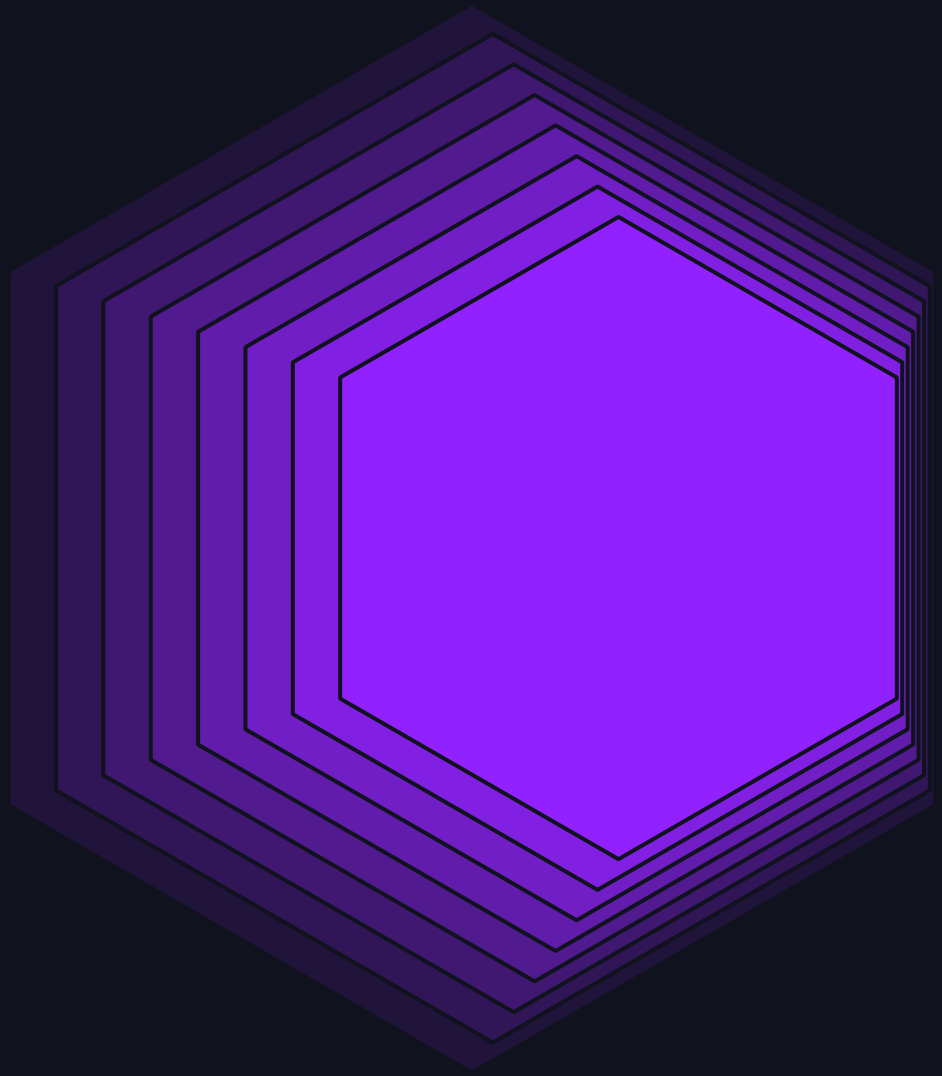
DATA+AI SUMMIT BY databricks

Open Sourcing Unity Catalog



Ramesh Chandra, Denny Lee, Michelle Leon
June, 2024

Who Are We?

Michelle Leon

- Staff Product Manager
  - Previously Webflow, Airbnb
- Based in San Francisco
- Talk to me about
  - Delta Lake
  - Unity Catalog interoperability
  - Best burritos in the Mission neighborhood 🌯

Who Are We?

Ramesh Chandra

- Principal Software Engineer
  - Previously Google, Nutanix
- Based in Mountain View
- Talk to me about
  - Unity Catalog
  - Governance
  - Sharing

Who Are We?

Denny Lee

- Sr. Staff Developer Advocate
  - Previously Microsoft, SAP Concur
- Based in Seattle-area (Kirkland)
- Talk to me about
  - Delta Lake
  - Apache Spark™
  - MLflow
  - Unity Catalog
  - Coffee
  - Cycling

Agenda

Overview

API spec

Server

Client

What’s next

Overview

Challenges today

Most cloud data platforms lack open access

Data and AI assets are arbitrarily siloed

Governance across Data + AI is inconsistent and hard

We built Unity Catalog to address these problems

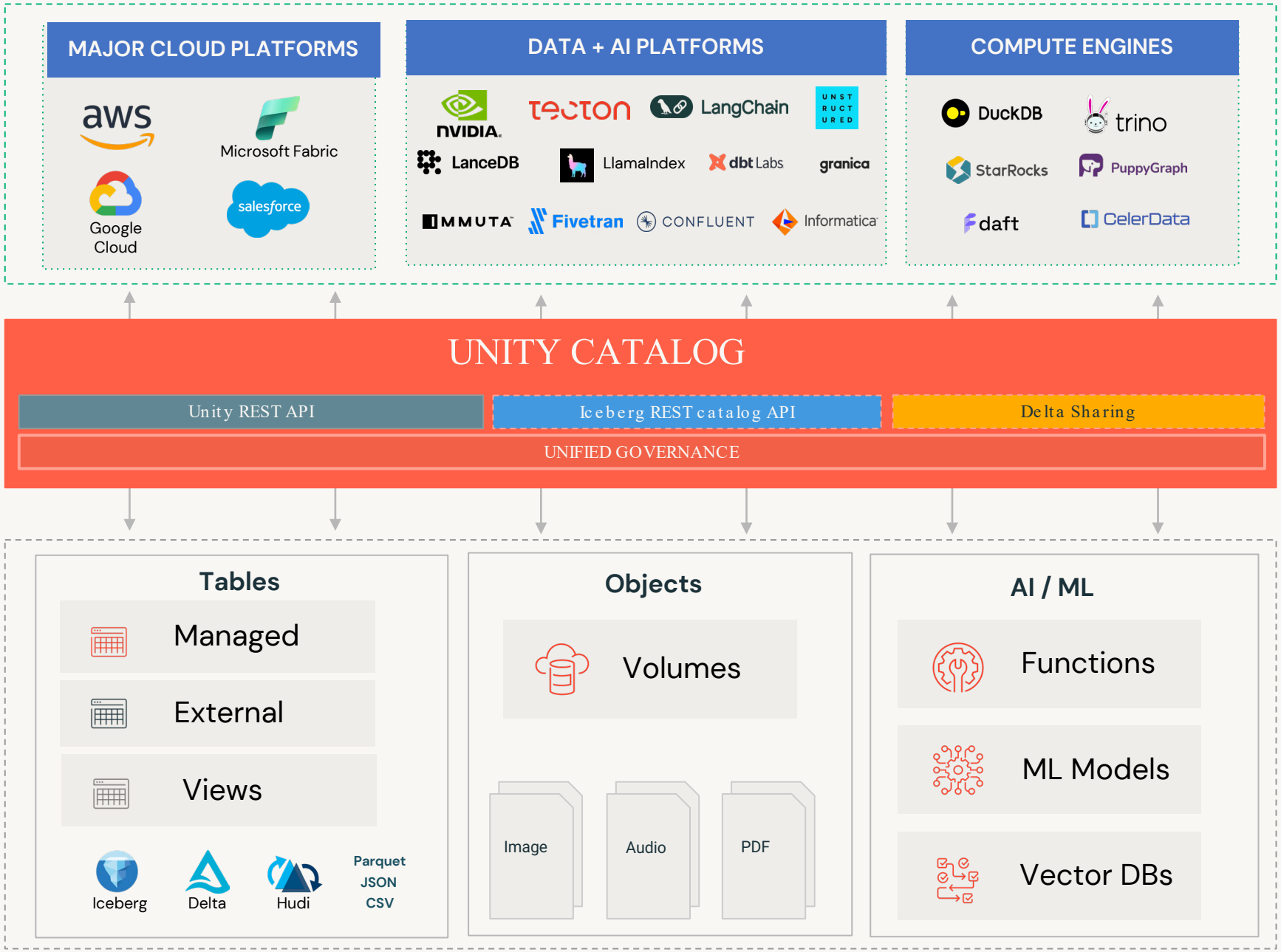
Unity Catalog: The industry's only universal catalog for Data and AI

Any engine
Client ecosystem

Any client
Universal standard

Any asset
Data + AI assets

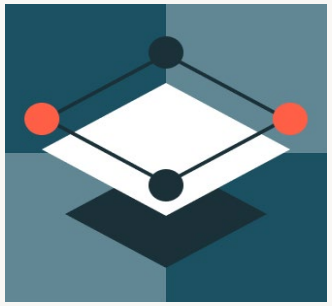
Any format
UniForm



And now Unity Catalog is open source!

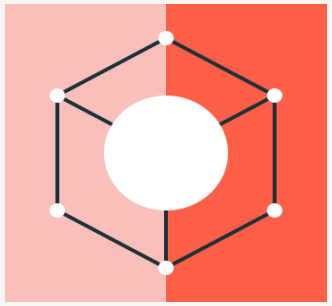
Introducing

Unity Catalog Open Source



Open

Open APIs and OSS server maximize flexibility and customer choice



Interoperable

Universal interface supports any format, engine, data and AI asset

Unified

Unified governance across tabular, non-tabular data and AI assets

Available today

OpenAPI spec

Managed tables APIs

External tables APIs

Volumes APIs

Functions APIs

Credential vending APIs

OSS server

OSS server

Available in new Unity Catalog Github repo

New developer resources

Unity Catalog OSS SDK

REST API docs

Updated Databricks SDKs (Java, Python, Go)

Secure credential vending

Unity Catalog secure credential vending

Available on Databricks in Private Preview

LF AI & DATA

SANDBOX PROJECT

Unity Catalog

The only open and interoperable catalog for data and AI

{Iceberg REST catalog APIs }

ICEBERG ECOSYSTEM

- Amazon Web Services
- Microsoft Azure
- Google Cloud
- Snowflake
- Salesforce
- Apache Impala
- Clickhouse
- Daft
- Doris
- Dremio
- Flink
- Hive
- Presto
- Apache Spark
- StarRocks
- Trino

Unity Catalog

DELTA ECOSYSTEM

- Amazon Web Services
- Azure
- Google Cloud
- Airbyte
- Arrow
- Athena
- Apache Ballista
- aws-pandas-sdk
- beam
- Daft
- Dask
- Apache DataFusion
- Delta-RS
- DLT (spark-r)
- DuckDB
- EMR
- Flink
- Glue
- Hive
- Pandas
- Polars
- PrestoDB
- Pulsar
- PuppyGraph
- Python
- Ray
- Rust
- Apache Spark
- StarRocks
- Startree (pinot)
- Trino

UNITY CATALOG PARTNERS

- aws
- Azure
- Google Cloud
- salesforce
- NVIDIA
- CONFLUENT
- daft
- databricks
- dbt Labs
- DuckDB
- Fivetran
- granica
- IMMUTA
- Informatica
- LanceDB
- LangChain
- LlamaIndex
- ONEHOUSE
- PuppyGraph
- StarRocks
- tecton
- UNSTRUCTURED
- Apache XTable

Architecture Overview

Guiding Principles

Built with openness in mind from get-go

- Careful separation of engine from catalog

Designed to be universal for Data+AI

- Support for tabular, unstructured, and AI/ML assets

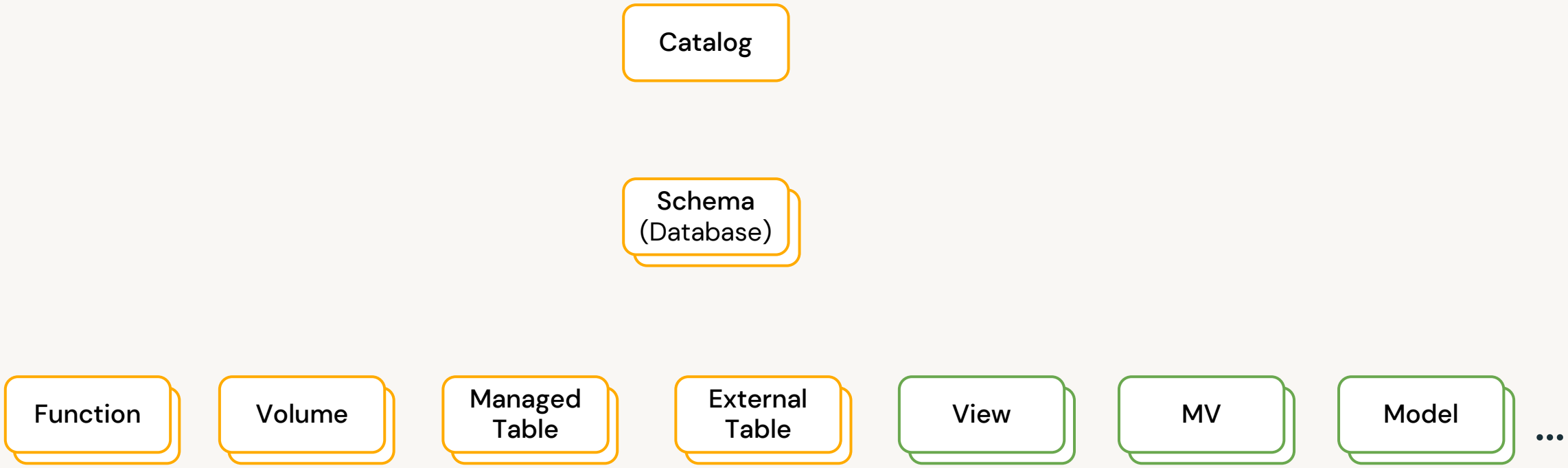
Governance from ground up

- Foundational building blocks like credential vending

Prove in real world before opening it

- Battle tested with thousands of customers

Three-level Namespace



API Overview

For each resource type: Create, Read, Update, Delete, List APIs

Temporary credential vending for resources with storage (tables, volumes)

Commit API for table creates and updates (coming soon)

Let's browse API docs

User Experience

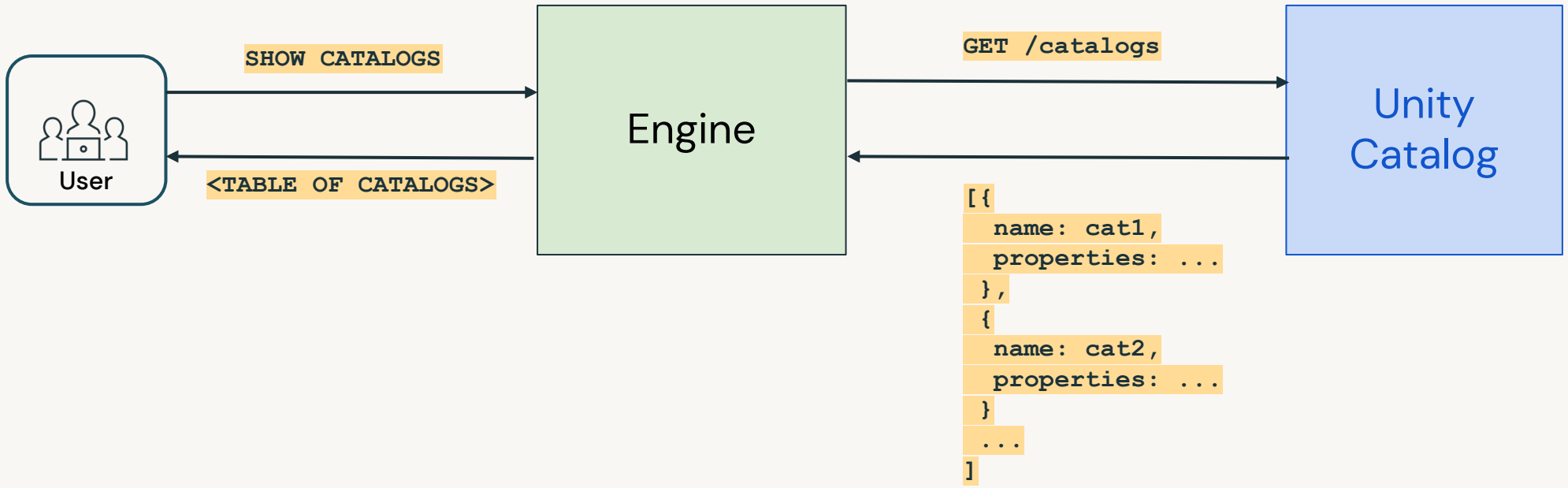
Spark

```
bin/spark-shell \
--packages org.apache.open_catalog:open-catalog-spark-connector:1.0.0 \
--conf spark.sql.extensions=org.apache.open_catalog.spark.extensions.OpenCatalogSparkSessionExtensions \
--conf spark.sql.catalog.my_catalog=org.example.unitycatalog.spark.SparkCatalog \
--conf spark.sql.catalog.my_catalog.uri=<unity-catalog-api-endpoint> \
--conf spark.sql.catalog.my_catalog.credential=<personal-access-token>
```

DuckDB

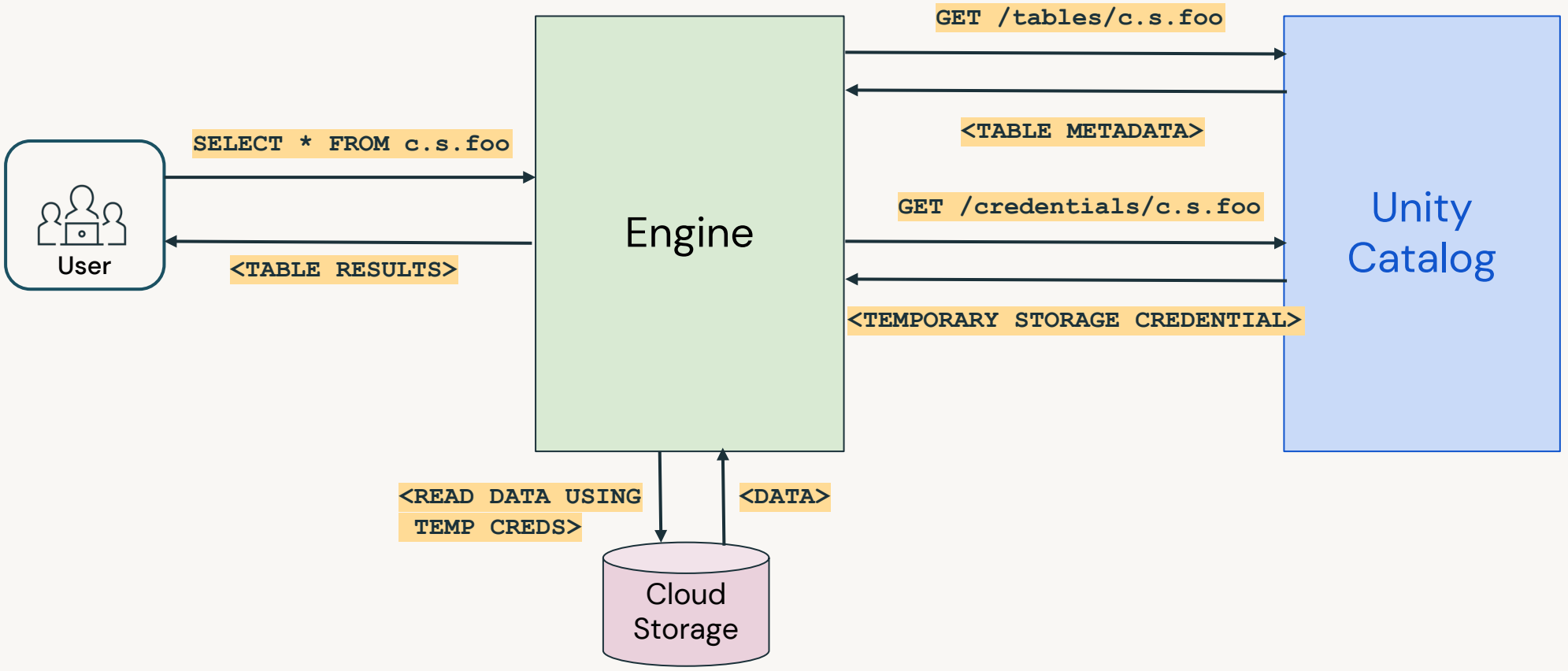
```
CREATE SECRET (
      TYPE UC,
      TOKEN <personal-access-token>,
      ENDPOINT '<unity-catalog-api-endpoint>'
);
ATTACH '<remote_catalog_name>' AS <local_catalog_name> (TYPE UC_CATALOG);
```

Life of a command



*representative APIs

Life of a *more interesting* command



*representative APIs

Demo Time: Table reads with DuckDB

Temporary Credential Vending

What was before?

jane@acme.co

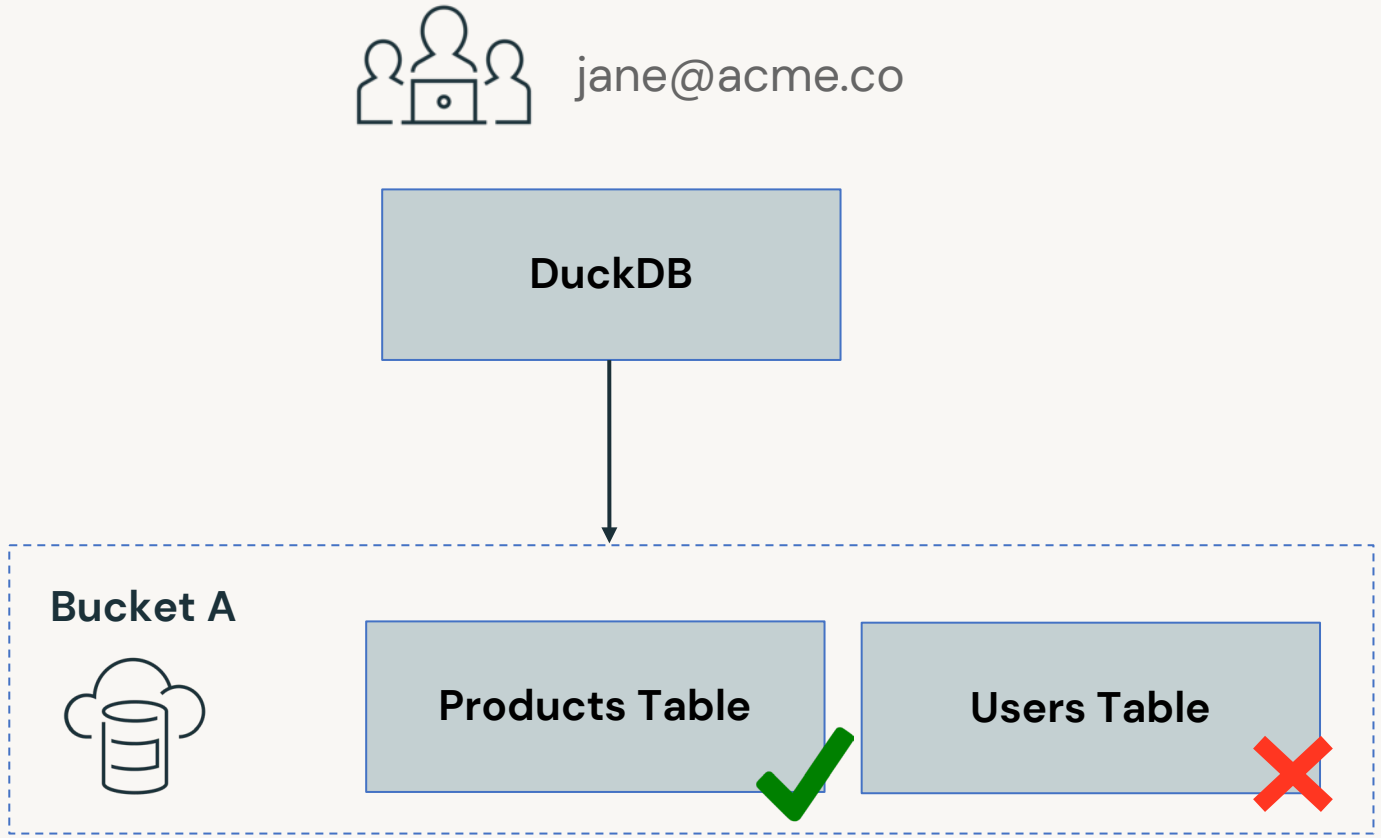
DuckDB

Bucket A

Products Table

Users Table

What was before?



Jane needs SELECT access to only the Products Table

Granting storage access also gives access to Users Table

What is a temporary credential?

- Time-limited, downscoped credential generated on demand
  - Can be scoped down to a single higher-level object (table, volume, model)
  - Cannot access files outside of the object
- Foundational building block for higher-level governance mechanisms
  - Grants, RBAC, ABAC etc.,
  - Powerful – scales access management
- Leverages mechanisms of cloud storage providers
  - AWS session tokens, Azure delegation SAS credentials, etc.,

Code Deep Dive

API – https://go.unitycatalog.io/apidocs

Server – https://github.com/unitycatalog/unitycatalog/tree/main/server/src/main/java/io/unitycatalog/server

CLI – https://github.com/unitycatalog/unitycatalog/blob/main/examples/cli/src/main/java/io/unitycatalog/cli/TableCli.java

Let’s take it for a spin!

What’s next

OSS Community Engagement

- Iceberg REST Catalog and Unity Catalog discussions
- Slack
- GitHub
- Distribution Lists: User & Dev
- Documentation
- API/Swagger
- Community Meetings
- Events

Mapping Unity Catalog and Iceberg REST Catalog

Work with the community to map out differences, for example:

operationId	Iceberg REST Catalog API	Unity Catalog API
createTable	✓	✓
listTables	✓	✓
loadTable	✓	
deleteTable		✓
createView	✓	
createVolume		✓

go.unitycatalog.io/<>

go.unitycatalog.io/slack

go.unitycatalog.io/GitHub

go.unitycatalog.io/user

go.unitycatalog.io/dev


Unity Catalog