

# Agenda - 2 Days

## Day 1

### General

- 0900 - Welcome & Introduction
- 0910 - Databricks Overview
- 0930 - Lakehouse / Demo



### Data Engineering

- 0950 - Data Engineering Overview
- 1020 - Lab Setup
- 1030 - Morning Tea
- 1045 - Data Engineering Lab
- 1300 - Lunch



### Analytics/BI

- 1400 - Analytics/BI Overview
- 1420 - Databricks SQL Lab
- 1445 - Afternoon Tea
- 1500 - Databricks SQL Lab Cont'd
- 1530 - Databricks Roadmap

You Are Here

## Day 2

### Data Science

- 0950 - Databricks ML Overview
- 1020 - Lab Setup & Break
- 1030 - Databricks ML Lab

### Wrap Up

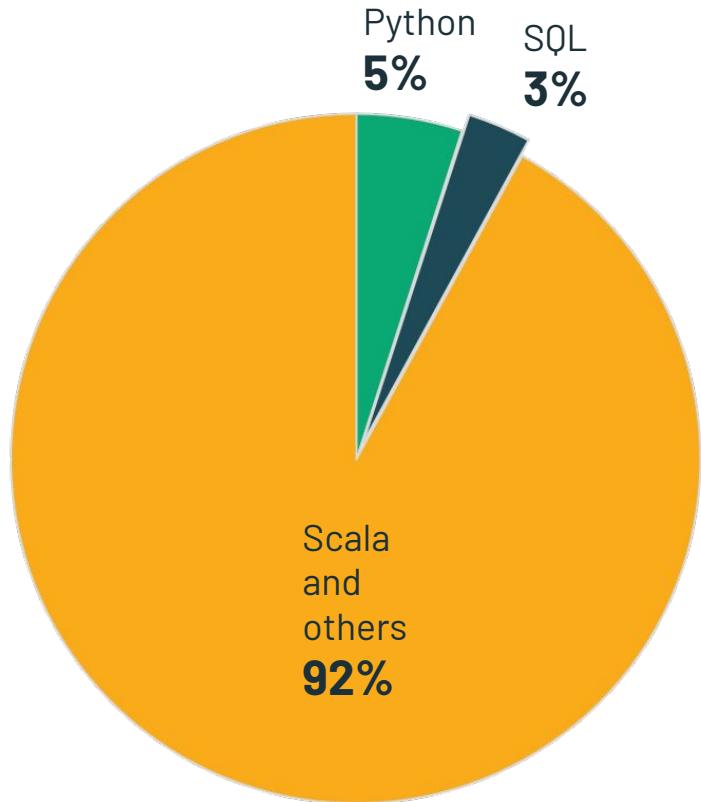
- 1200 - Closing Comments
- 1210 - Q&A

# Databricks SQL Overview

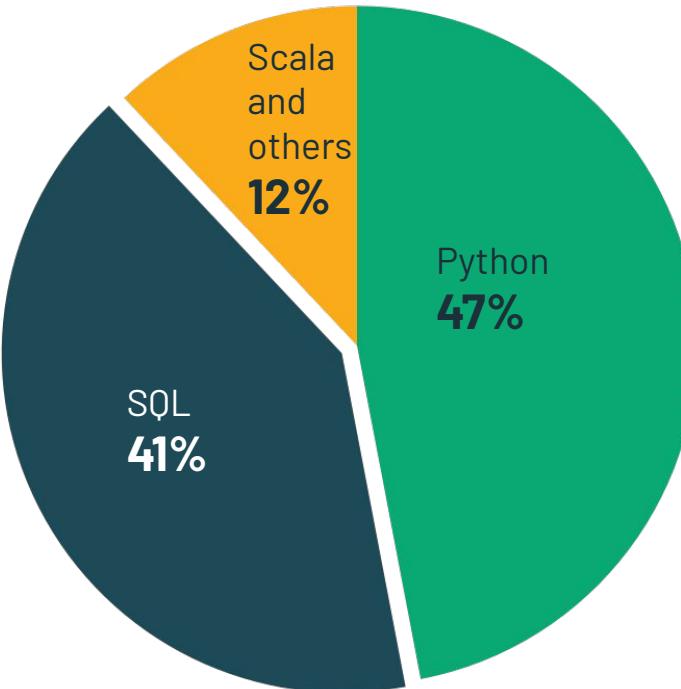


# Spark usage among Databricks customers

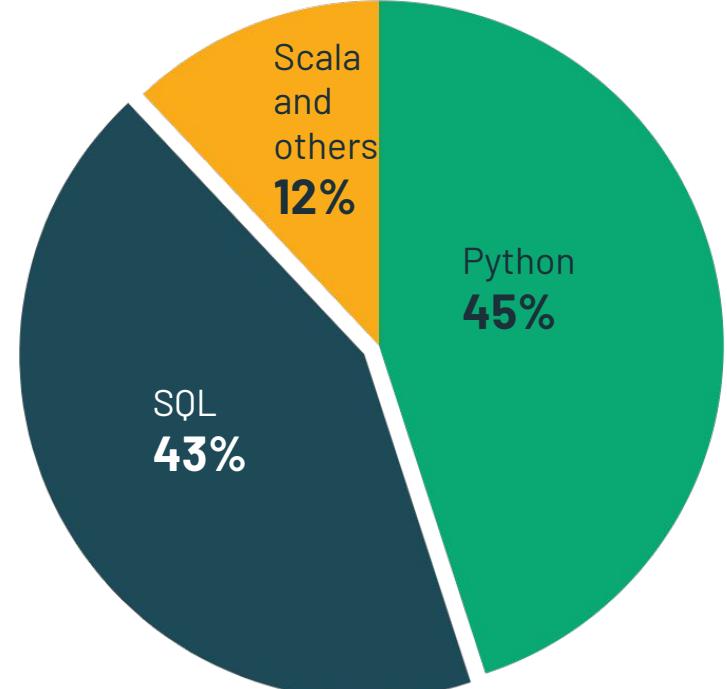
2013



2020



2021

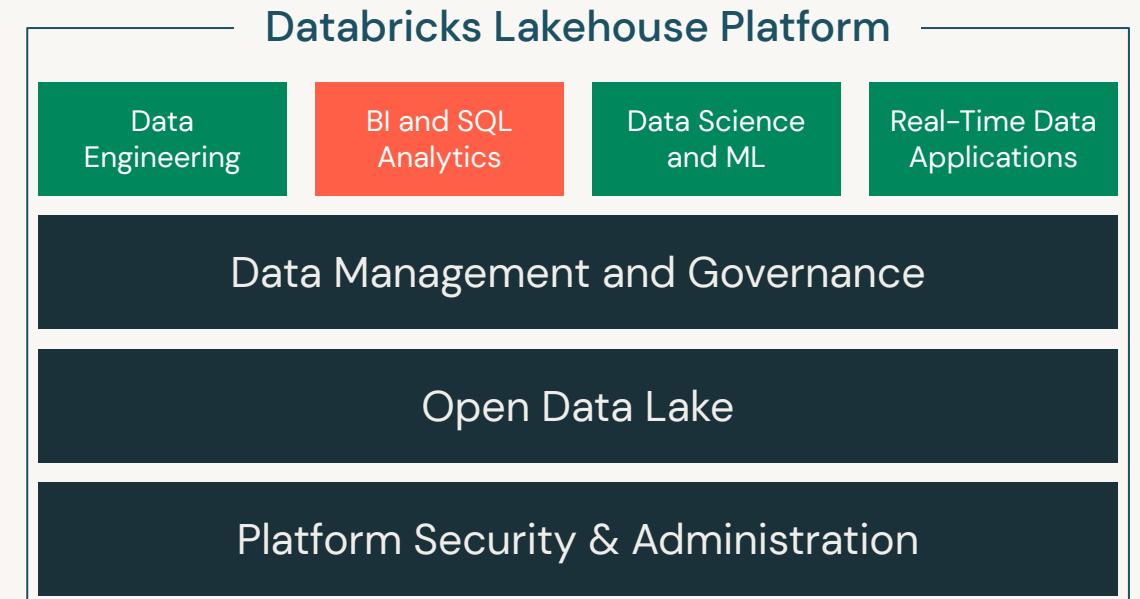


# Databricks SQL

## Analytics on the Lakehouse

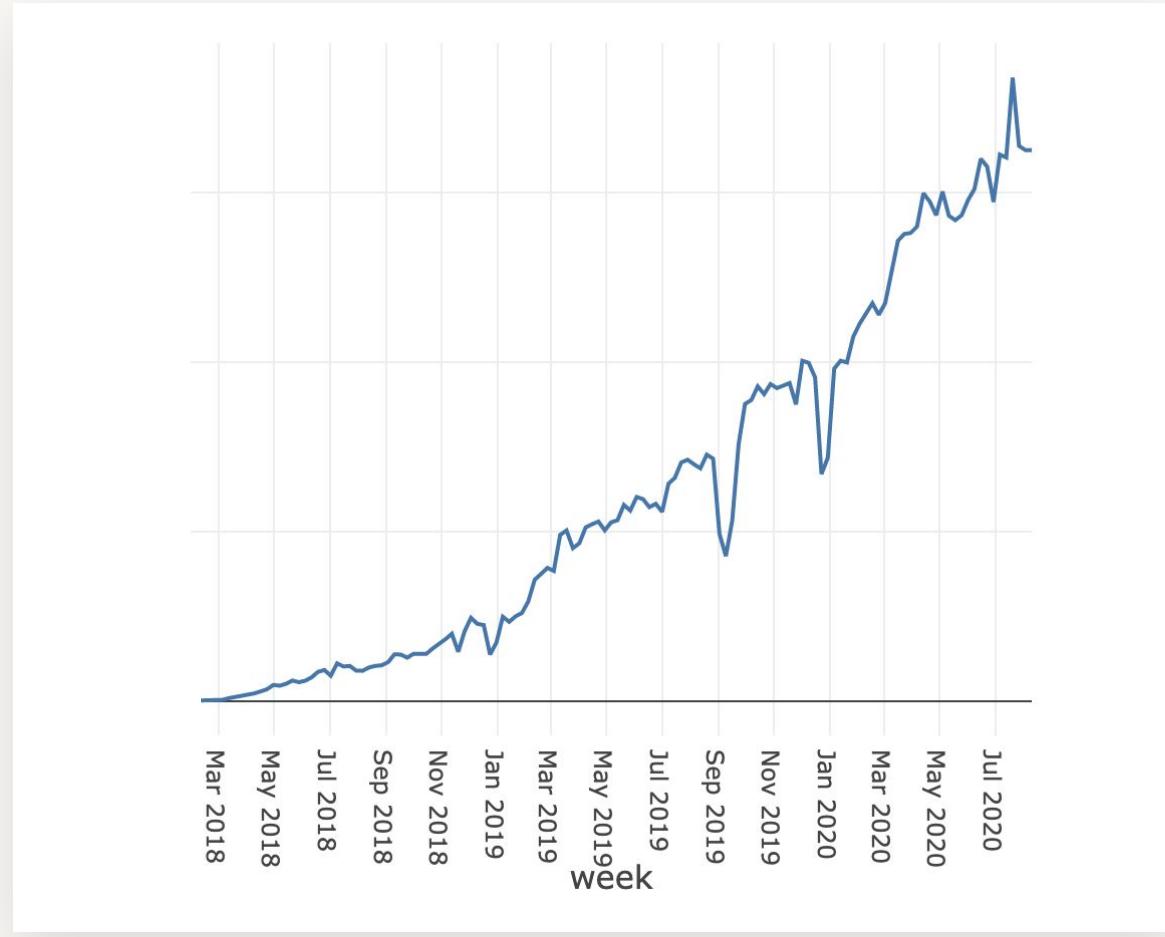
World-Class Performance at  
Data Lake Economics

- Fast and predictable performance for all queries
- Simplified administration and fine-grained governance
- Analytics on the freshest data with your tools of choice



# Why building the Lakehouse ?

## What we heard from our customers



- 1 Delta Lake enabled robust data management on the data lake.
- 2 Customers increasingly use SQL to directly query data lake data.
- 3 All the data is going to the lake. Only a portion gets into the data warehouse.

*"We do not want  
another data warehouse"*

– Our customers

# Myth or Reality ?

# Databricks sets official data warehousing performance record and outperforms cloud data warehouses with lakehouse



Learn more at <https://dbricks.co/benchmark>



# How to build a Lakehouse?



# Inner Workings of Lakehouse

**Consumption Layer**

**Compute Layer**

**Storage Layer**



# Inner Workings of Lakehouse

**Consumption Layer**

**Compute Layer**

**Storage Layer**



# Delta Lake is the foundation of the Lakehouse

An open format storage layer built for lake-first architecture



ACID Transactions, Time travel, Highly available

Advanced indexing, Caching, Auto-tuning

Fine-grained, role-based access controls

Streaming & batch, Analytics & ML

Python, SQL, R, Scala

# Building the foundation of the Lakehouse

Greatly improve data quality for end users – on your existing data lake



# Inner Workings of Lakehouse

**Consumption Layer**

**Compute Layer**

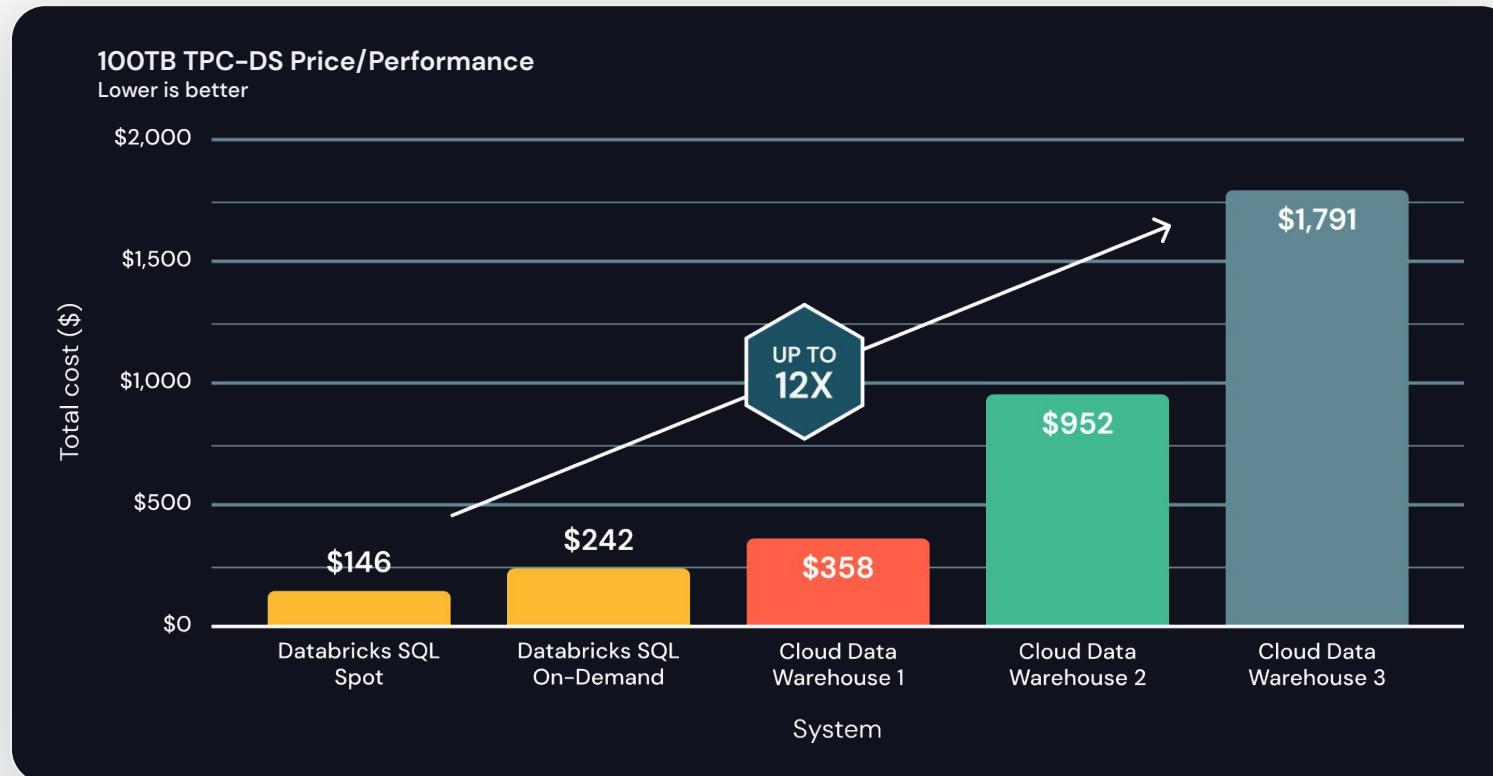
**Storage Layer**



# Best price / performance

## Lightning fast analytics

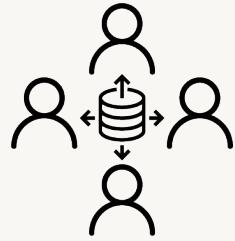
Query and analyze your most complete and freshest data with  
**up to 12x better price/performance** than legacy cloud data warehouses.



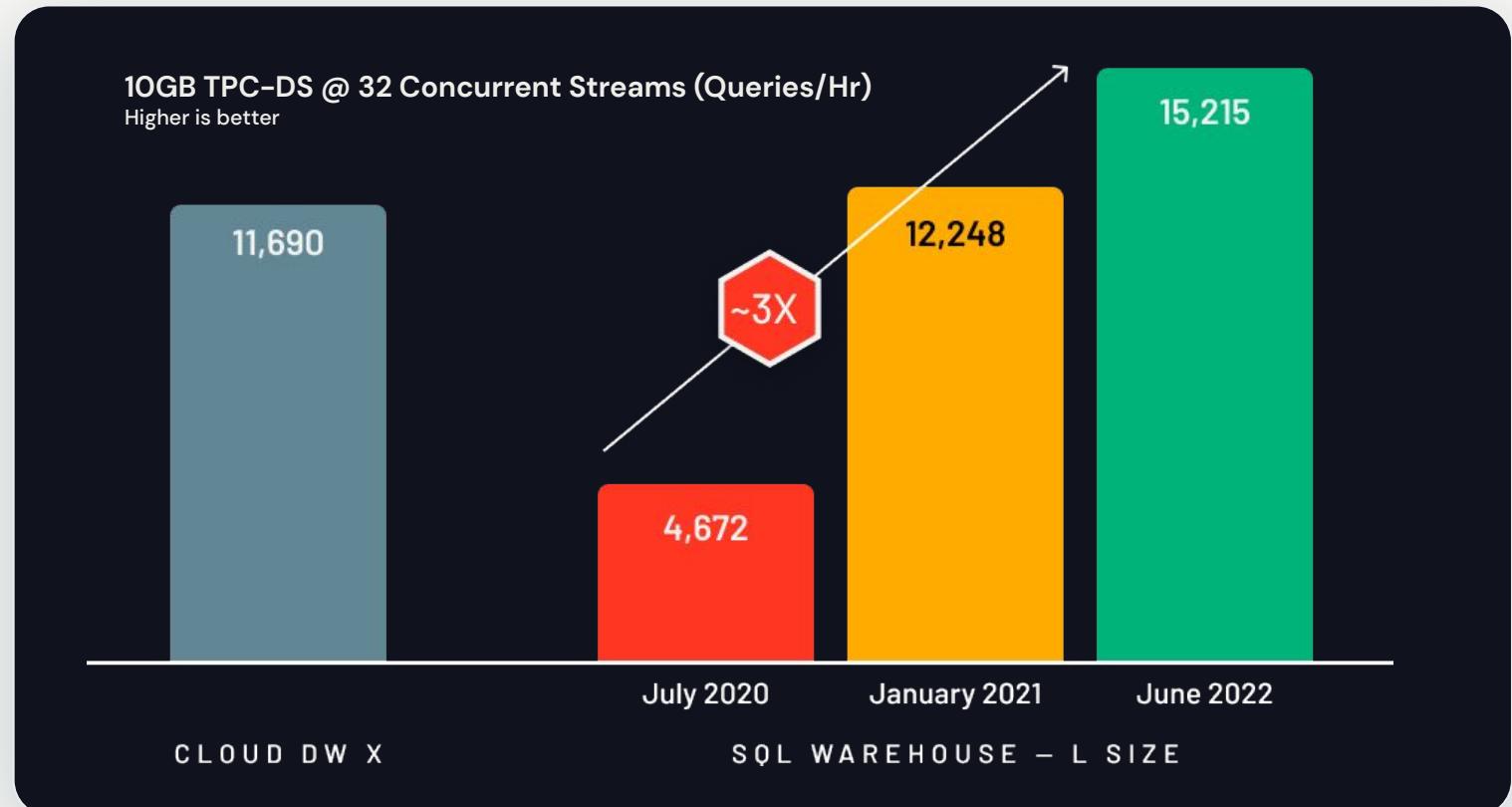
# Fast and predictable performance for all queries

## Beyond large query performance

### High Concurrency



Shaving system overhead off and improving QpH by 3X

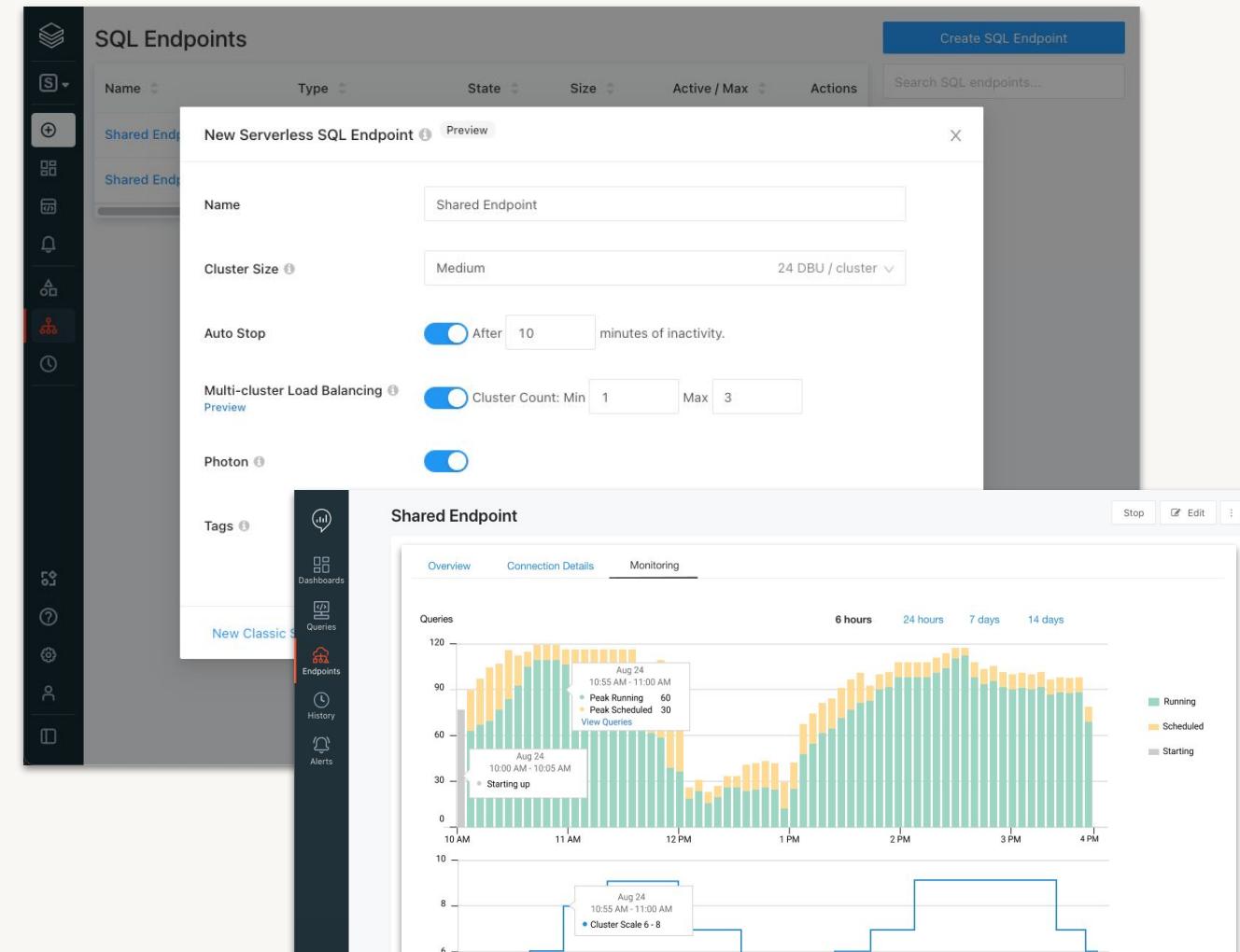


# Simple administration – no management required

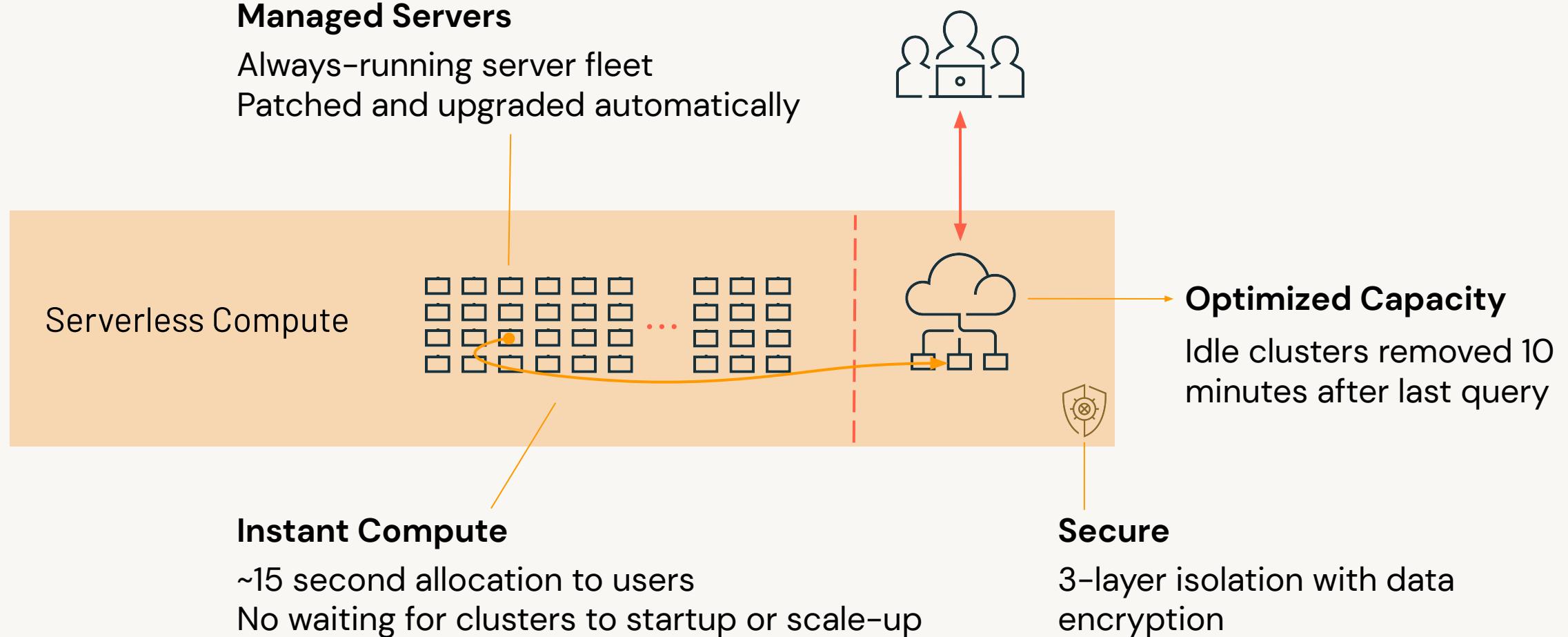
Elastic, instant compute decoupled from storage – now in Serverless

Quickly setup instant, elastic SQL compute decoupled from storage. Databricks automatically determines instance types and configuration for the best price/performance.

Then, easily manage users, data, and resources with endpoint monitoring, query history, and data explorer.



# Databricks SQL Serverless



# Inner Workings of Lakehouse

Consumption Layer

Compute Layer

Storage Layer



# Fine-grained governance on the Lakehouse

## Introducing Databricks Unity Catalog

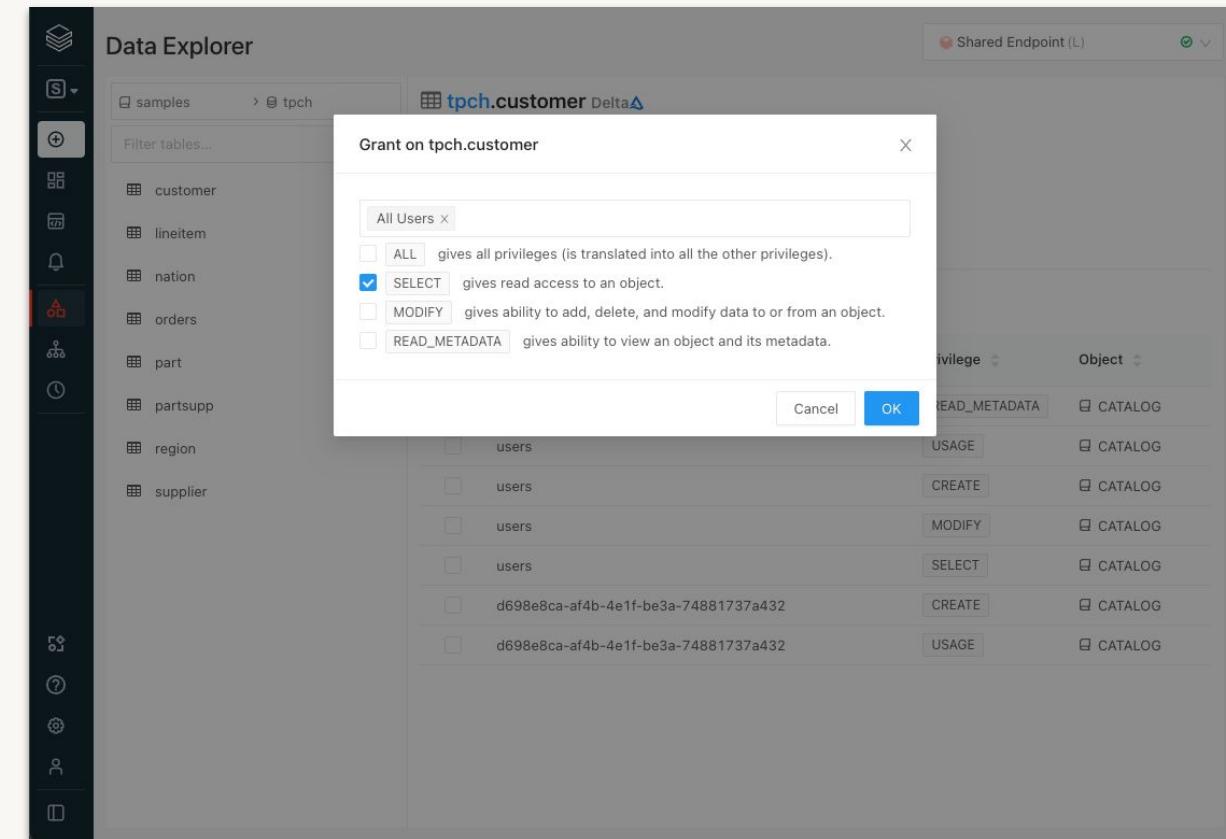
- **Centrally catalog, search, and discover** data and AI assets
- Simplify governance with a **unified cross-cloud governance model**
- **Easily integrate** with your existing Enterprise Data Catalogs
- **Securely share live data across platforms** with delta sharing

The screenshot shows the Databricks Unity Catalog interface for a table named 'Firehose'. The left sidebar includes navigation icons and a search bar. The main panel displays the following sections:

- Description:** A table containing raw events from across the LOC platform. Use this table to find all reported game-play events.
- Recommendations:** Optimizing metric may improve performance. Learn more. Table\_name has not been vaccinated in 90 days. Learn more.
- Owners:** A list of users associated with the table.
- Frequent users:** A list of users who frequently interact with the table.
- ABAC Policies:** A list of policies applied to the table, including PII, Cost, and Inventory.
- Top Queries:** A list of top queries executed against the table, such as 'Champions stats 1H 2021', 'Gameplay analysis Q2 2021', 'Gameplay hours LOC semi-finals', and 'Purchase prediction model 2021'.
- Statistics:** Information about the table's format (Delta), last update (5 hours ago), creation date (1 year ago), and size (500 GB).
- Schema:** Details for columns: ProductID (integer), Status (string), PricingTier (float), DistributionTier (string), and AccountInfo (string). Each column has an 'Inventory' tag.
- Lineage:** A diagram showing the flow of data from 'inbindcall' through 'Firehose' to 'disp\_rec' and 'inventory'.
- Privileges:** A table showing permissions for 'User / Group' (bi\_team, dev) and 'Permissions' (Select, Modify, Manage, Select) at different times.
- Data Sample:** A preview of the table data.

# Easy to govern self-served analytics

- Confidently onboard new users, discover, secure, and govern data
- Manage costs and usage effectively with endpoints monitoring and query history
- Meet compliance needs with built-in audit trail



# First-class SQL development experience

## Analytics & BI on the Lakehouse

Query data lake data using familiar **ANSI SQL**, and find and share new insights faster with the built-in SQL query editor, alerts, visualizations, and interactive dashboards.

The screenshot displays the Databricks platform interface. On the left, the Schema Browser shows tables like customer, lineitem, nation, and orders with their respective column details. The main area features a query editor with the following SQL code:

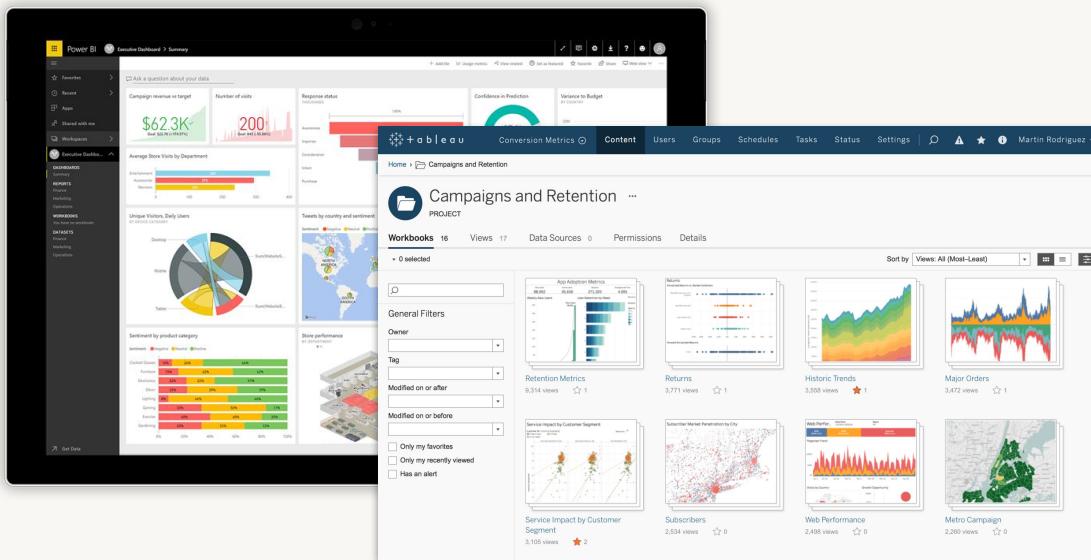
```
1 SELECT
2     o_orderdate AS Date,
3     o_orderpriority AS Priority,
4     sum(o_totalprice) AS 'Total Price'
5 FROM
6     `samples`.`tpch`.`orders`
7 WHERE
8     o_orderdate > '1994-01-01'
9     AND o_orderdate < '1994-01-31'
10 GROUP BY
11     1
```

Below the query editor are several dashboards and charts. One chart shows "Retail Revenue & Supply Chain" with counts of 750,000 Unique Customers and 50,000 Unique Suppliers. Another chart, "National Revenue Trends", is a stacked bar chart showing revenue from 1994 to 1998 across countries: ARGENTINA, BRAZIL, CHINA, JAPAN, UNITED KINGDOM, FRANCE, JORDAN, and UNITED STATES. A third chart, "Shifts in Pricing Priorities", is a line graph showing price trends over time from Jan 2 to Jan 30, 1994, with a legend for priority levels: 1-URGENT (red), 2-HIGH (orange), 3-MEDIUM (yellow), 4-NOT SPECIFIED (blue), and 5-LOW (grey). A "Global Revenue Analysis" map shows revenue distribution across continents.

# A platform for your tools of choice

## Analytics & BI on the Lakehouse

Get critical business data in with one click integrations, and benefit from fast performance, low latency, and high user concurrency for **your existing BI tools**.



Ingest & ETL

BI & Analytics



# Databricks SQL Lab Overview



©2021 Databricks Inc. — All rights reserved



# Lab Context



- As part of the lab today, we will be working with APJuice data from different juice outlets across APJ
- Dataset consists of sales, stores, customer and product info
- The lab will focus on analysing data that has been processed using batch to build a star schema data model in Delta Lake Gold Layer and then aggregating the data for dashboarding



# databricks Lakehouse Platform

SIMPLE ◇ OPEN ◇ COLLABORATIVE

Data Engineering

BI & SQL  
Analytics

Real-time Data  
Applications

Data Science  
& Machine Learning

Data Management & Governance



DELTA LAKE

Open Data Lake



Structured



Semi-structured



Unstructured



Streaming



# databricks Lakehouse Platform

SIMPLE ◇ OPEN ◇ COLLABORATIVE

Data Engineering

BI & SQL  
Analytics

Real-time Data  
Applications

Data Science  
& Machine Learning

Data Management & Governance



DELTA LAKE

Open Data Lake



Structured



Semi-structured



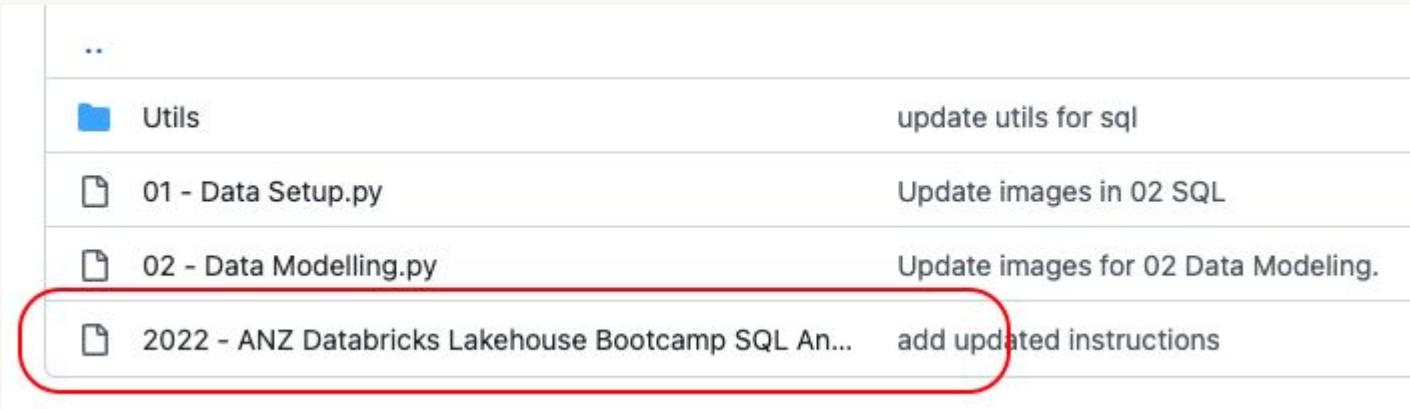
Unstructured



Streaming

# Download The Workbook

The workbook is in the lab git repo in



**<https://bit.ly/3NQiFLS>**

[https://github.com/zivelenorkunaite/apjbootcamp2022/LAB O2-SQL/](https://github.com/zivelenorkunaite/apjbootcamp2022/LAB%20O2-SQL/)

# Databricks SQL Lab



©2021 Databricks Inc. — All rights reserved



# Afternoon Tea & Feedback

