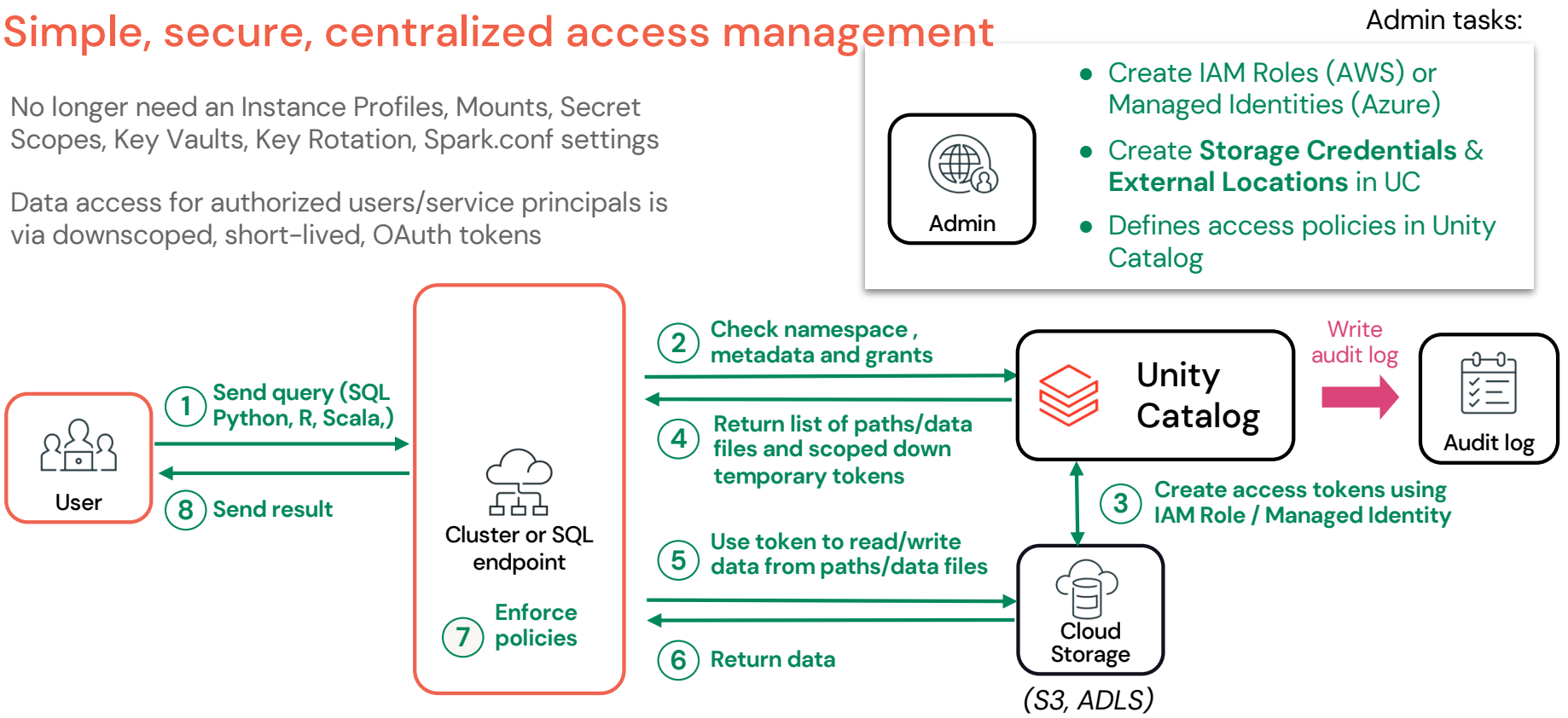
Need to rotate keys, configure spark.conf properties/instance profiles on clusters, configure table ACLs on each workspace, mounts are inflexible...

Life of a query with Unity Catalog

Simple, secure, centralized access management

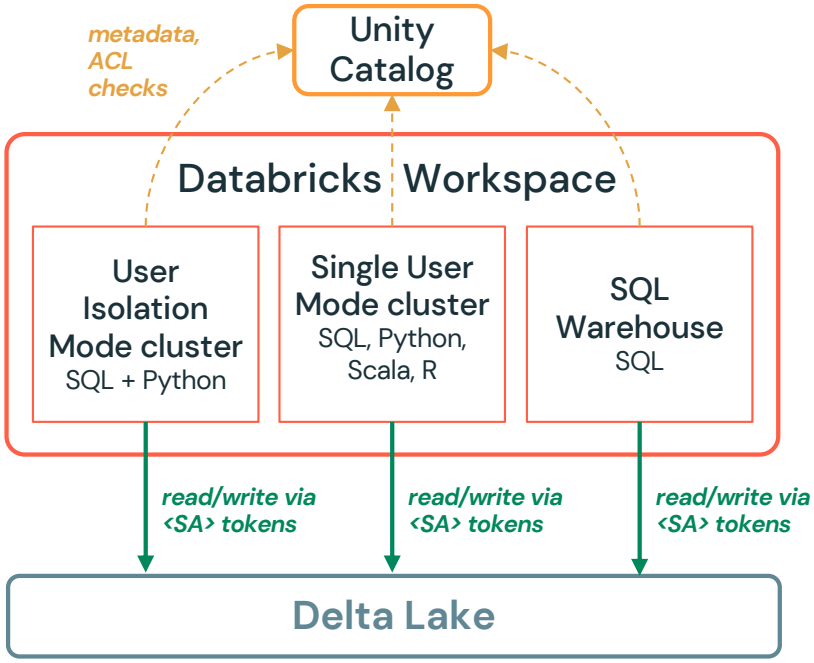
No longer need an Instance Profiles, Mounts, Secret Scopes, Key Vaults, Key Rotation, Spark.conf settings

Data access for authorized users/service principals is via downscoped, short-lived, OAuth tokens



Migrating Workloads

Cluster options in Unity Catalog



<SA> **System Account** (Service Principal or Managed Identity for Azure, IAM Role for AWS, Service Account for GCP)

All Purpose clusters with User Isolation mode

- Use User isolation mode for general workloads (ETL, data exploration, …) using SQL and Python
- Multiple users can work on the same cluster

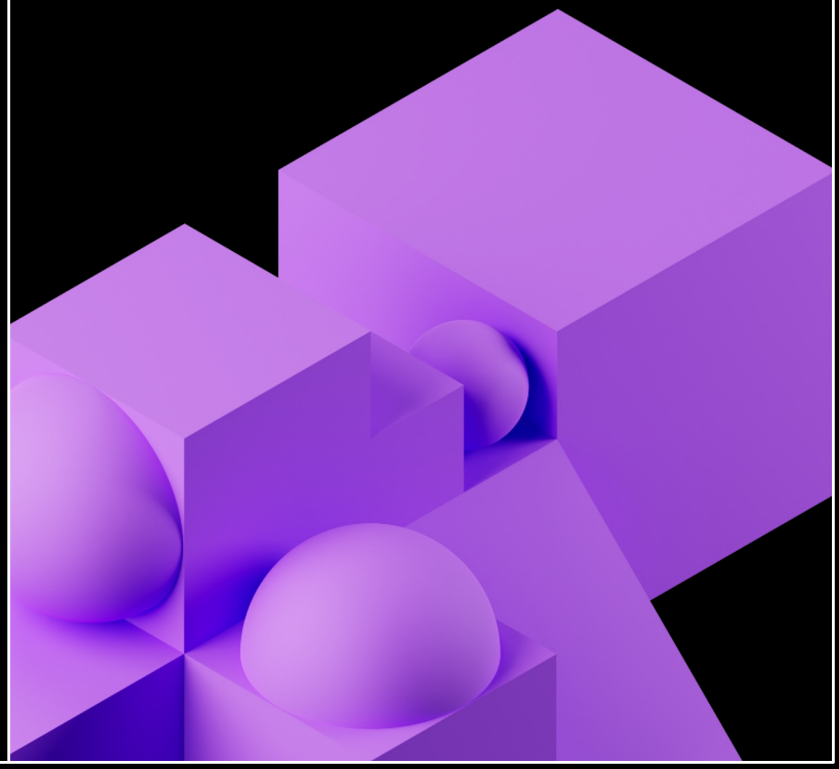
All Purpose clusters with Single User mode

- Use Single User mode for Scala users and for Data Scientists
- ML Runtime with MLflow is supported
- Only the owner can execute code on this cluster, so for notebook collaboration, co-workers can see everything, but they cannot execute cells.

SQL Warehouse

- Use SQL Warehouses for Business Analysts either using Databricks SQL Editor or external BI tools like Power BI, Tableau, …

Unity Catalog In Action



Things We Took into Consideration

Inquiry Leads to Requirements

Catalogs Structure - What is the best structure for us?

- By environment?
- By product?
- Is there a cost impact?

Things We Took into Consideration

Inquiry Leads to Requirements

Catalogs Structure - What is the best structure for us?

- By environment?
- By product?
- Is there a cost impact?

Data Engineering Team's Role - How is it going to impact our day-to-day work?

- Govern w/o being a bottleneck
- Interaction with stakeholders
- Minimize administrative burden

Things We Took into Consideration

Inquiry Leads to Requirements

Catalogs Structure - What is the best structure for us?

- By environment?
- By product?
- Is there a cost impact?

Data Engineering Team's Role - How is it going to impact our day-to-day work?

- Govern w/o being a bottleneck
- Interaction with stakeholders
- Minimize administrative burden

Data Location - Where should we store the data?

Mapping Requirements to Design Decisions #1

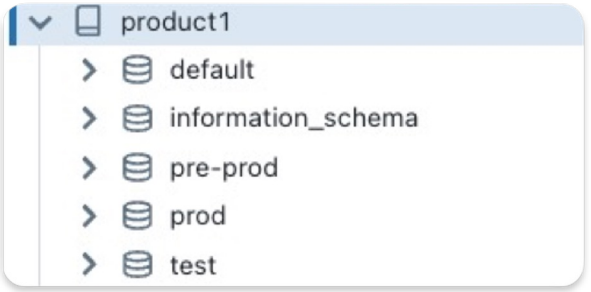
Catalogs Structure

Requirement

- Teams are segregated by product

Design Decisions

- Each catalog
  - Represents a product
  - Contains a schema for each environment



Mapping Requirements to Design Decisions #2

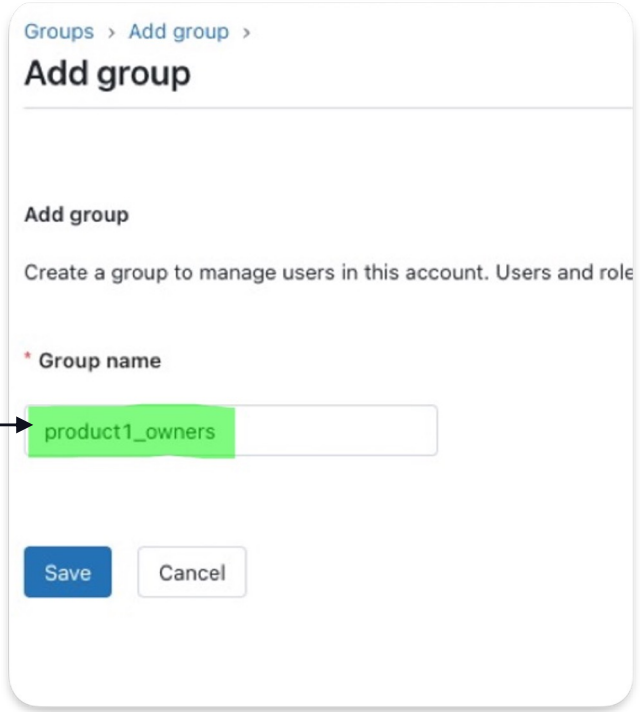
Governance Model

Requirements

- Product owners responsible for their own data
- Govern w/o being a bottleneck

Design Decisions

- Apply a distributed data governance
- Each catalog has a product owner group



Mapping Requirements to Design Decisions #3

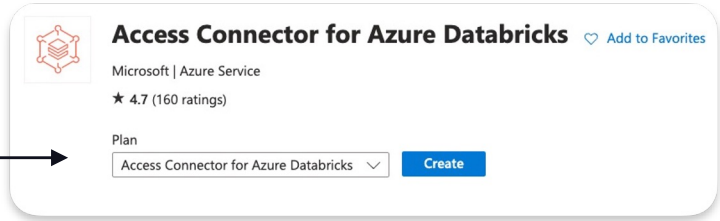
Data Location

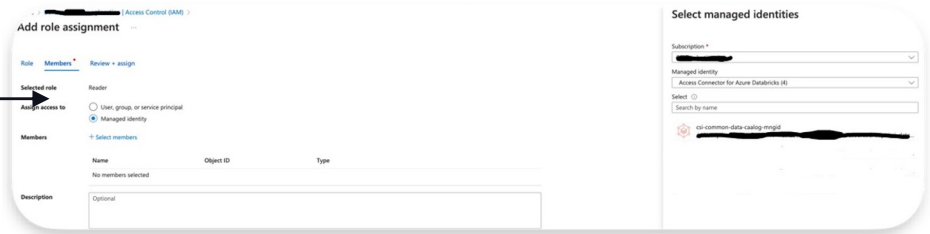
Requirements

- Most of the data comes from external data sources
- Data should be stored separately for each product and environment

Design Decisions

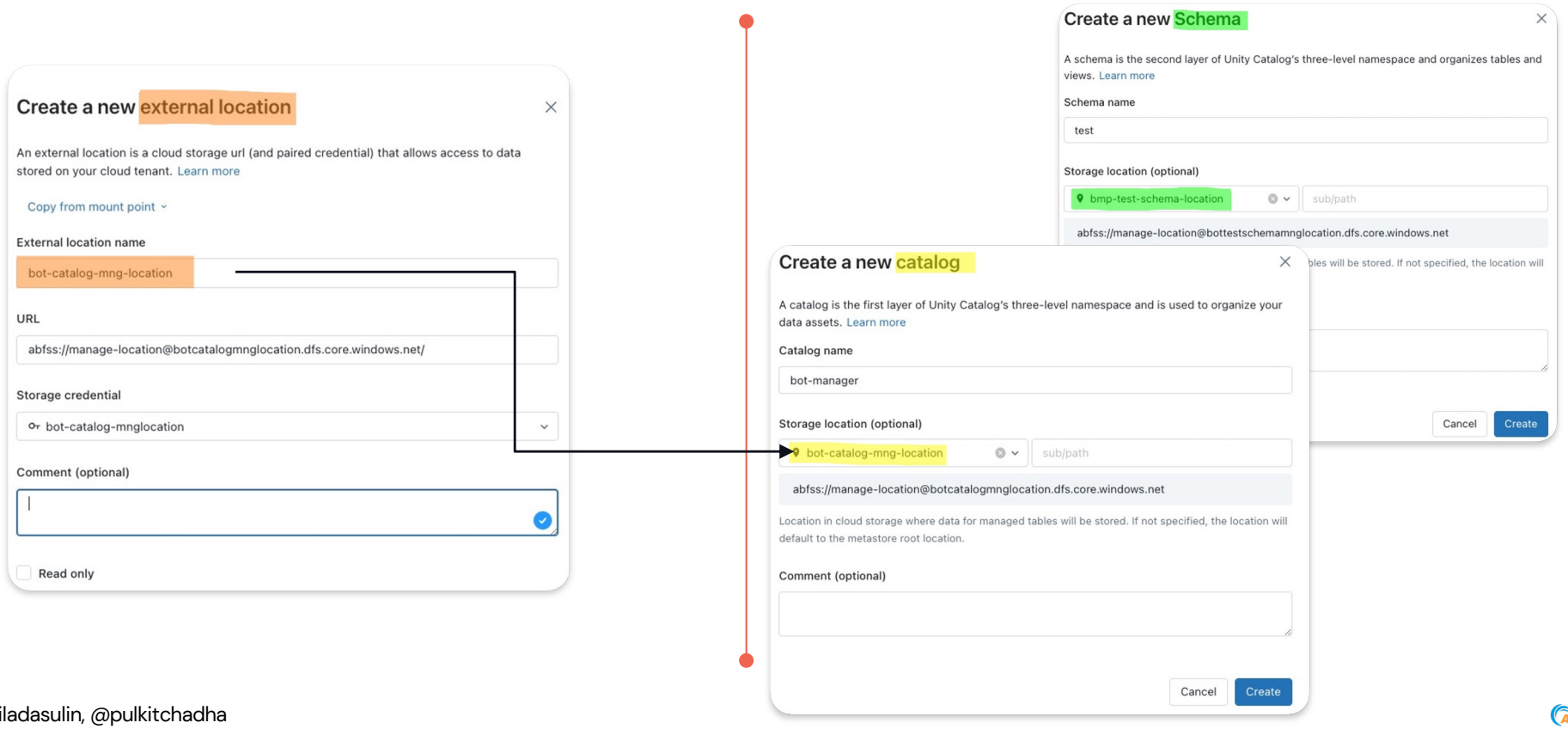
- Use Managed Identities to create External tables
- Use Managed Location feature





Managed Location for Catalogs & Schemas

Differentiate between products and between prod/non-prod environments



Mapping Requirements to Design Decisions #4

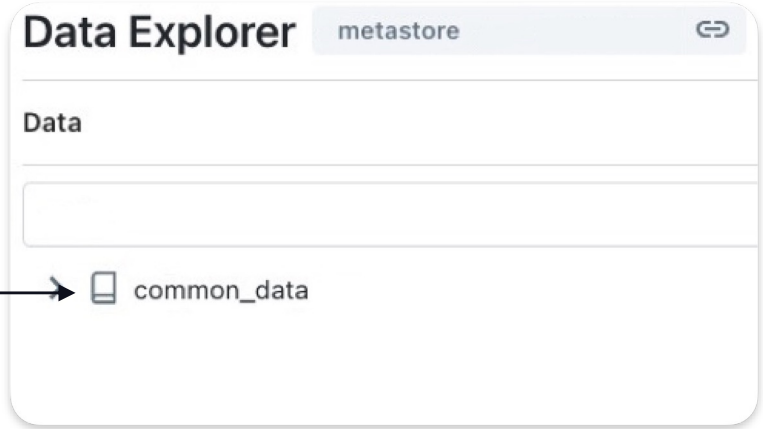
Data Engineering Team's Role

Requirements

- Ownership of all shared data sources
- Products' data is governed by high level standards and utilities
- Enable access to metadata

Design Decisions

- Create a common_data catalog
- Create utility that expose metadata
- Create dashboards and alerts to gain a high level perspective



Mapping Requirements to Design Decisions #5

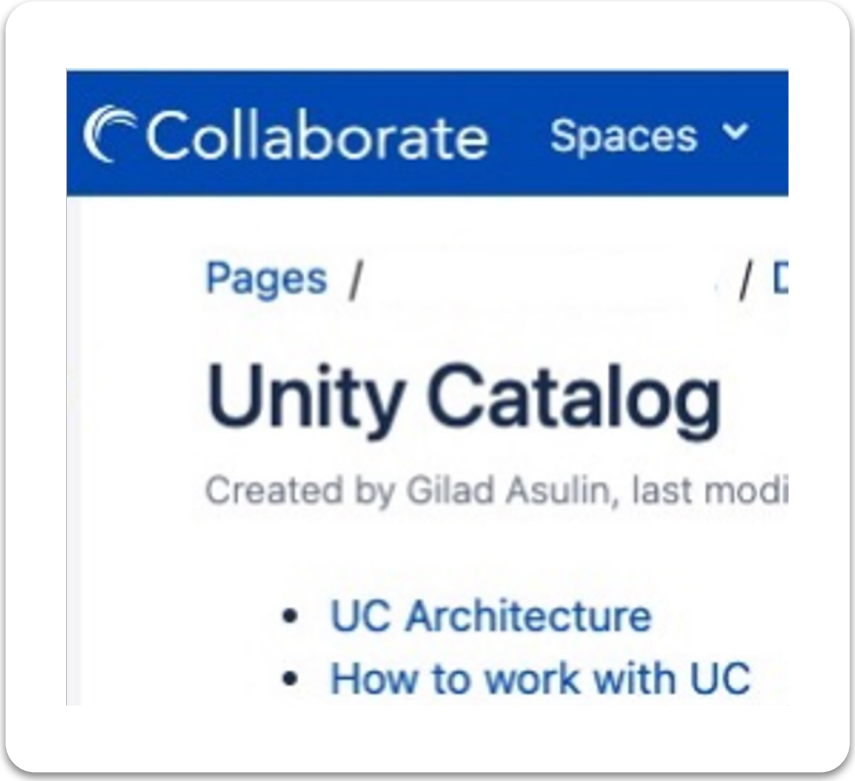
Stakeholders Impact

Requirement

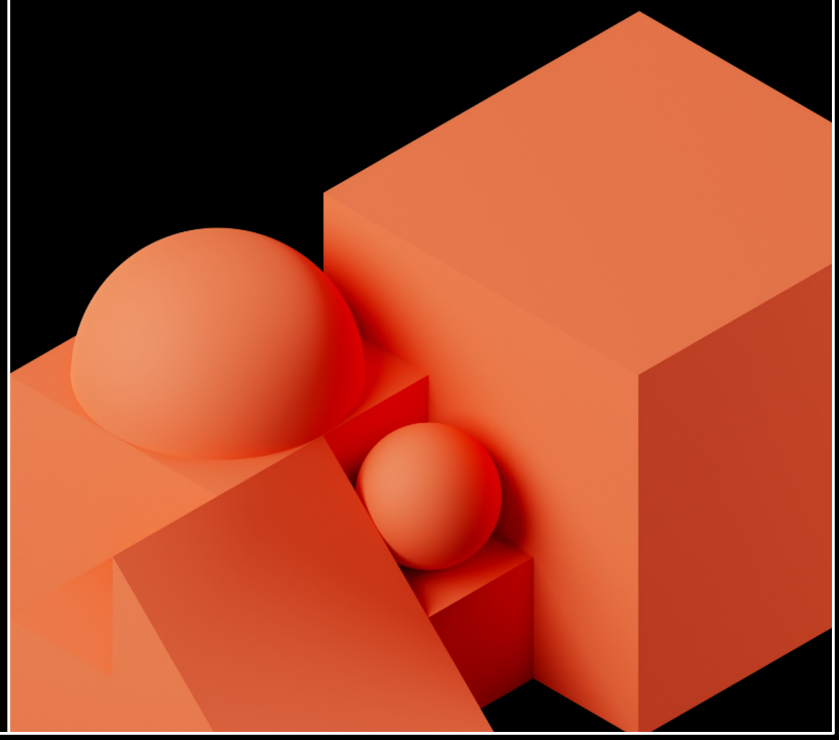
- Identify and map all the potential impacts on stakeholders

Design Decisions

- Document all the scenarios that can impact the roles in the organization
- Communicate with stakeholders to make them engaged



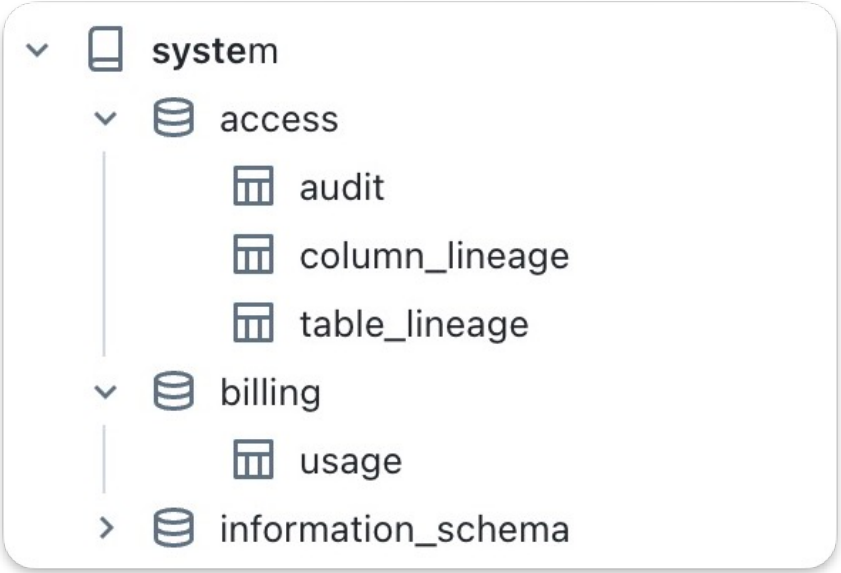
Utilities Built using Unity Catalog's Metadata



SYSTEM Catalog

Unity Catalog's Metadata

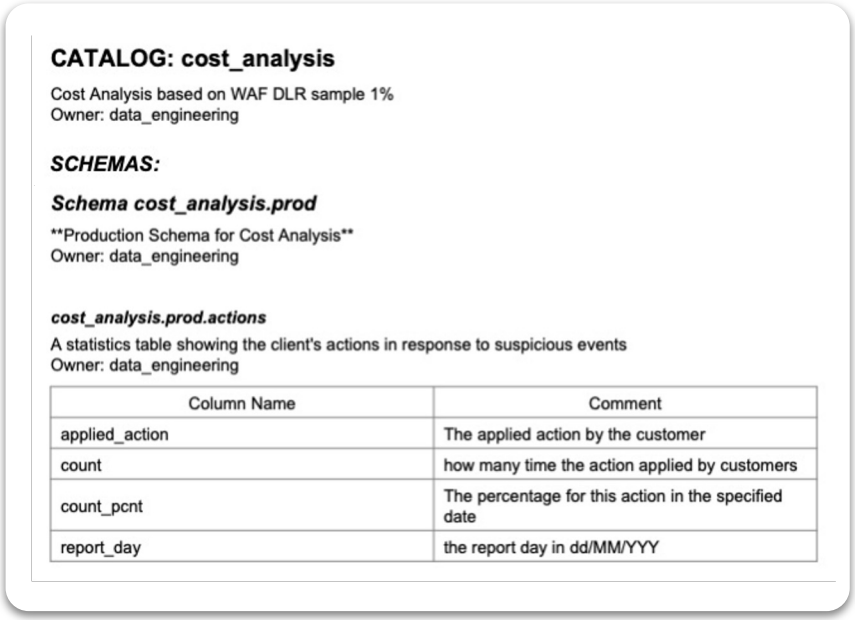
- Stores all Unity Catalog's metadata and Databricks workspaces activities
- `INFORMATION_SCHEMA` stores metastore metadata.
  - Hive_metastore is not part of metastore
- `ACCESS SCHEMA` stores the account level activities
- `BILLING SCHEMA` stores the billing logs in the account



Expand Data Assets Visibility

Catalogs as PDF – Utility for non Unity Catalog users

- Contains all the catalogs\schemas\tables information
  - Comments\tags on each entity.
  - Who owns this catalog?
  - What entities we have?
- The audience
  - Employees that don't have permissions to UC like
    - External departments
    - Product managers
    - Restricted users



CATALOG: cost_analysis

Cost Analysis based on WAF DLR sample 1%
Owner: data_engineering

SCHEMAS:

Schema cost_analysis.prod

Production Schema for Cost Analysis
Owner: data_engineering

cost_analysis.prod.actions

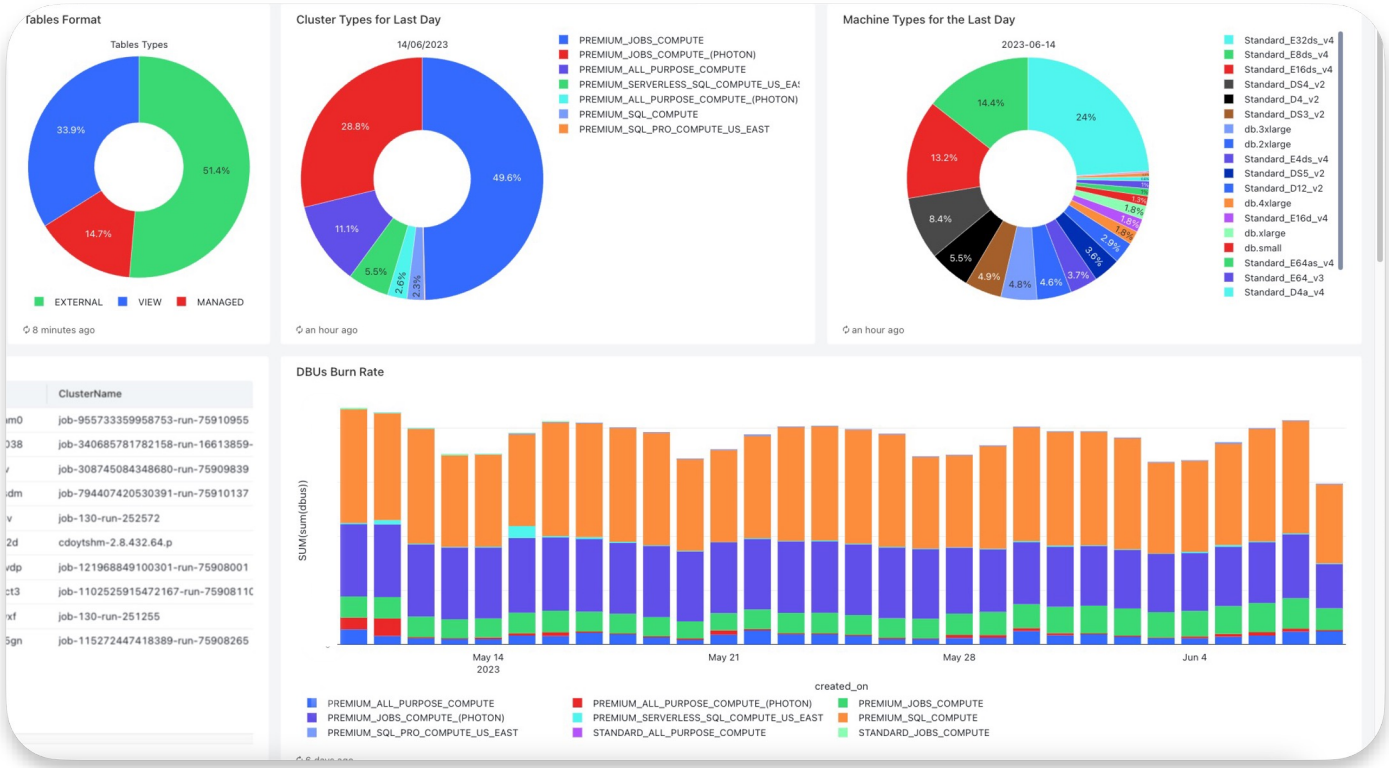
A statistics table showing the client's actions in response to suspicious events
Owner: data_engineering

Column Name	Comment
applied_action	The applied action by the customer
count	how many time the action applied by customers
count_pcnt	The percentage for this action in the specified date
report_day	the report day in dd/MM/YYY

@giladasulin, @pulkitchadha

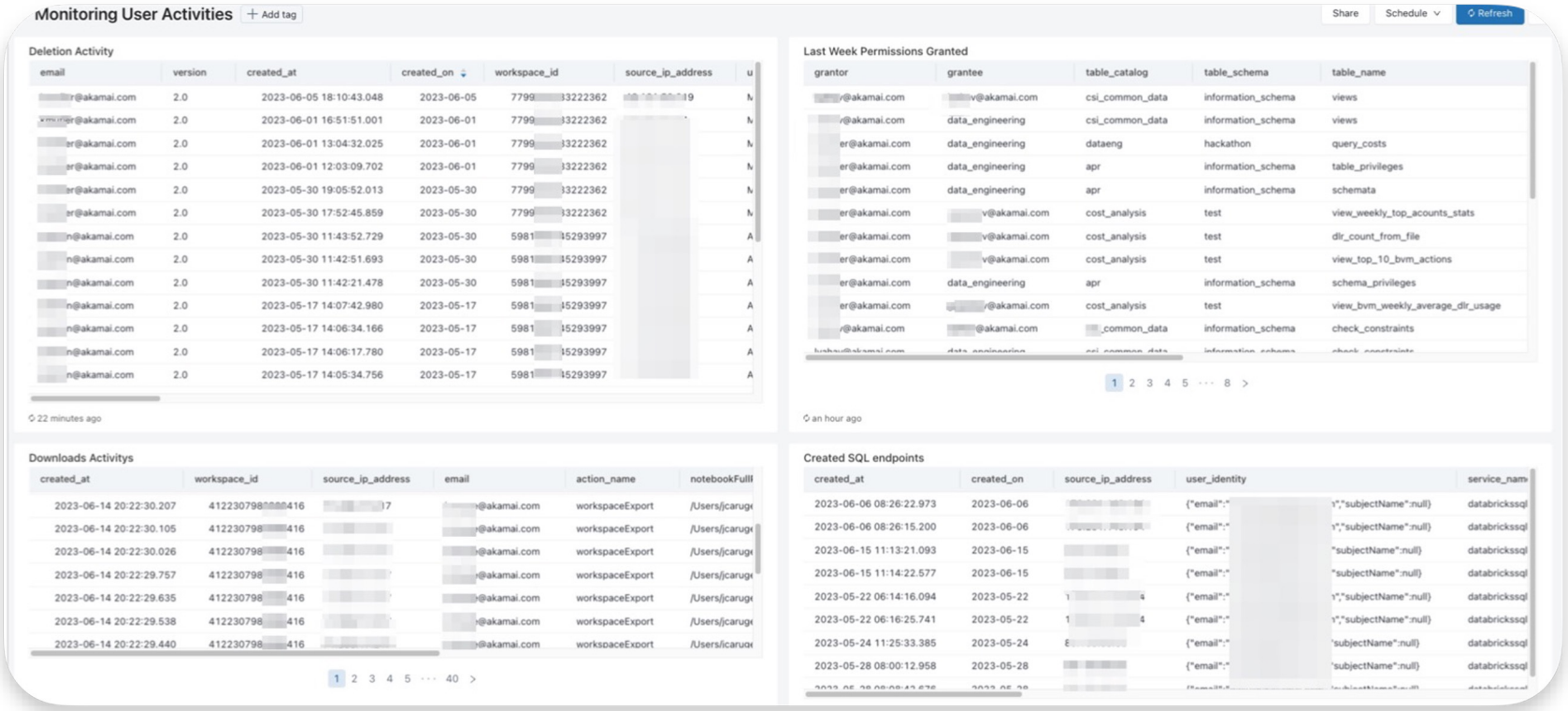
High Level Governance Dashboards

State of the Lakehouse



High Level Governance Dashboards

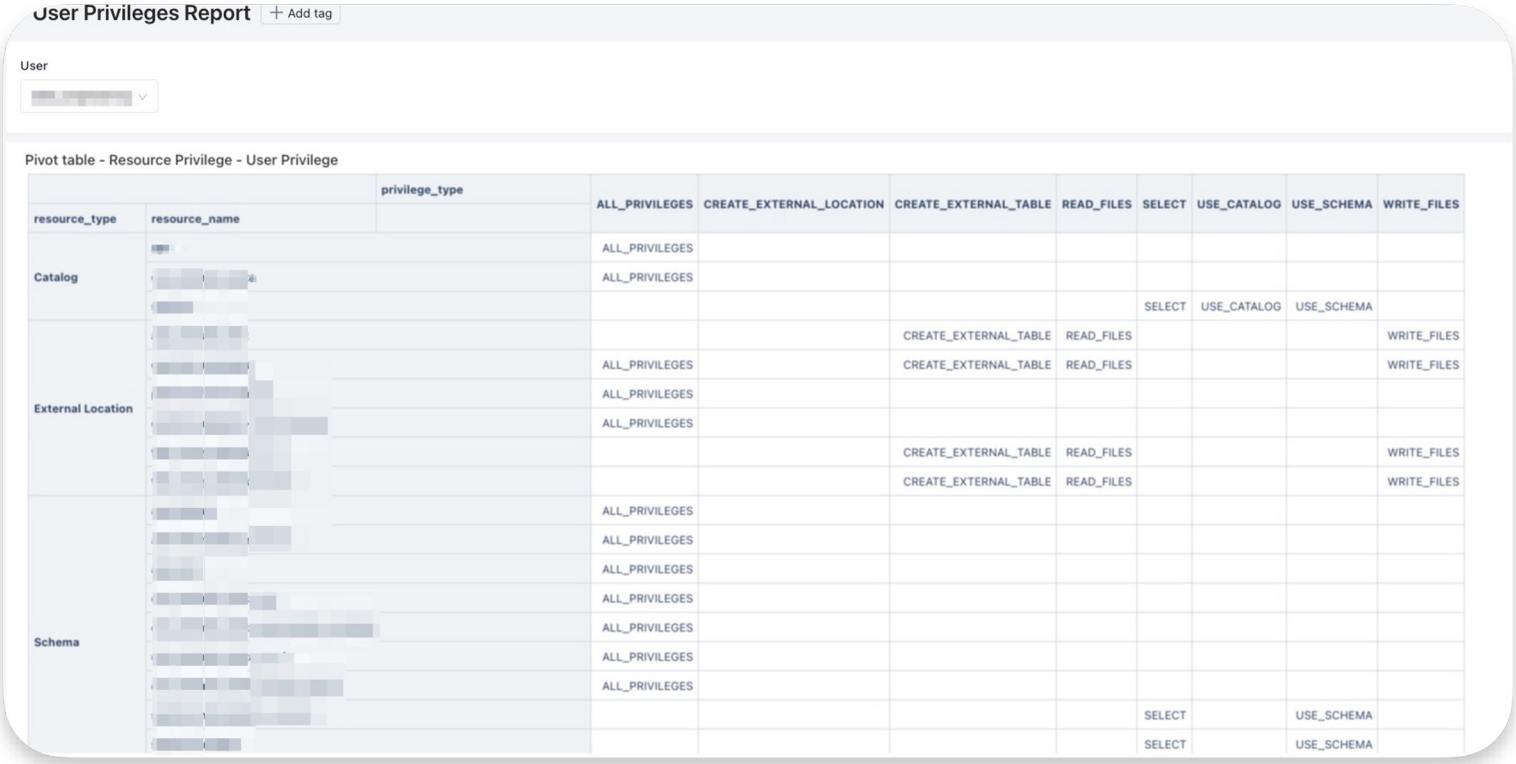
Users' Activity Monitor



@giladasulin, @pulkitchadha

High Level Governance Dashboards

User Privileges Report



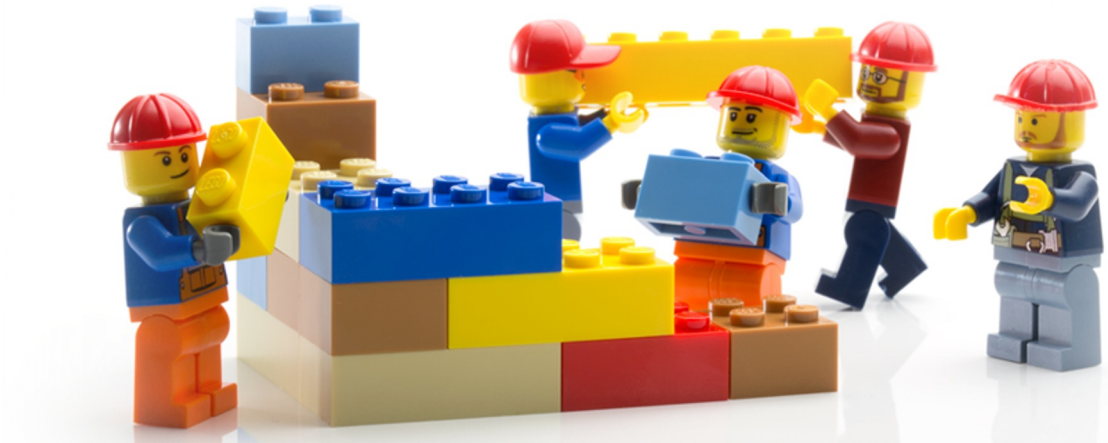
resource_type	resource_name	privilege_type	ALL_PRIVILEGES	CREATE_EXTERNAL_LOCATION	CREATE_EXTERNAL_TABLE	READ_FILES	SELECT	USE_CATALOG	USE_SCHEMA	WRITE_FILES
Catalog			ALL_PRIVILEGES							
			ALL_PRIVILEGES							
							SELECT	USE_CATALOG	USE_SCHEMA	
External Location					CREATE_EXTERNAL_TABLE	READ_FILES				WRITE_FILES
			ALL_PRIVILEGES		CREATE_EXTERNAL_TABLE	READ_FILES				WRITE_FILES
			ALL_PRIVILEGES							
			ALL_PRIVILEGES							
					CREATE_EXTERNAL_TABLE	READ_FILES				WRITE_FILES
					CREATE_EXTERNAL_TABLE	READ_FILES				WRITE_FILES
Schema			ALL_PRIVILEGES							
			ALL_PRIVILEGES							
			ALL_PRIVILEGES							
			ALL_PRIVILEGES							
			ALL_PRIVILEGES							
			ALL_PRIVILEGES							
			ALL_PRIVILEGES							
							SELECT		USE_SCHEMA	
							SELECT		USE_SCHEMA	

@giladasulin, @pulkitchadha

Akamai

Unity Catalog Migration Recipe

Putting the pieces together



Akamai's Unity Catalog Migration Journey

Foundational building blocks

One Time Setup

- Identity federation
- Create account groups
- Create account level dashboards

@giladasulin, @pulkitchadha

Akamai's Unity Catalog Migration Journey

Foundational building blocks

One Time Setup

- Identity federation
- Create account groups
- Create account level dashboards

Per Region

- Create a metastore
- Assign metastore to Workspaces
- Create common-data catalog
- Create metastore level utilities & dashboards

Akamai's Unity Catalog Migration Journey

Foundational building blocks

One Time Setup

- Identity federation
- Create account groups
- Create account level dashboards

Per Region

- Create a metastore
- Assign to Workspaces
- Create common-data catalog
- Create metastore level utilities & dashboards

Per Product

- Create catalog, schemas (environments) & tables
- Set product owners
- Migrate legacy metadata & managed tables
- Migrate clusters, jobs & notebooks
- Revoke mounts & Hive_metastore

@giladasulin, @pulkitchadha

Lessons Learned From the Journey

(So far…)

- Invest in requirements collection early.
- Use automation for efficiency
- Stay updated on new features.
  - Some noteworthy features:
    - Serverless capabilities
    - Audit logs
- Based on our experience, UC performs optimally with Delta Lake format

What You Learned Today?

Unity Catalog is essential
for Databricks users

- Central access management
- Share data between workspaces easily
- Enhanced security

What You Learned Today?

Unity Catalog is essential for Databricks users

- Central access management
- Share data between workspaces easily
- Enhanced security

Starting your journey towards using Unity Catalog

- Collect requirements.
- Derive design decisions based on requirements
- Follow the migration journey slide and do adjustments for your use case

@giladasulin, @pulkitchadha

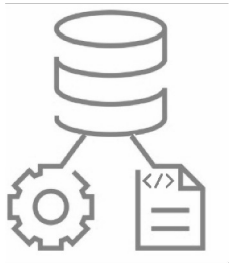
What You Learned Today?

Unity Catalog is essential for Databricks users

- Central access management
- Share data between workspaces easily
- Enhanced security

Starting your journey towards using Unity Catalog

- Collect requirements.
- Derive design decisions based on requirements
- Follow the migration journey slide and do adjustments for your use case



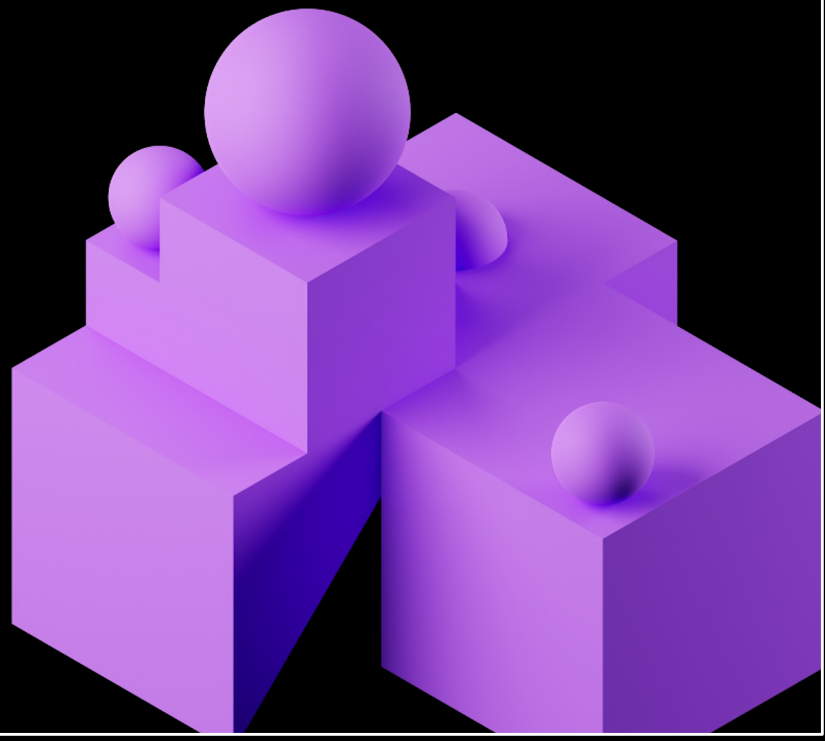
Building utilities using Unity Catalog's metadata

- Catalog as PDF
- State of the Lakehouse Dashboard
- Users' Activity Monitor
- User Privileges Report

DATA+AI SUMMIT
BY databricks

Thank You!

Your feedback is important!
Feel free to reach out 🙂



Gilad Asulin

Pulkit Chadha