iykra Data Fellowship

IYKRA
Data Fellowship Program

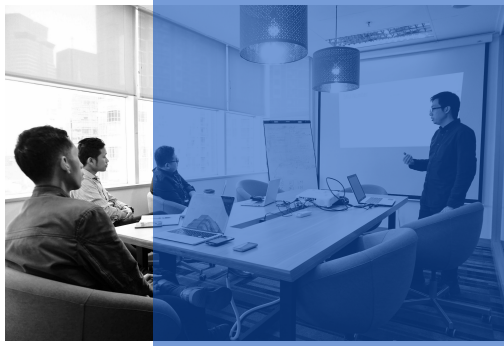
BigQuery and Modern Data Warehousing

by (Bangun Sasongko)

Trainer Profile

Bangun W. Sasongko

Data Engineer at BRI
Ex. Devops Engineer
Bachelor of Physics, Electrical and Instrumentations
@sasongkobgn
sasongkobgn@gmail.com



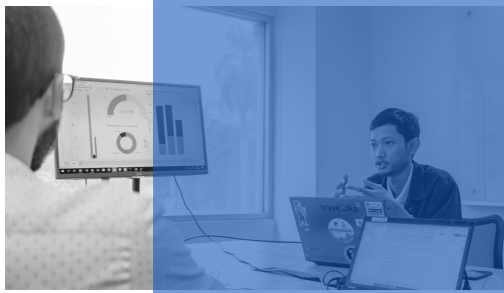
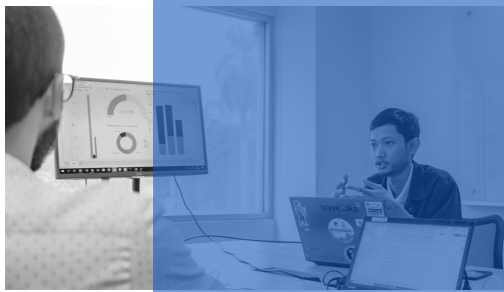
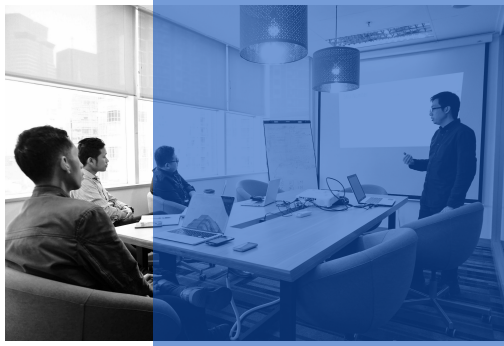


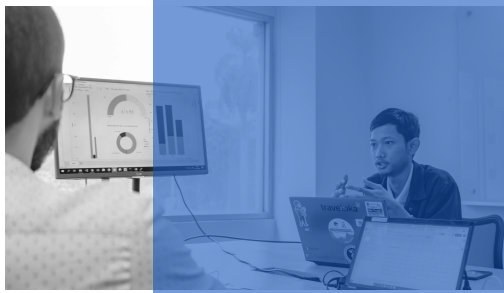
Table of Content



About this Course

This course delves into advanced functionalities of BigQuery, empowering you to optimize data management, processing, and analysis for large datasets

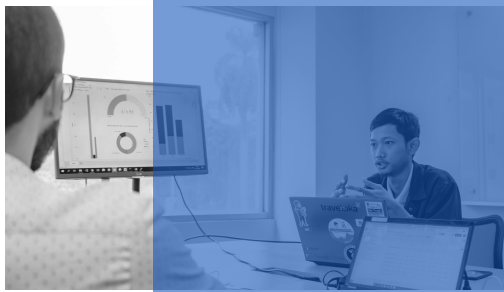




The Objectives

By the end of this course, you will be able to:

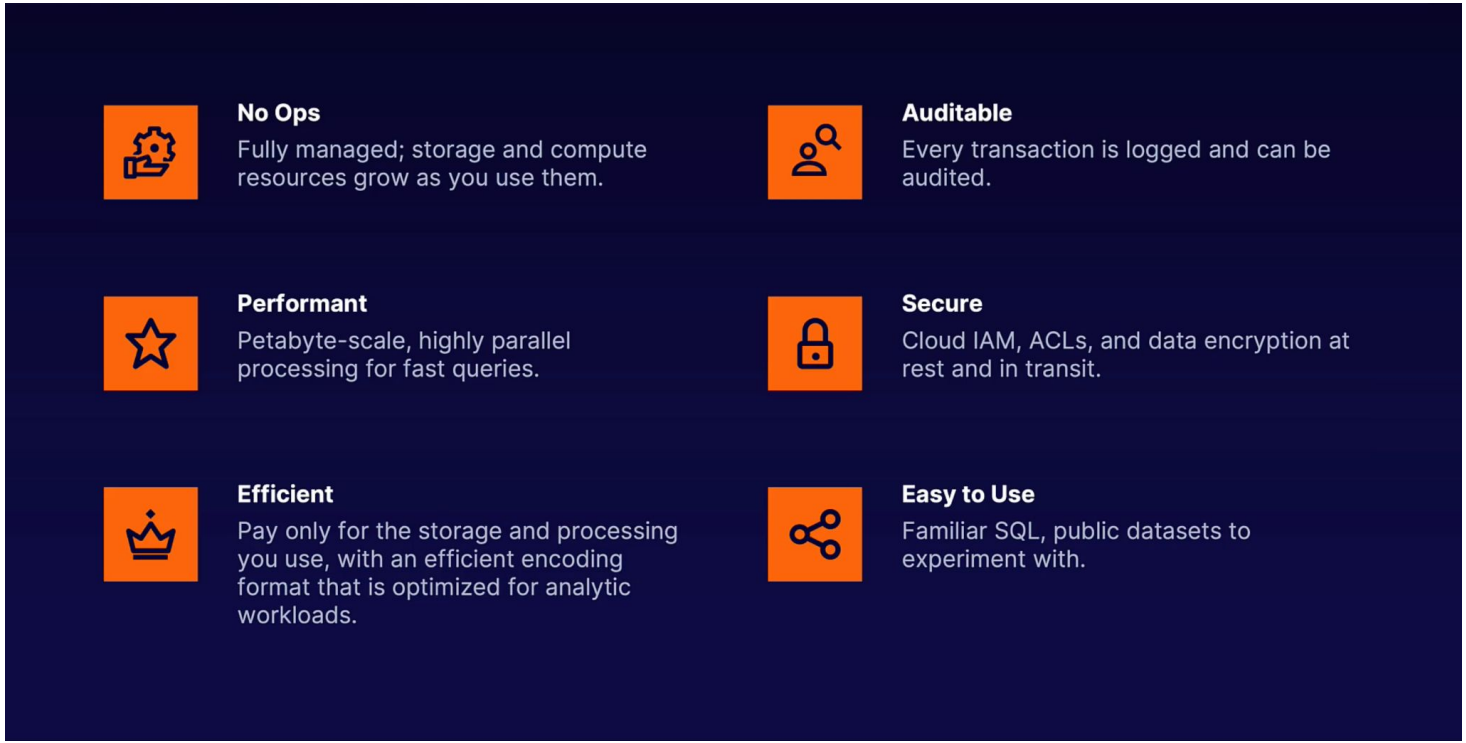
1. Optimize Data Management in BigQuery
2. Enhance Query Performance
3. Bridge the Data Lake and Data Warehouse Gap



Data Fellowship

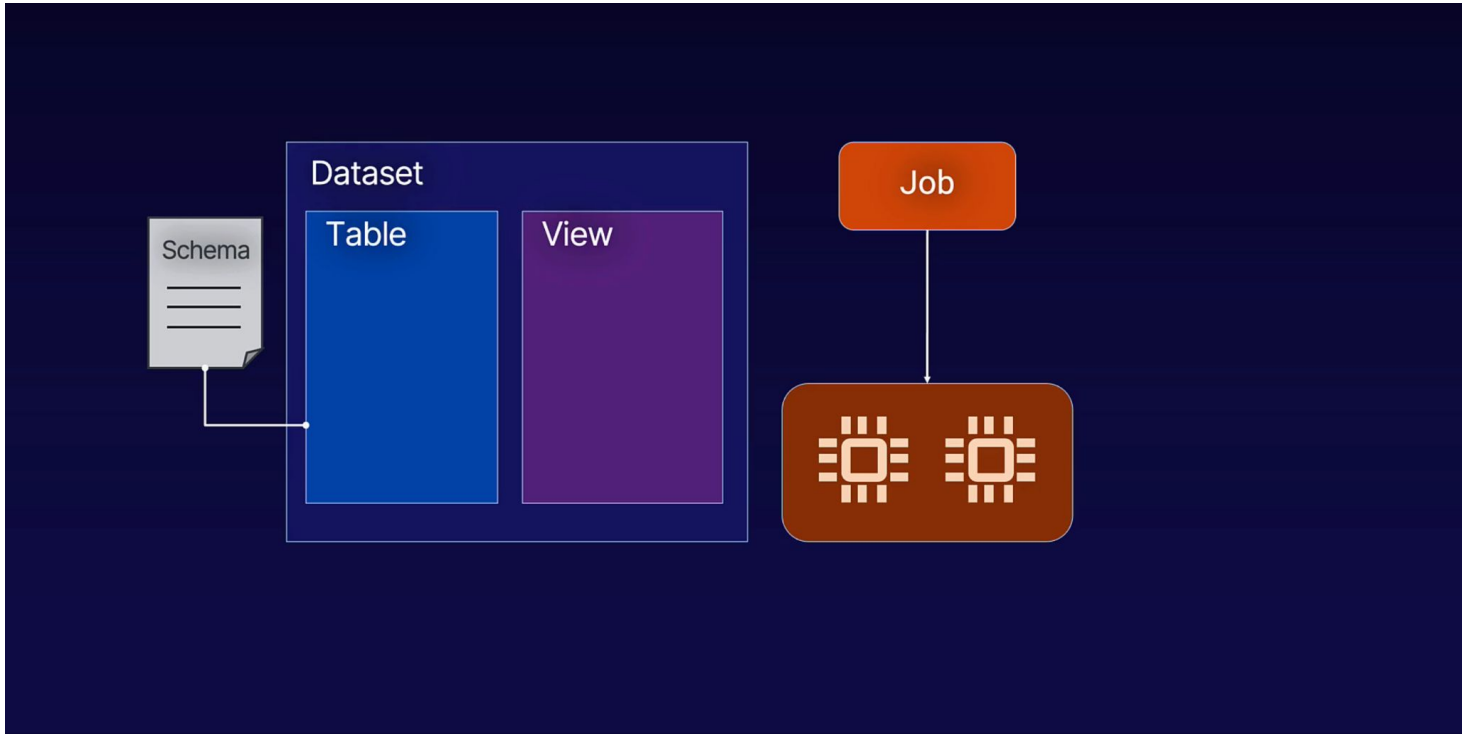
Deep Dive BigQuery

What makes BigQuery Modern?



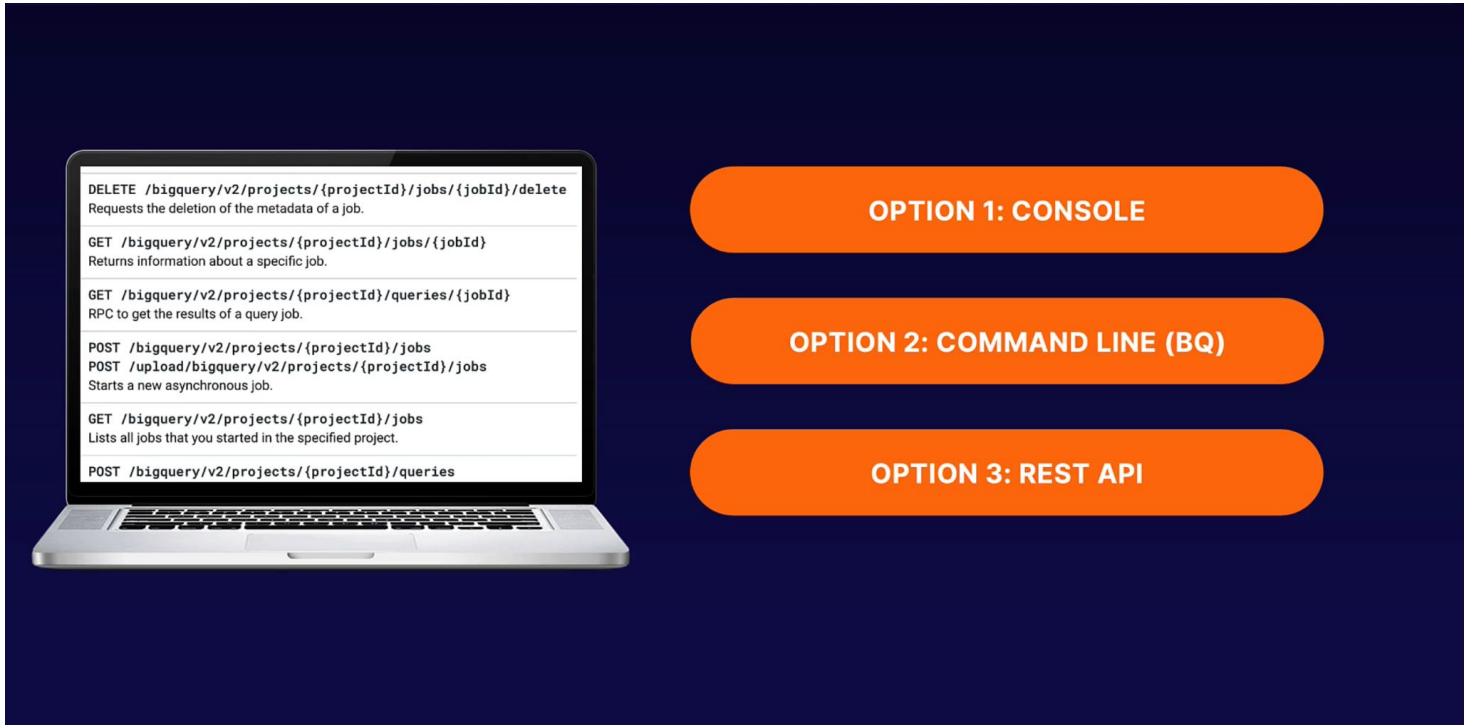
What makes BigQuery Modern?

BigQuery Basics Concepts



BigQuery Basics Concepts

BigQuery Basics Concepts



BigQuery Basics Concepts

BigQuery Use Cases

1. **General Analytics**
   Driving insights from data through online analytical processing (OLAP).

2. **Predictive Analytics**
   Anomaly/fraud detection, recommendation systems, sales forecasting, etc.

3. **Data Monetization**
   You can list your data for sale in the Google Cloud Marketplace.

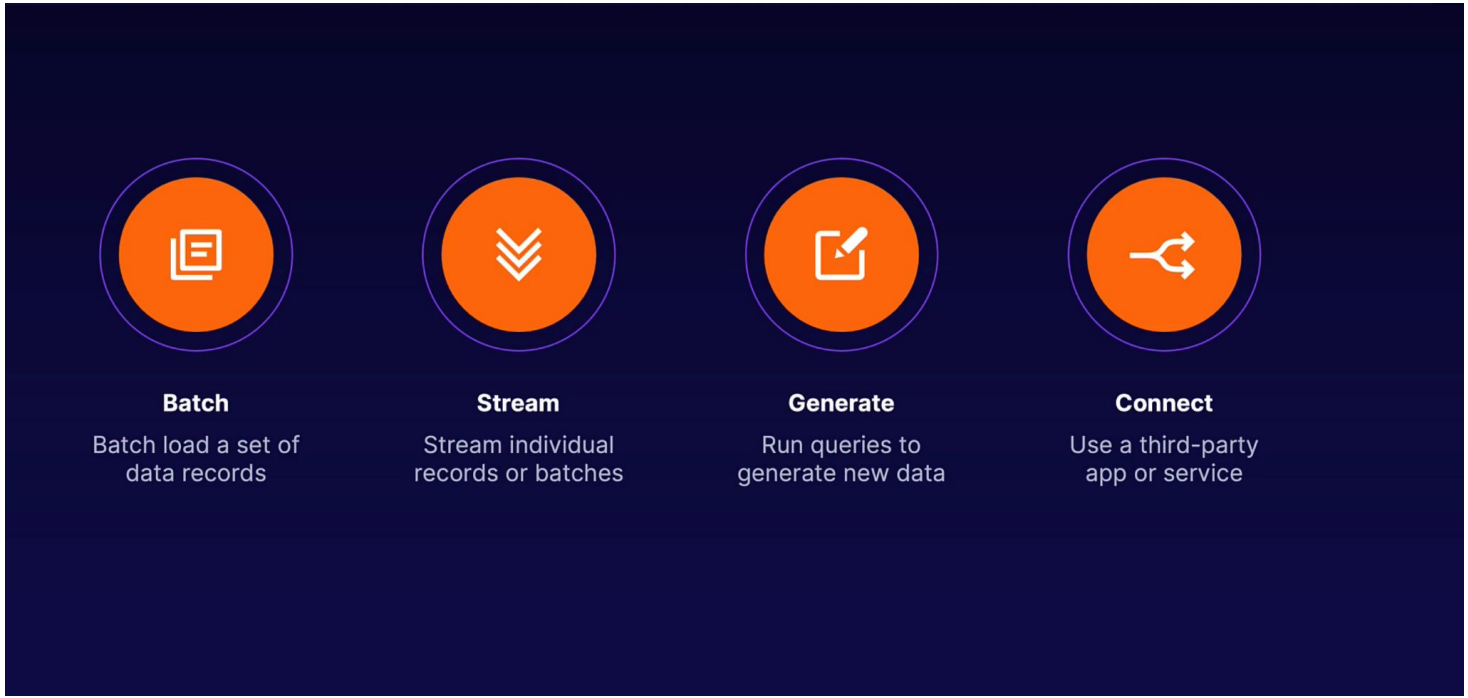
Takeaways

- BigQuery is a fully managed, modern, and easy to use data warehousing services
- BigQuery separation of storage and compute makes it resource and cost efficient
- There are 3 ways to interact with BigQuery service: console, command line, and REST APIs

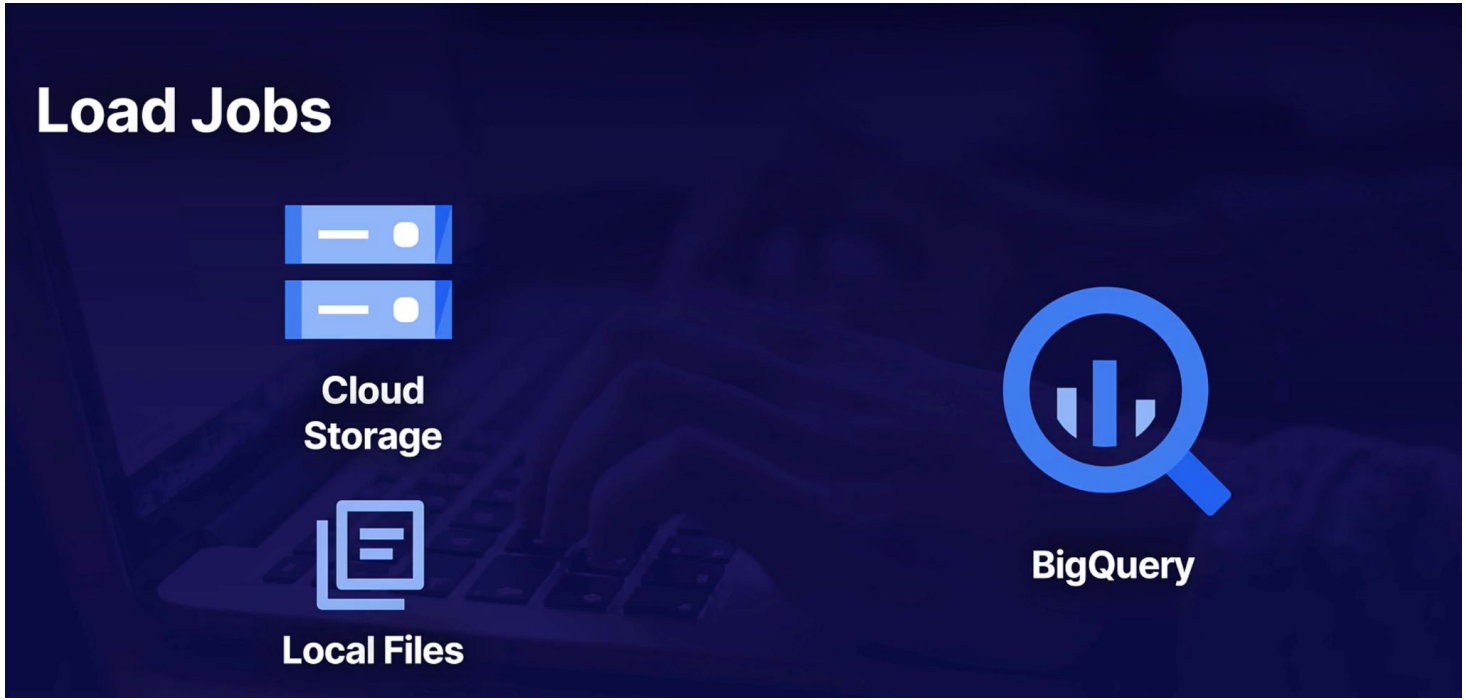
Data Fellowship

Loading Data into BigQuery

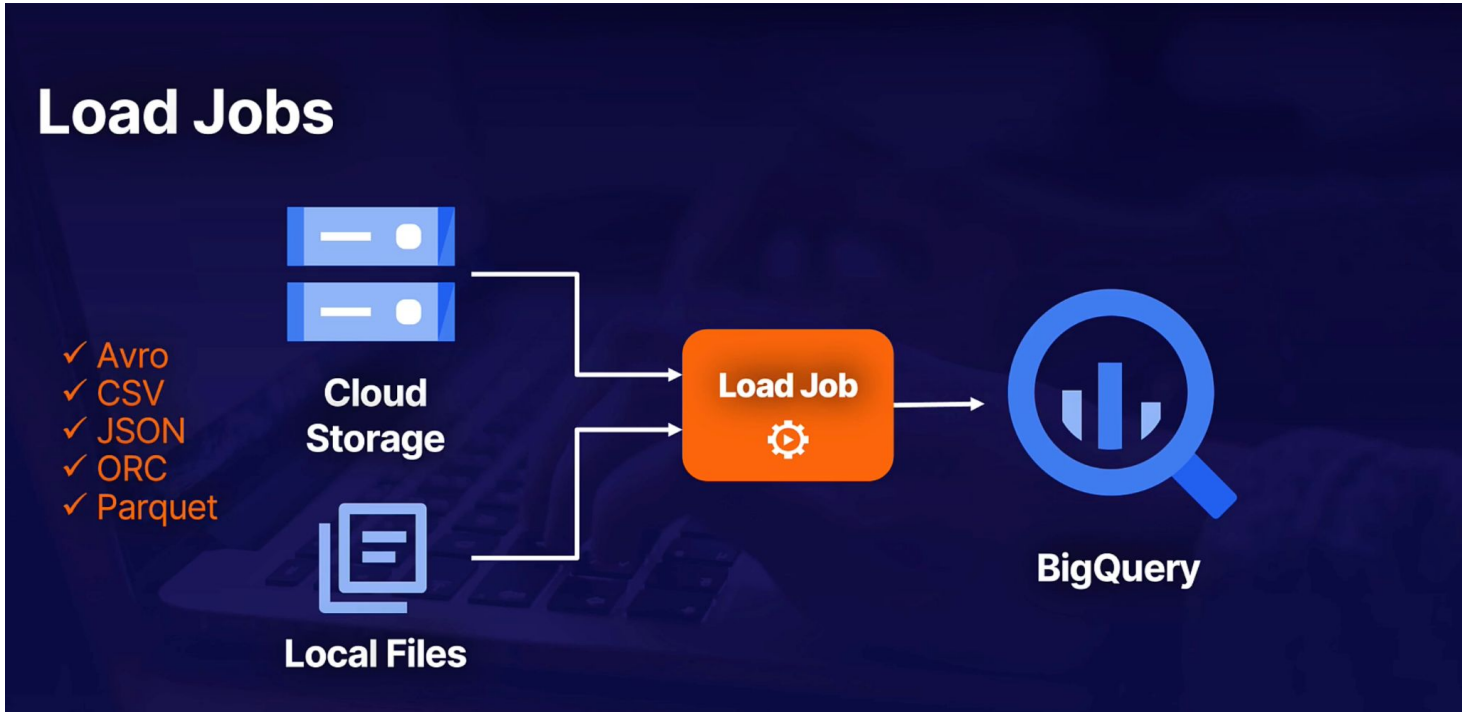
Data Ingestions and Options



Batch loading

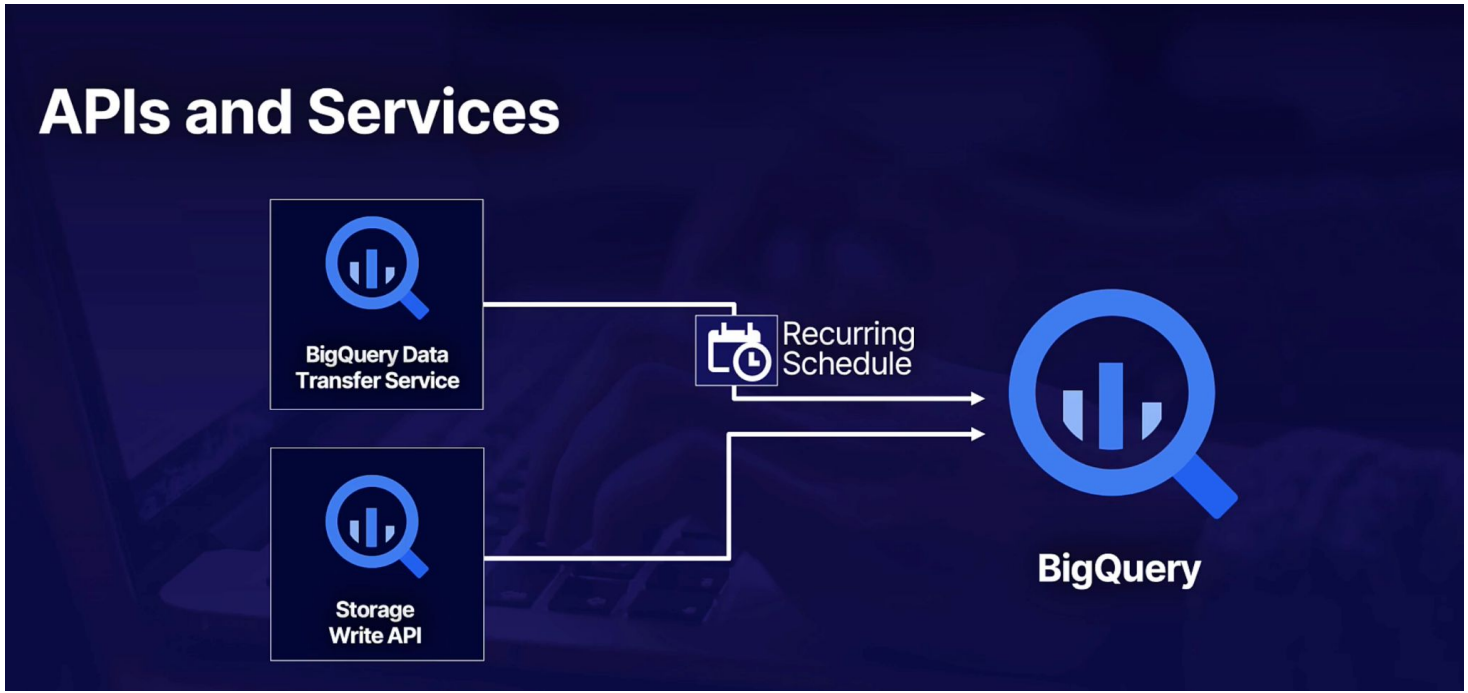


Batch loading



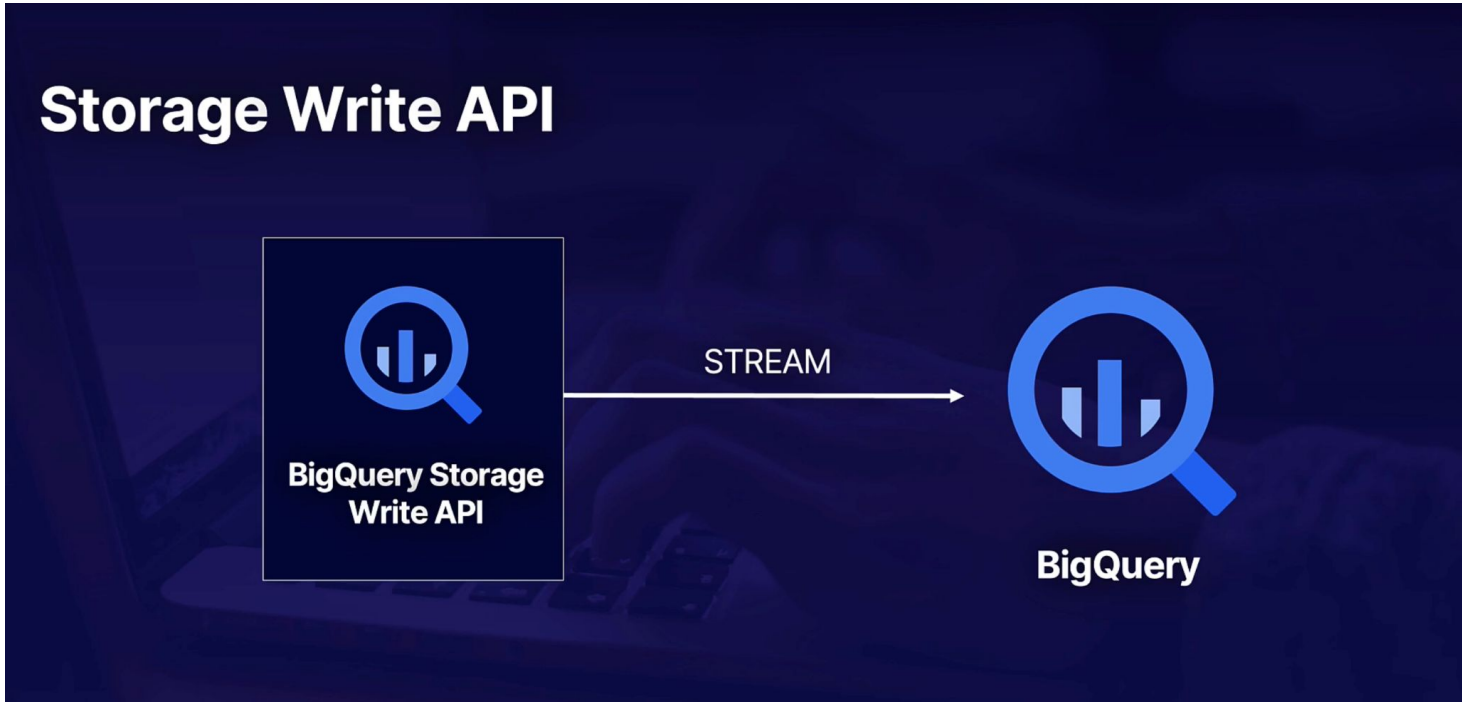
Batch loading

Batch loading

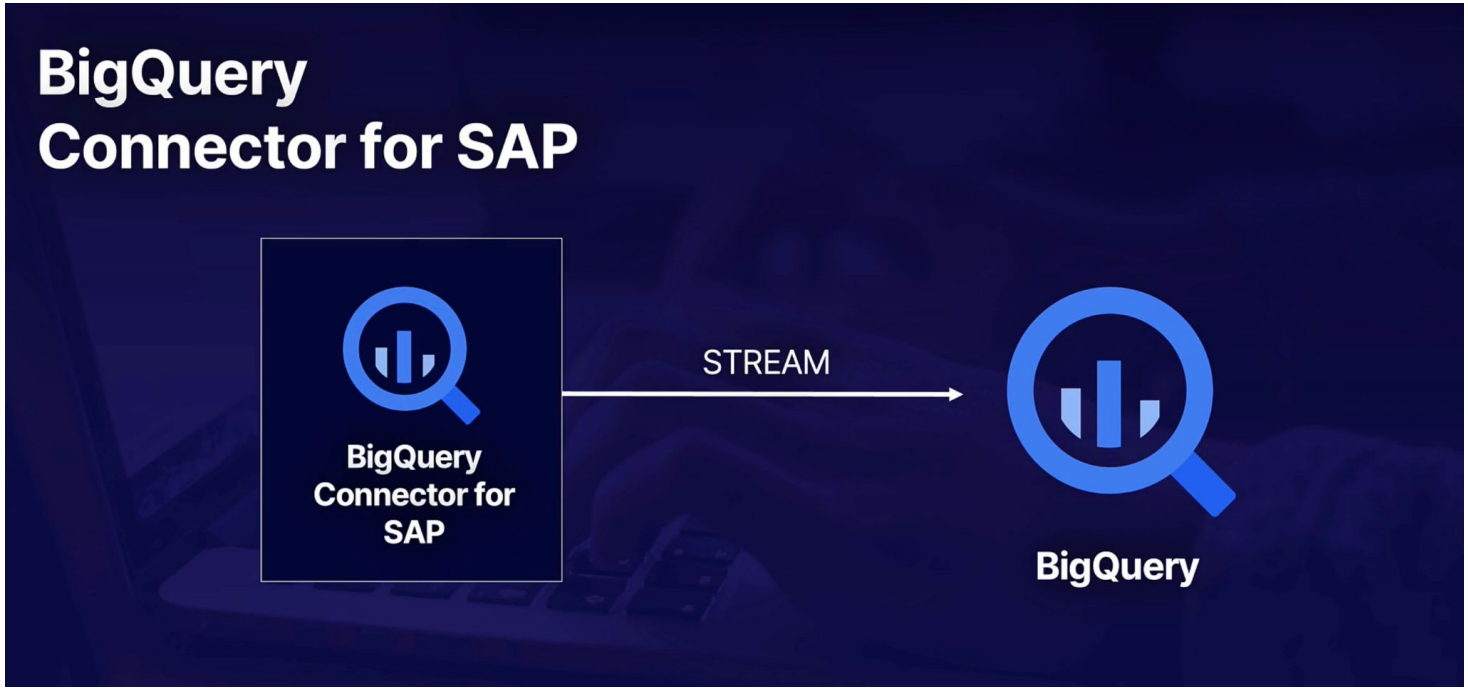


Batch loading

Streaming ingestion

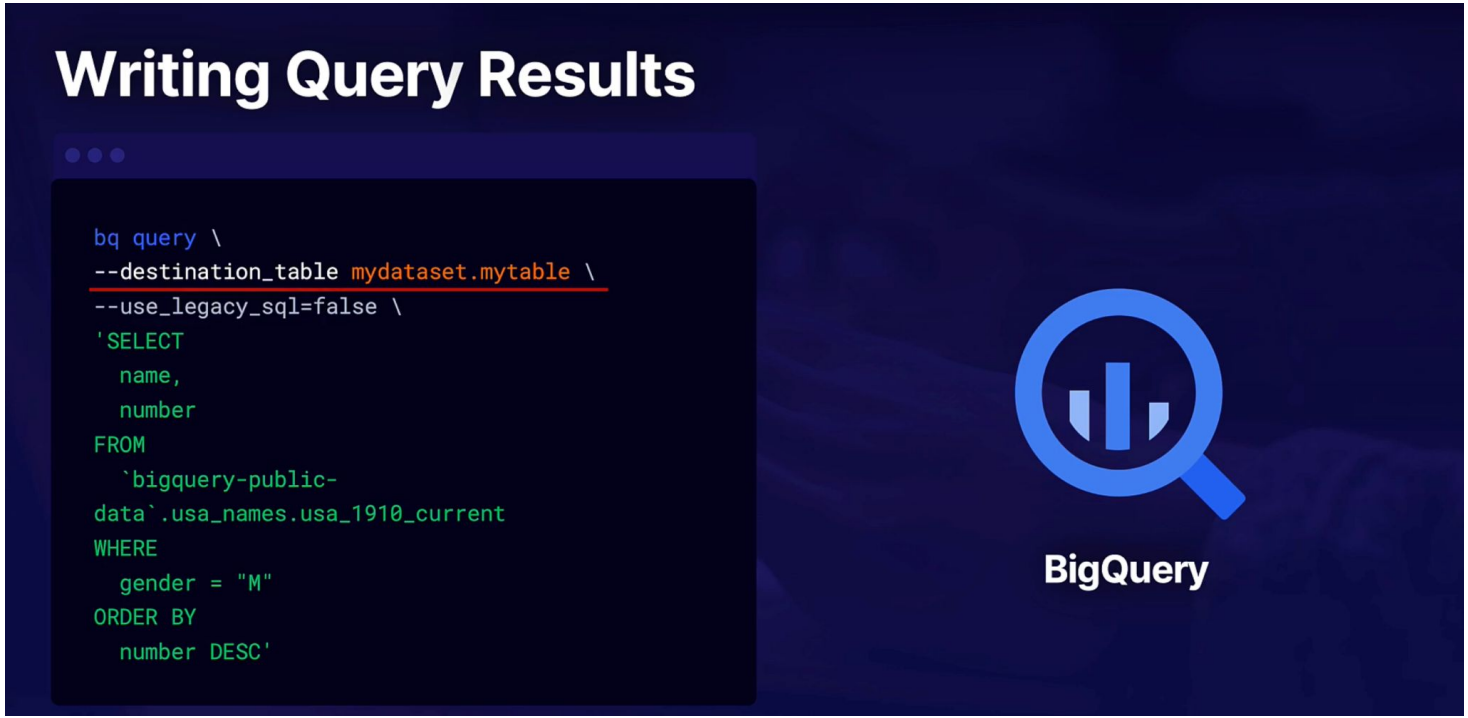


Streaming ingestion

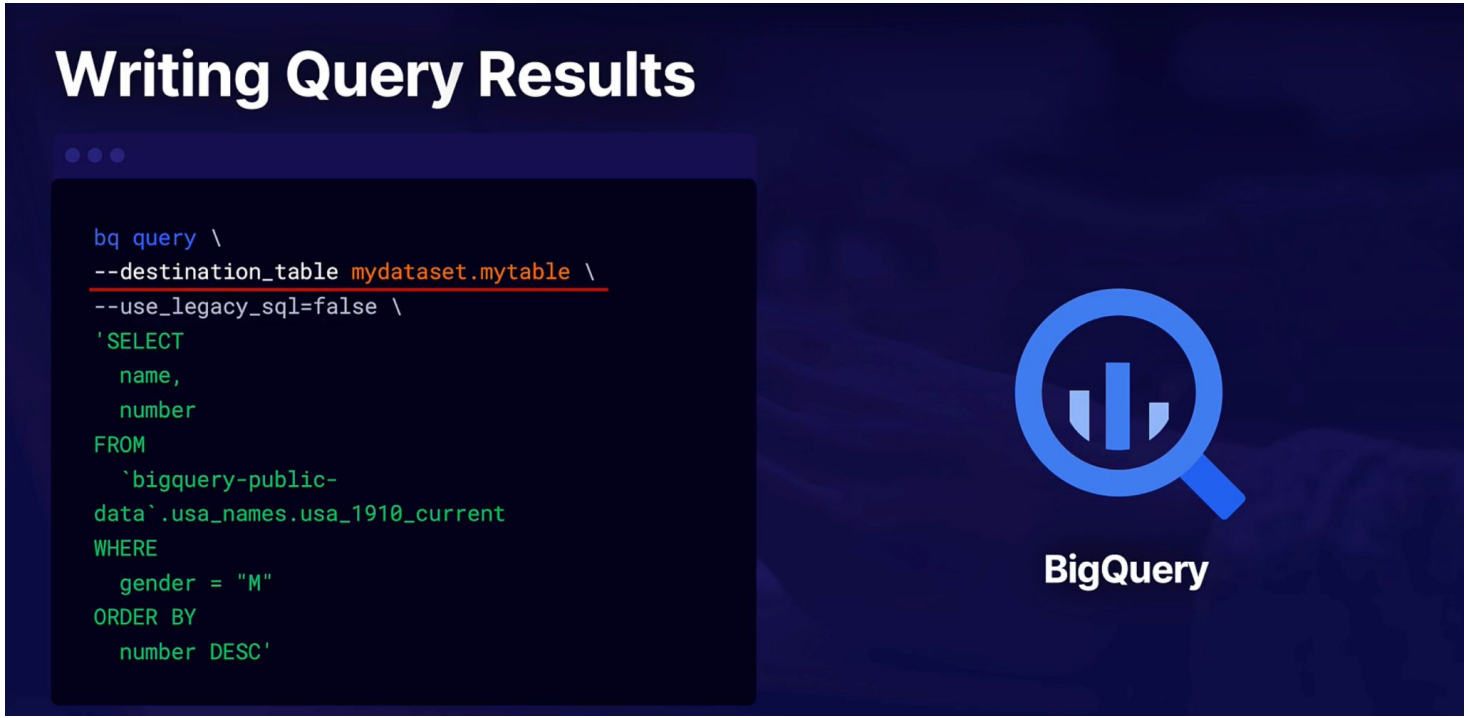


Streaming ingestion

Generating data with SQL



Generating data with SQL



Choosing how to load data

1. **Real-time vs. Offline Analysis**

   If you need real-time analysis, consider streaming. For offline analysis, use batch loading.

2. **Data Source**

   For local files or Cloud Storage, consider batch load jobs. If the data source is Cloud SQL, consider federated queries.

3. **Latency Requirements**

   Streaming load options will have the lowest latency.

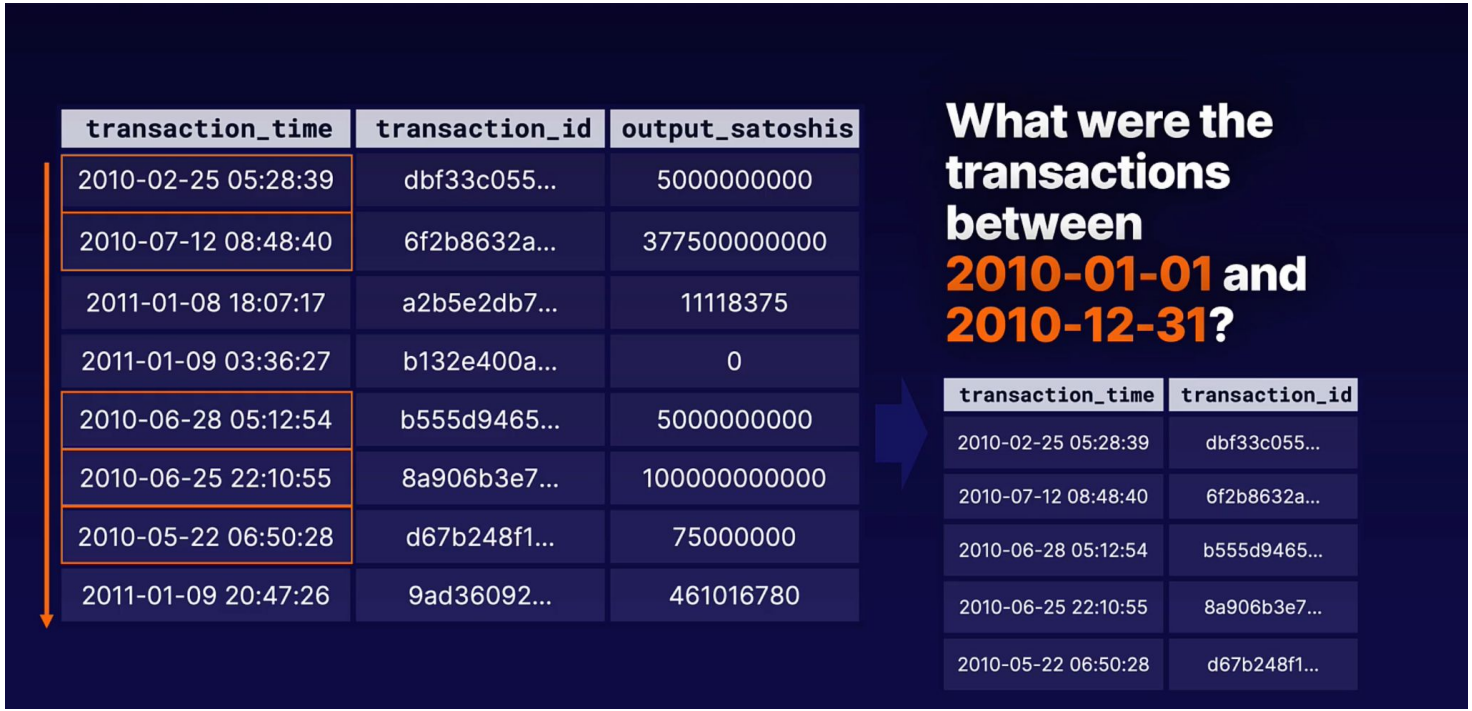
Takeaways

- You have broad option to load data into BigQuery: batch, stream, generate, connect
- Specific methods to ingest data include: batch load jobs, SQL, APIs, Servicers and Connectors
- The best option for loading data is depends on data sources as well as analytics and latency requirements

Data Fellowship

Partitioning and Clustering in BigQuery

Partitioned Tables

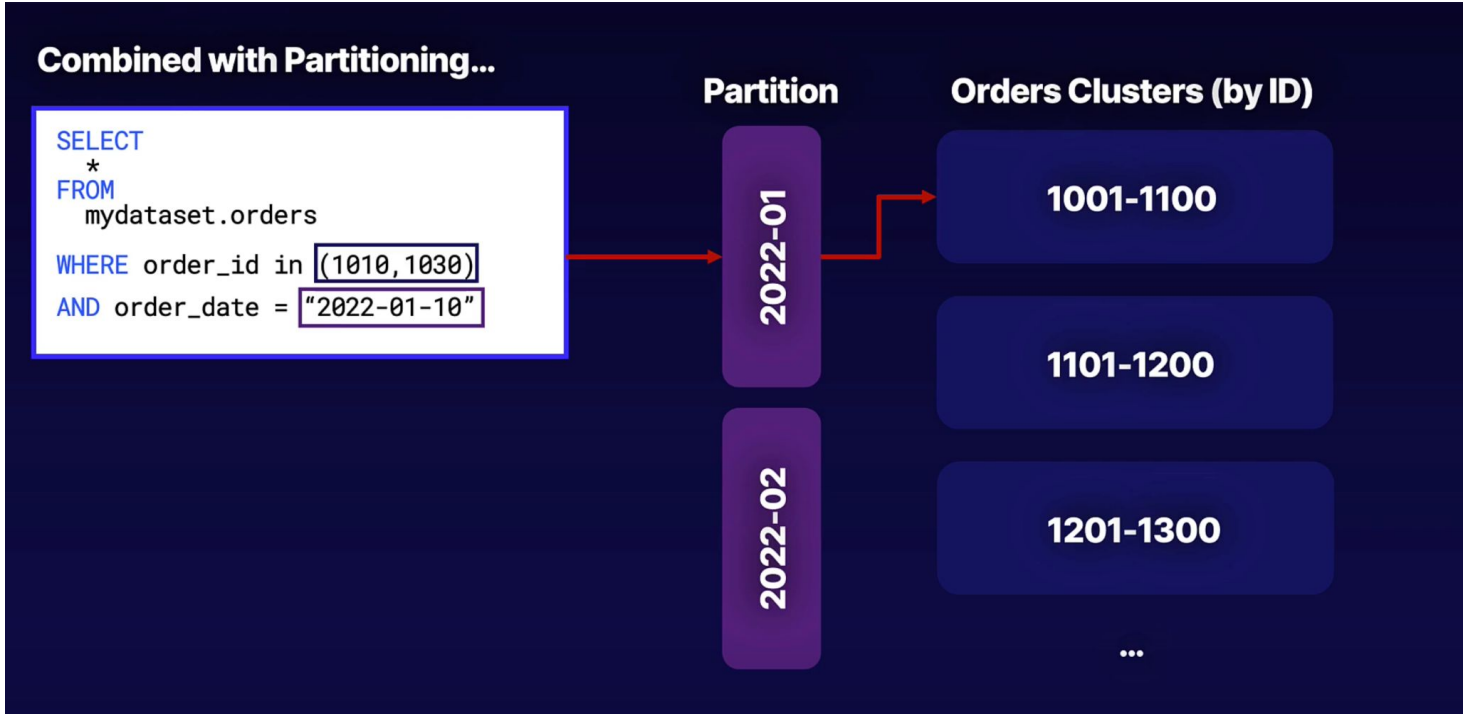


You can partitions table by

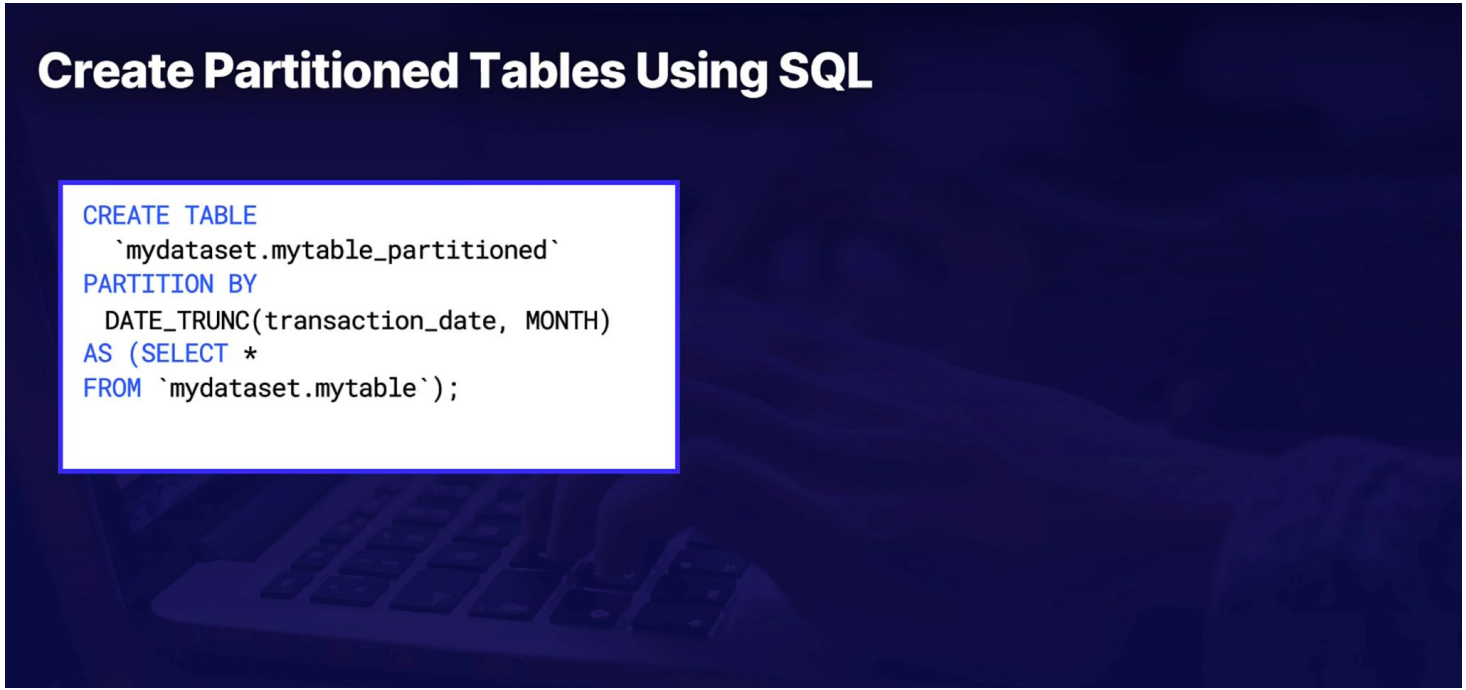
Clustering Overview

Clustering Overview

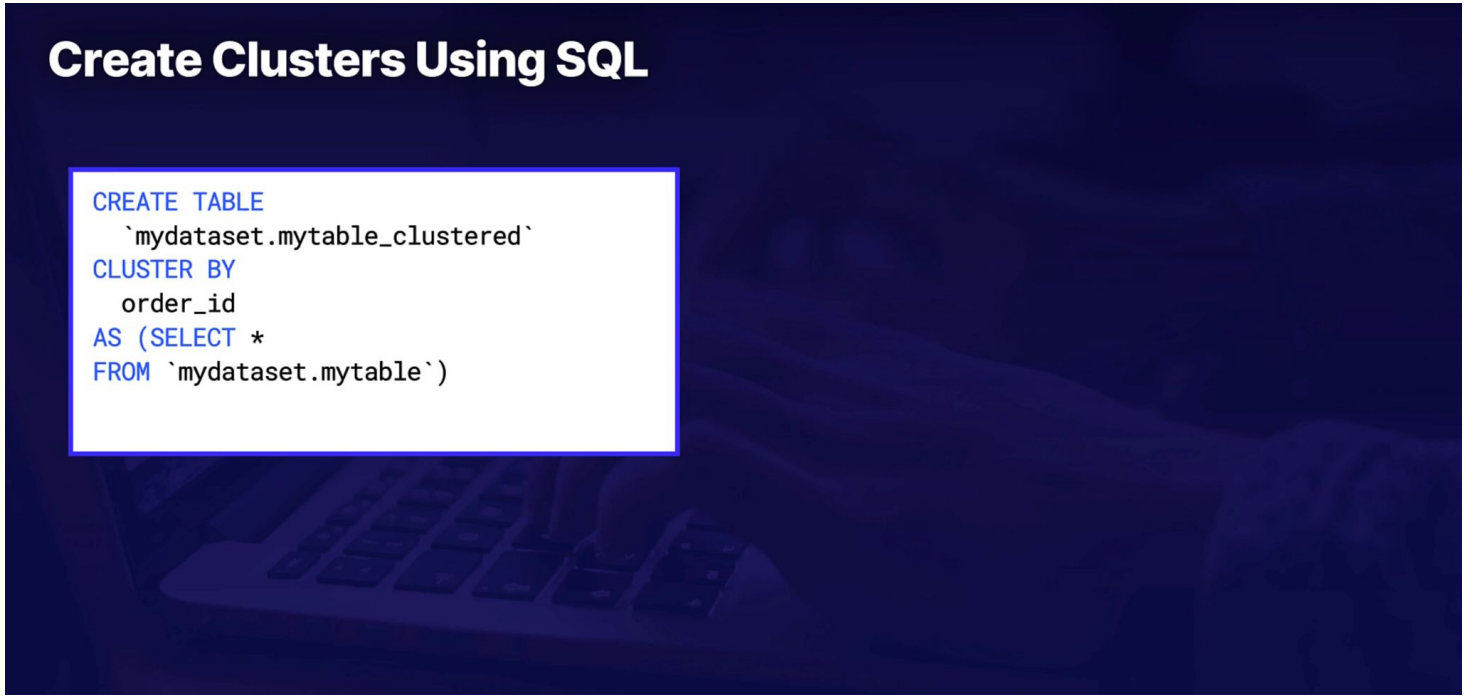
Clustering Overview



Using Partitioned and Tables Using SQL



Using Partitioned and Tables Using SQL



When to use Partitioning?

When to Use Partitioning

1. **Time- or Numeric-Based Segmentation**
   You need to segment your data based on time or numeric ranges.

2. **Low-Cardinality Column**
   There aren't too many distinct values.

3. **Partitions Wouldn't Be Too Small**
   Streaming load options will have the lowest latency.

When to Use Clustering

1. **Colocate Related Data**
   The column(s) contents can help organize the data for common filtering actions.

2. **High-Cardinality Column**
   There are many distinct values.

3. **Filters and Aggregations Over Multiple Columns**
   Your queries often use filters or aggregations against certain columns.

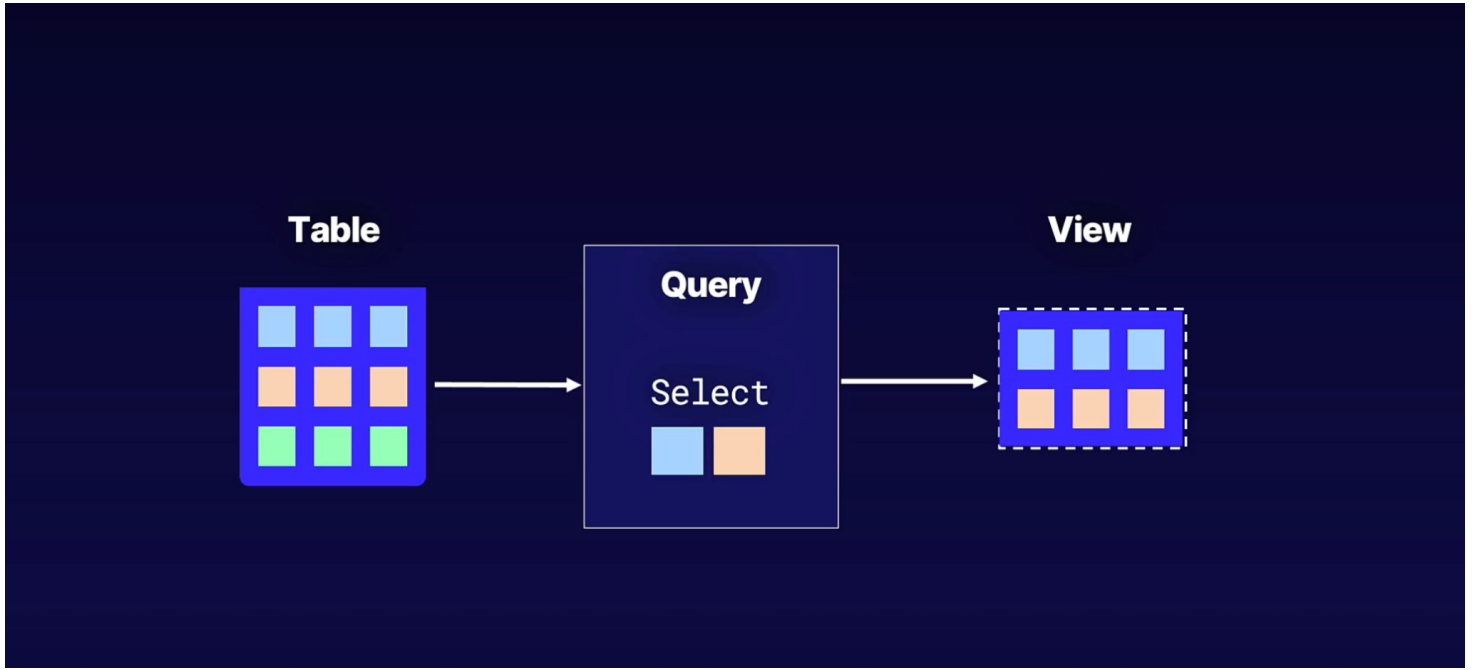
Takeaways

- Partitions table are table divided into segments that simplify data management and improve query performance`
- You can partitions using time and numeric related columns with different level of granularity
- Clustering is used to colocate related data that can improve performance of filter and aggregation queries

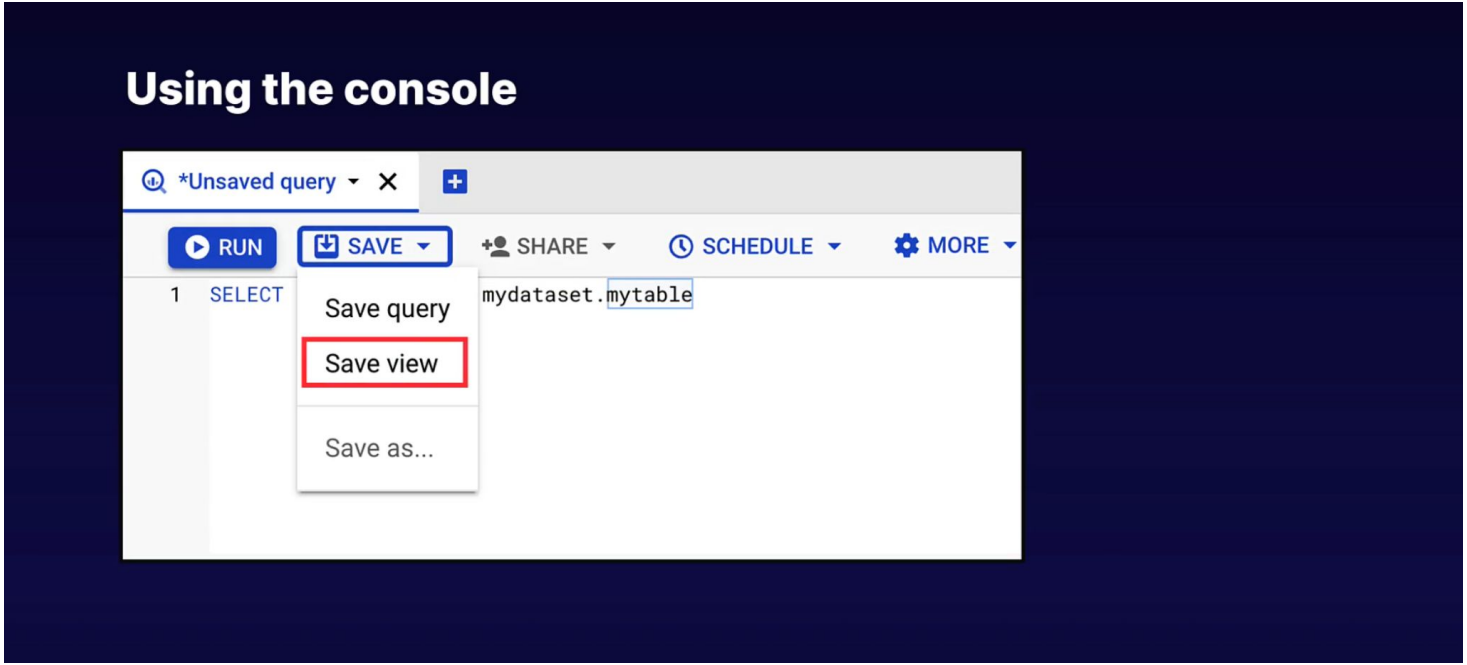
Data Fellowship

Working with Views and Materialized Views

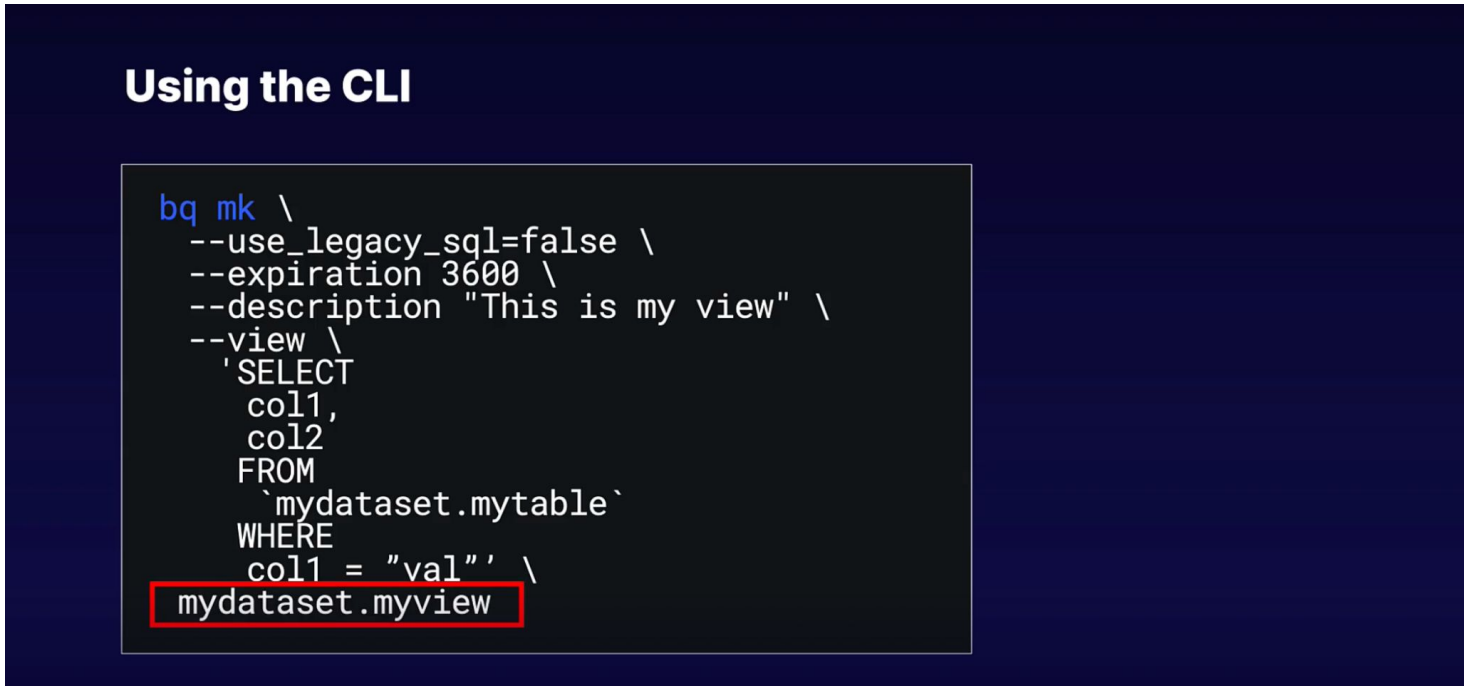
What is views?



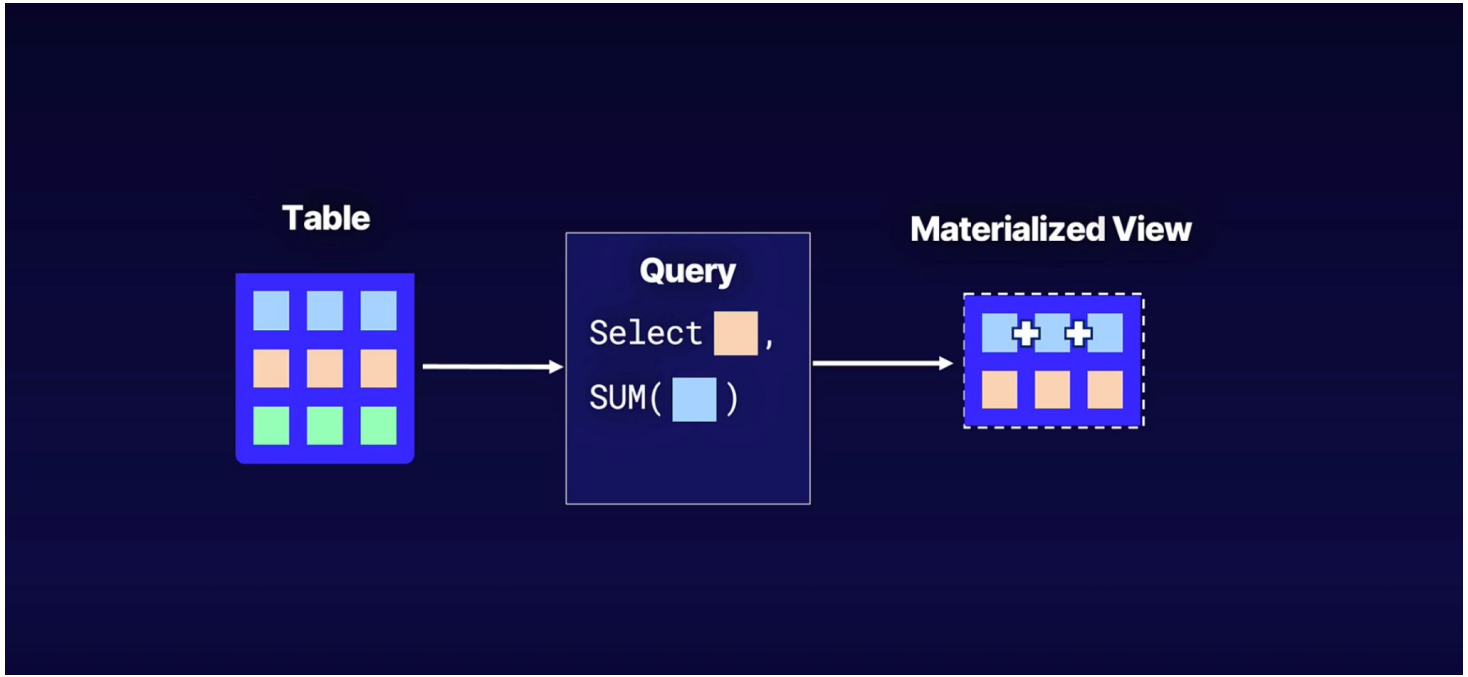
What is views?



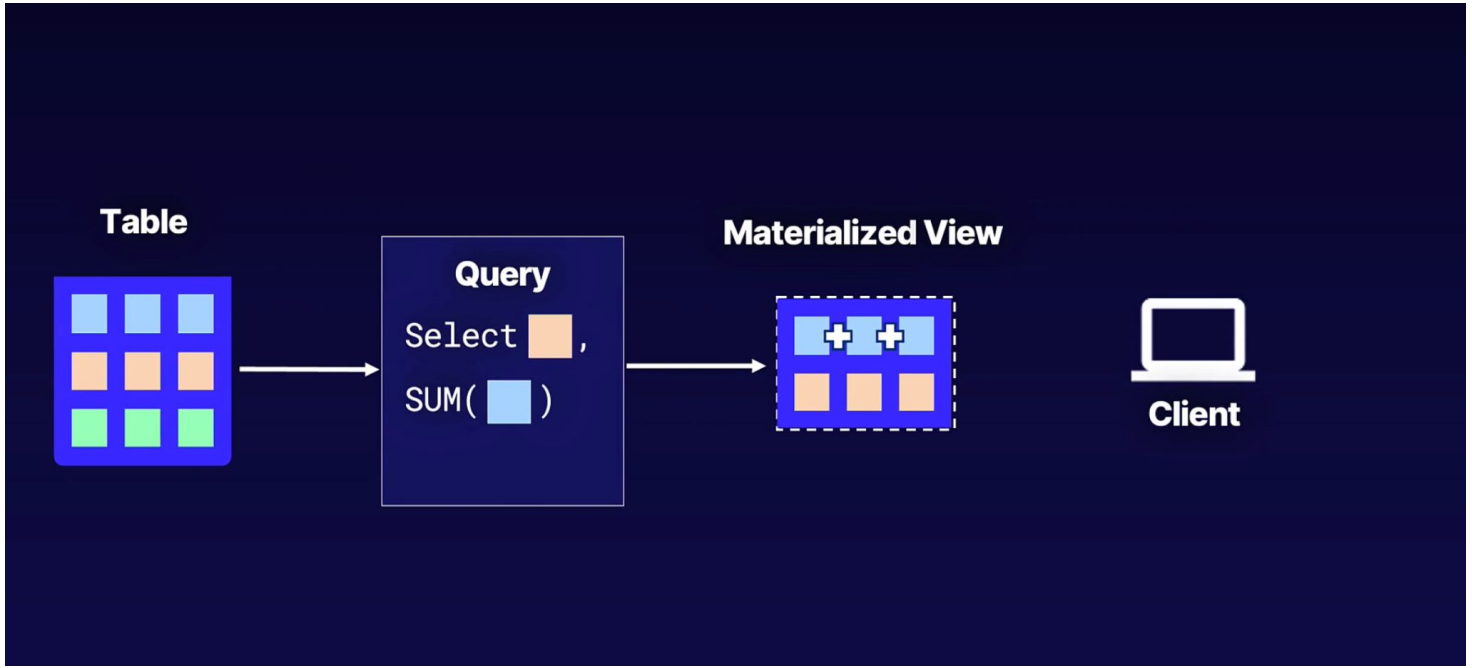
What is views?



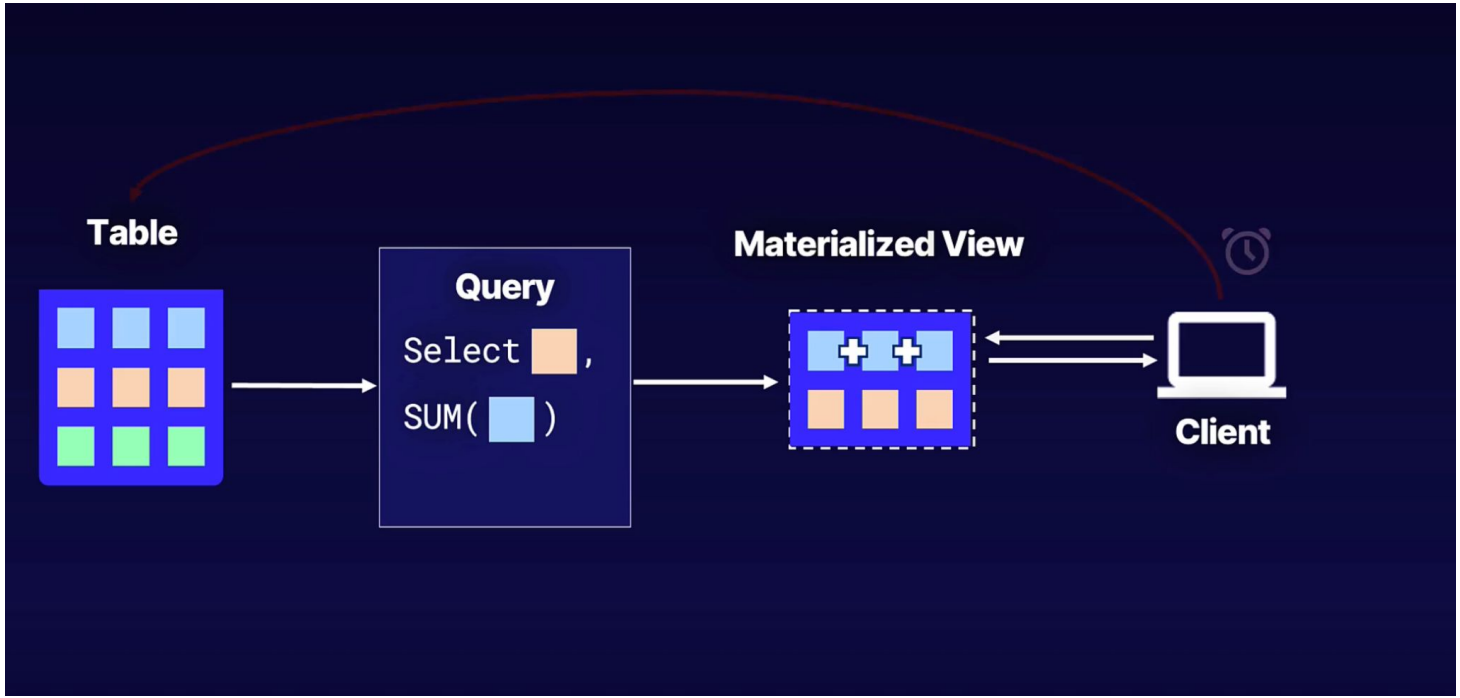
Materialized Views



Materialized Views



Materialized Views

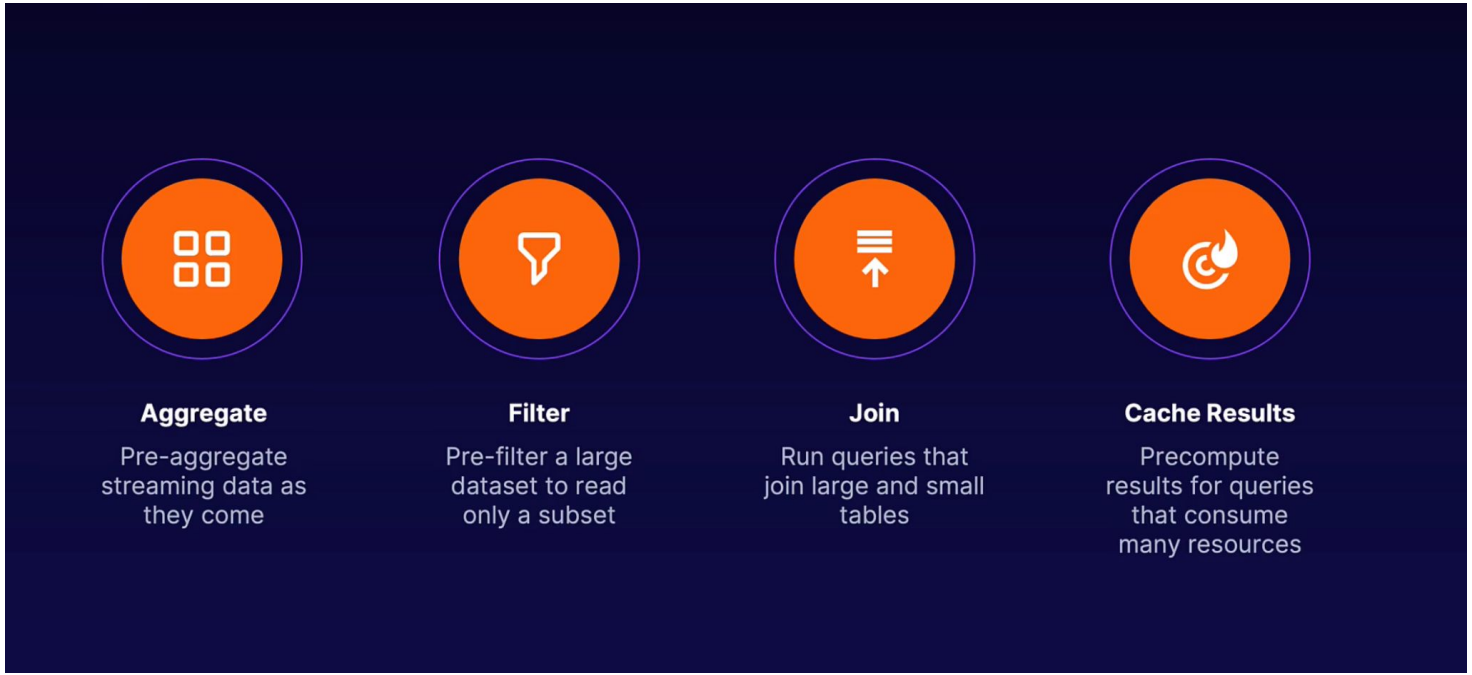


Materialized Views

Using SQL

```
CREATE MATERIALIZED VIEW
myproject.mydataset.my_mat_view AS (
  SELECT
    product_id,
    SUM(clicks) AS sum_clicks
  FROM
    myproject.mydataset.my_table
  GROUP BY
    product_id
);
```

Materialized Views



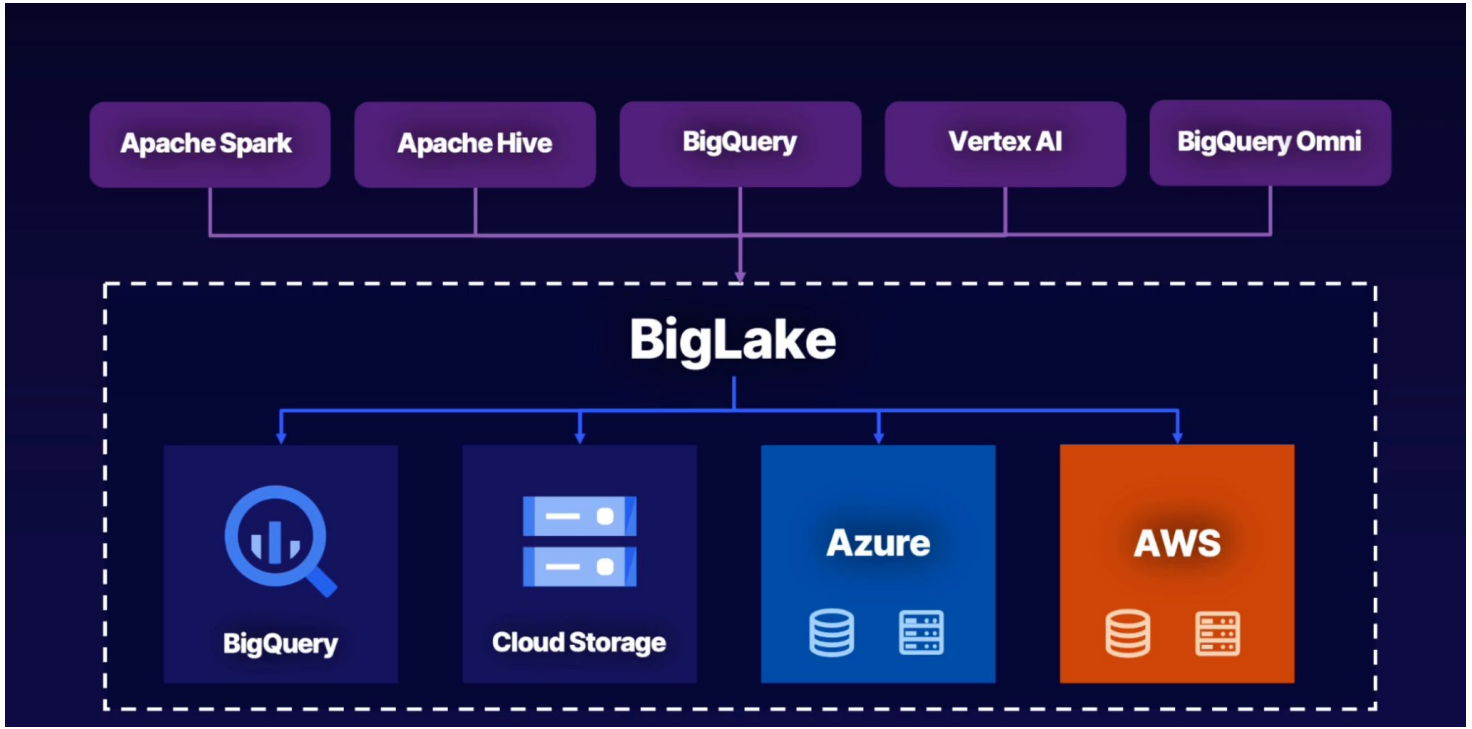
Takeaways

- Views are virtual tables defined by SQL query and can be used in the same way table are used.
- A materialized view is precomputed view that caches result of a query
- materialized views are ideal for when the query consumes a lot of resources, while the base table does not change often

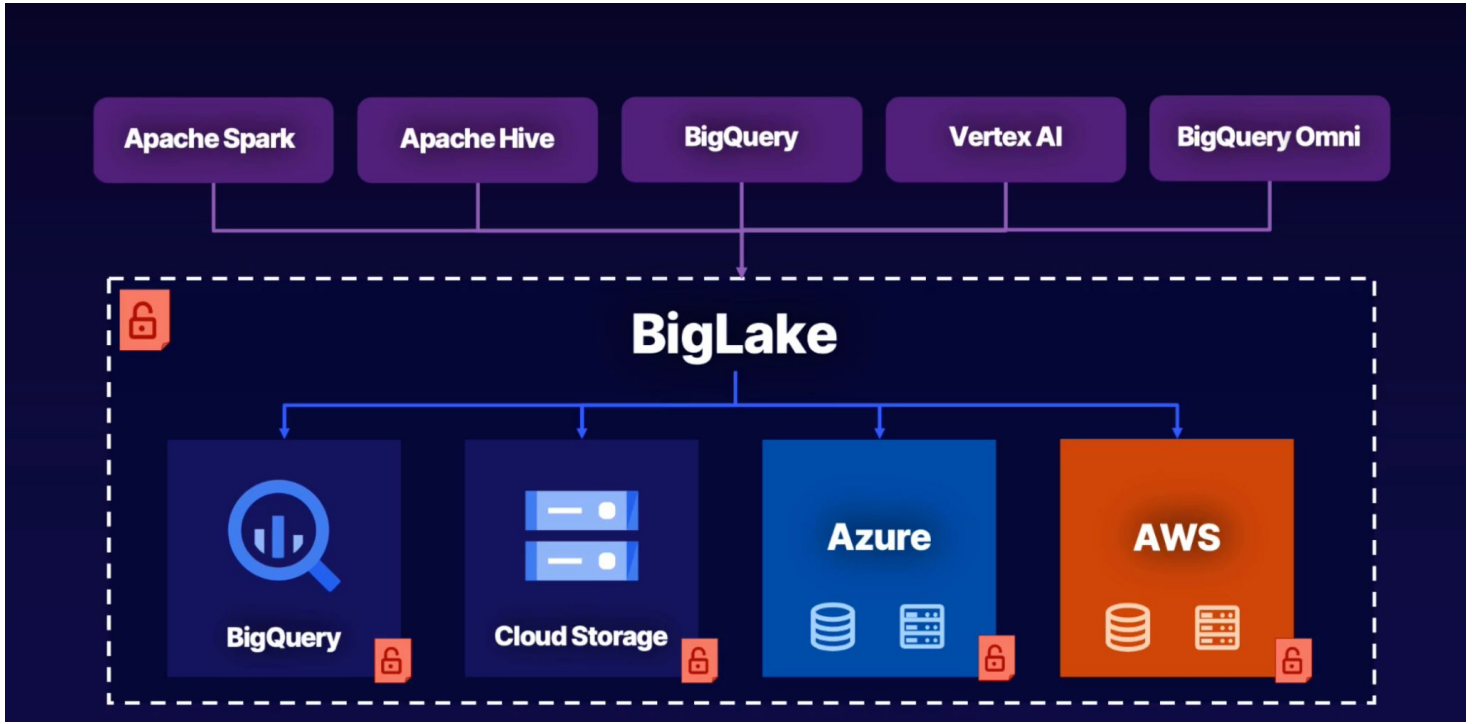
Data Fellowship

Unifying data lake and data warehouses

Introductions to Big Lake

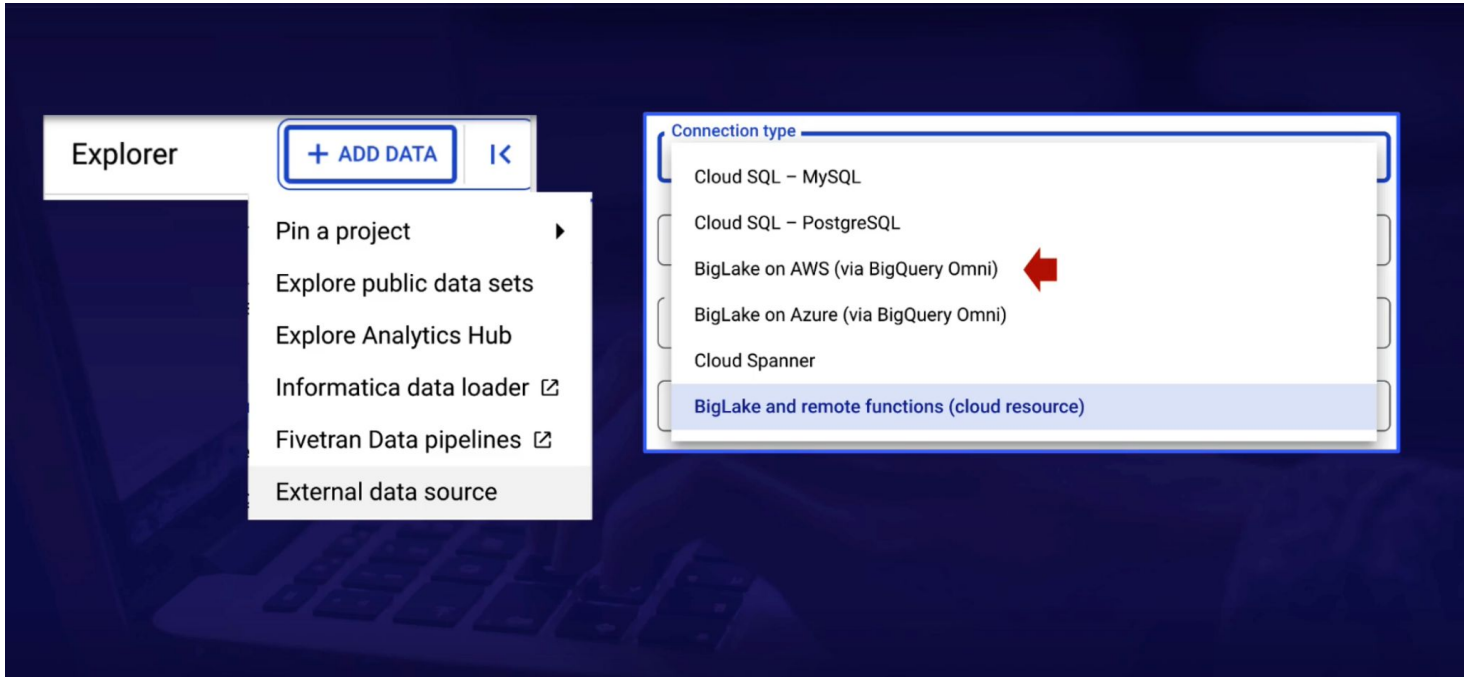


Introductions to Big Lake

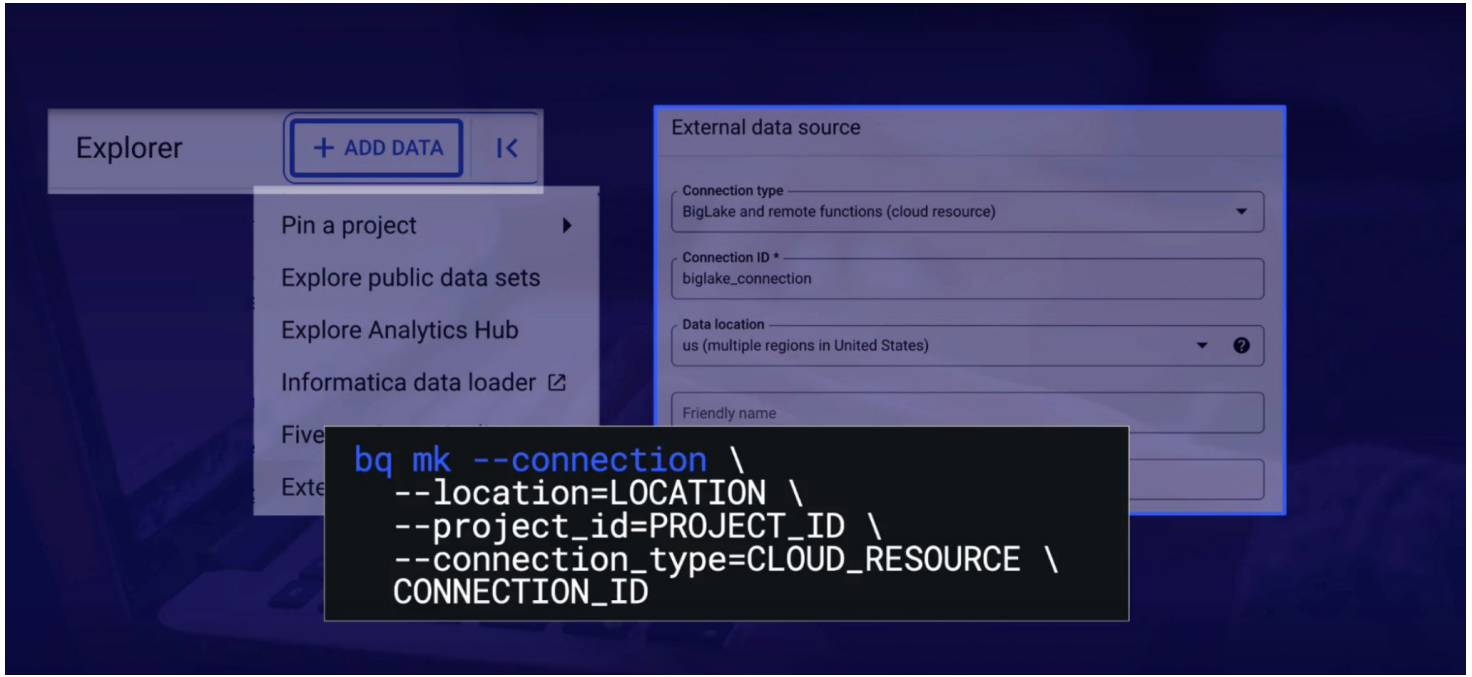


Big Lake Features

Big Lake Features

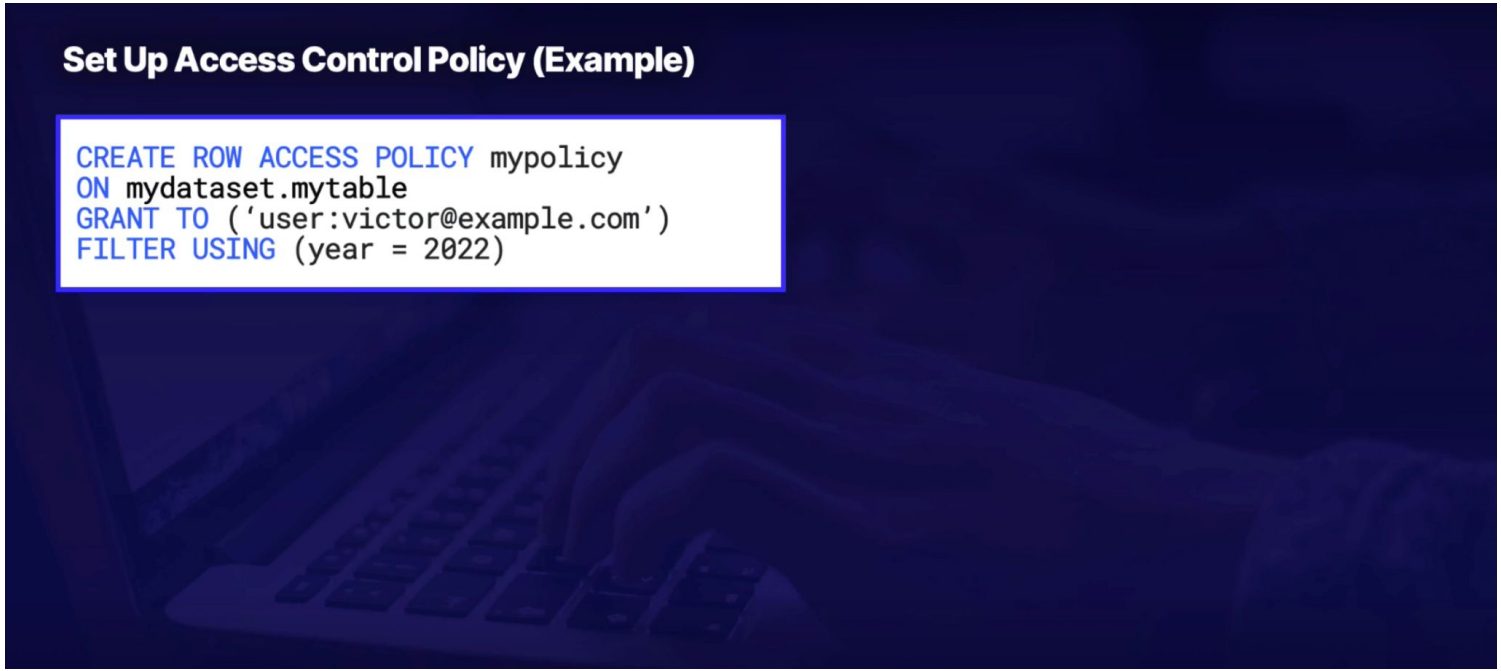


Big Lake Features



Big Lake Features

Big Lake Features



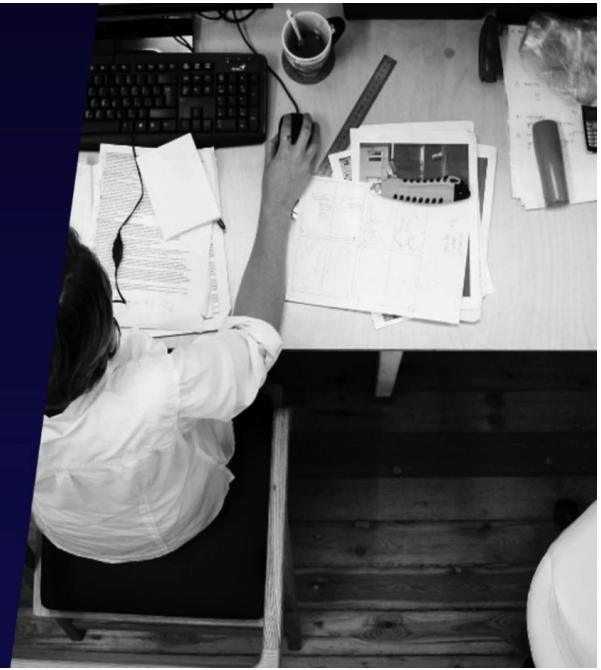
Big Lake Features

BigLake Use Cases

1. **Unified Data Lake and Warehouse**
   Interoperability between data lakes and warehouses, with a single copy of data and uniform management.

2. **Centralized Data Governance**
   Enforce fine-grained access controls and governance over multicloud data lakes.

3. **Extend the Power of BigQuery into Data Lakes**
   Run performant analytics and build ML models without moving data.



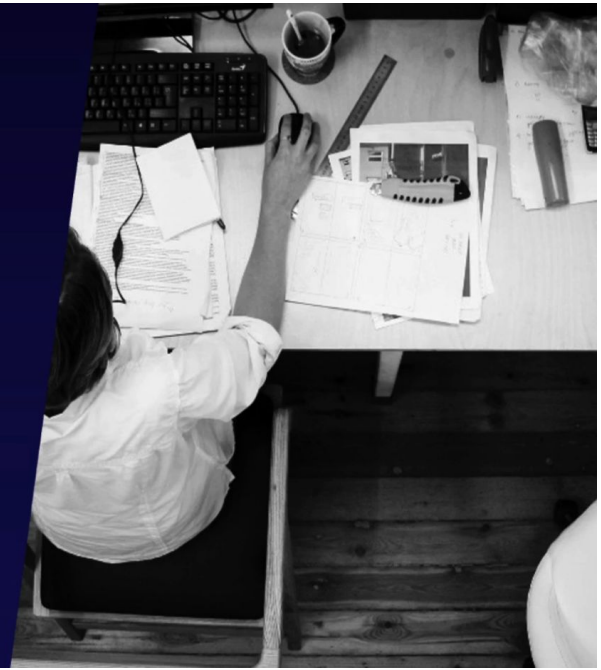
Big Lake Features

BigLake Use Cases

1. **Unified Data Lake and Warehouse**
   Interoperability between data lakes and warehouses, with a single copy of data and uniform management.

2. **Centralized Data Governance**
   Enforce fine-grained access controls and governance over multicloud data lakes.

3. **Extend the Power of BigQuery into Data Lakes**
   Run performant analytics and build ML models without moving data.



Takeaways

- BigLake is storage engine that unified the data lake and data warehouses
- BigLake is builts on open data formats and offers unified, multi cloud data governance and fine grained security.
- A BigLake tables can be use like another BigQuery Table, while the data may be in the Cluud Storage, AWS, or Azure

References

A cloud guru : BigQuery and the modern data warehousing

iykra Data Fellowship

Thank you!

nothin’ is impossible until its done