As the title suggests, this is a demo heavy session.
The slides are more for the talktrack, which will be pretty minimal.
Will potentially have the intro slides and then will jump into demos.

DATA+AI SUMMIT
BY databricks

DATABRICKS WORKFLOWS: PRACTICAL HOW-TOS AND DEMOS

Author Name
Date

Product safe harbor statement

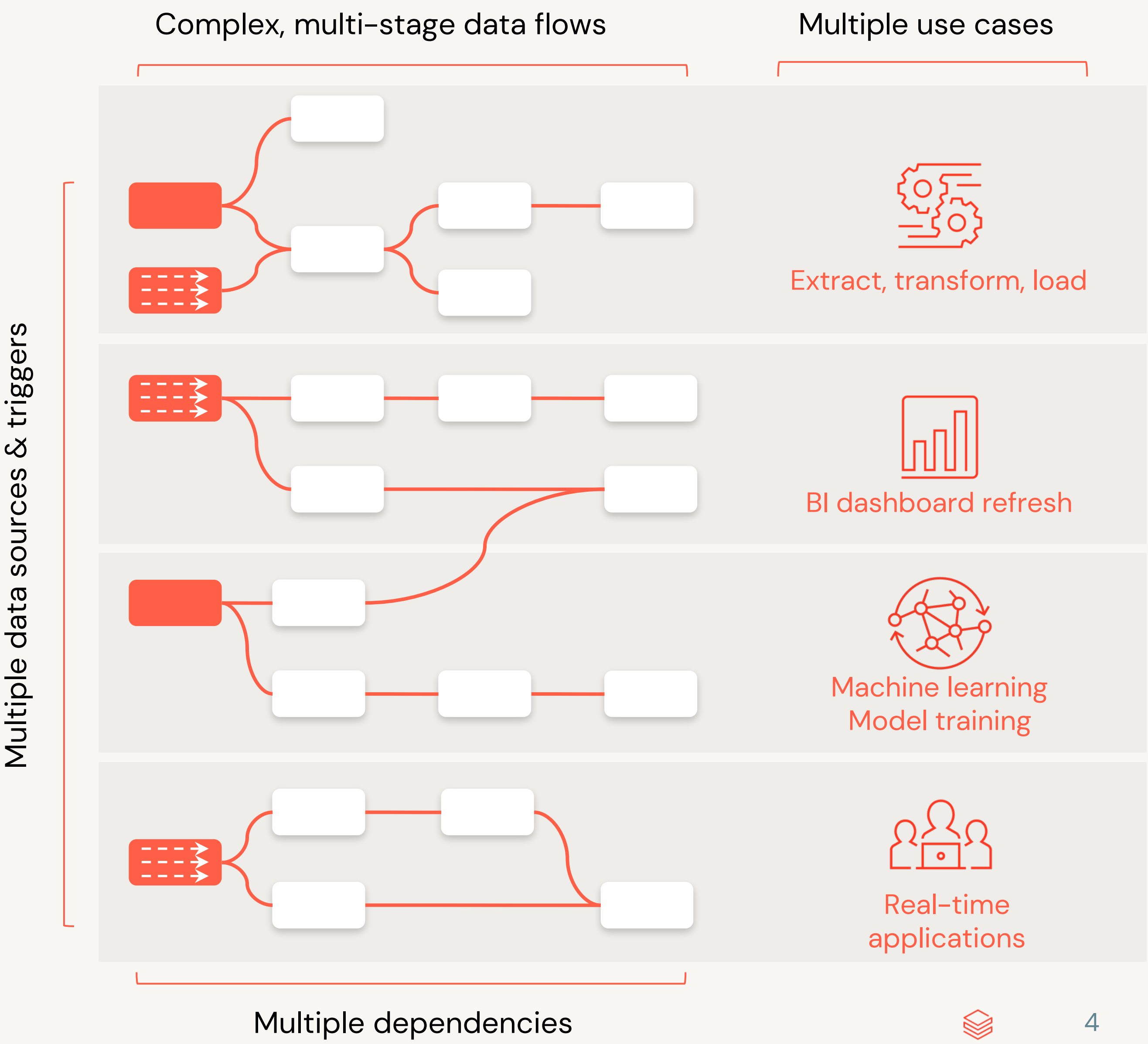
This information is provided to outline Databricks’ general product direction and is for **informational purposes only**. Customers who purchase Databricks services should make their purchase decisions relying solely upon services, features, and functions that are currently available. Unreleased features or functionality described in forward-looking statements are subject to change at Databricks discretion and may not be delivered as planned or at all

Modern data engineering requires modern data orchestration

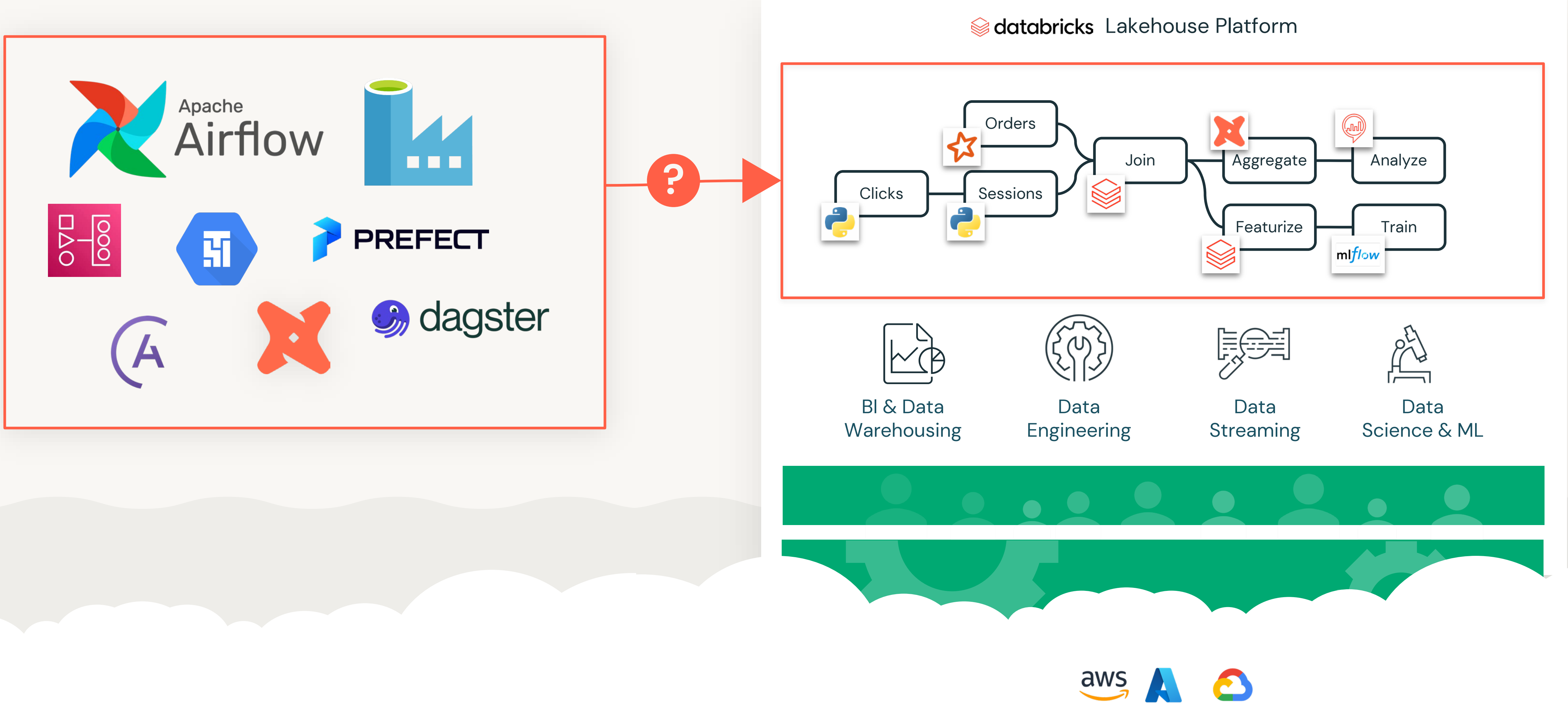
Orchestrating processes across all data, analytics and AI use cases is business critical

"Data pipelines are growing in size, volume, and complexity, with multistage processing and dependencies between various data assets."*

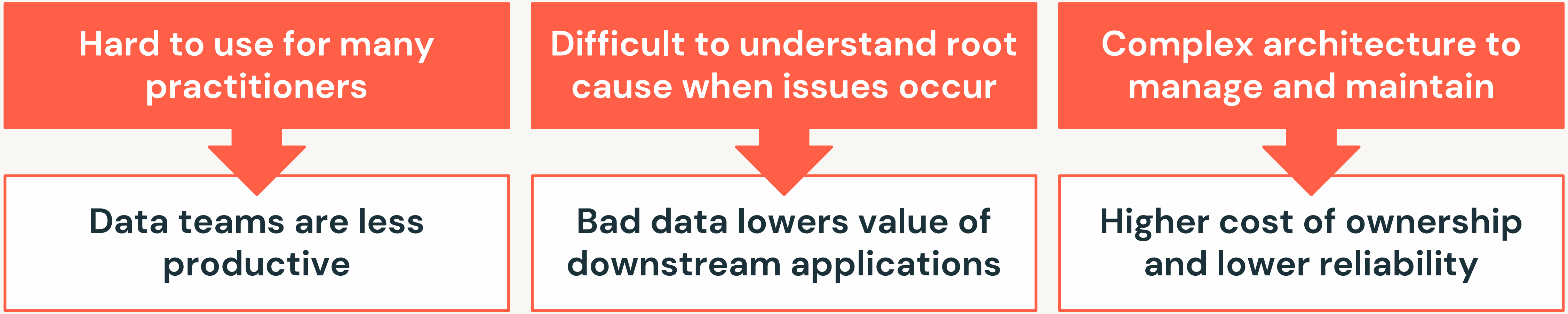
**Gartner Data Engineering Essentials, Patterns and Best Practices, September 2022*



There are many ways to orchestrate your Lakehouse workloads



External orchestrators create challenges



Hard to use for many practitioners

Data teams are less productive

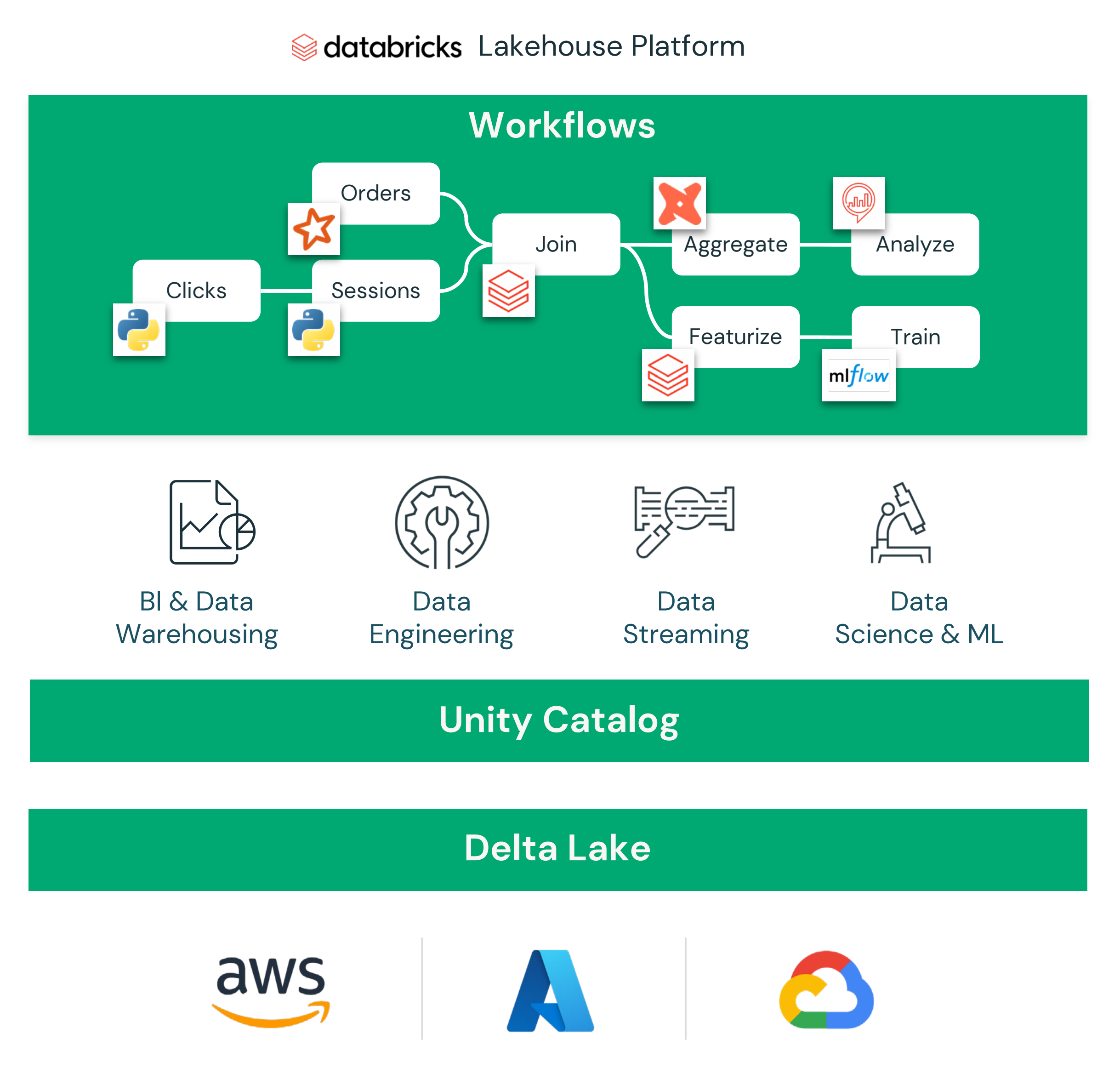
Difficult to understand root cause when issues occur

Bad data lowers value of downstream applications

Complex architecture to manage and maintain

Higher cost of ownership and lower reliability

These tools are not unified with your Lakehouse



databricks Lakehouse Platform

Workflows

Orders

Clicks

Sessions

Join

Aggregate

Analyze

Featurize

Train

mlflow

BI & Data Warehousing

Data Engineering

Data Streaming

Data Science & ML

Unity Catalog

Delta Lake

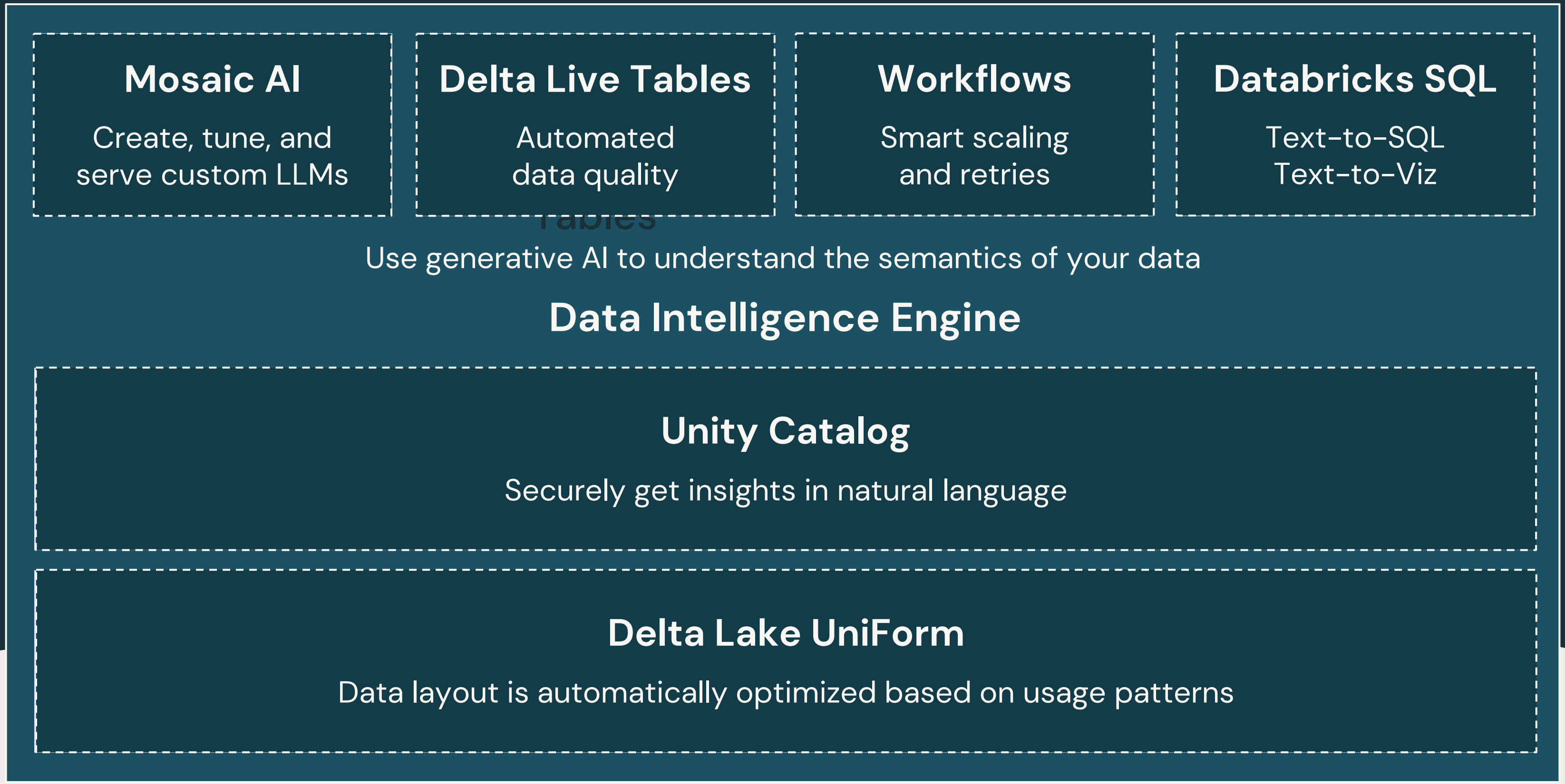
aws

Databricks Workflows

Unified orchestration for data, analytics, and AI on the Lakehouse Platform

- Simple authoring
- Actionable insights
- Proven reliability

Databricks Data Intelligence Platform



Mosaic AI

Create, tune, and
serve custom LLMs

Delta Live Tables

Automated
data quality

Workflows

Smart scaling
and retries

Databricks SQL

Text-to-SQL
Text-to-Viz

Tables

Use generative AI to understand the semantics of your data

Data Intelligence Engine

Unity Catalog

Securely get insights in natural language

Delta Lake UniForm

Data layout is automatically optimized based on usage patterns

Open Data Lake

All Raw Data
(Logs, Texts, Audio, Video, Images)

Top 3 reasons why customers love Databricks Workflows

Simple authoring

for all data practitioners

Any data practitioner accelerate development by easily orchestrating workflows from inside their Databricks workspace in just a few clicks. Advanced users can use their favorite IDE's with full support for CI/CD.

Actionable insights

from real-time monitoring

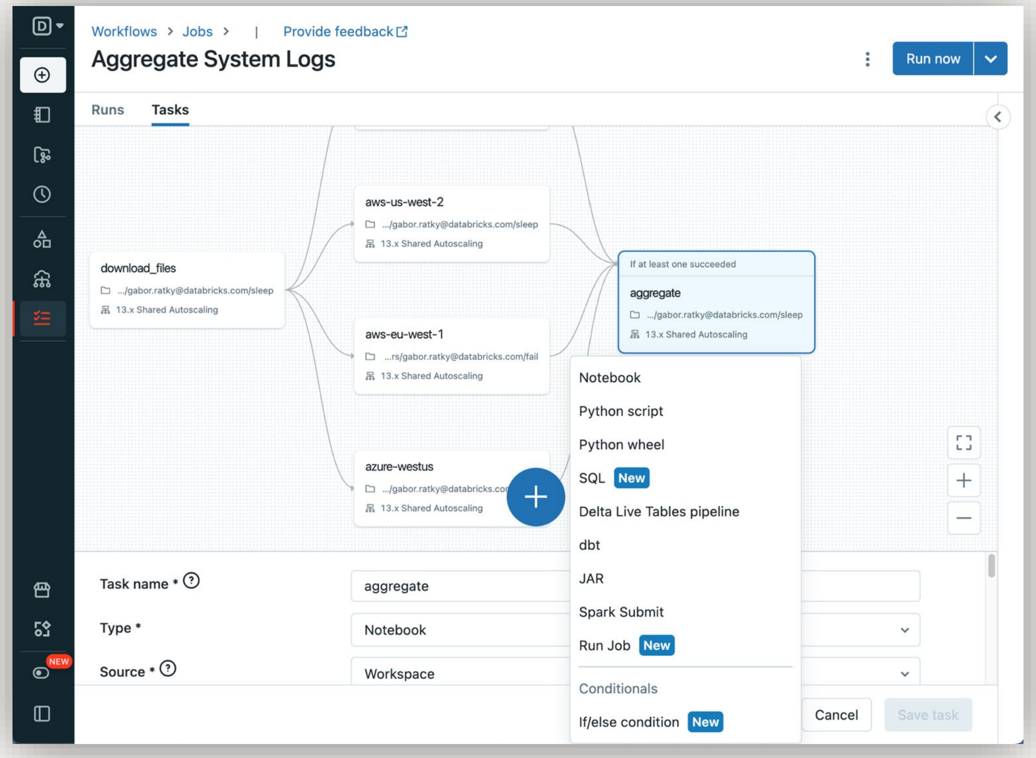
Full visibility into every task in every workflow. See the health of all your production workloads in real-time with detailed metrics and analytics to identify, troubleshoot, and fix issues fast.

Proven reliability

for production workloads

A fully managed orchestration service with serverless data processing and a history of 99.95% uptime. Workflows is trusted by thousands of Databricks customers running millions of production workloads.

Simple authoring for all data practitioners

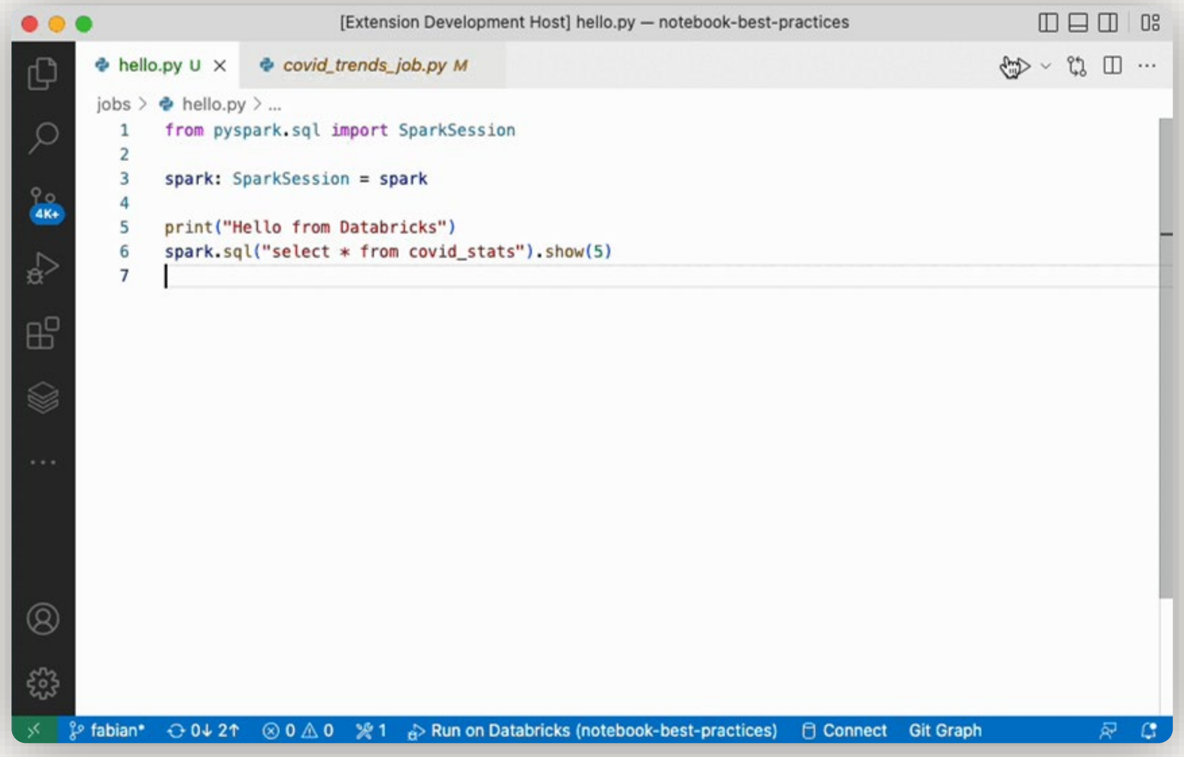


Task name * aggregate

Type * Notebook

Source * Workspace

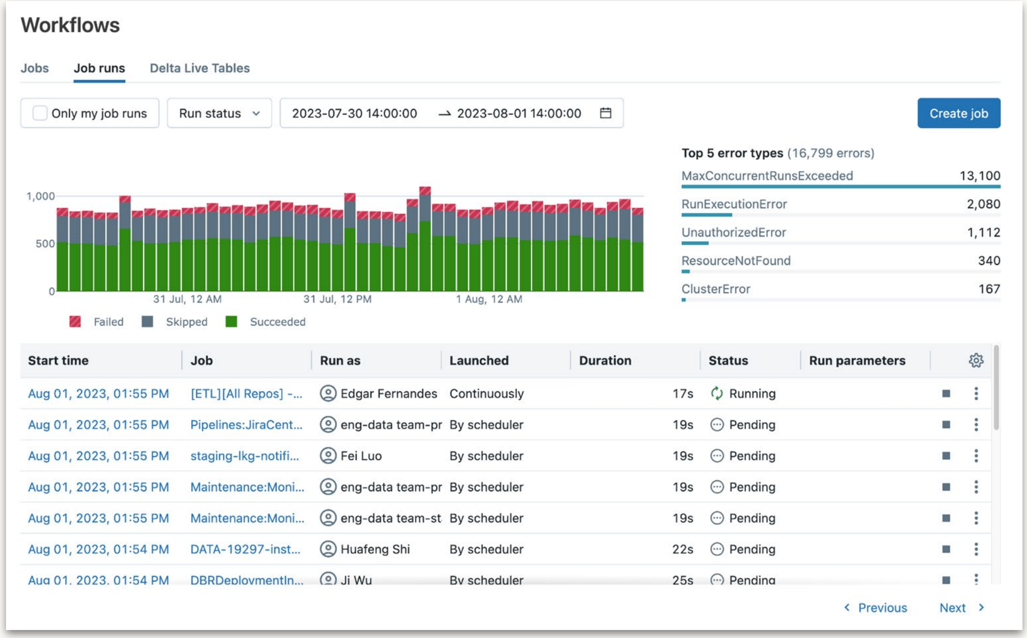
Build sophisticated workflows inside your Databricks workspace with a few clicks


[Extension Development Host] hello.py — notebook-best-practices
hello.py U
covid_trends_job.py M
jobs > hello.py > ...
4K+
fabian*
0↓ 2↑
0 ⚠ 0
1
Run on Databricks (notebook-best-practices)
Connect
Git Graph

```
from pyspark.sql import SparkSession

spark: SparkSession = spark

print("Hello from Databricks")
spark.sql("select * from covid_stats").show(5)

```

Or connect your favorite IDE to develop workflows locally and run them on Databricks

Actionable insights from real-time monitoring



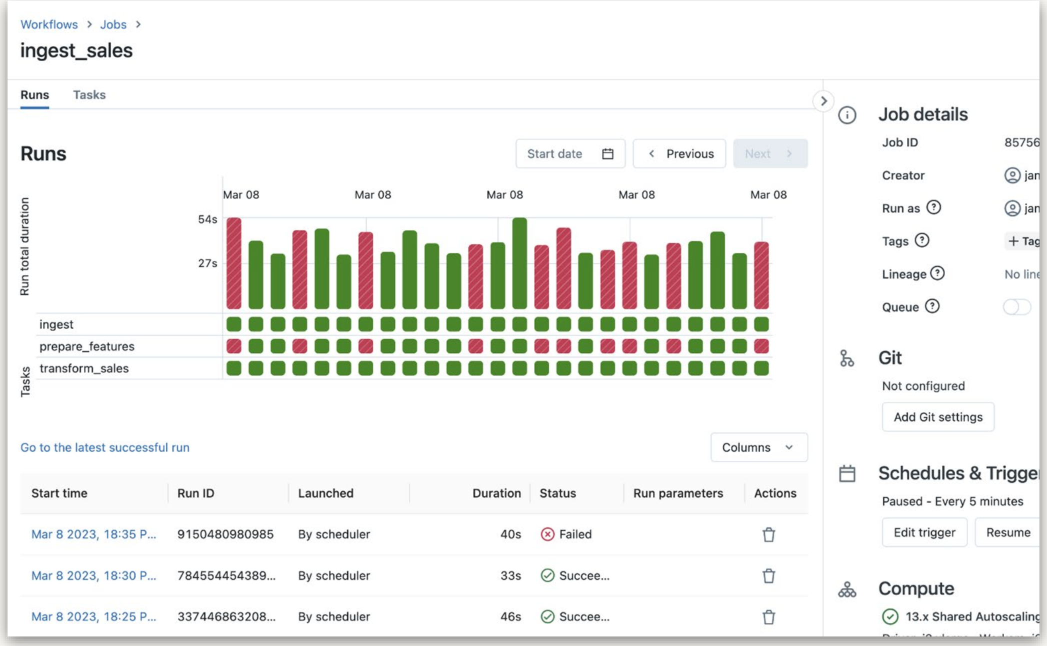
□ Only my job runs | Run status | 2023-07-30 14:00:00 → 2023-08-01 14:00:00

Top 5 error types (16,799 errors)

MaxConcurrentRunsExceeded	13,100
RunExecutionError	2,080
UnauthorizedError	1,112
ResourceNotFound	340
ClusterError	167

Start time	Job	Run as	Launched	Duration	Status	Run parameters
Aug 01, 2023, 01:55 PM	[ETL][All Repos] -...	Edgar Fernandes	Continuously	17s	Running	
Aug 01, 2023, 01:55 PM	Pipelines:JiraCent...	eng-data team-pr	By scheduler	19s	Pending	
Aug 01, 2023, 01:55 PM	staging-lkg-notifi...	Fei Luo	By scheduler	19s	Pending	
Aug 01, 2023, 01:55 PM	Maintenance:Moni...	eng-data team-pr	By scheduler	19s	Pending	
Aug 01, 2023, 01:55 PM	Maintenance:Moni...	eng-data team-st	By scheduler	19s	Pending	
Aug 01, 2023, 01:54 PM	DATA-19297-inst...	Huafeng Shi	By scheduler	22s	Pending	
Aug 01, 2023, 01:54 PM	DBRDeploymentIn...	Ji Wu	By scheduler	25s	Pending	

A simple and intuitive monitoring UI provides real-time metrics and detailed analytics for every workflow run



Job details

Job ID	85756
Creator	jan
Run as	jan
Tags	+ Tag
Lineage	No line
Queue	

Git

Not configured

Schedules & Trigge

Paused - Every 5 minutes

Compute

13.x Shared Autoscaling

Start time	Run ID	Launched	Duration	Status	Run parameters	Actions
Mar 8 2023, 18:35 P...	9150480980985	By scheduler	40s	Failed		
Mar 8 2023, 18:30 P...	784554454389...	By scheduler	33s	Succee...		
Mar 8 2023, 18:25 P...	337446863208...	By scheduler	46s	Succee...		

Drill down to understand which tasks are failing and why. Troubleshoot issues before your customers are impacted

Proven reliability for production workloads

99.95

%uptime

Trusted by thousands of customers running millions of production workloads

Fully managed service

Reduce maintenance costs and let your teams focus on innovation instead of resource management tasks

[In Preview]

Serverless data processing

Massively scalable compute resources for task execution remove even more of the maintenance burden from your teams and further reduces costs

Building Blocks of Databricks Workflows

A unit of orchestration in Databricks Workflows is called a **Job.**

Jobs consist of one or more **Tasks**

- Databricks Notebooks
- Python Scripts
- Python Wheels
- SQL Files/Queries
- DBSQL Dashboards
- Delta Live Tables Pipeline
- dbt
- Java JAR file
- Spark Submit

Control flows can be established between **Tasks.**

- Sequential
- Parallel
- Conditionals (Run If)
- Jobs-as-a-Task (Modular)
- For-Each Loop

Jobs supports different **Triggers**

- Manual Trigger
- Scheduled (Cron)
- API Trigger
- File Arrival Triggers
- Coming Soon Table Triggers
- Continuous (Streaming)

More on Operations and Cost Efficiencies

Resource Isolation

Dedicated, tailored Job clusters enables each task to run independently **without resource sharing bottlenecks!**

Job Cluster Re-Use

Cluster re–use enables users to run tasks in a Databricks Job on the same cluster for more **efficient cluster utilization and decreased job latency!**

Late Running Jobs

Tasks can **trigger emails** to stakeholders enabling you to **proactively monitor and take action against late or long running jobs!**

Repair and Re-Run

You can configure **when and how many times failed runs are retried!**

Serverless Compute

SIMPLE and FAST

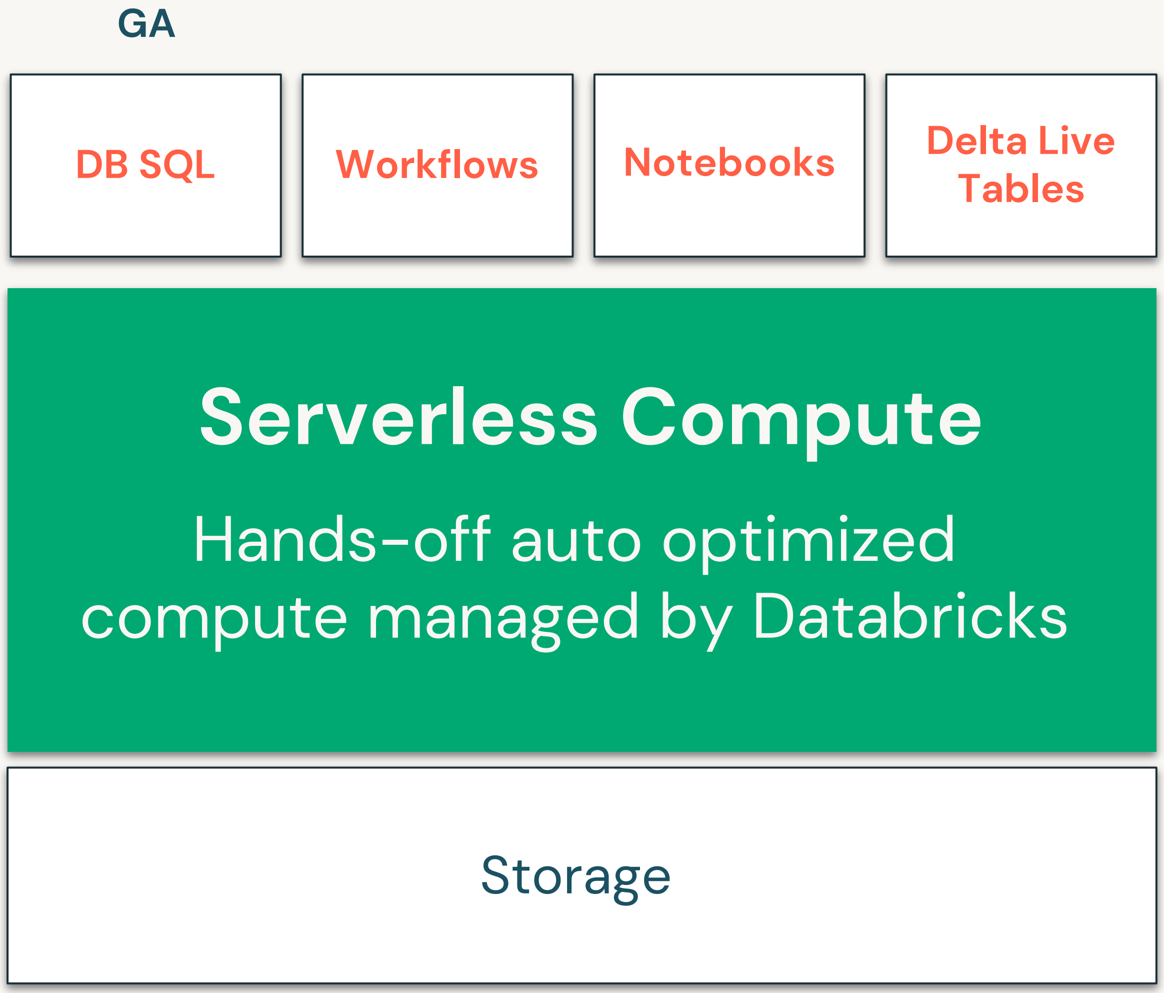
No knobs
Fast startup
For any practitioner

EFFICIENT

Fully managed and versionless
Paying only what you use
Strong cost governance

RELIABLE

Secure by default
Stable with smart fail-overs



GA

DB SQL

Workflows

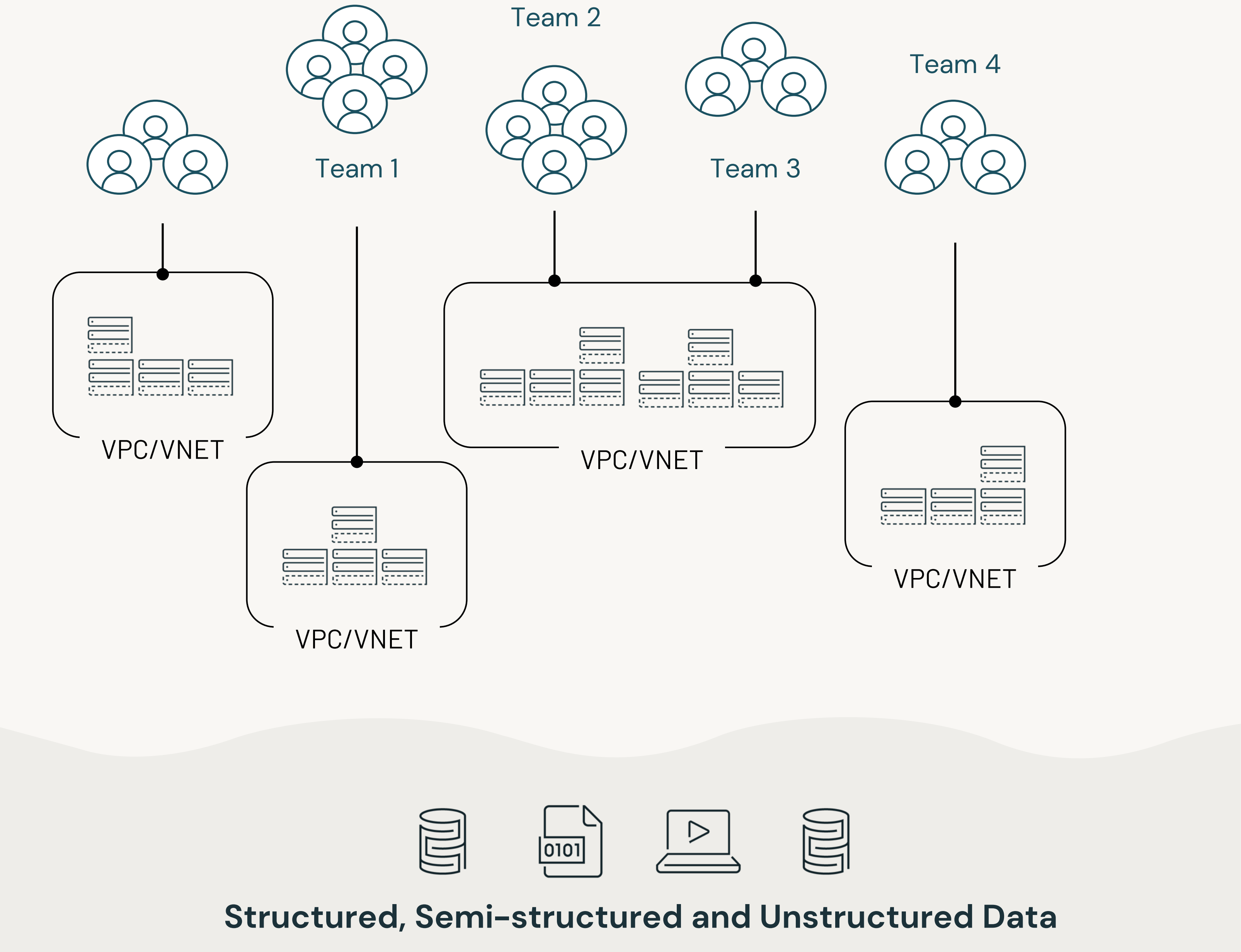
Notebooks

Delta Live Tables

Serverless Compute

Hands-off auto optimized compute managed by Databricks

Storage



STOP spending time on…

Setting up networks

Create and configure VNets
Set up gateways and firewall rules
Setup and manage private endpoints
X-tenant identities
IP address / subnet management

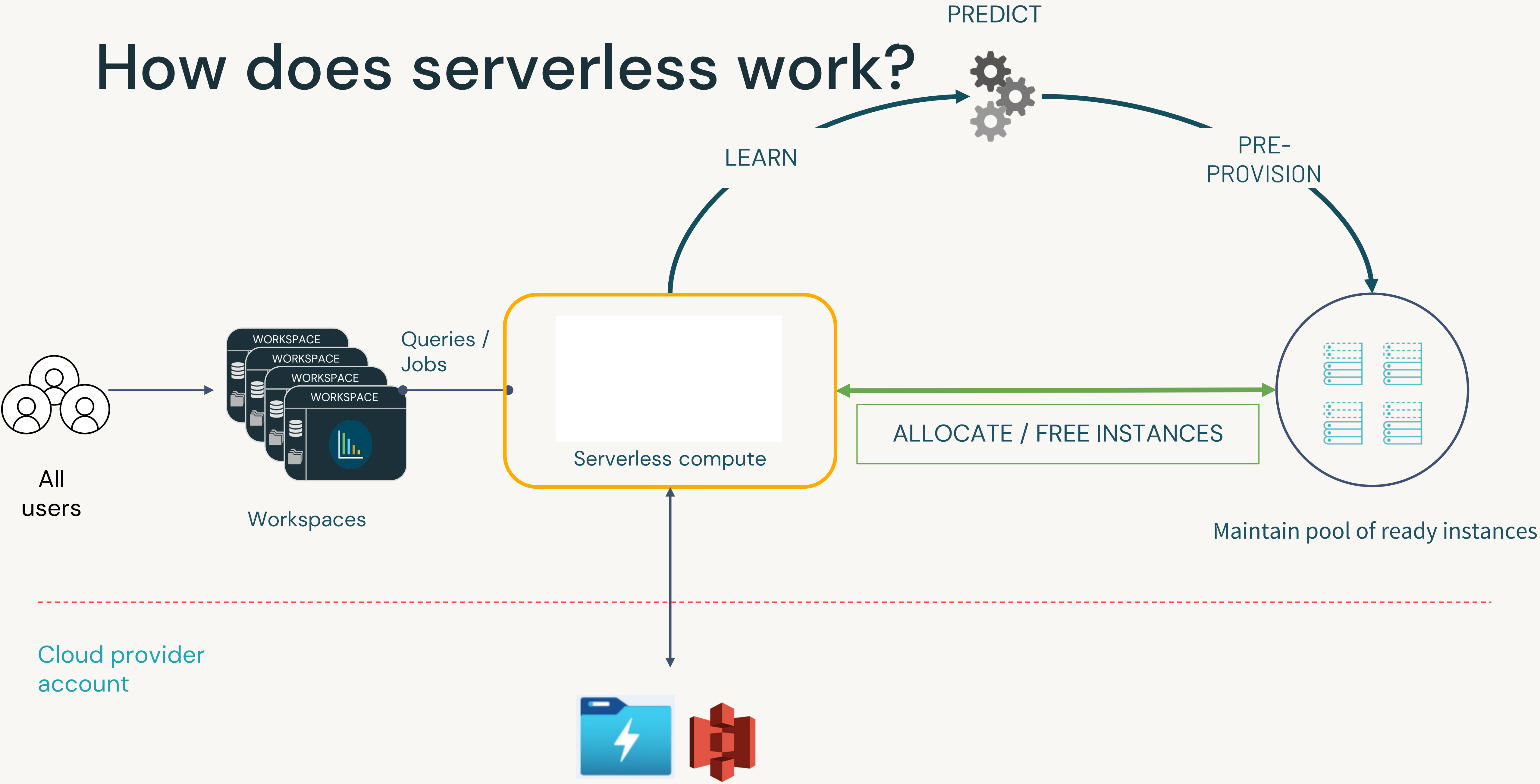
Security and Compliance

Vulnerability management
Encryption and key management
Intrusion detection and monitoring Data exfiltration protection

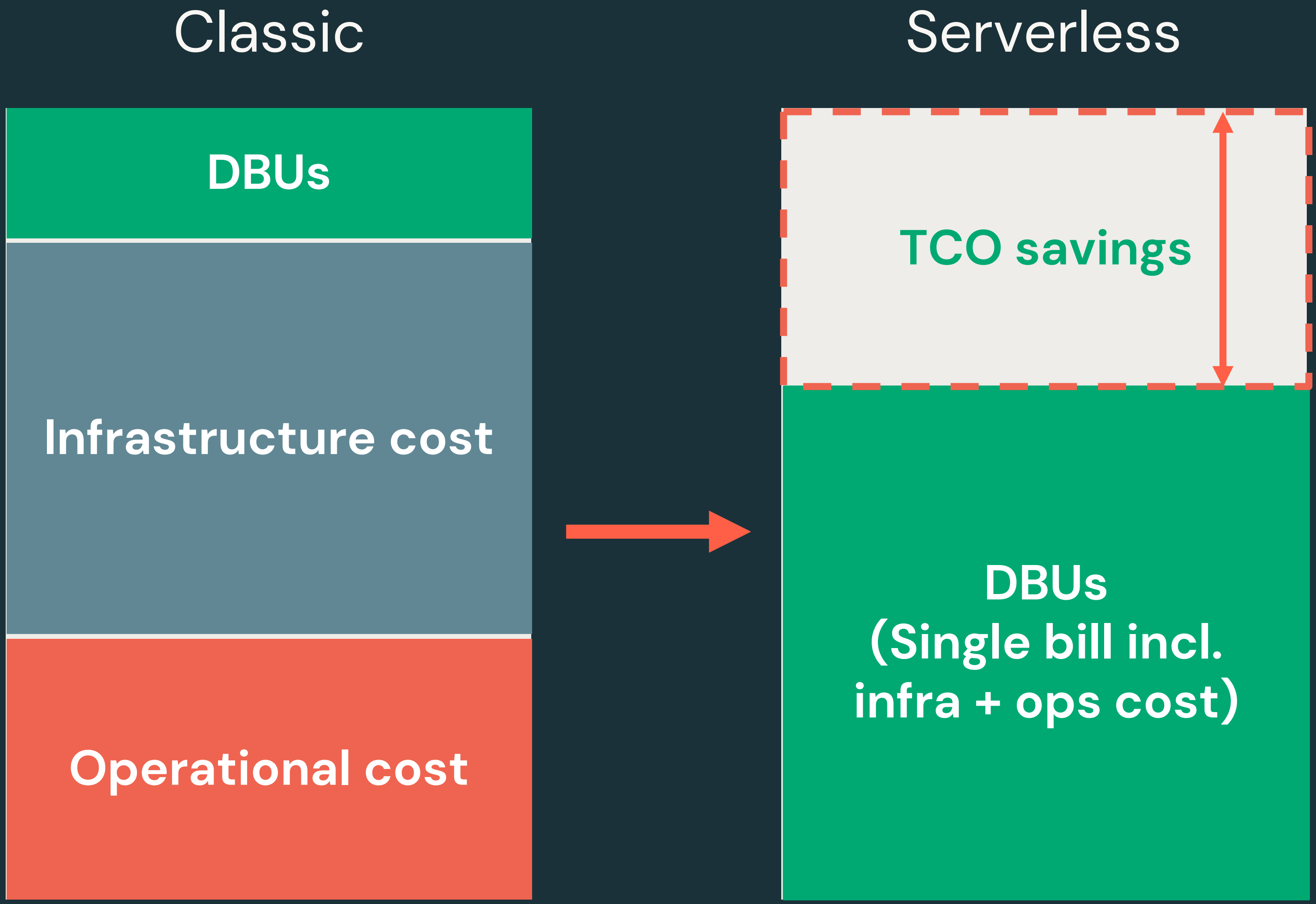
Managing efficiency

Capacity projections and reservations
Right sizing instances for workloads
Maintaining high utilization
Managing instance pools
Vacuum / compaction of Delta tables

How does serverless work?



Serverless reduces TCO

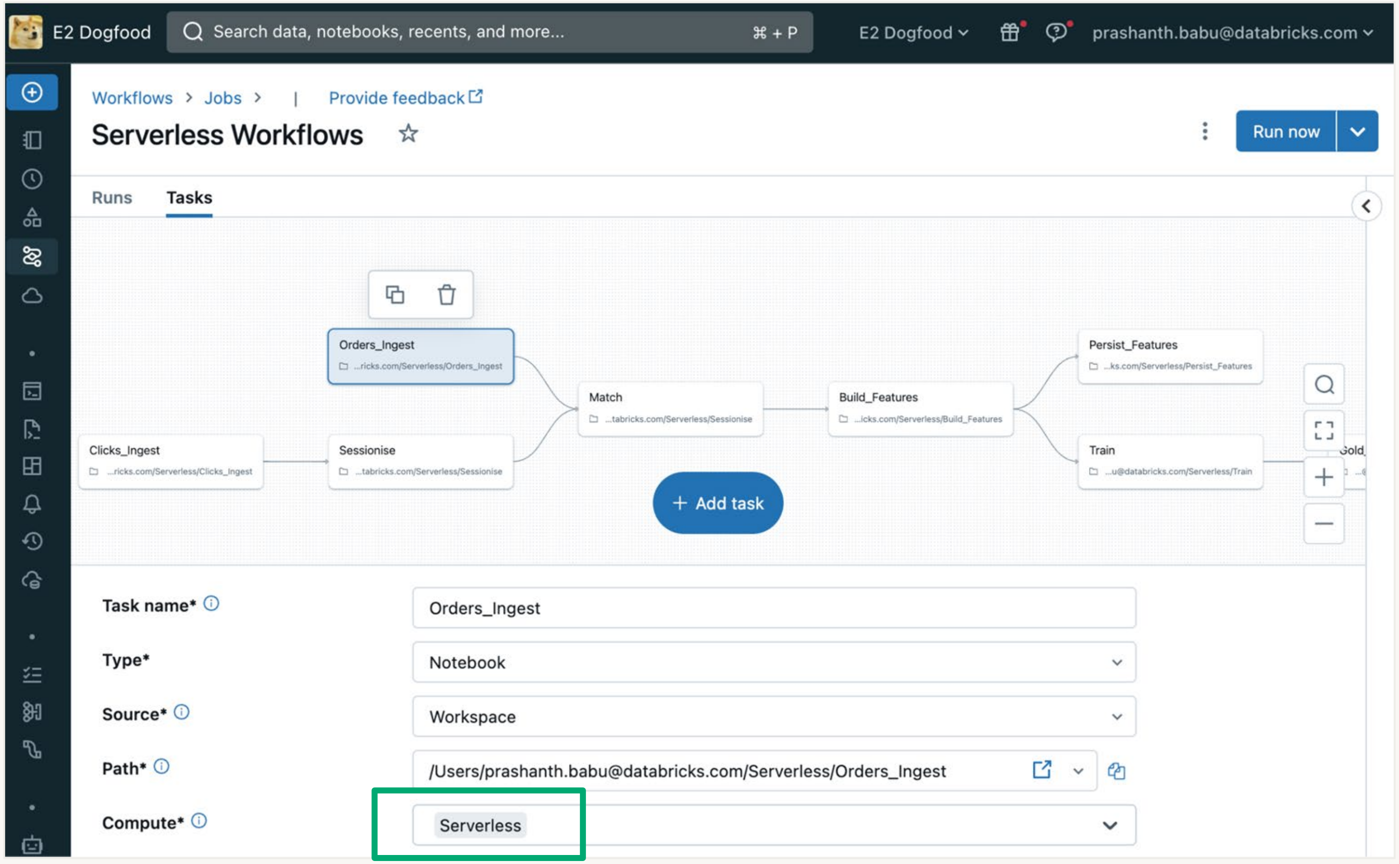


Serverless Compute for Workflows

Hands-off, auto-optimizing compute

Fully managed and reliable

- <60s startup (at GA)
- Versionless with auto-update
- System tables for cost observability
- Budgets for cost control (coming)
- Automatic instance type failover (coming)



E2 Dogfood
Search data, notebooks, recents, and more...
⌘ + P
E2 Dogfood
prashanth.babu@databricks.com
Workflows > Jobs >
Provide feedback
Run now
Runs
Tasks
Orders_Ingest
...ricks.com/Serverless/Orders_Ingest
Clicks_Ingest
...ricks.com/Serverless/Clicks_Ingest
Sessionise
...tabricks.com/Serverless/Sessionise
Match
...tabricks.com/Serverless/Sessionise
Build_Features
...icks.com/Serverless/Build_Features
Persist_Features
...ks.com/Serverless/Persist_Features
Train
...u@databricks.com/Serverless/Train
Gold
+ Add task

Serverless Workflows

Task name*: Orders_Ingest

Type*: Notebook

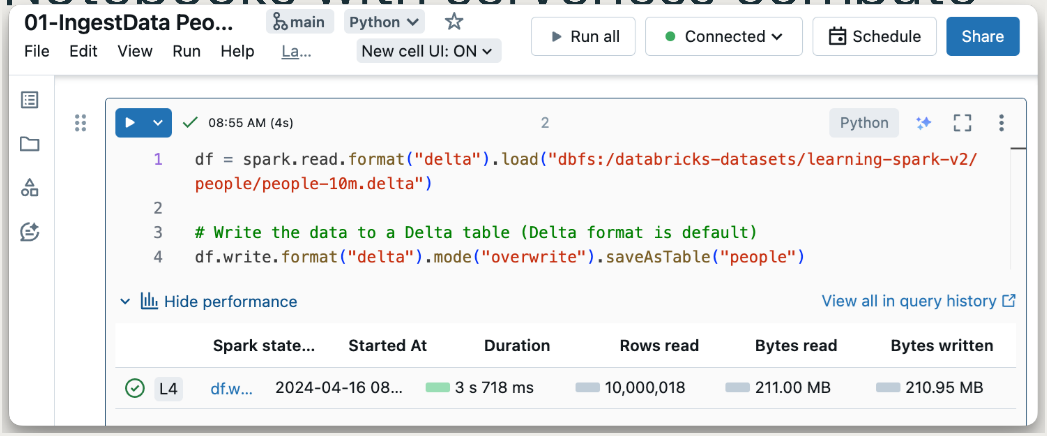
Source*: Workspace

Path*: /Users/prashanth.babu@databricks.com/Serverless/Orders_Ingest

Compute*: Serverless

Secure, multi-user, serverless Spark

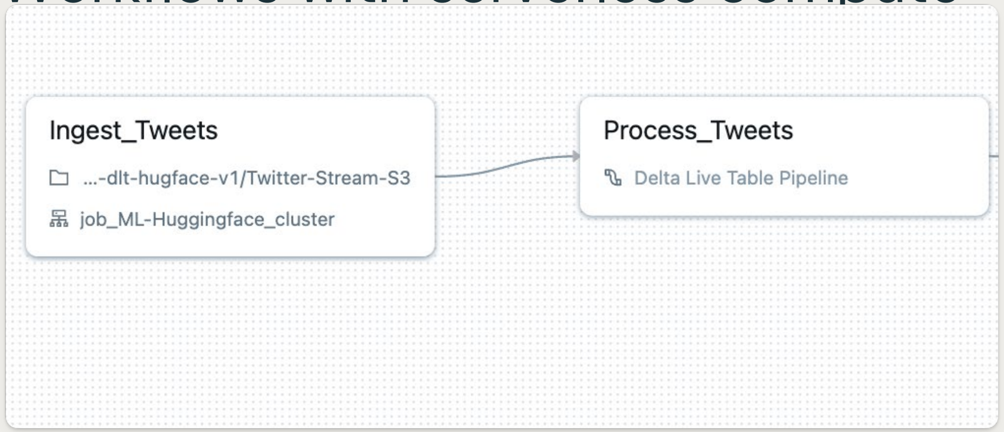
Notebooks with serverless compute



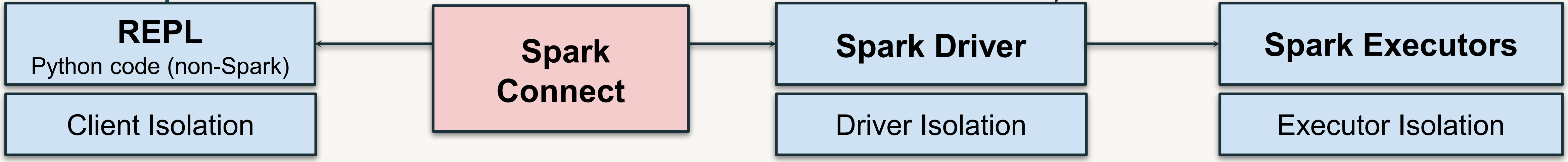
```
df = spark.read.format("delta").load("dbfs:/databricks-datasets/learning-spark-v2/
people/people-10m.delta")

# Write the data to a Delta table (Delta format is default)
df.write.format("delta").mode("overwrite").saveAsTable("people")
```

		Spark state...	Started At	Duration	Rows read	Bytes read	Bytes written
✓	L4	df.w...	2024-04-16 08...	3 s 718 ms	10,000,018	211.00 MB	210.95 MB

Workflows with serverless compute


Ingest_Tweets
...-dlt-hugface-v1/Twitter-Stream-S3
job_ML-Huggingface_cluster
Process_Tweets
Delta Live Table Pipeline

Built on Spark Connect: user code runs in full isolation on client, driver and executors



HowTos and Demos with emphasis on best practices

databricks