

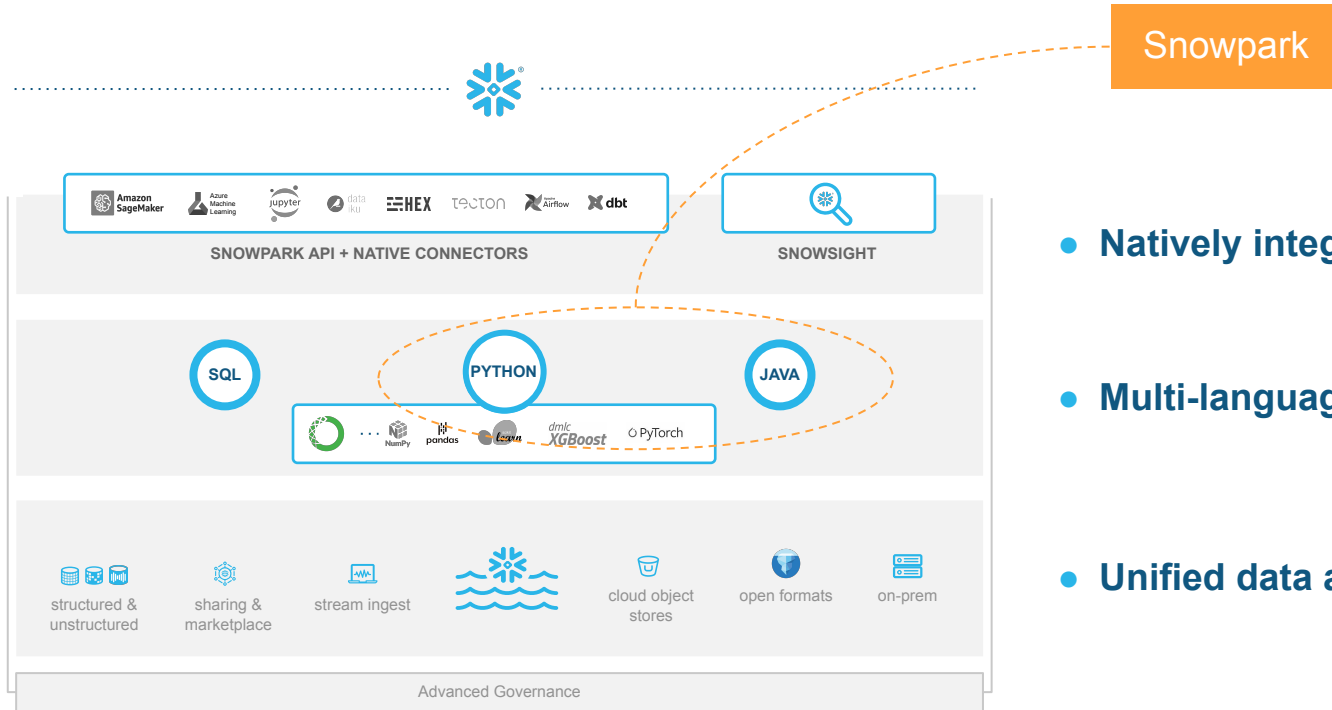


# TRAIN, DEPLOY & RUN ML MODELS USING PYTHON, SNOWPARK & STREAMLIT

Ahmad Khan  
Head of AI/ML Strategy, Snowflake

March 2023

# Snowflake Platform for Data Science & ML



- **Natively integrated ecosystem**
- **Multi-language, elastic engine**
- **Unified data access**

# WHY SNOWPARK



Data Engineer

Python/Scala



Data Engineer

SQL

```
snowdf =  
session.table("products").  
filter(col("id") === 1)
```



**Snowpark:** Native Dataframes  
for Snowflake

```
select * from products  
where id = 1;
```



# SNOWPARK



Program in Java, Scala and **Python**  
against data in Snowflake



# CODE THE SAME WAY, EXECUTE FASTER WITH SNOWPARK

Dataframes Library in your dev environment

CLIENT-SIDE

SQL

SCALA

PYTHON

JAVA

OTHER

SERVER-SIDE

SQL

JAVA

PYTHON

JAVASCRIPT

EXTERNAL

SNOWFLAKE PROCESSING ENGINE



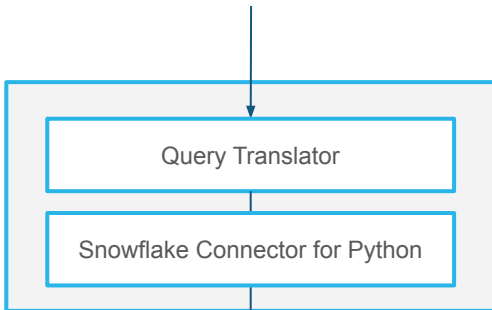
Native Python UDFs and SPROCs



## DataFrame API Query

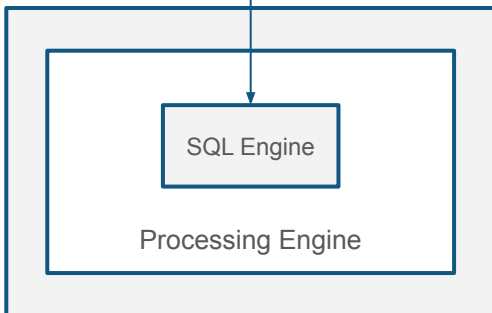
```
df.filter(df.state == 'WA')  
.avg(df.amount)
```

SNOWPARK  
CLIENT API



{...} SQL Query

SNOWFLAKE



# DataFrame API



Query Snowflake data  
with Python



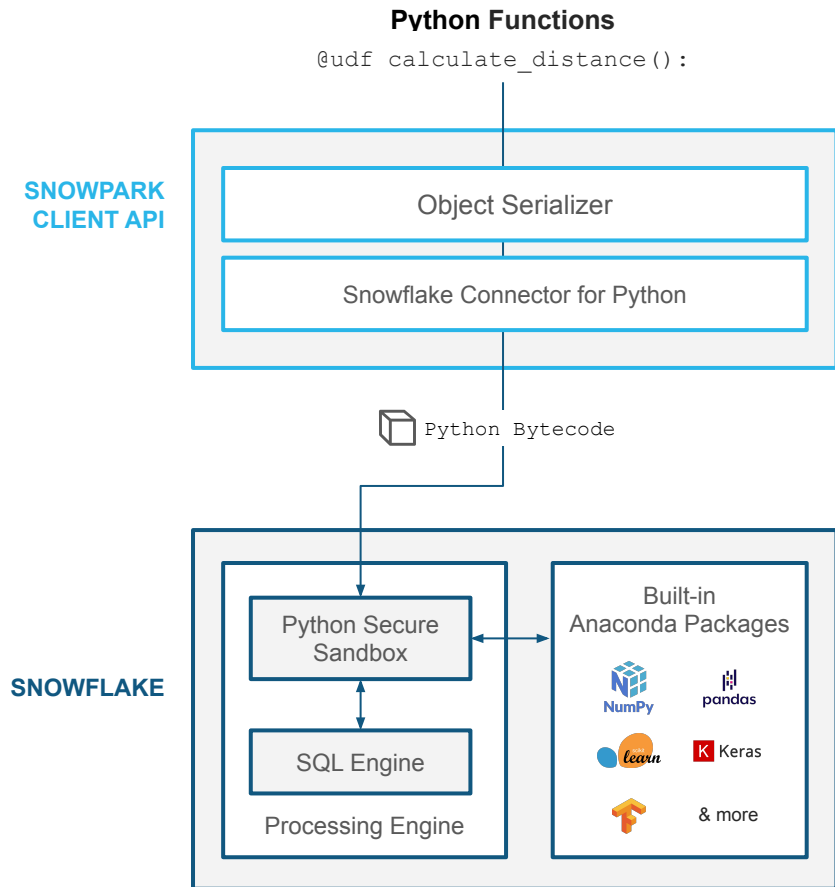
Familiar DataFrame API



100% push-down to Snowflake

Native Snowflake performance  
and scale





# Python Functions

- > Bring custom Python code to Snowflake as User Defined Functions (UDFs)
- > Code is serialized and pushed down to run in a secure sandboxed environment
- > Seamlessly access third-party packages with Anaconda integration





# SNOWPARK FOR PYTHON

SNOWPARK  
CLIENT API

DataFrame Query

Python Functions & Objects

```
predict(df.filter(  
df.state == 'WA'))
```

```
@udf def predict()
```

Query Translator

Object Serializer

Snowflake Connector for Python

{...} SQL Query

Python Bytecode

SNOWFLAKE

Processing Engine

SQL Engine

Python Secure  
Sandbox

Built-in Anaconda  
Packages

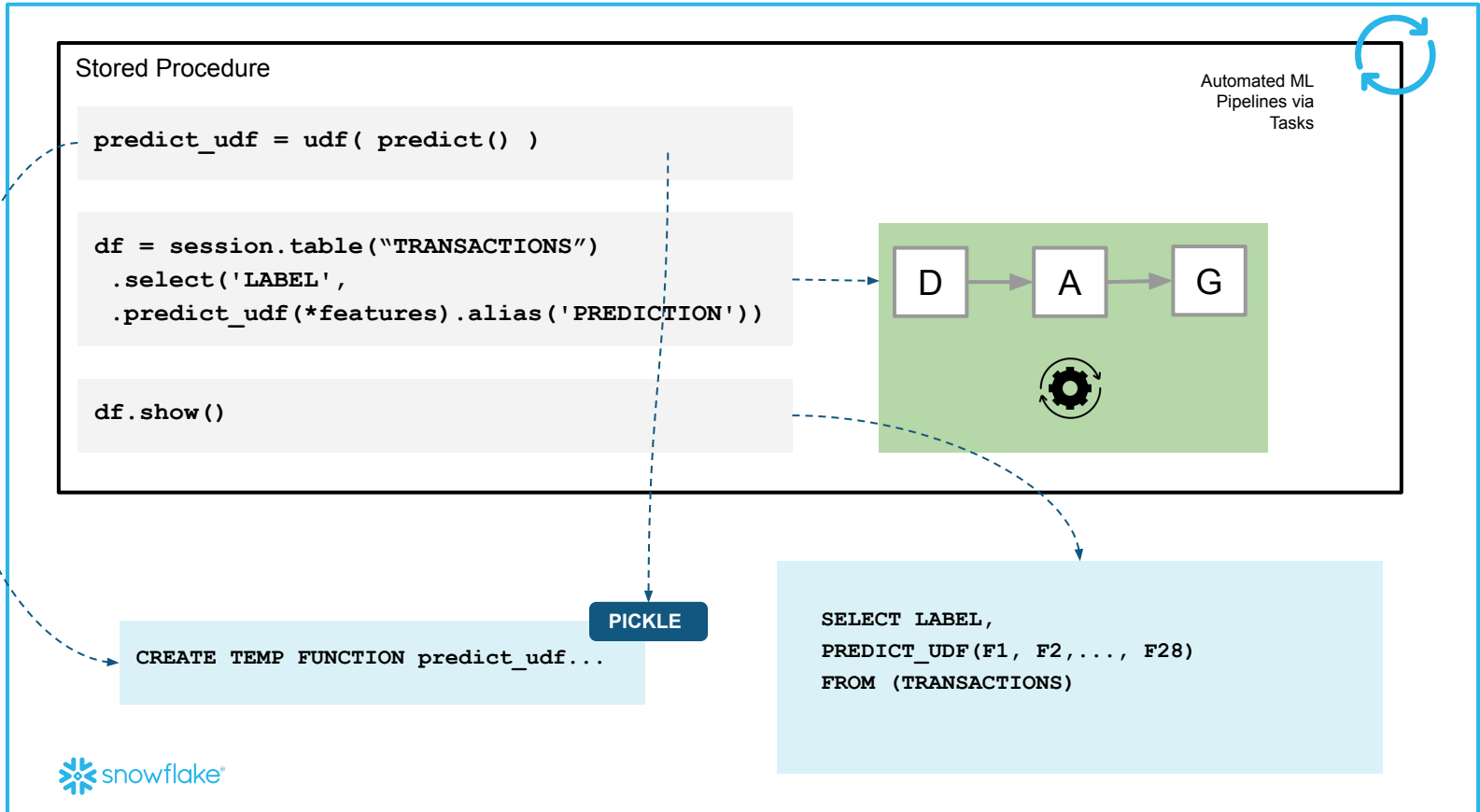


& more





# EXAMPLE: SNOWPARK INFERENCE PIPELINE



# Streamlit in Snowflake

## Build Interactive Apps with Python in Snowflake

The screenshot shows a Streamlit application running in Snowflake. The interface is divided into three main sections:

- Left Panel (File Explorer):** Shows a tree view of the Snowflake database structure, including schemas like APP\_DB, INFORMATION\_SCHEMA, and ML\_TRAINING. The 'ROI\_PREDICTION' streamlit object is highlighted.
- Code Editor:** Displays Python code for the application. The code includes a function to predict ROI based on advertising budgets and a chart to visualize the predicted revenue.
- Application View:** Shows the rendered application titled 'SportsCo Ad Spend Optimizer'. It features:
  - Advertising budgets:** A table with columns for Search engine, Video, Social media, and Email. Each cell contains a value and a slider control.
 

Search engine	Video	Social media	Email
70	85	60	72
  - Predicted revenue:** A large text display showing '\$ 0.30 million' and a trend indicator '-96.5% vs last month'.

 First-class integration of Streamlit

 Python-based app development

 Side-by-side editor in your browser



# Snowpark Guided Workshop

<https://tinyurl.com/2skz2s4a>



Getting Started with Data Engineering and ML using Snowpark for Python

13 mins remaining

- 1 Overview
- 2 Setup Environment
- 3 Clone GitHub Repository
- 4 Data Engineering
- 5 Data Pipelines
- 6 Machine Learning
- 7 Streamlit Application**
- 8 Conclusion And Resources

```
streamlit run Snowpark_Streamlit_Revenue_Prediction.py
```

If all goes well, you should see a browser window open with the app loaded as shown below.



**THANK YOU**

