



Snowpark Snowflake's Developer Framework

Speakers

Phani Raj

Senior Data Cloud Architect - GSI Partners
Snowflake

Prathamesh Nimkar

Senior Data Cloud Architect - GSI Partners
Snowflake



Agenda

Snowflake Overview

Snowpark Overview

Demo

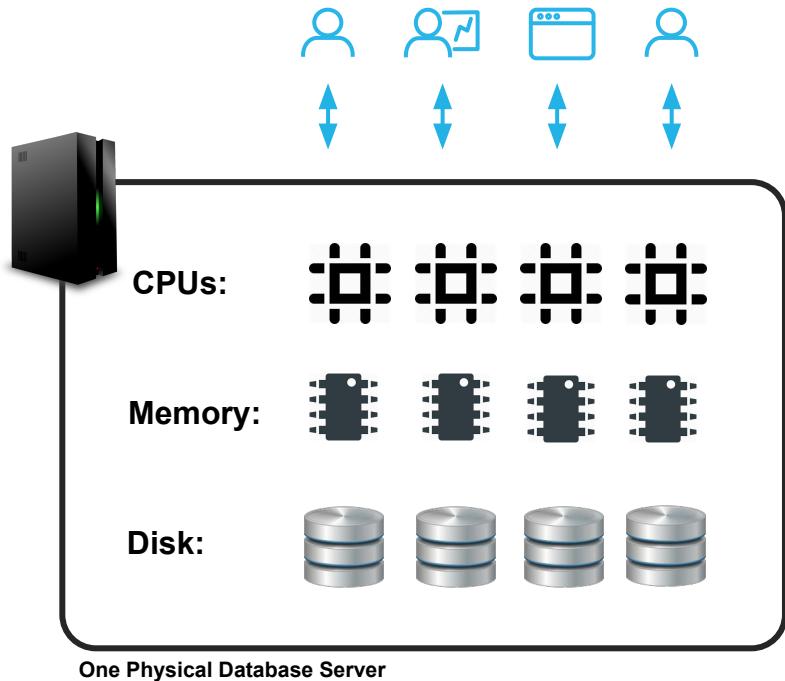
Q&A



Snowflake Overview



Traditional Architecture

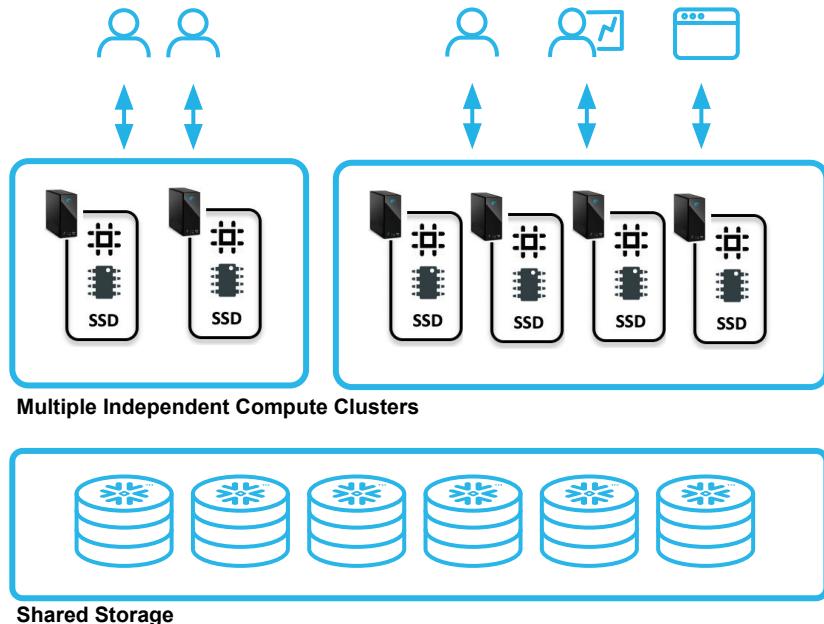


Outcome

- A single database server
- 100% shared resources
- Contention: A "tug of war"
- Hard to scale
- Option: Bigger server (months)



Snowflake Architecture

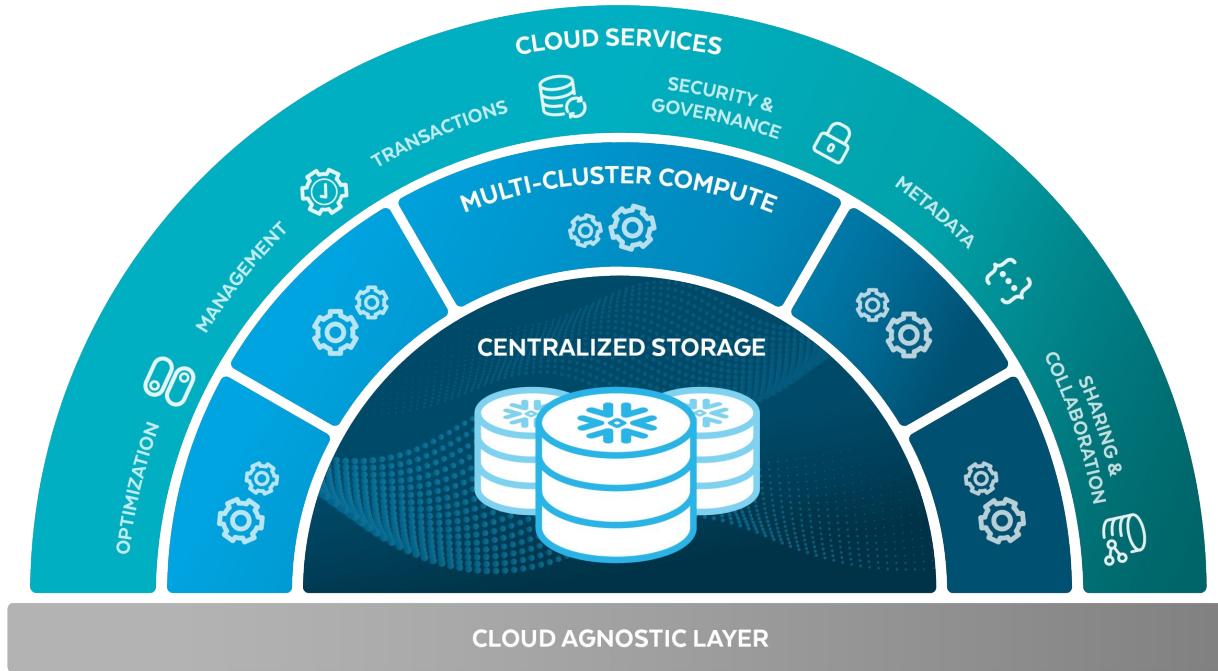


Outcome

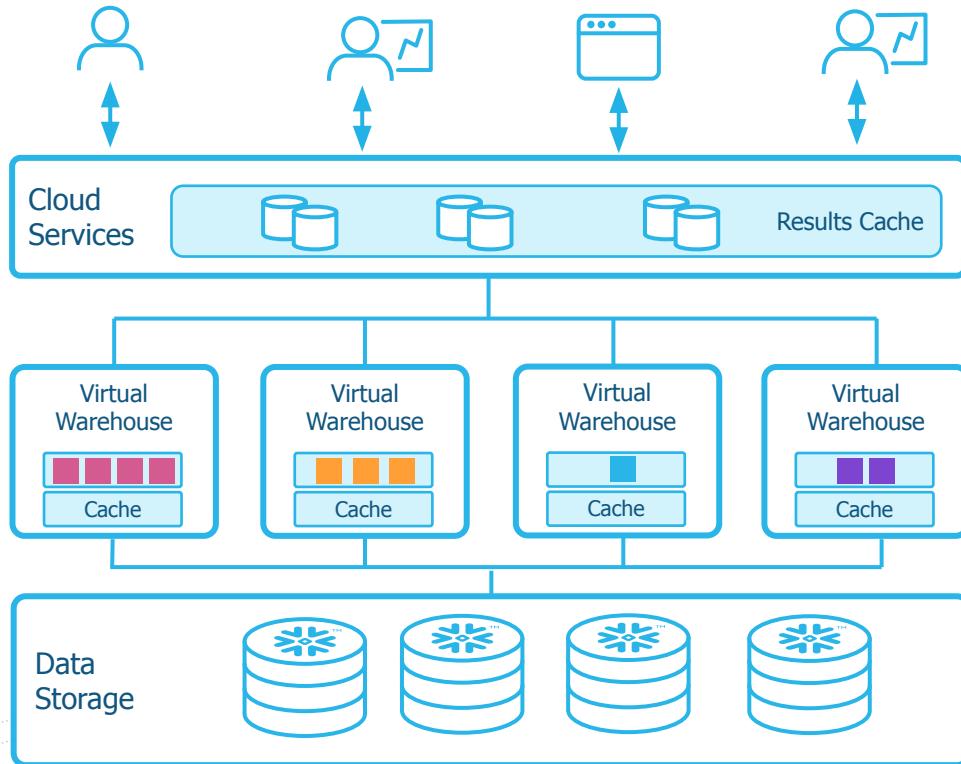
- Multiple Virtual Warehouses
- Compute entirely dynamic
- Can scale Up, Down or Suspend
- Milliseconds
- Unlimited Compute
- Unlimited Storage



Snowflake Architecture



Snowflake Architecture

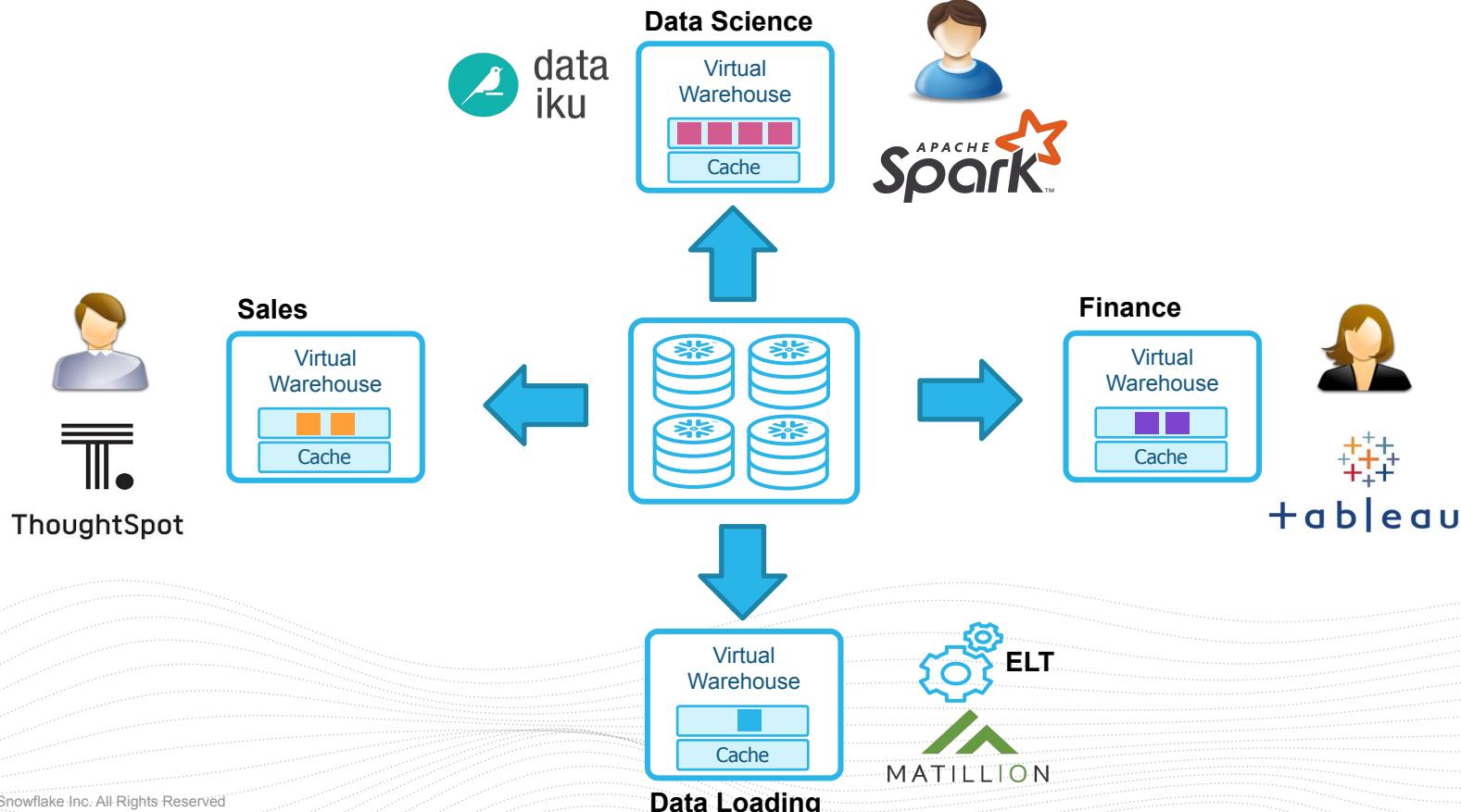


Cloud Services

Query Processing

Database Storage

Snowflake Virtual Warehouses



Snowpark Overview



WHAT IS SNOWPARK?



Python • Java • Scala • SQL

Developer framework that allows data engineers and other developers to code in their preferred way and execute pipeline from a single Snowflake platform.

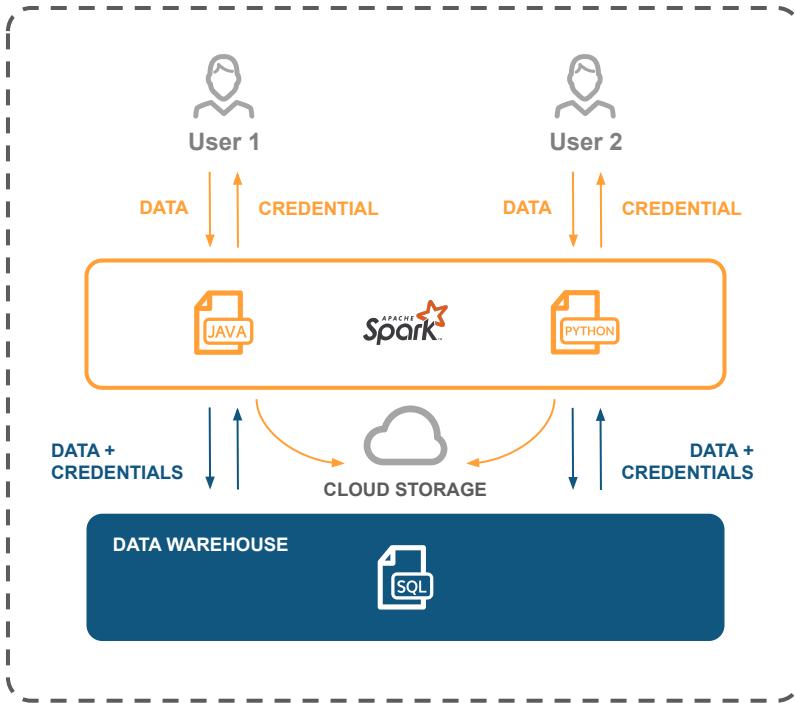
Language of Choice

No Governance Trade-offs

Faster & Cheaper Pipelines

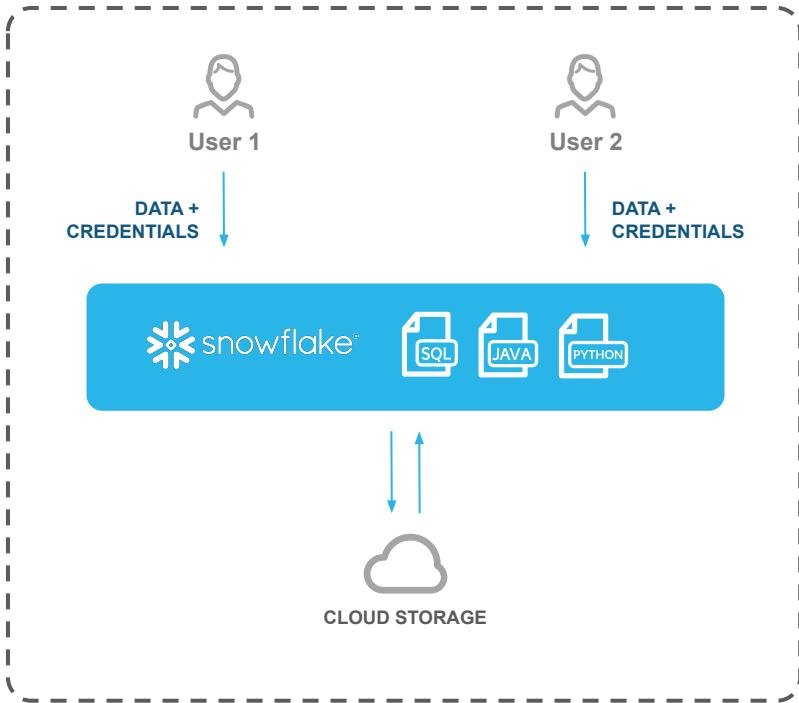


Complexity with Traditional Approach



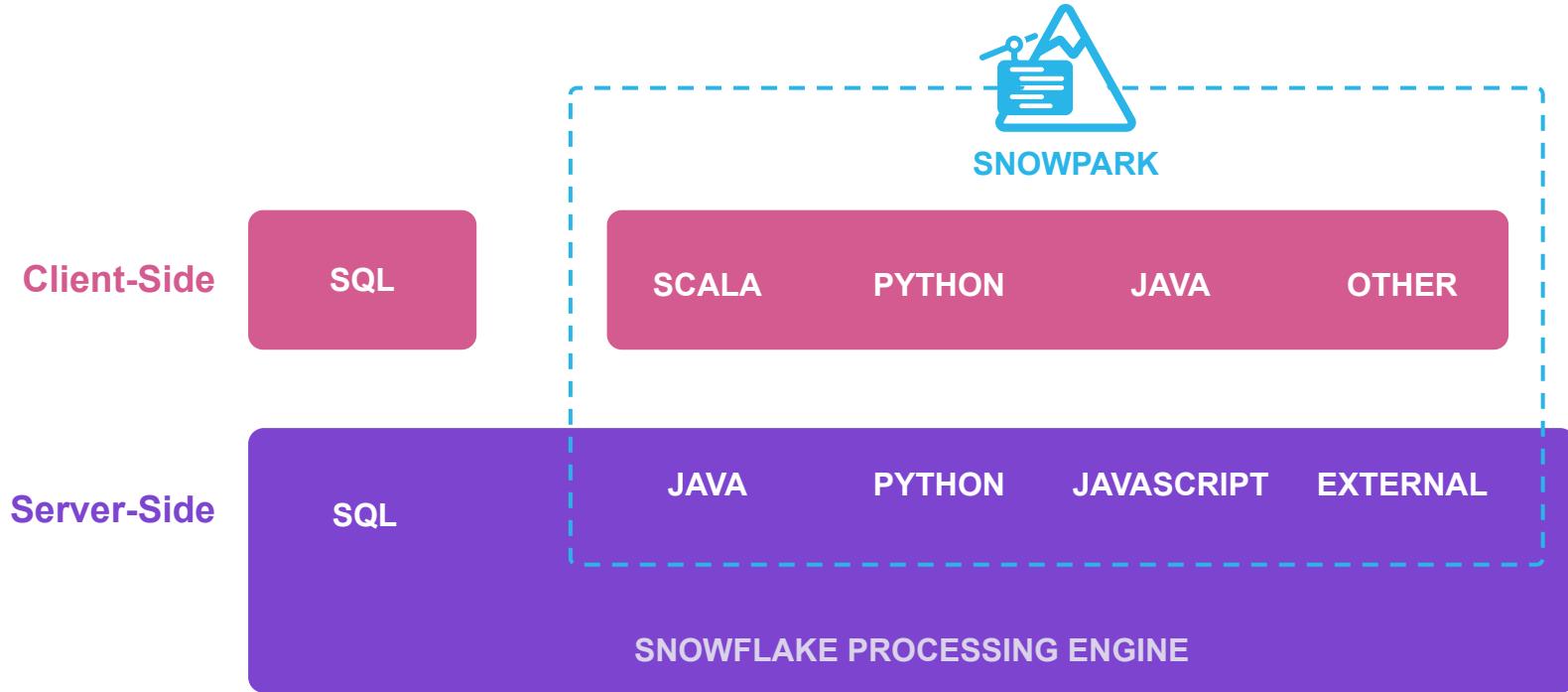
- Customers often run separate processing clusters for different languages
- Complex capacity management and resource sizing
- Lots of data movement and data silos
- Loose governance control and security loopholes

Streamlined Architecture with Snowflake

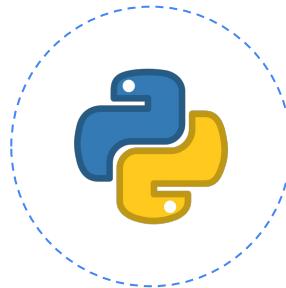


- One single platform with native support for different languages
- Simpler capacity management and resource sizing
- Streamlined architecture and collaboration on the same data
- Consistent governance and security policies

Code the Same Way, Execute Faster With Snowpark



Snowpark for Python



Familiar Programming Constructs

Use familiar syntax
with DataFrame
abstraction



Rich Ecosystem

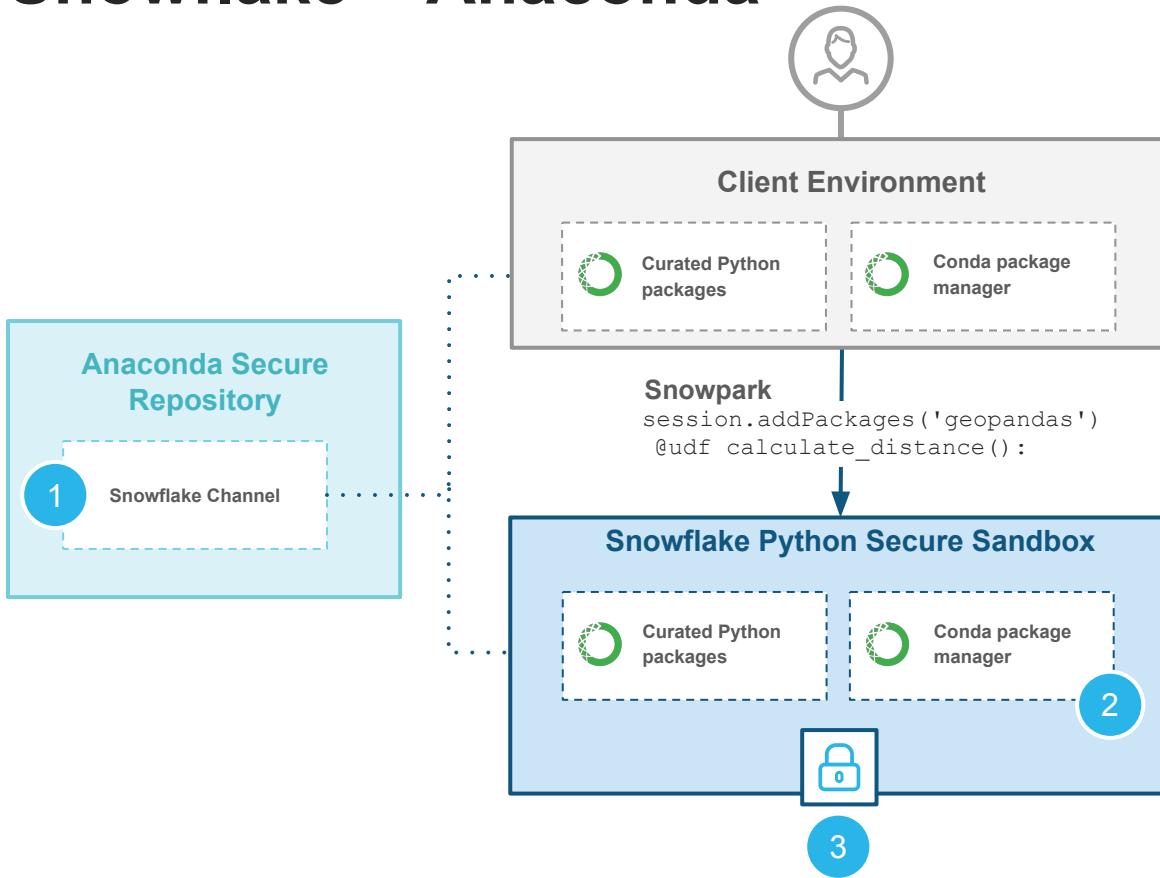
Easy access to hundreds of
packages with automated
dependency management



Secure Processing

Build with confidence
in a highly secure,
sandboxed environment

Snowflake + Anaconda



1 Easy Access

Curated packages pre-installed in Snowflake also available for local development

2 No Dependency Hell

Conda package manager integrated in Snowflake secure sandbox

3 Scalable and Secure

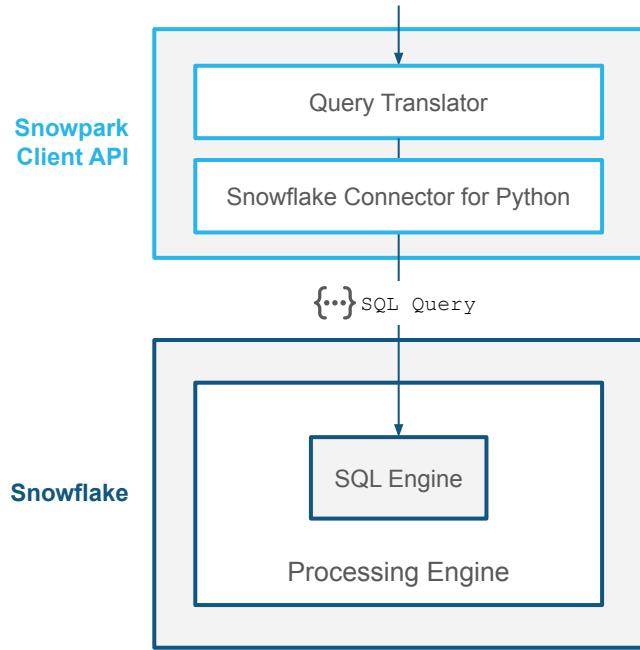
Process with secure sandbox integrated into Snowflake processing engine

All of this with no additional charges beyond warehouse usage



DataFrame API Query

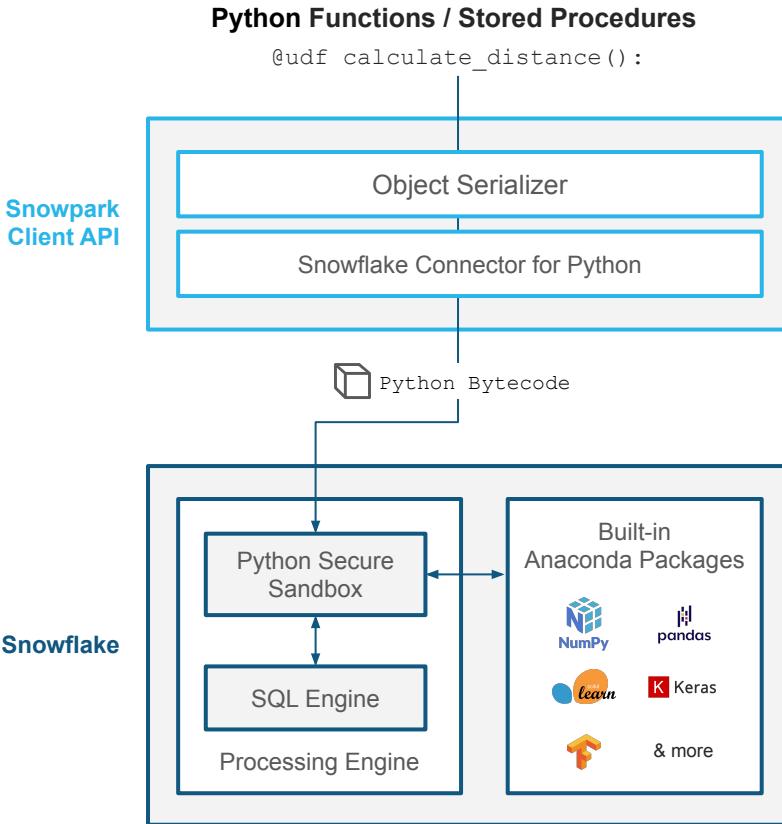
```
df.filter(df.state == 'WA').avg(df.amount)
```



DataFrame API

- Query Snowflake data with familiar Python DataFrame API
- 100% push-down to Snowflake
- Native Snowflake performance and scale



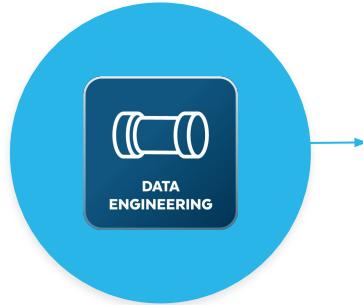


Server-Side Runtime

- Bring custom Python code to Snowflake as UDFs / UDTFs / Stored Procs
- Code is serialized and pushed down to run in a secure sandboxed environment
- Seamlessly access third-party packages with Anaconda integration



Snowpark Use Cases



ETL/ELT
Ingestion and transformation to bring raw data into modeled format

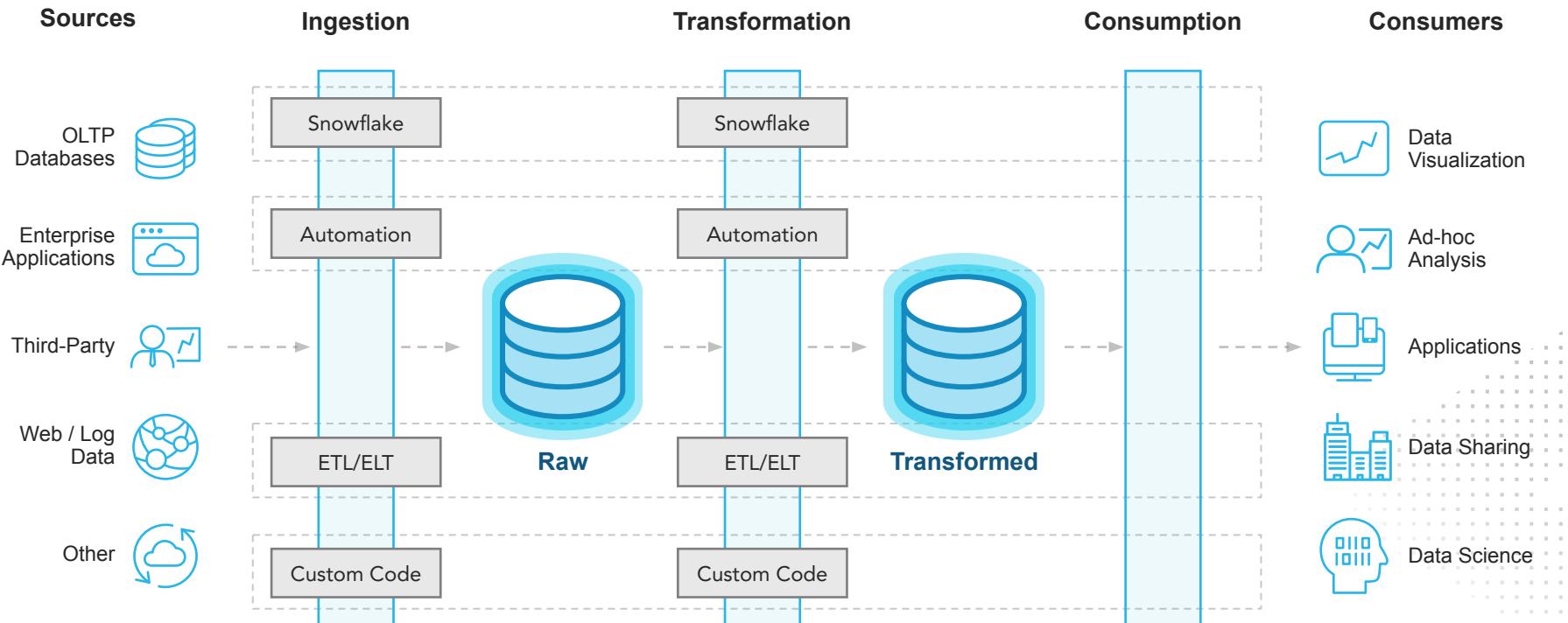
Custom Logic
Custom code for industry specific transformations, quality checks or automation rules

Data Science & Machine Learning Pipelines
Pipelines feeding ML models that require custom transformations and commonly use third-party OS libraries

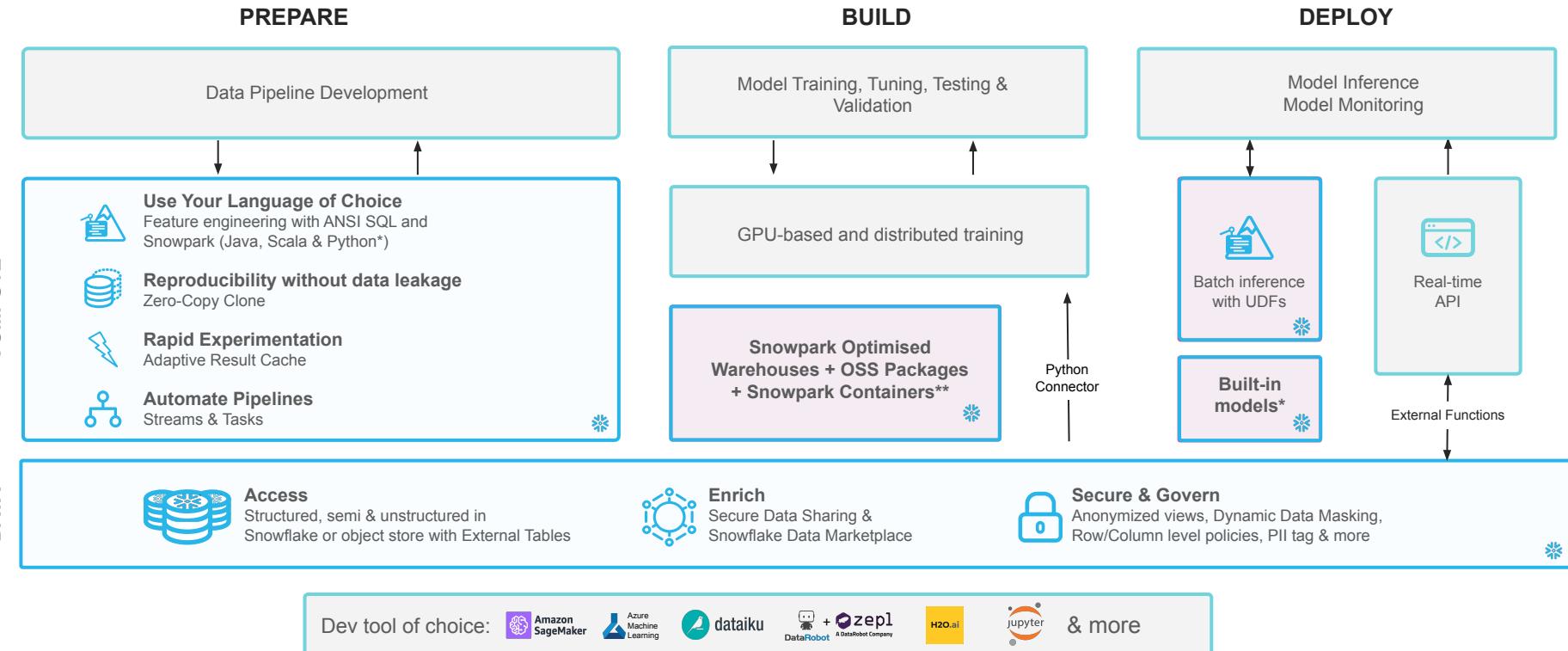


Data Engineering Implementation

A deeper look at the types of tools used during each phase



Machine Learning with Snowflake



Demo



Q&A

