

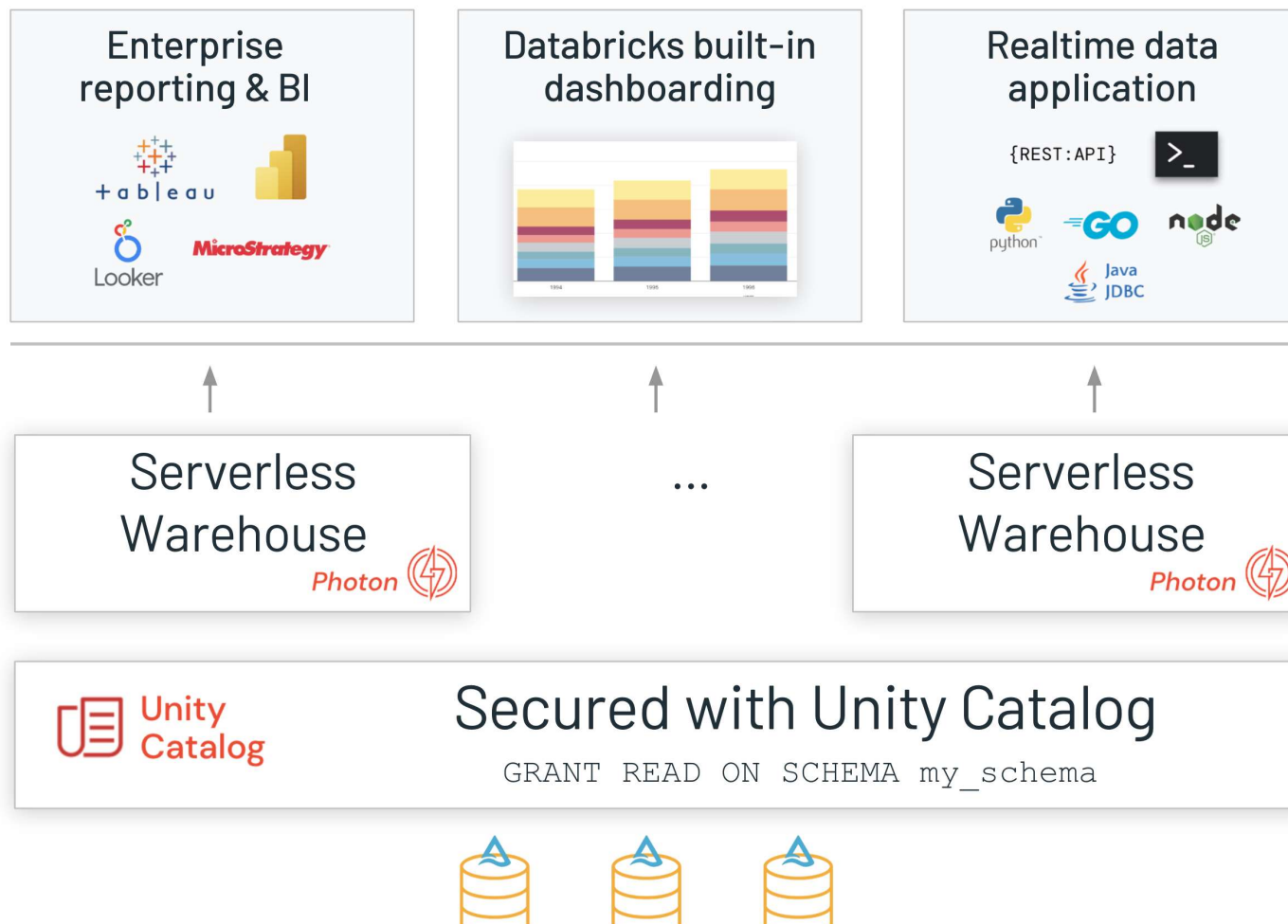
databricks03 - BI and Data Warehousing

(<https://databricks.com>)

Your Lakehouse is the best Warehouse

Traditional Data Warehouses can't keep up with the variety of data and use cases. Business agility requires reliable, real-time data, with insight from ML models.

Working with the lakehouse unlock traditional BI analysis but also real time applications having a direct connection to your entire data, while remaining fully secured.





Instant, elastic compute
Lower TCO with Serverless
Zero management
Governance layer - row level

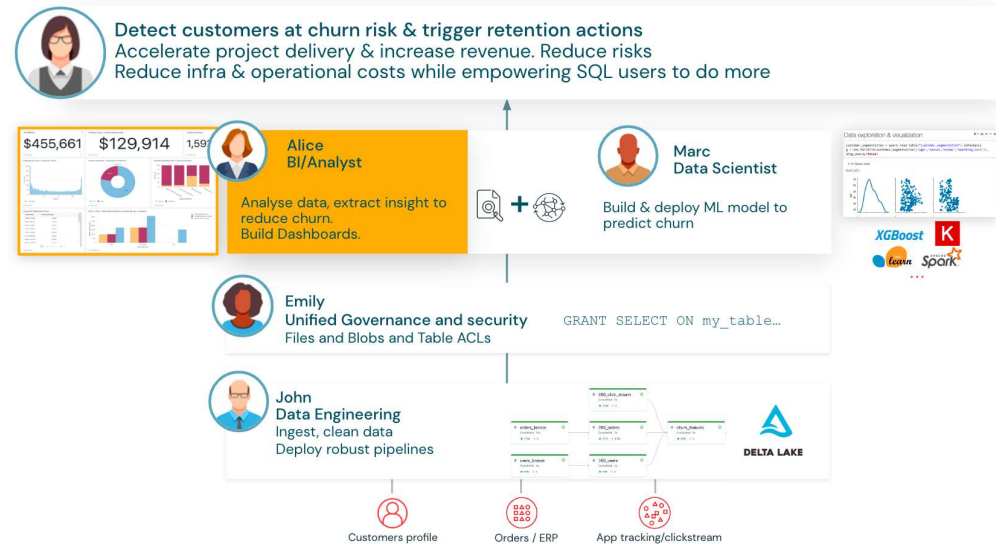
BI & Datawarehousing with Databricks SQL

Our datasets are now properly ingested, secured, with a high quality and easily discoverable within our organization.

Let's explore how Databricks SQL support your Data Analyst team with interactive BI and start analyzing our customer Churn.

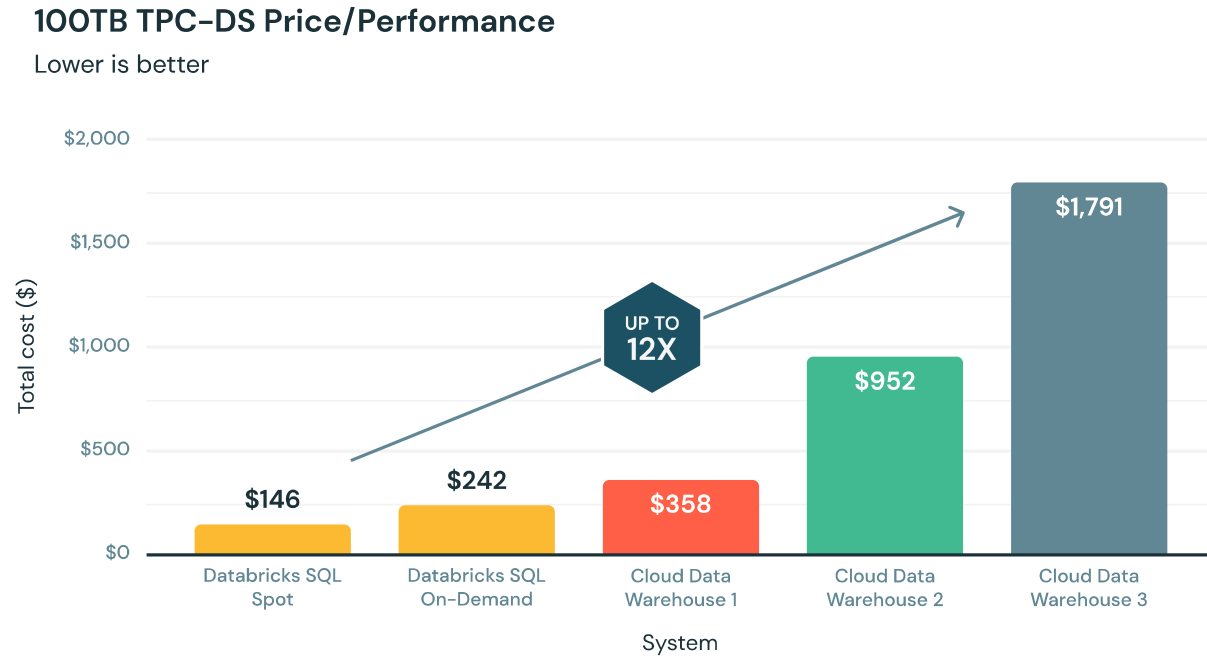
To start with Databricks SQL, open the SQL view on the top left menu.

You'll be able to:



Databricks SQL Warehouses: best-in-class BI engine

Databricks SQL is a warehouse engine packed with thousands of optimizations to provide you with the best performance for all your tools, query types and real-world applications. It holds the Data Warehousing Performance Record.



(<https://www.databricks.com/blog/2021/11/02/databricks-sets-official-data-warehousing-performance-record.html>)

This includes the next-generation vectorized query engine Photon, which together with SQL warehouses, provides up to 12x better price/performance than other cloud data warehouses.

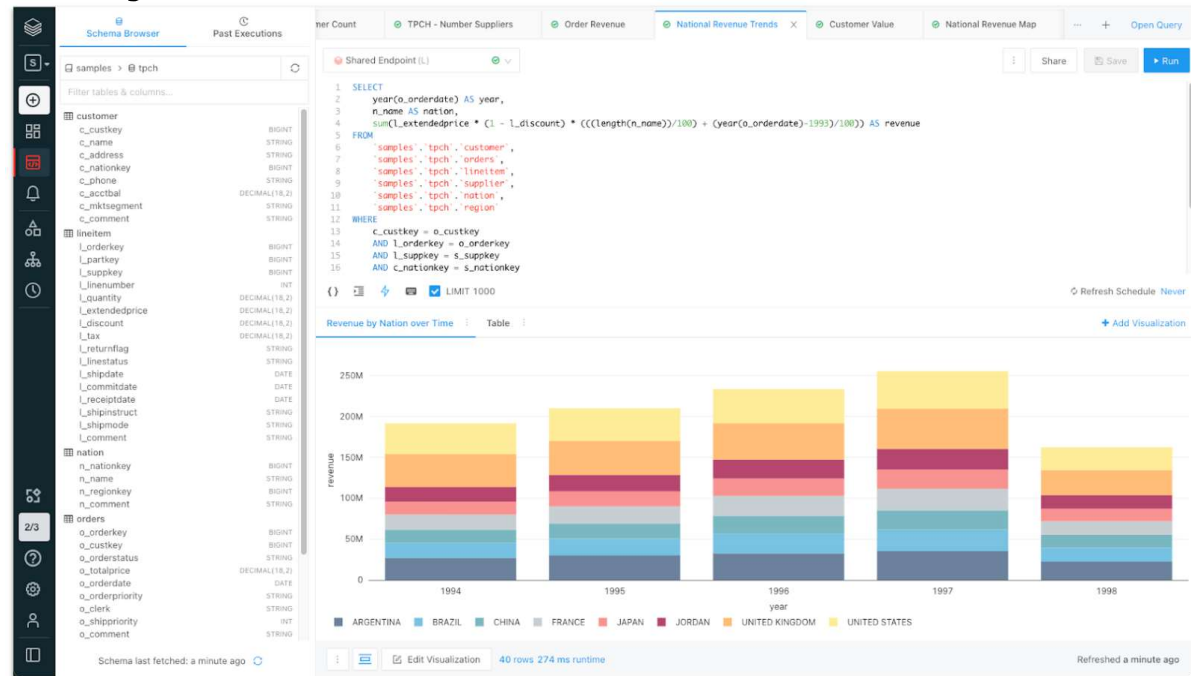
Serverless warehouse provide instant, elastic SQL compute — decoupled from storage — and will automatically scale to provide unlimited concurrency without disruption, for high concurrency use cases.

Creating your first Query

Our users can now start running SQL queries using the SQL editor and add new visualizations.

By leveraging auto-completion and the schema browser, we can start running adhoc queries on top of our data.

While this is ideal for Data Analyst to start analysing our customer Churn, other personas can also leverage DBSQL to track our data inspection pipeline the data



```
%run ./includes/SetupLab
```

```
print("For the following exercise use the following catalog.schema : \n" + labContext.catalogAndSchema() )
```

For the following exercise use the following catalog.schema :

```
cloud_lakehouse_labs.odl_user_1237583_databricks_labs_com_retail
```

Lab exercise

Create the following queries and visualisations using the above catalog and schema

1. Total MRR

```
SELECT
  sum(amount)/1000 as MRR
FROM churn_orders
WHERE
  month(to_timestamp(transaction_date, 'MM-dd-yyyy HH:mm:ss')) =
  (
    select max(month(to_timestamp(transaction_date, 'MM-dd-yyyy HH:mm:ss')))
    from churn_orders
  );
```

Create a *counter* visualisation

2. MRR at Risk

```
SELECT
  sum(amount)/1000 as MRR_at_risk
FROM churn_orders
WHERE month(to_timestamp(transaction_date, 'MM-dd-yyyy HH:mm:ss')) =
  (
    select max(month(to_timestamp(transaction_date, 'MM-dd-yyyy HH:mm:ss')))
    from churn_orders
  )
and user_id in
  (
    SELECT user_id FROM churn_prediction WHERE churn_prediction=1
  )
```

Create a *counter* visualisation

3. Customers at risk

```
SELECT count(*) as Customers, cast(churn_prediction as boolean) as `At Risk`  
FROM churn_prediction GROUP BY churn_prediction;
```

For 4 and 5 switch to the schema in the hive metastore where the DLT tables are created

4. Customer Tenure - Historical

```
SELECT cast(days_since_creation/30 as int) as days_since_creation, churn, count(*) as customers  
FROM churn_features  
GROUP BY days_since_creation, churn having days_since_creation < 1000
```

Create a *bar* visualisation

5. Subscriptions by Internet Service - Historical

```
select platform, churn, count(*) as event_count  
from churn_app_events  
inner join churn_users using (user_id)  
where platform is not null  
group by platform, churn
```

Create a *horizontal bar* visualisation

For the rest switch back to the original catalog and schema (if applicable)

6. Predicted to churn by channel


```
SELECT channel, count(*) as users  
FROM churn_prediction  
WHERE churn_prediction=1 and channel is not null  
GROUP BY channel
```

Create a *pie chart* visualisation

7. Predicted to churn by country

```
SELECT country, churn_prediction, count(*) as customers  
FROM churn_prediction  
GROUP BY country, churn_prediction
```

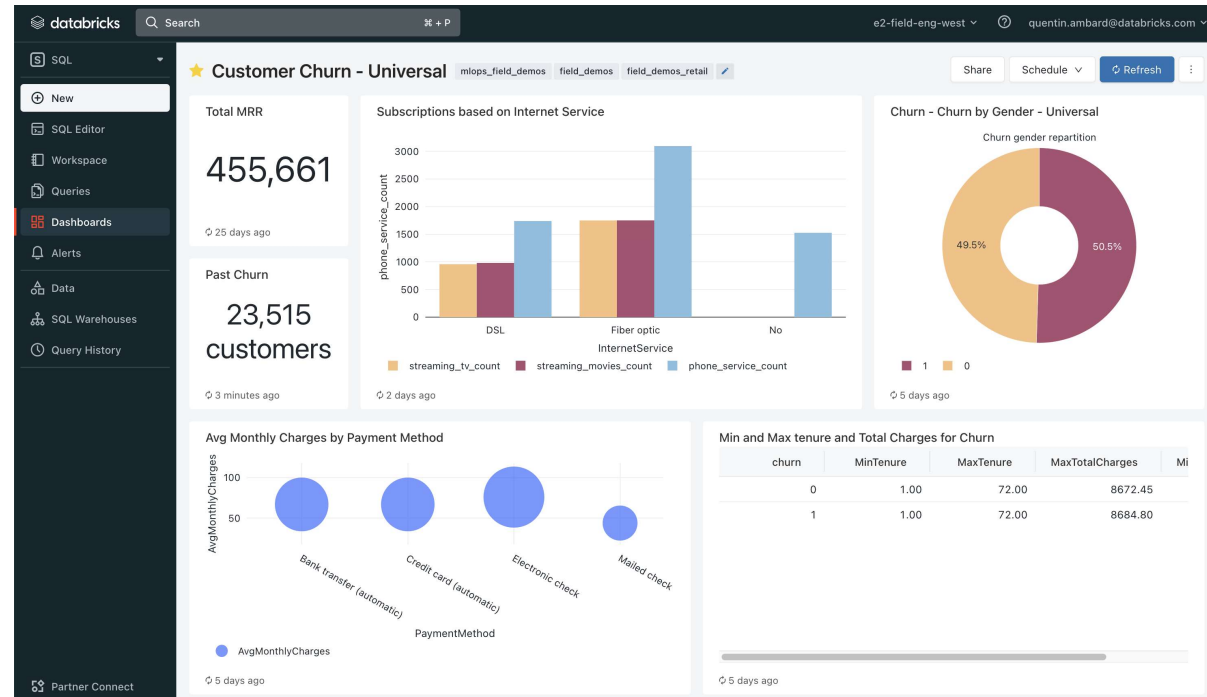
Create a *bar* visualisation

Creating our Churn Dashboard

The next step is now to assemble our queries and their visualization in a comprehensive SQL dashboard that our business will be able to track.

Lab exercise

Assemble the visualisations defined with the above queries into a dashboard



Using Third party BI tools

SQL warehouse can also be used with an external BI tool such as Tableau or PowerBI.

This will allow you to run direct queries on top of your table, with a unified security model and Unity Catalog (ex: through SSO). Now analysts can use their favorite tools to discover new business insights on the most complete and freshest data.

Tableau - [Tableau Databricks Connector](#)

Going further with DBSQL & Databricks Warehouse

Databricks SQL offers much more and provides a full warehouse capabilities

Data modeling

Comprehensive data modeling. Save your data based on your requirements: Data vault, Star schema, Inmon...

Databricks let you create your PK/FK, identity columns (auto-increment)

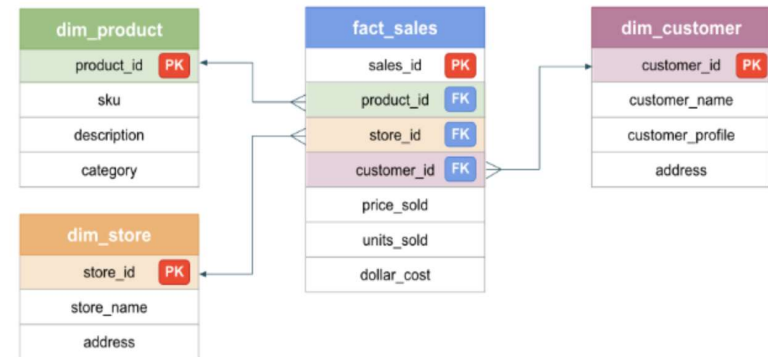
Data ingestion made easy with DBSQL & DBT

Turnkey capabilities allow analysts and analytic engineers to easily ingest data from anything like cloud storage to enterprise applications such as Salesforce, Google Analytics, or Marketo using Fivetran. It's just one click away.

Then, simply manage dependencies and transform data in-place with built-in ETL capabilities on the Lakehouse (Delta Live Table), or using your favorite tools like dbt on Databricks SQL for best-in-class performance.

Query federation

Need to access cross-system data? Databricks SQL query federation let you define datasources outside of databricks (ex: PostgreSQL)



Next up

Orchestrating and automating with Workflows ([\\$.04%20-%20Orchestrating%20with%20Workflows](#))

