UNITY CATALOG

1. What is Unity Catalog?

Unity Catalog is a **centralized data governance and management solution** in Databricks. It acts as a **single place to organize, secure, and manage access to all your data and AI assets** across your entire data lakehouse environment.

Overview of Unity Catalog

Unity Catalog is a centralized data catalog that provides access control, auditing, lineage, quality monitoring, and data discovery capabilities across Azure Databricks workspaces.

Key features of Unity Catalog include:

- **Define once, secure everywhere**: Unity Catalog offers a single place to administer data access policies that apply across all workspaces in a region.
- **Standards-compliant security model**: Unity Catalog's security model is based on standard ANSI SQL and allows administrators to grant permissions in their existing data lake using familiar syntax.
- **Built-in auditing and lineage**: Unity Catalog automatically captures user-level audit logs that record access to your data. Unity Catalog also captures lineage data that tracks how data assets are created and used across all languages.
- **Data discovery**: Unity Catalog lets you tag and document data assets, and provides a search interface to help data consumers find data.
- **System tables**: Unity Catalog lets you easily access and query your account's operational data, including audit logs, billable usage, and lineage.

Why Unity Catalog?

- **Unified Governance:** Manage access policies consistently across different data assets like tables, files, and machine learning models.
- **Fine-Grained Access Control:** Control who can view or modify data down to columns in tables.
- **Data Discovery:** Make it easy for users to find and understand the data available.
- **Audit and Compliance:** Track who accessed or changed data for security and regulatory requirements.

2. Core Concepts: Metastore, Catalog, Schema, and Table

Unity Catalog uses a **3-level namespace hierarchy** within a **Metastore** to organize data:

2.1 Metastore

- The **Metastore** is the top-level container in Unity Catalog.
- It holds all catalogs and metadata about your data.
- You create a Metastore once and connect one or more Databricks workspaces to it.
- It's like a "master catalog" for all your data assets.

2.2 Catalog

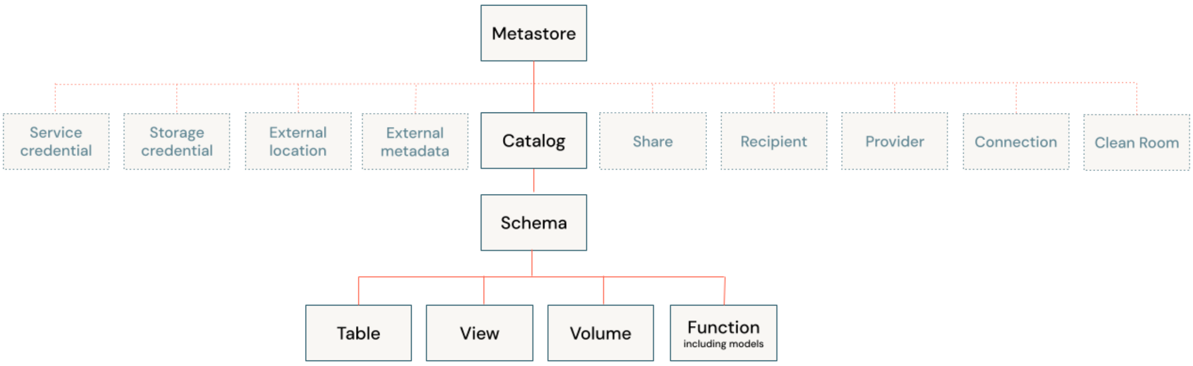
- A **Catalog** is a container within the Metastore.
- Think of it as a big folder grouping related datasets.
- Example: sales_data, marketing_data, or hr_data.

2.3 Schema (Database)

- Inside a Catalog, there are **Schemas**.
- Schemas organize data into smaller, logical groups.
- Example: Inside sales_data catalog, you could have customer_info, transactions, and products schemas.

2.4 Table

- The **Table** is the actual data storage object inside a Schema.
- It contains rows and columns like a spreadsheet or database table.
- Tables can be permanent, temporary, or views.



3. Benefits of Unity Catalog

- **Single Source of Truth:** Data assets managed in one place.
- **Cross-Workspace Sharing:** Share data securely across multiple Databricks workspaces.
- **Simplified Data Access Management:** Permissions set once, inherited down the hierarchy.
- **Column-Level Security:** Restrict access to sensitive data within tables.
- **Support for Multiple Clouds:** Works with AWS, Azure, and Google Cloud.

4. File format support

Unity Catalog supports the following table formats:

- Managed tables must use the delta table format.
- External tables can use delta , CSV, JSON, parquet, text.

5. Summary of Unity Catalog Namespace

Level	Description	Example Name
Metastore	Top-level container	my_metastore
Catalog	Group of related schemas	sales_catalog
Schema	Group of related tables	customer_info
Table	Actual data table or view	customers

6. Why Use Unity Catalog?

- Simplifies data management for multiple teams.
- Improves security with fine-grained controls.
- Makes sharing and collaboration easier.
- Supports compliance and auditing requirements.