DISCLAIMER

This course is independently created.
It is not affiliated with or endorsed by Databricks Inc.
All explanations, concepts, and practice materials are original.
They are designed only for educational purposes.
This course does NOT contain any actual certification exam questions.
The content is based on publicly available documentation, real-world scenarios, and personal experience.
All product names, logos, and trademarks are property of their respective owners.
Their use here is purely for identification and educational purposes.
Always refer to the official Databricks documentation for the most accurate and up-to-date information.

https://www.databricks.com/learn/certification/data-engineer-associate

https://docs.databricks.com/

What is a LAKEHOUSE?

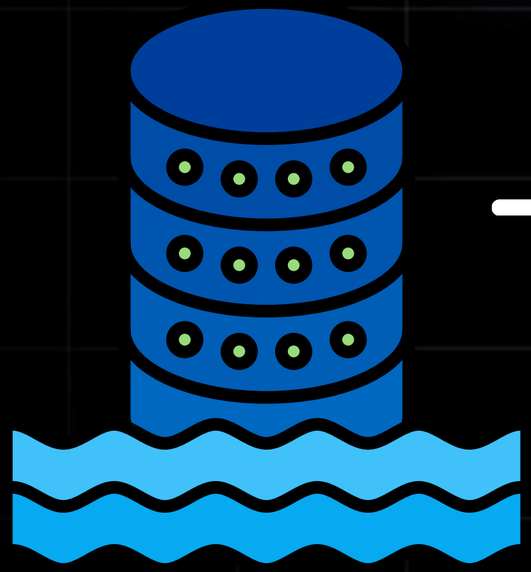
1980

3 Vs

- VOLUME
- VELOCITY
- VARIETY

What is a LAKEHOUSE?

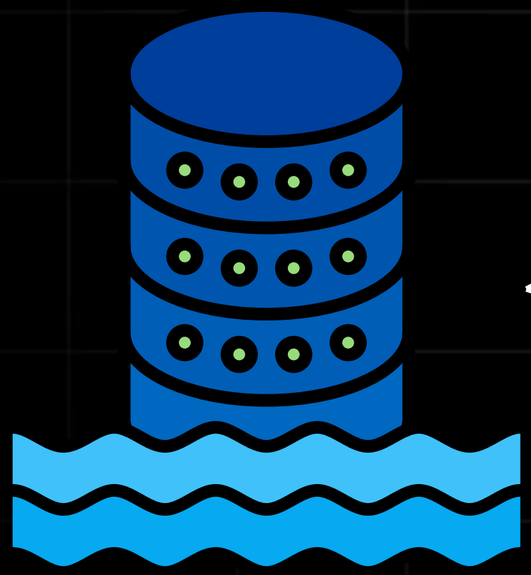
2000



3 Vs

- VOLUME
- VELOCITY
- VARIETY

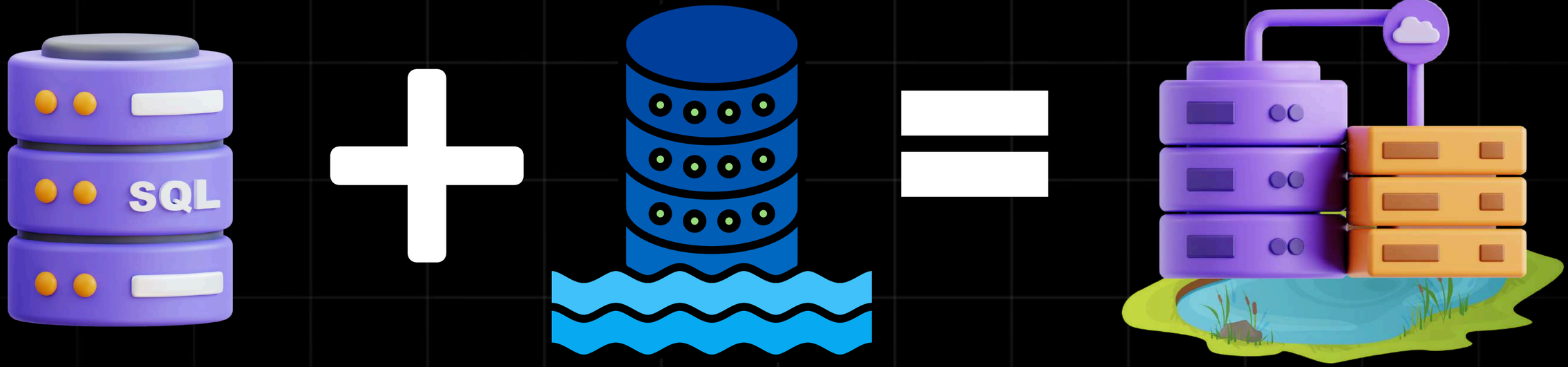
What is a LAKEHOUSE?



Data Analyst

Data Scientist

What is a LAKEHOUSE?



What is a DELTA LAKE ?

What is a DELTA LAKE?

raw_dataday1.parquet

raw_dataday2.parquet

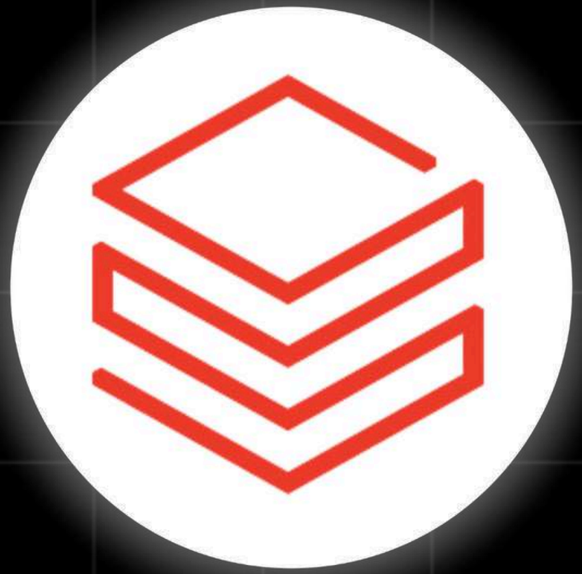
_delt_log

APACHE SPARK

Apache Spark is a fast, general-purpose distributed computing engine designed for big data analytics.

DATABRICKS

- DLT
- ALERTS
- SQL WAREHOUSE
- ETL WORKFLOW



- CLUSTER
- NOTEBOOK
- OPTIMIZED ENGINE
- UNITY CATALOG

DATABRICKS Architecture

CONTROL PLANE

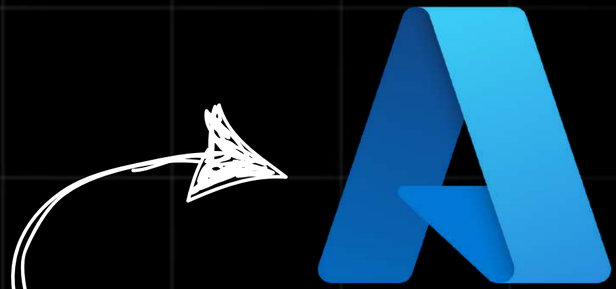
The control plane includes the backend services that Azure Databricks manages in your Azure Databricks account. The web application is in the control plane.

COMPUTE PLANE

The compute plane is where your data is processed.

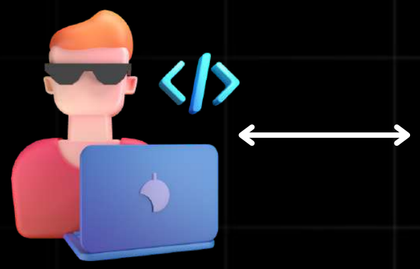
DATABRICKS Architecture

COMPUTE PLANE



SERVERLESS COMPUTE

CLASSIC COMPUTE



Control Plane

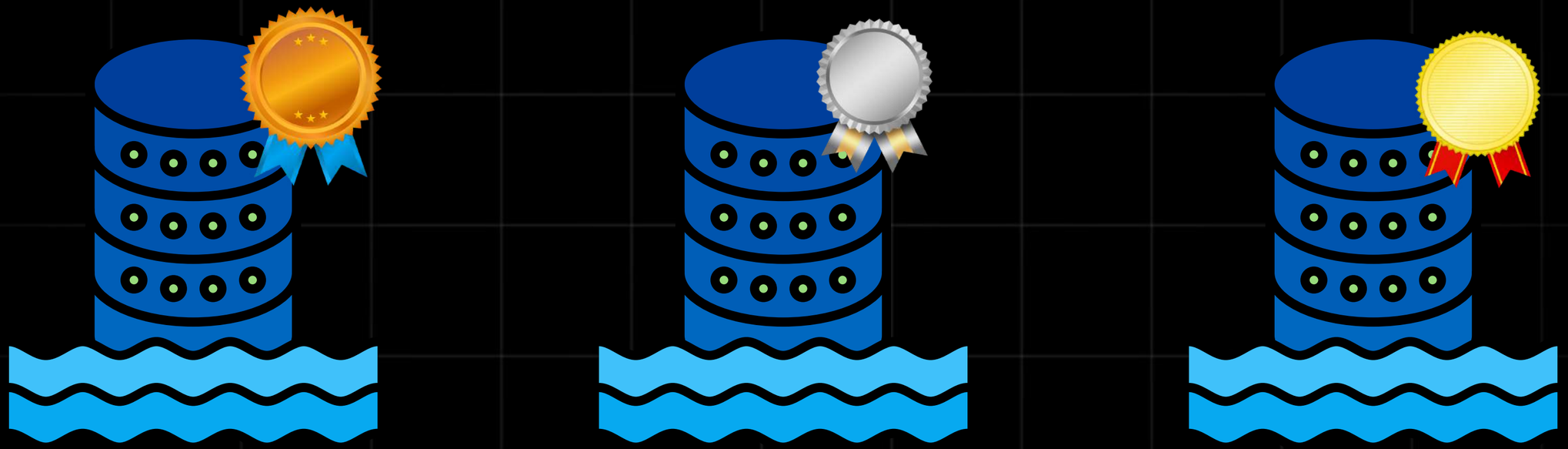
Serverless Compute

Storage Account

Your Resources

Classic Compute

MEDALLION Architecture



DATABRICKS COMPUTE

ALL PURPOSE

Use Case: For development, exploration, notebooks, and ad-hoc analysis.

Features:

Supports multiple users working together.

Can be used with notebooks, dashboards, and APIs.

Costs more because it's always running while in use.

Good for data exploration and testing pipelines.

DATABRICKS COMPUTE

JOB COMPUTE

Use Case: For running production jobs or scheduled pipelines.

Features:

- Spins up when a job starts, terminates when the job ends.
- More cost-effective than all-purpose clusters.
- Can’t be used for interactive notebooks.

DATABRICKS COMPUTE

POOLS

Use Case: Reduce startup time and cost for clusters.

Features:

- Pre-warmed compute resources.
- Saves cost by avoiding cold-start times.
- Can be used for both job and all-purpose clusters.

DATABRICKS COMPUTE

SQL WAREHOUSE

Use Case: For running SQL queries and powering dashboards.

Features:

- Connects easily to BI tools like Power BI, Tableau.
- Optimized for SQL queries, not Spark jobs.

DATABRICKS COMPUTE

SERVERLESS

Use Case: Hands-off resource management—best for cost and scaling.

Features:

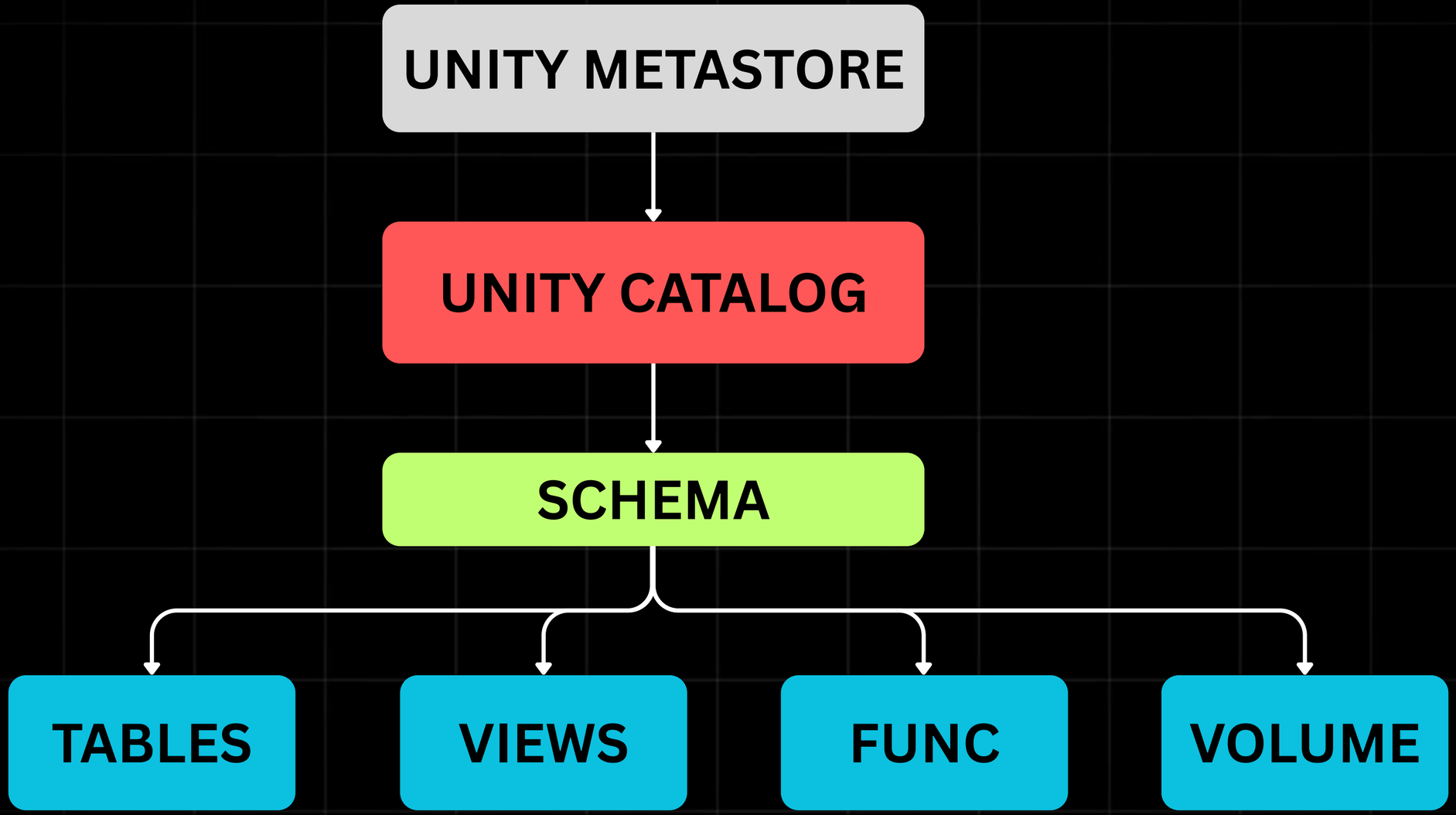
- Auto-scales and auto-terminates.
- You don’t manage instances—Databricks does.
- Ideal for unpredictable workloads.

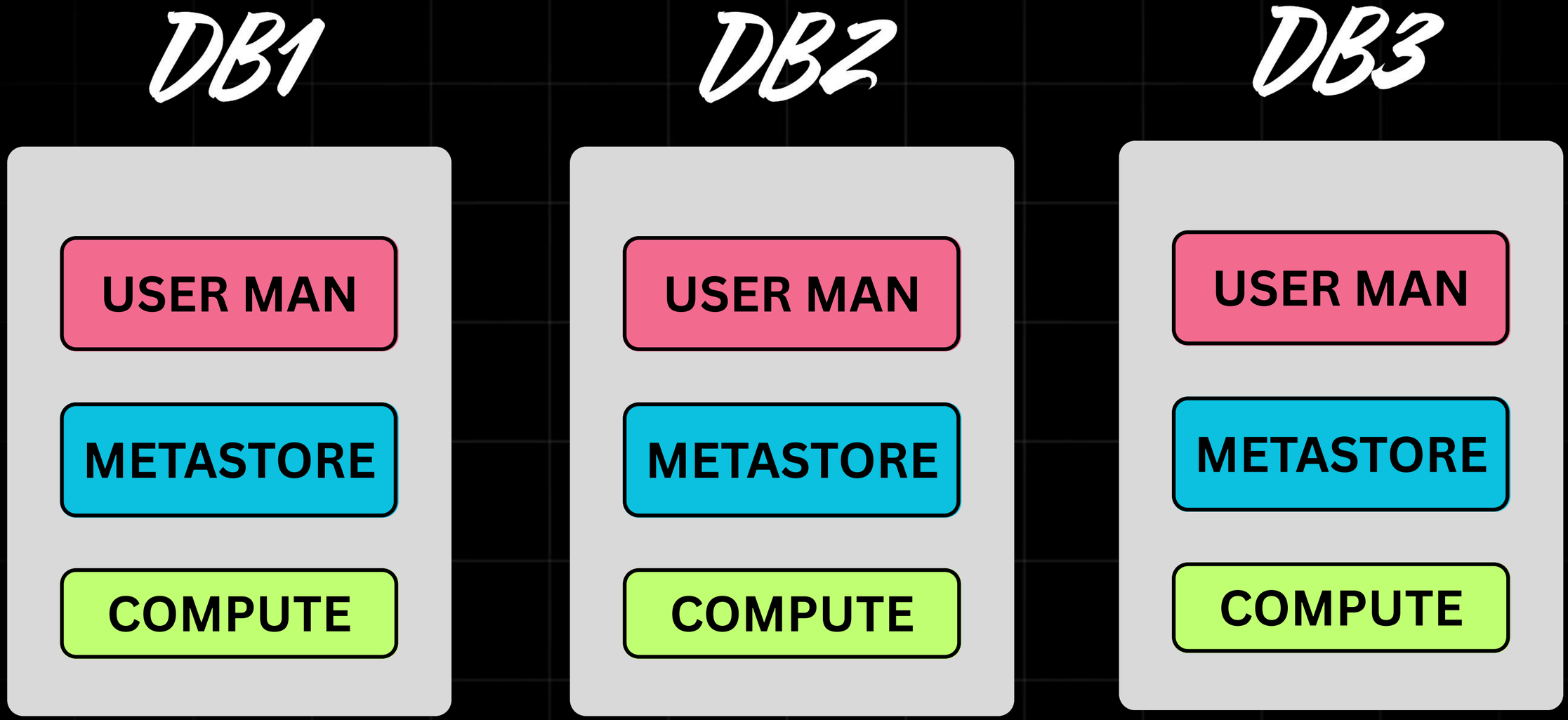
UNITY CATALOG

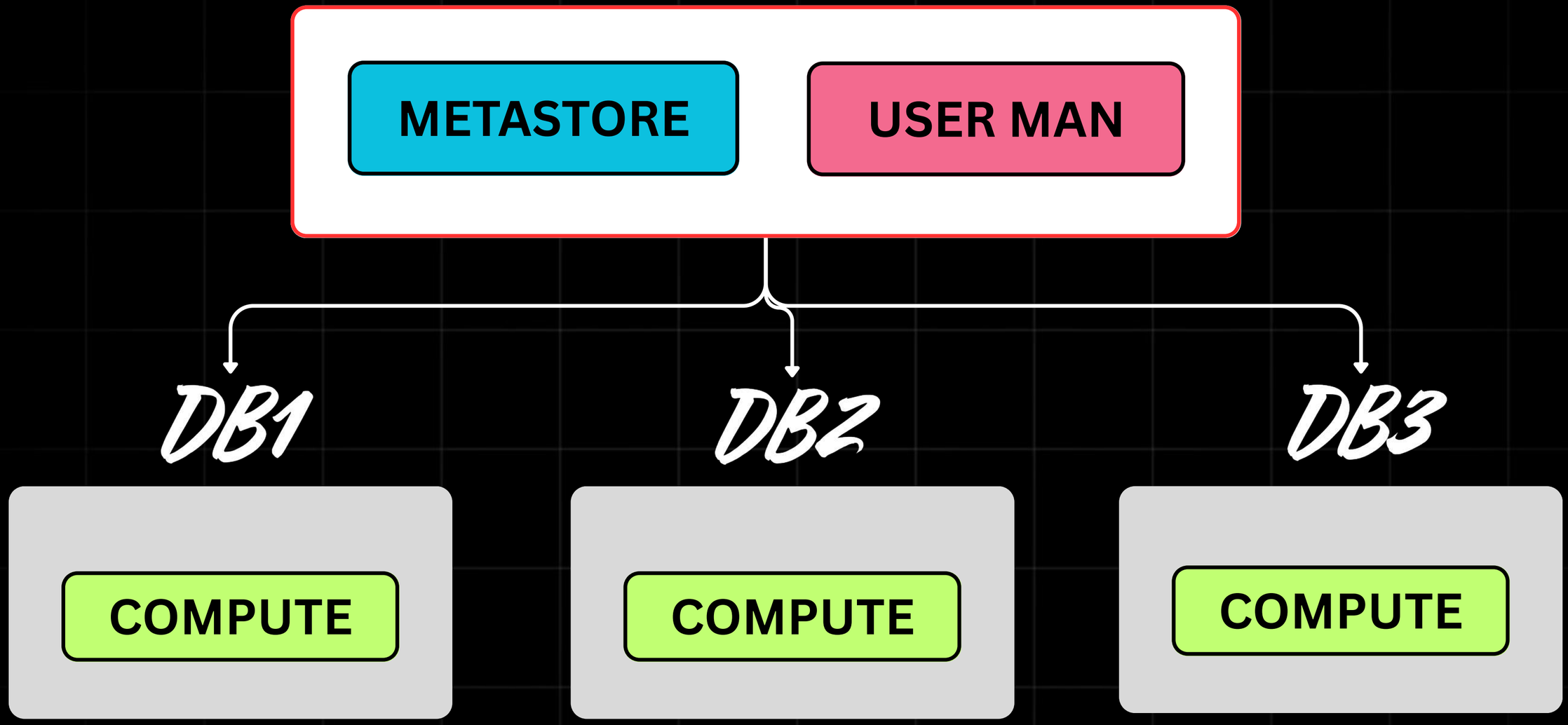
Unity Catalog is a centralized data governance solution in Azure Databricks that helps manage data access, track usage, and enable discovery across all workspaces.

Key highlights:

- Centralized access control: Define data policies once and enforce them across all workspaces.
- Auditing and lineage: Automatically logs data access and tracks how data flows and is transformed.
- Data discovery: Supports tagging, documentation, and search to help users find relevant datasets.


UNITY METASTORE
UNITY CATALOG
SCHEMA
TABLES
VIEWS
FUNC
VOLUME


DB1
USER MAN
METASTORE
COMPUTE
DB2
USER MAN
METASTORE
COMPUTE
DB3
USER MAN
METASTORE
COMPUTE


METASTORE
USER MAN
DB1
DB2
DB3
COMPUTE
COMPUTE
COMPUTE

MANAGED Tables

METASTORE

DEFAULT STORAGE

CLOUD

EXTERNAL STORAGE

EXTERNAL Tables

METASTORE

DEFAULT STORAGE

CLOUD

EXTERNAL STORAGE

VOLUMES

CLOUD

EXTERNAL STORAGE

DBUTILS

“dbutils” is a powerful utility library provided by Databricks to help you interact with your Databricks environment more effectively. It allows you to perform a variety of tasks like:

- Managing files and directories
- Working with secrets
- Handling widgets (for parameterization)
- Managing libraries

Databricks FILES

The Databricks Data Intelligence Platform helps data teams across your organization work together and turn data ideas into real solutions by using shared and secure data assets and tools.

Lakehouse Federation

Lakehouse Federation in Databricks is a powerful feature that allows you to query external data sources without moving the data into your Databricks Lakehouse. This means you can run SQL queries on data stored in external systems like MySQL, PostgreSQL, Azure SQL DB, Snowflake, and more — directly from Databricks.

Lakehouse Federation

- You define read-only connections to external databases.
- These connections are registered as catalogs in Unity Catalog.
- The actual data stays in the external system — only the results are returned to Databricks.

SPARK STREAMING

AUTOLOADER

Auto Loader in Databricks is a powerful feature designed to efficiently and incrementally load data from cloud storage into Databricks, especially when new files keep arriving continuously. It’s ideal for building streaming data pipelines without manually tracking new files.

Idempotency

As new files are detected, their metadata is saved in a scalable key-value store (RocksDB) at the checkpoint location of your Auto Loader pipeline. This helps ensure that each file is processed only once.

Schema Inference

When Auto Loader reads data for the first time, it infers the schema by sampling up to 50 GB or 1000 files—whichever comes first. It then saves this schema information in a _schemas folder at the specified cloudFiles.schemaLocation, so it can keep track of any schema changes over time.

Schema Evolution

Auto Loader checks for new columns while processing your data. If it finds any, it stops the stream and throws an UnknownFieldException. But before this happens, Auto Loader performs schema inference on the latest micro-batch and updates the schema location by adding the new columns at the end. It keeps the data types of the existing columns unchanged.

COPY INTO

The COPY INTO SQL command is used to load data from a file path into a Delta table. It’s a repeatable and reliable operation—any files that have already been processed are automatically ignored.

DATABRICKS SQL

Databricks SQL provides data warehousing functionality directly within your existing data lakes, offering high performance and compatibility with open data formats and ANSI SQL standards. It features a built-in SQL editor and tools for creating dashboards, enabling seamless collaboration within the Databricks workspace.

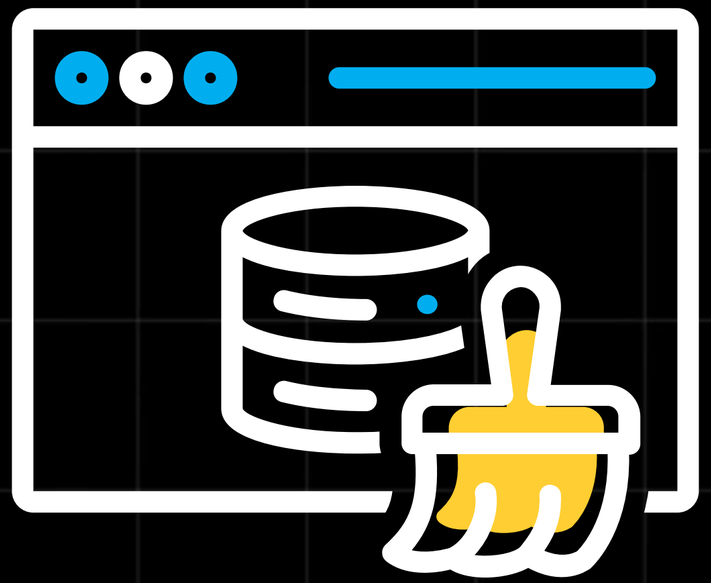
SQL WAREHOUSE

SQL EDITOR

Databricks provides a built-in SQL editor within its user interface that allows you to write queries, explore datasets, and build visualizations. Additionally, you can save your queries and share them with others in your workspace.

SQL QUERIES

QUERY Caching



ALERTS

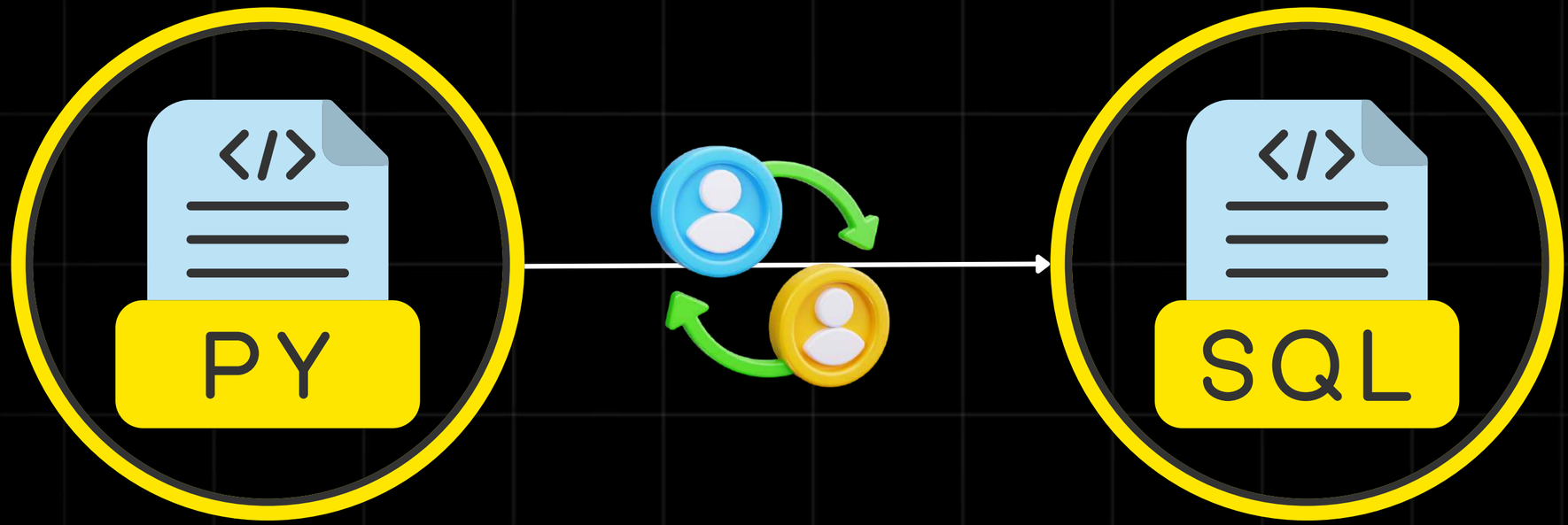
LAKEFLOW JOBS

LAKEFLOW JOBS

Jobs consist of one or more tasks, and support custom control flow logic like branching (if / else statements) or looping (for each statements) using a visual authoring UI. Tasks can load or transform data in an ETL workflow, or build, train and deploy ML models in a controlled and repeatable way as part of your machine learning pipelines.

First Lakeflow Job

Dynamic Reference



SET TASK VALUES

No Task Values are required. The resulting rows will be returned in the form of a LIST. It will be the LIST of DICTIONARIES [for multiple columns]. We can use the data of any specific column from that list using the “KEY” [column name]

SET TASK VALUES

```
dbutils.jobs.taskValues.set(key = "key", value = "value")
```

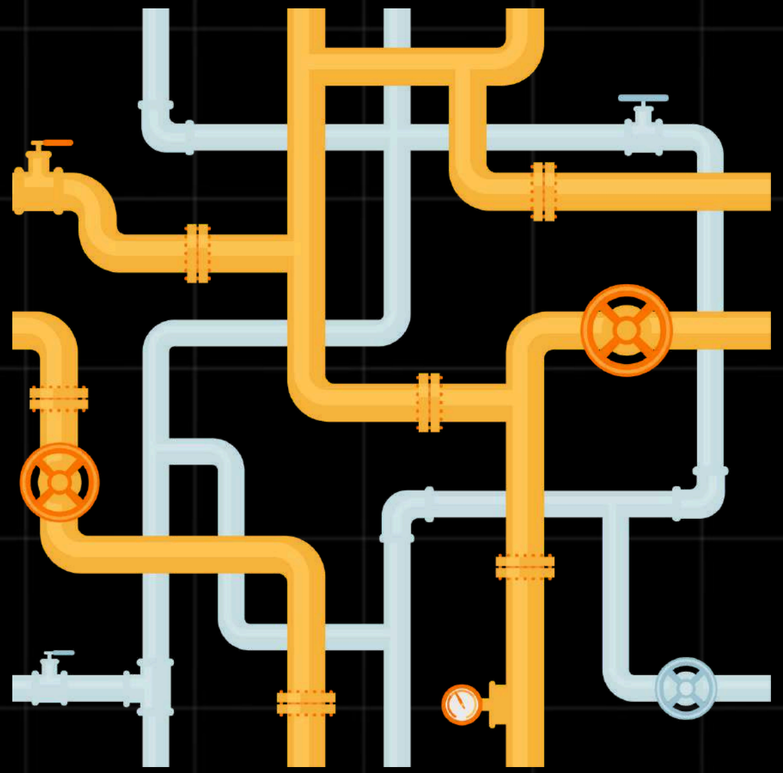
GET TASK VALUES

```
dbutils.jobs.taskValues.get(taskKey = "taskname", key = "outputkey")
```

GET TASK VALUES

“rows”, “first_row”, etc

DELTA LIVE TABLES



DELTA LIVE TABLES

Delta Live Tables is a declarative framework in Databricks that lets you build reliable, automated data pipelines using SQL or Python.

DELTA LIVE TABLES

Data ingestion
Data transformation
Pipeline management
Quality checks
Incremental processing

DLT VS JOBS

Delta Live Tables is a declarative framework in Databricks that lets you build reliable, automated data pipelines using SQL or Python.

DLT CORE COMPONENTS

STREAMING TABLES

MATERIALIZED VIEWS

TEMPORARY VIEWS

DLT APPEND FLOW

DLT AUTO_CDC_FLOW

AUTO_CDC_FLOW is a declarative method used to implement Change Data Capture (CDC) logic.
It helps you apply INSERTs, UPDATEs, and DELETEs automatically from a streaming source (like data with CDC events) to a target Delta table.

SLOWLY CHANGING DIM

Product_ID	Name	Prod_Cat
1	Honey	Food
2	Shirt	Clothing
3	Comb	Clothing

SCD TYPE-1

Product_ID	Name	Prod_Cat
1	Honey	Food
2	Shirt	Clothing
3	Comb	Clothing

BEFORE

Product_ID	Name	Prod_Cat
1	Honey	Food
2	Shirt	Clothing
3	Comb	Hair

AFTER

SCD TYPE-2

Product_ID	Name	Prod_Cat	Eff_StartDate	Eff_ExpDat	InUSE
1	Honey	Food	1/1/2024	1/1/3000	Yes
2	Shirt	Clothing	1/1/2024	1/1/3000	Yes
3	Comb	Clothing	1/1/2024	1/1/3000	Yes

BEFORE

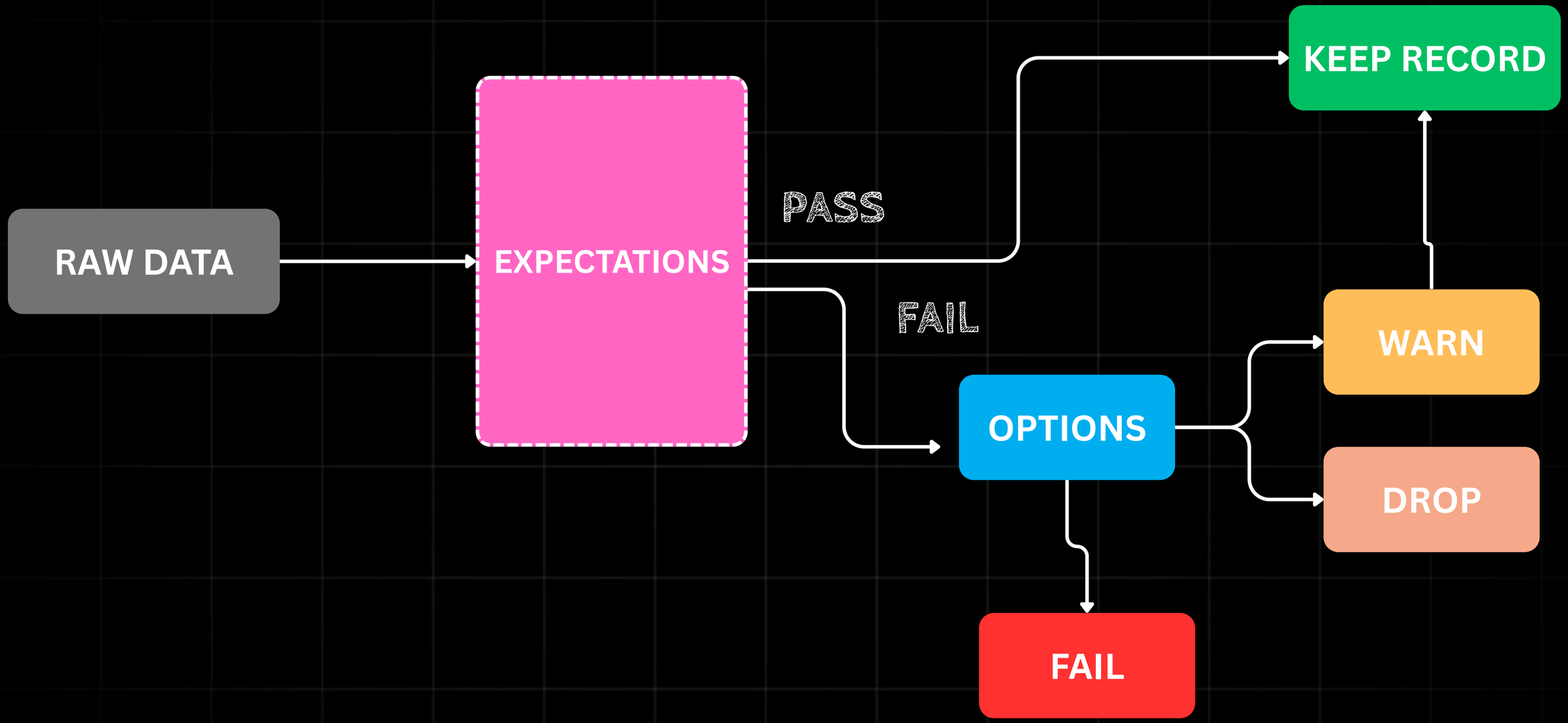
AFTER

Product_ID	Name	Prod_Cat	Eff_StartDa	Eff_ExpDat	InUSE
1	Honey	Food	1/1/2024	1/1/3000	Yes
2	Shirt	Clothing	1/1/2024	1/1/3000	Yes
3	Comb	Clothing	1/1/2024	1/2/2024	No
4	Comb	Hair	1/2/2024	1/1/3000	Yes

EXPECTATIONS

Expectations are like data quality rules that you define within your pipeline to check whether your data meets certain conditions.

Think of it as:
“I expect this column to never be null”
“I expect values in this column to be within a certain range”


RAW DATA
EXPECTATIONS
PASS
FAIL
KEEP RECORD
OPTIONS
WARN
DROP
FAIL

Continuous VS Triggered

ORCHESTRATE

DLT Monitoring

Unity Catalog FUNCTIONS

SCALAR

Unity Catalog FUNCTIONS

TABLE

DATA GOVERNANCE

DATA DISCOVERABILITY

DATA GOVERNANCE

DATA QUALITY

DATA GOVERNANCE

DATA COLLABORATION

DATA GOVERNANCE

DELTA SHARING

DATA GOVERNANCE

DATA ACCESS CONTROL

DATA MASKING

ROW LEVEL SECURITY

DATABRICKS APPS

Databricks Apps let you build interactive web applications inside Databricks, using tools like Dash, Streamlit, or Plotly.
You can turn your notebooks or dashboards into apps that others can easily open and use — without needing to understand or edit the code.

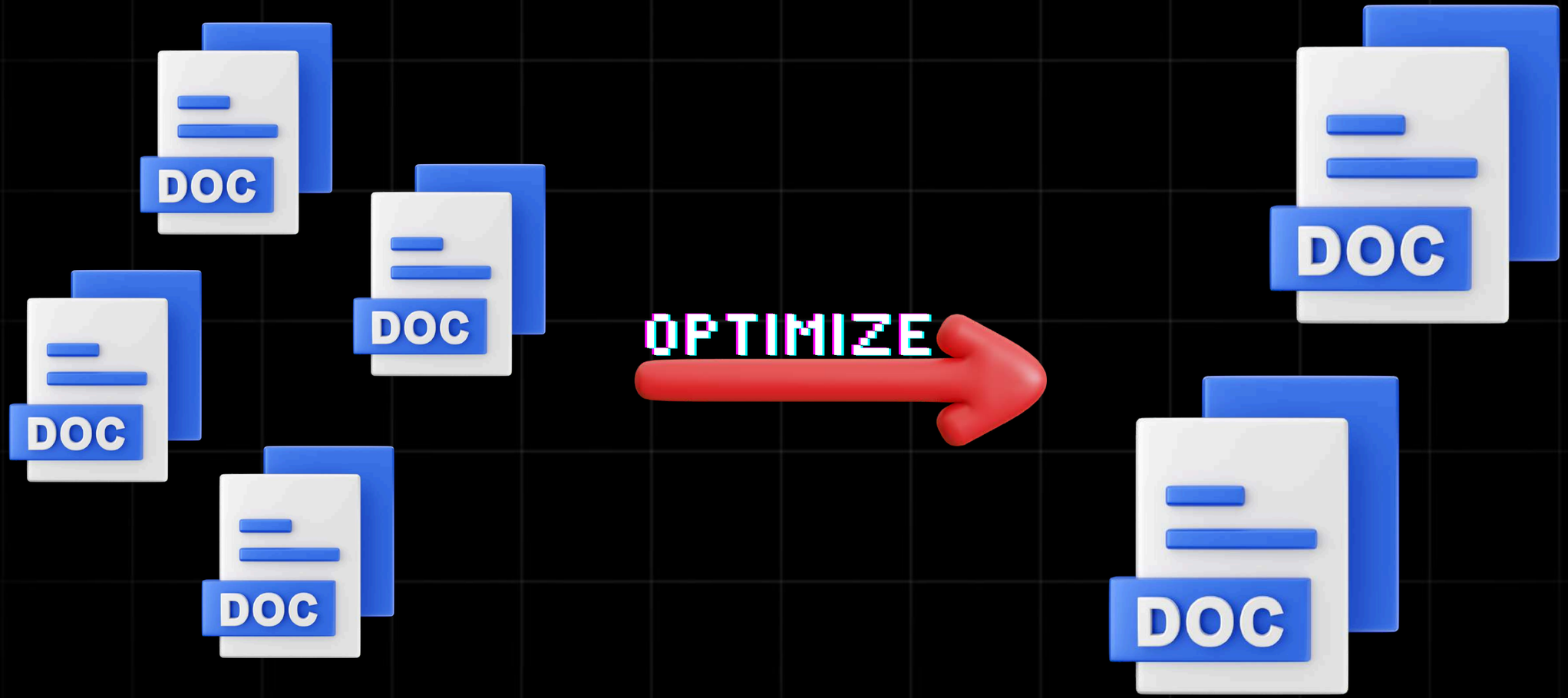
DATABRICKS APPS

Key Features

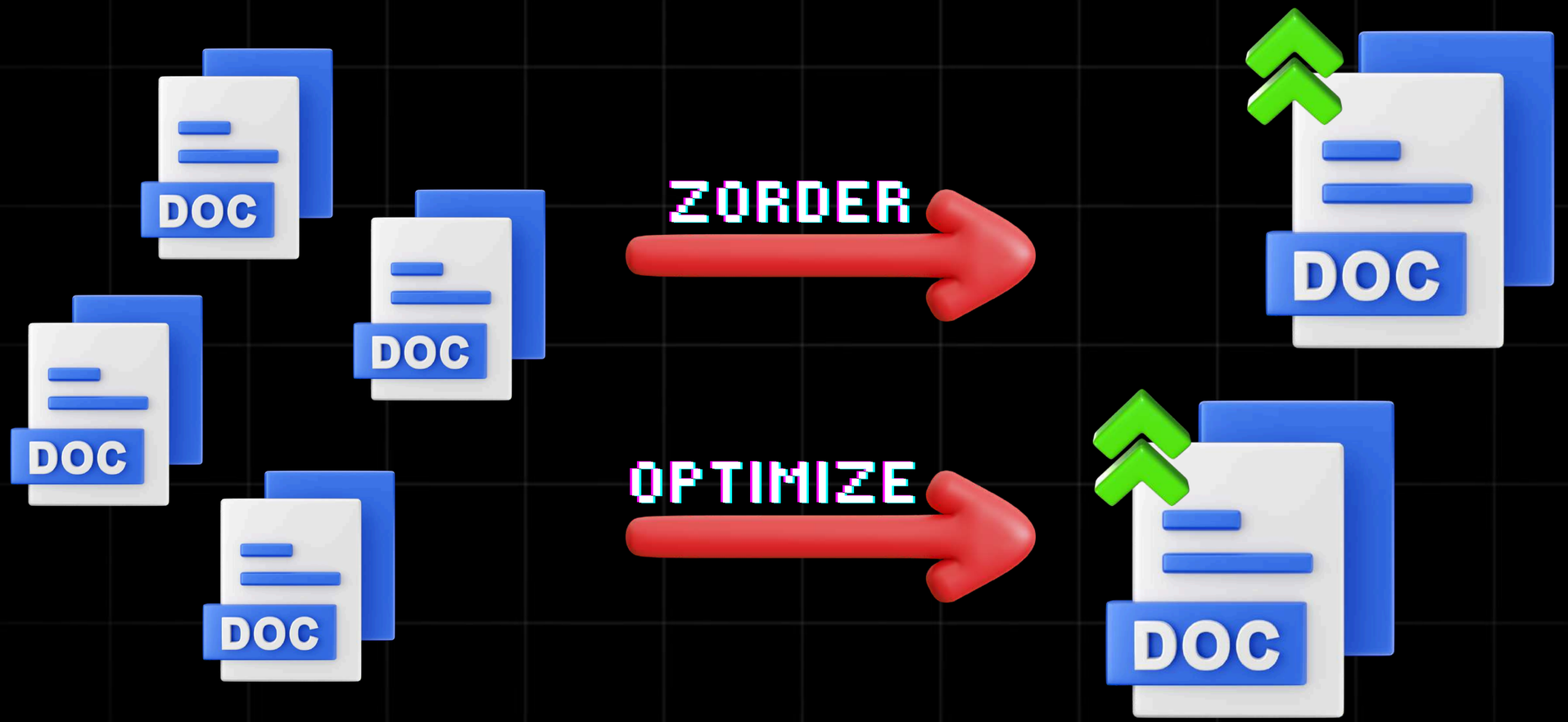
- No extra setup: You don't need to spin up a separate web server.
- Supports popular tools: Use Python-based tools like Streamlit, Dash, and Panel.
- UI built-in: Add sliders, dropdowns, checkboxes, etc. to create a UI for your data.
- Secure & shareable: You can share the app with your team using workspace permissions.

DELTA LAKE FEATURES

OPTIMIZE



ZORDERBY



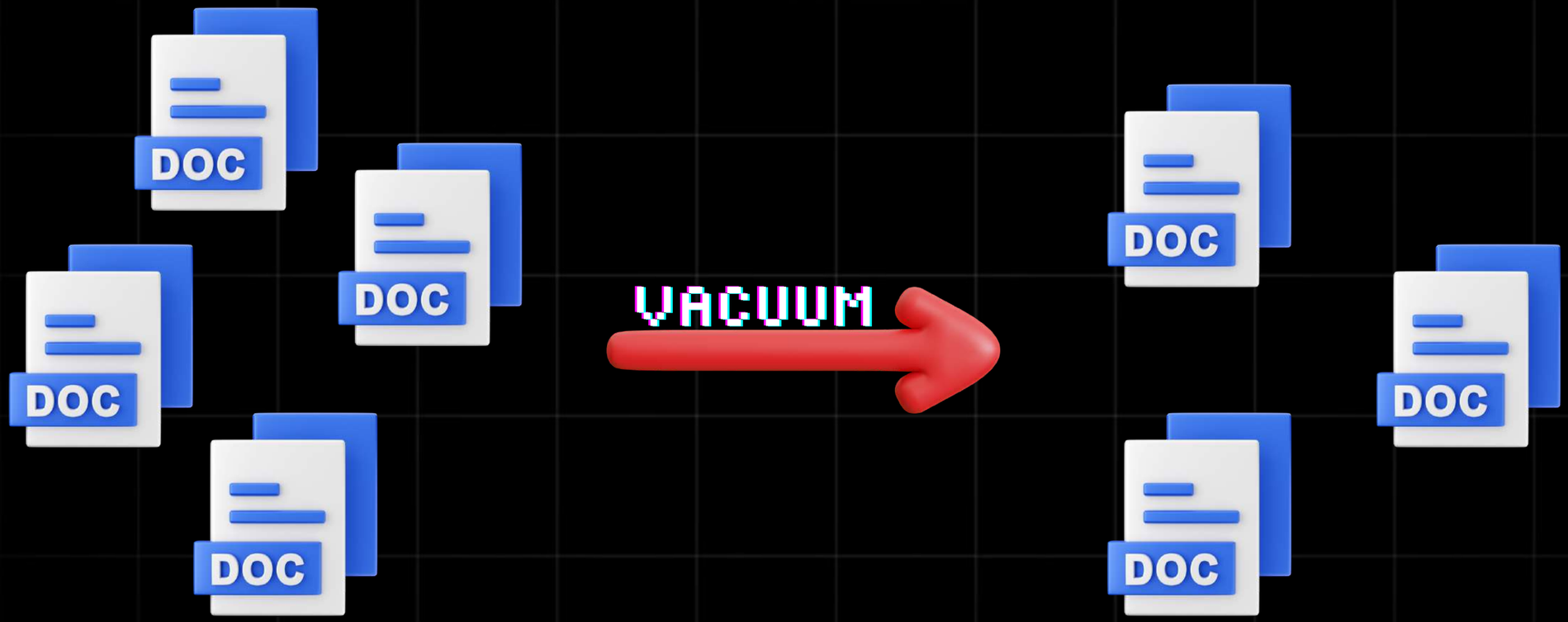
LIQUID CLUSTERING

SAY NO TO PARTITIONS

DATA VERSIONING

TIME TRAVEL

VACUUM



CTAS

DEEP CLONE

SHALLOW CLONE

YOUR NEXT STEPS