**UNITY CATALOG:**

It is a Databricks control center for all data.

It helps to:

* Organize data
* Control access
* Track usage

Why to use Unity Catalog?

* Single control point for all data assets
* Easy permission management using AAD groups
* Works across multiple Databricks workspaces
* Automatically tracks lineage

| **Component** | **Description** | **Azure Example** |
| --- | --- | --- |
| **Metastore** | The top-level storage and governance unit | company\_metastore in East US region |
| **Catalog** | Top-level container inside a metastore | sales\_data |
| **Schema** | Logical grouping of tables/views | usa |
| **Table** | Actual dataset | q1\_orders |

UNITY CATALOG IN AZURE – ARCHITECTURE DIAGRAM:

Azure Active Directory (AAD)

Unity Catalog Metastore

Catalog: sales\_data Catalog: marketing\_data

Schema: usa Schema: Europe

Table: q1\_orders Table: campaign\_stats

**METASTORE:**

The Metastore is the storage brain of Unity Catalog. It stores:

* All the names of catalogs, schemas, and tables
* Permissions & data lineage

EXAMPLE:

Metastore Name: company\_metastore

Cloud Storage: s3://my-company-metastore

Workspaces attached: sales\_workspace, analytics\_workspace

**THREE LEVEL NAMESPACE:**

catalog.schema.table

means:

Catalog → Big top-level folder

Schema → Subfolder inside a catalog

Table → Data inside a schema

EXAMPLE:

from pyspark.sql import SparkSession

spark = SparkSession.builder \

.appName("UnityCatalogExample") \

.getOrCreate()

# Create Catalog

spark.sql("CREATE CATALOG IF NOT EXISTS sales")

# Create Schema

spark.sql("CREATE SCHEMA IF NOT EXISTS sales.north\_region")

# Create Table

spark.sql("""

CREATE TABLE IF NOT EXISTS sales.north\_region.q1\_orders (

order\_id STRING,

amount DOUBLE

)

USING DELTA

""")

# Insert Data

spark.sql("""

INSERT INTO sales.north\_region.q1\_orders VALUES

('O-001', 100.5),

('O-002', 250.75)

""")

# Query Data

df = spark.sql("SELECT \* FROM sales.north\_region.q1\_orders")

df.show()