

eBook

The Data Team's Guide to the Databricks Data Intelligence Platform



Contents

CHAPTER 1	The data lakehouse	4
CHAPTER 2	The Databricks Data Intelligence Platform	11
CHAPTER 3	Data reliability and performance	17
CHAPTER 4	Unified governance and sharing for data, analytics and AI	27
CHAPTER 5	Security	40
CHAPTER 6	Instant compute and serverless	47
CHAPTER 7	Data warehousing	50
CHAPTER 8	Data engineering	54
CHAPTER 9	Data streaming	66
CHAPTER 10	Data science, AI and machine learning	71
CHAPTER 11	Databricks Technology Partners and the modern data stack	77
CHAPTER 12	Get started with the Databricks Data Intelligence Platform	79

INTRODUCTION

The Data Team's Guide to the Databricks Data Intelligence Platform

The Data Team's Guide to the Databricks Data Intelligence Platform is designed for data practitioners and leaders who are embarking on their journey into the data lakehouse architecture.

In this eBook, you will learn the full capabilities of the data lakehouse architecture and how the Databricks Data Intelligence Platform allows your entire organization to use data and AI.

You will see how it's built on a lakehouse to provide an open, unified foundation for all data and governance, and is powered by a Data Intelligence Engine that understands the uniqueness of your data. From ETL to data warehousing to generative AI, Databricks helps you simplify and accelerate your data and AI goals.

CHAPTER

01

The data
lakehouse

The evolution of data architectures

Data has moved front and center within every organization as data-driven insights have fueled innovation, competitive advantage and better customer experiences.

However, as companies place mandates on becoming more data-driven, their data teams are left in a sprint to deliver the right data for business insights and innovation. With the widespread adoption of cloud, data teams often invest in large-scale complex data systems that have capabilities for streaming, business intelligence, analytics and machine learning to support the overall business objectives.

To support these objectives, data teams have deployed cloud data warehouses and data lakes.

Traditional data systems: The data warehouse and data lake

With the advent of big data, companies began collecting large amounts of data from many different sources, such as weblogs, sensor data and images. Data warehouses — which have a long history as the foundation for decision support and business intelligence applications — cannot handle large volumes of data.

While data warehouses are great for structured data and historical analysis, they weren't designed for unstructured data, semi-structured data, and data with high variety, velocity and volume, making them unsuitable for many types of data.

This led to the introduction of data lakes, providing a single repository of raw data in a variety of formats. While suitable for storing big data, data lakes do not support transactions, nor do they enforce data quality, and their lack of consistency/isolation makes it almost impossible to read, write or process data.

For these reasons, many of the promises of data lakes never materialized and, in many cases, reduced the benefits of data warehouses.

As companies discovered new use cases for data exploration, predictive modeling and prescriptive analytics, the need for a single, flexible, high-performance system only grew. Data teams require systems for diverse data applications including SQL analytics, real-time analytics, data science and machine learning.

To solve for new use cases and new users, a common approach is to use multiple systems — a data lake, several data warehouses and other specialized systems such as streaming, time-series, graph and image databases. But having multiple systems introduces complexity and delay, as data teams invariably need to move or copy data between different systems, effectively losing oversight and governance over data usage.

You have now duplicated data in two different systems and the changes you make in one system are unlikely to find their way to the other. So you are going to have data drift almost immediately, not to mention paying to store the same data multiple times.

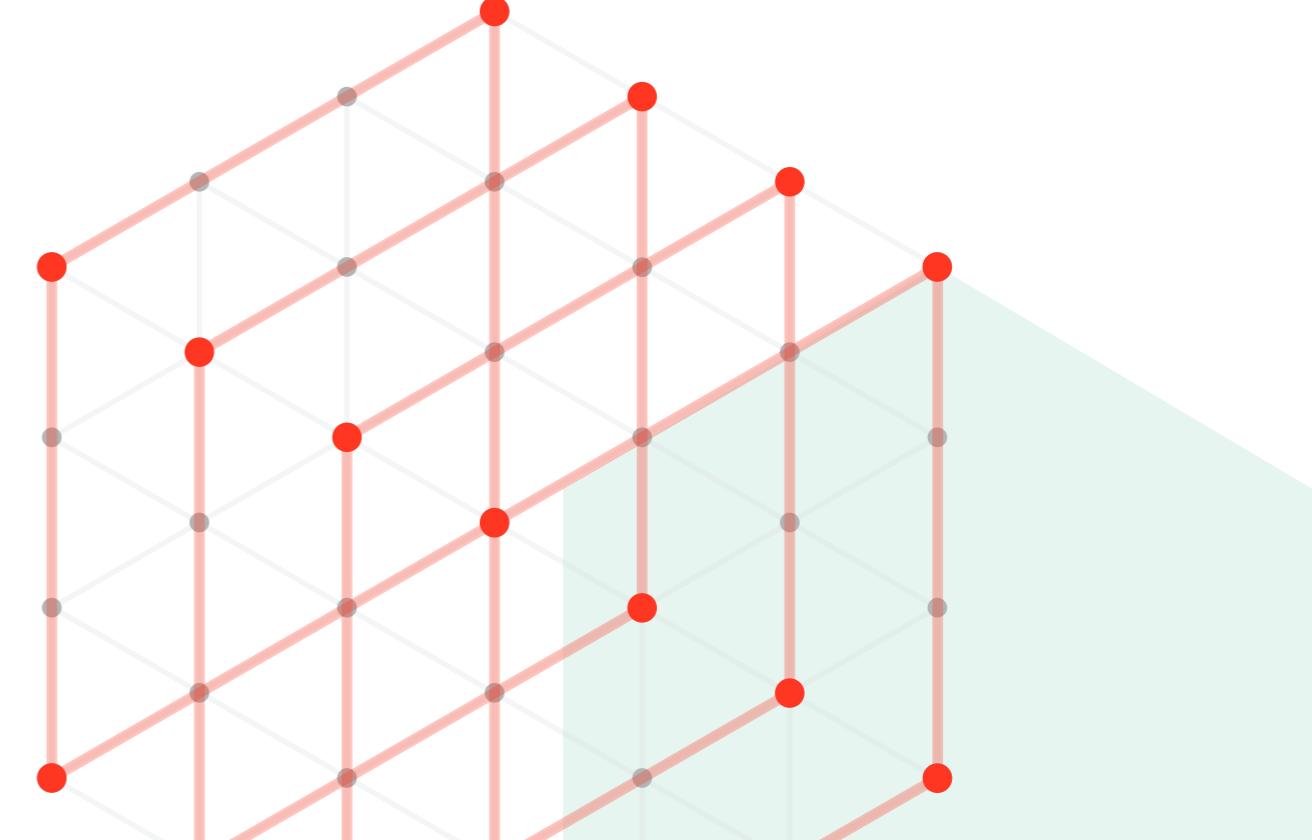
Then, because governance is happening at two distinct levels across these platforms, you are not able to control things consistently.

Challenges with data, analytics and AI

In a recent [Accenture](#) study, only 32% of companies reported tangible and measurable value from data. The challenge is that most companies continue to implement two different platforms: data warehouses for BI and data lakes for AI. These platforms are incompatible with each other, but data from both systems is generally needed to deliver game-changing outcomes, which makes success with AI extremely difficult.

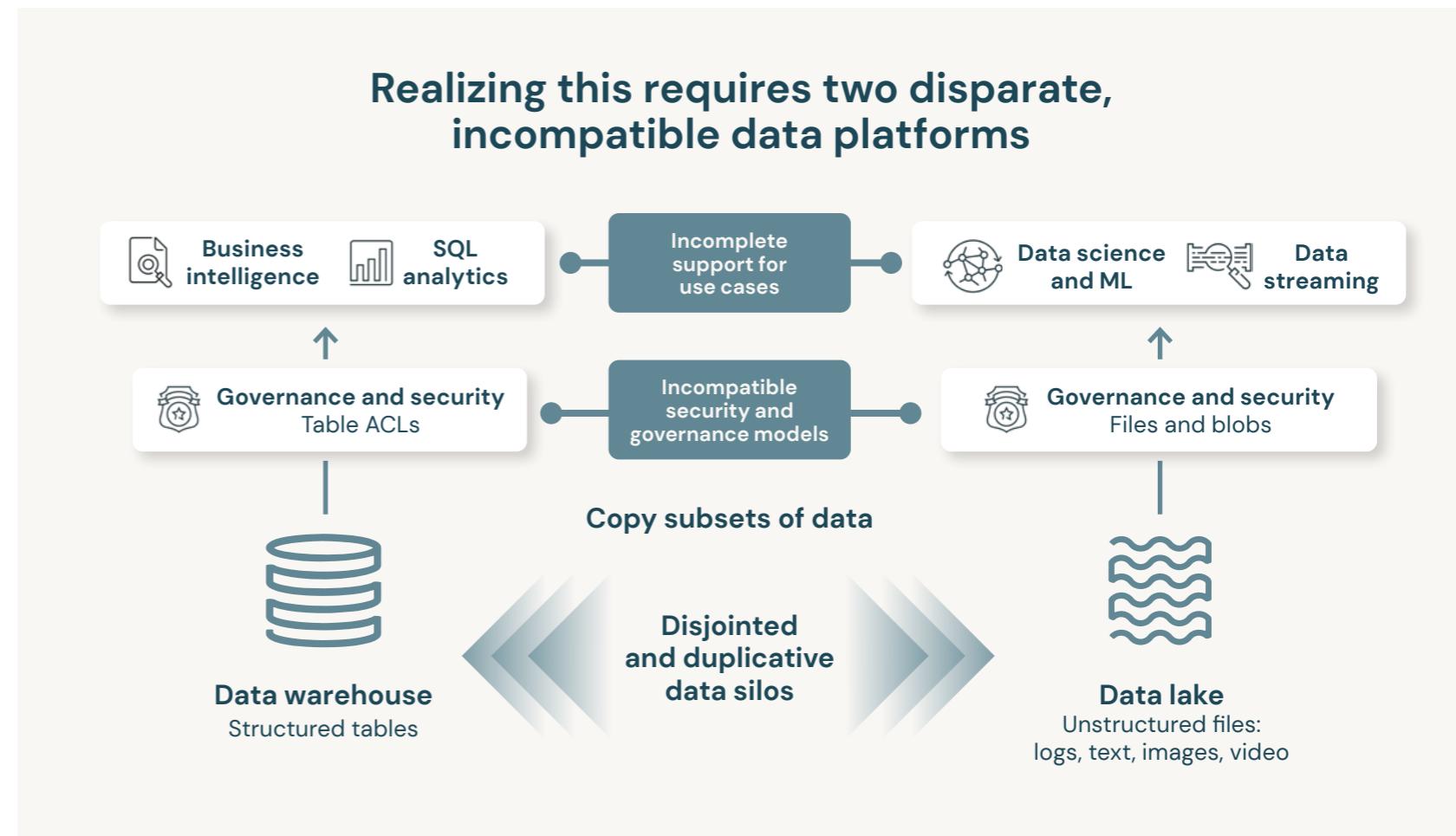
Today, most of the data is landing in the data lake, and a lot of it is unstructured. In fact, according to [IDC](#), about 80% of the data in any organization will be unstructured by 2025. But, this data is where much of the value from AI resides. Subsets of the data are then copied to the data warehouse into structured tables, and back again in some cases.

You also must secure and govern the data in both warehouses and offer fine-grained governance, while lakes tend to be coarser grained at the file level. Then, you stand up different stacks of tools on these platforms to do either BI or AI.



Finally, the tool stacks on top of these platforms are fundamentally different, which makes it difficult to get any kind of collaboration going between the teams that support them.

This is why AI efforts fail. There is a tremendous amount of complexity and rework being introduced into the system. Time and resources are being wasted trying to get the right data to the right people, and everything is happening too slowly to get in front of the competition.

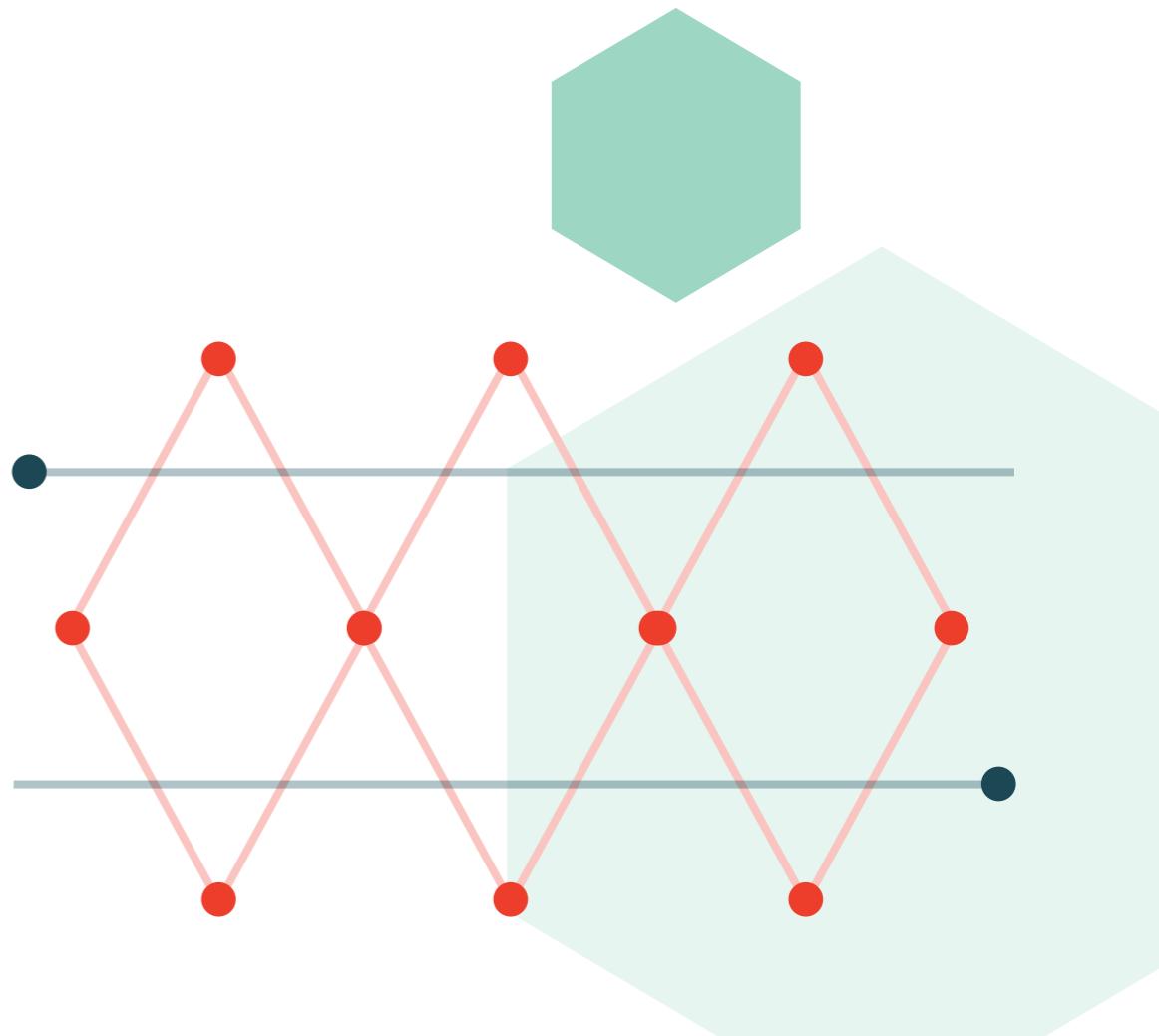


Lakehouse: A new generation of open platforms

Databricks is the inventor and pioneer of the data lakehouse architecture. We are continuously innovating on the lakehouse architecture to help customers deliver on their data, analytics and AI aspirations. The ideal data, analytics and AI platform needs to operate differently. Rather than copying and transforming data in multiple systems, you need one platform that accommodates all data types.

Ideally, the platform must be open, so that you are not locked into any walled gardens. You would also have one security and governance model. It would not only manage all data types, but it would also be cloud-agnostic to govern data wherever it is stored.

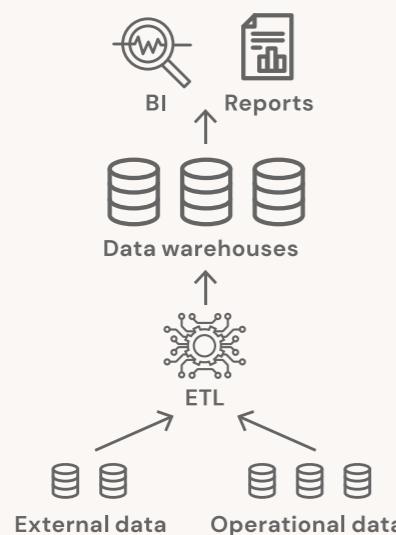
Last, it would support all major data, analytics and AI workloads, so that your teams can easily collaborate and get access to all the data they need to innovate. The lakehouse architecture is a prerequisite for data intelligence, because it unifies data and AI. Adding an integrated data intelligence engine that understands the semantics of your data and all the metadata to the lakehouse architecture is what creates a data intelligence platform.



Data lakehouse

One architecture to unify all your data, analytics and AI workloads

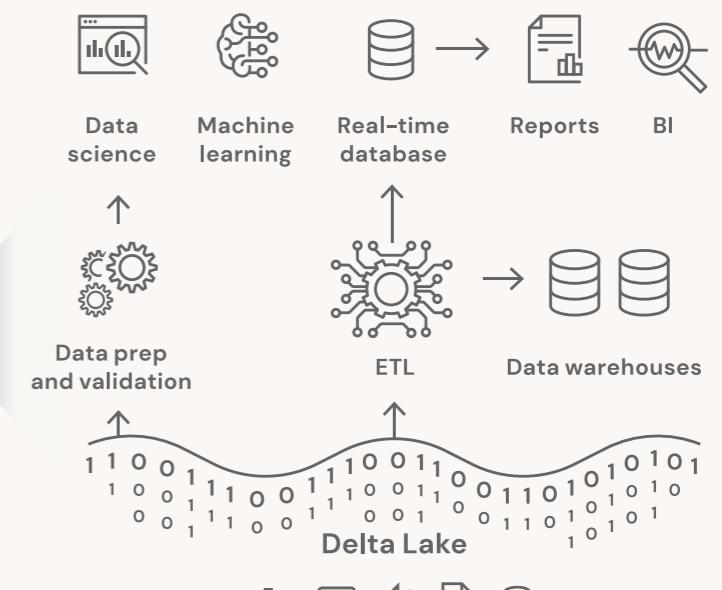
Data warehouse



Lakehouse Architecture

All machine learning, SQL, BI, and streaming use cases
One security and governance approach for all data assets on all clouds
An open and reliable data platform to efficiently handle all data types

Delta Lake



Structured, semi-structured and unstructured data

Key features for a lakehouse

Recent innovations with the data lakehouse architecture can help simplify your data and AI workloads, ease collaboration for data teams, and maintain the kind of flexibility and openness that allows your organization to stay agile as you scale. Here are key features to consider when evaluating data lakehouse architectures:

Transaction support: In an enterprise lakehouse, many data pipelines will often be reading and writing data concurrently. Support for ACID (Atomicity, Consistency, Isolation and Durability) transactions ensures consistency as multiple parties concurrently read or write data.

Schema enforcement and governance: The lakehouse should have a way to support schema enforcement and evolution, supporting data warehouse schema paradigms such as star/snowflake. The system should be able to reason about data integrity, and it should have robust governance and auditing mechanisms.

Data governance: Capabilities including auditing, retention and lineage have become essential, particularly considering recent privacy regulations. Tools that allow data discovery have become popular, such as data catalogs and data usage metrics.

BI support: Lakehouses allow the use of BI tools directly on the source data. This reduces staleness and latency, improves recency and lowers cost by not having to operationalize two copies of the data in both a data lake and a warehouse.

Storage decoupled from compute: In practice, this means storage and compute use separate clusters, thus these systems can scale to many more concurrent users and larger data sizes. Some modern data warehouses also have this property.

Openness: The storage formats, such as Apache Parquet, are open and standardized, so a variety of tools and engines, including machine learning and Python/R libraries, can efficiently access the data directly.

Support for diverse data types (unstructured and structured): The lakehouse can be used to store, refine, analyze and access data types needed for many new data applications, including images, video, audio, semi-structured data and text.

Support for diverse workloads: Use the same data repository for a range of workloads including data science, machine learning and SQL analytics. Multiple tools might be needed to support all these workloads.

End-to-end streaming: Real-time reports are the norm in many enterprises. Support for streaming eliminates the need for separate systems dedicated to serving real-time data applications.



Learn more

- [Lakehouse: A New Generation of Open Platforms That Unify Data Warehousing and Advanced Analytics](#)
- [Building the Data Lakehouse by Bill Inmon, Father of the Data Warehouse](#)
- [What Is a Data Lakehouse?](#)

CHAPTER

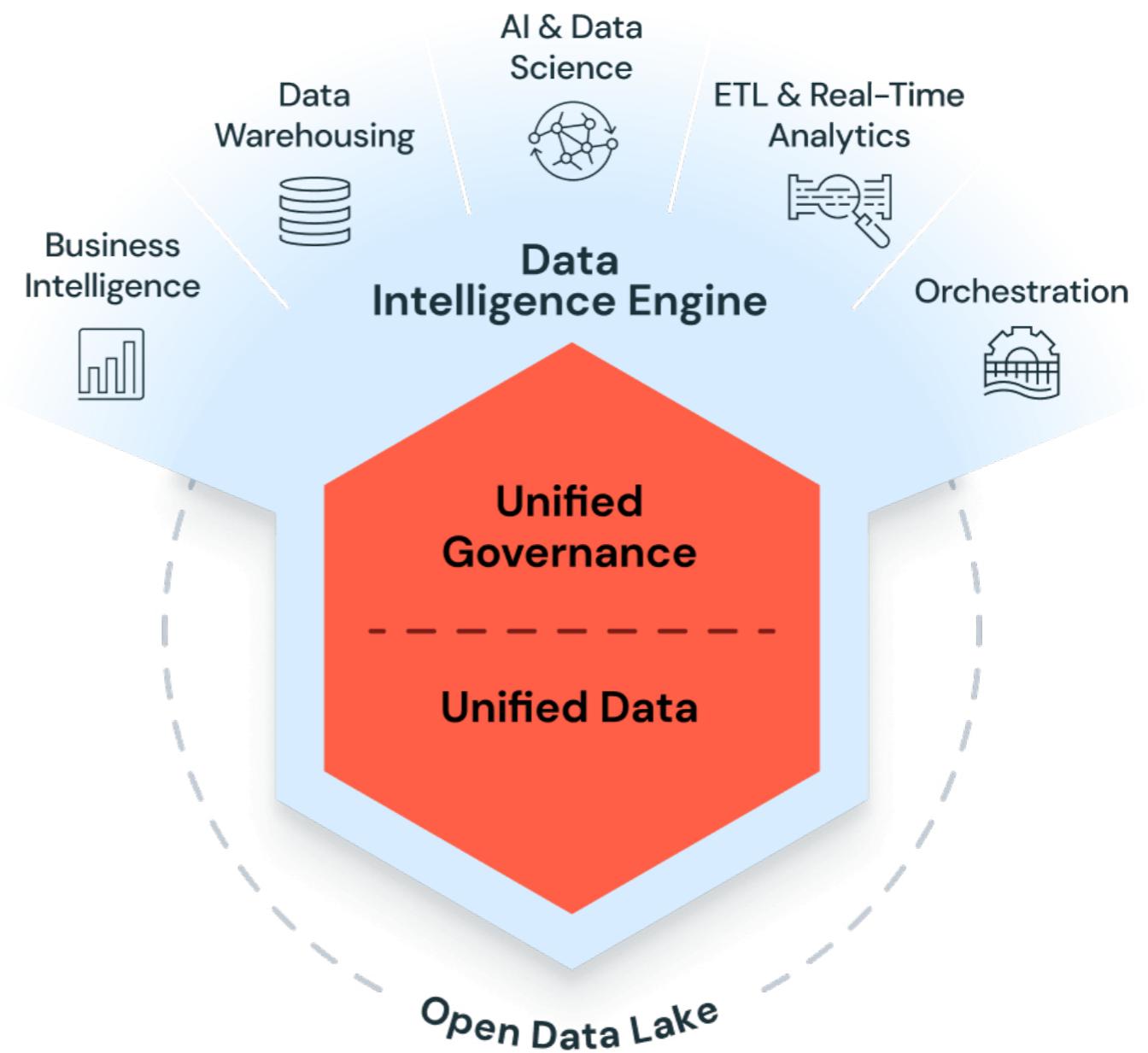
02

The Databricks Data Intelligence Platform

What is the Databricks Data Intelligence Platform

The Databricks Data Intelligence Platform allows your entire organization to use data and AI. It's built on a lakehouse to provide an open, unified foundation for all data and governance, and is powered by a Data Intelligence Engine that understands the uniqueness of your data.

The winners in every industry will be data and AI companies. From ETL to data warehousing to generative AI, Databricks helps you simplify and accelerate your data and AI goals.



Benefits of the Databricks Data Intelligence Platform

Intelligent

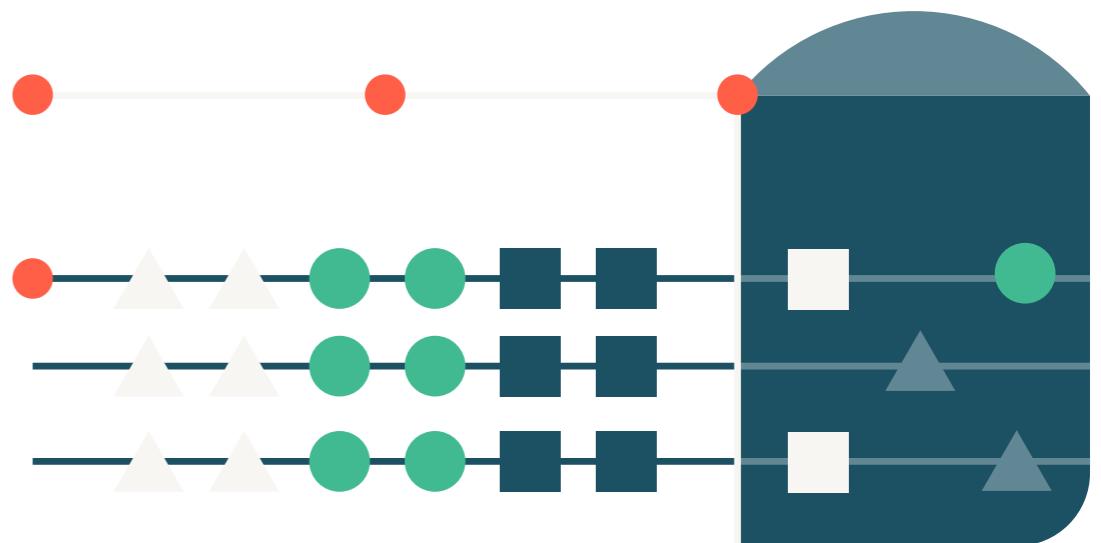
Databricks combines generative AI with the unification benefits of a lakehouse to power a Data Intelligence Engine that understands the unique semantics of your data. This allows the Databricks Platform to automatically optimize performance and manage infrastructure in ways unique to your business.

Simple

Natural language substantially simplifies the user experience on Databricks. The Data Intelligence Engine understands your organization's language, so search and discovery of new data is as easy as asking a question like you would to a coworker. Additionally, developing new data and applications is accelerated through natural language assistance to write code, remediate errors and find answers.

Private

Data and AI applications require strong governance and security, especially with the advent of generative AI. Databricks provides an end-to-end MLOps and AI development solution that's built upon our unified approach to governance and security. You're able to pursue all your AI initiatives — from using APIs like OpenAI to custom-built models — without compromising data privacy and IP control.



The Databricks Data Intelligence Platform architecture

Data reliability and performance

[Delta Lake](#) is an open format storage layer built for the data intelligence platform that integrates with all major analytics tools and works with the widest variety of formats to store and process data.

[Photon](#) is the next-generation query engine built for the lakehouse that leverages a state-of-the-art vectorized engine for fast querying and provides the best performance for all workloads in the lakehouse.

In [Chapter 3](#), we explore the details of data reliability and performance for the lakehouse.

Unified governance and security

The Databricks Data Intelligence Platform provides unified governance with enterprise scale, security and compliance. The [Databricks Unity Catalog](#) (UC) provides governance for your data and AI assets in the lakehouse — files, tables, dashboards and machine learning models — giving you much better control, management and security across clouds.

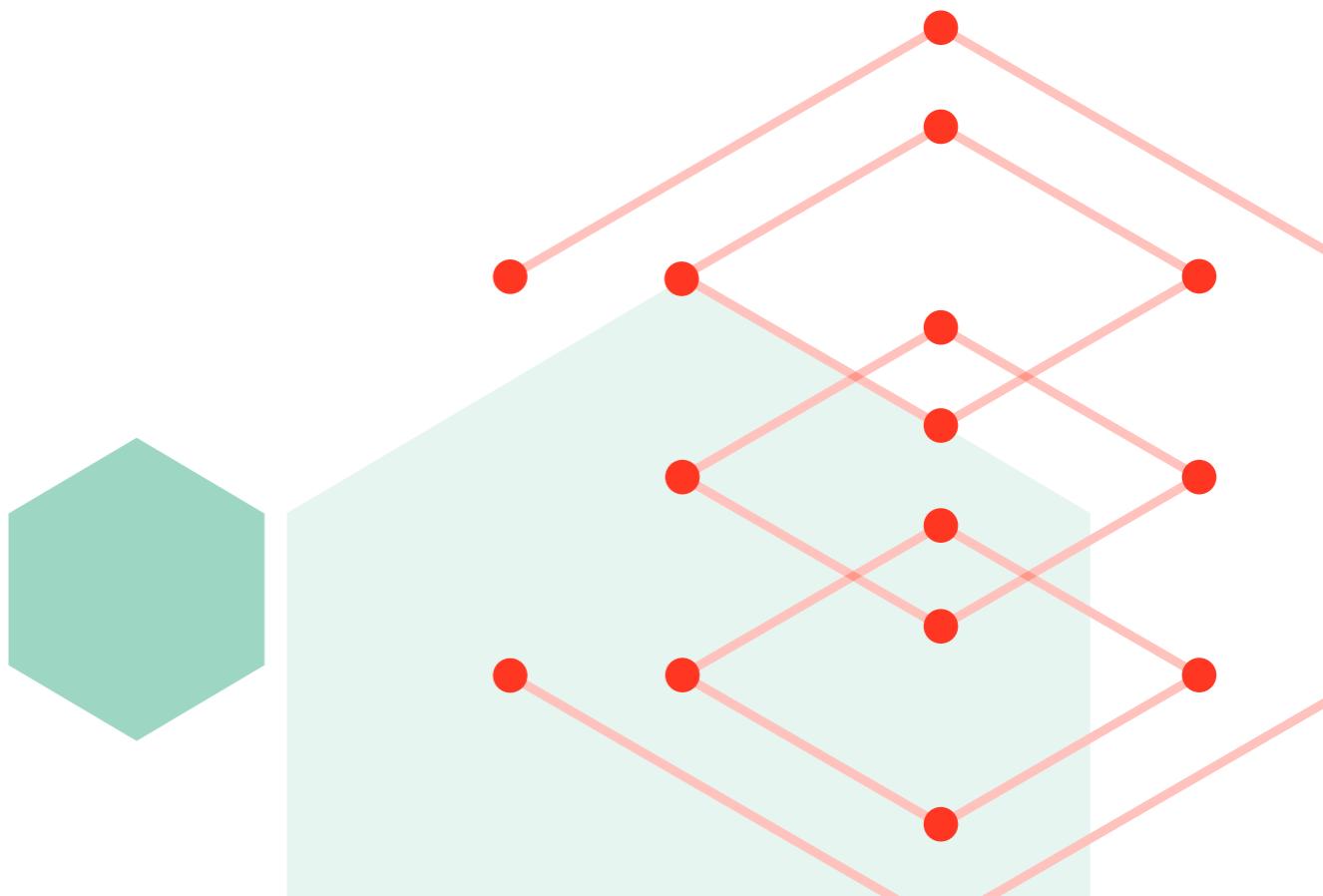
[Delta Sharing](#) is an open protocol that allows companies to securely share data across the organization in real time, independent of the platform on which the data resides.

In [Chapter 4](#), we go into the details of unified governance for lakehouse and, in [Chapter 5](#), we dive into the details of security for lakehouse.

Instant compute and serverless

Serverless compute is a fully managed service where Databricks provisions and manages the compute layer on behalf of the customer in the Databricks cloud account instead of the customer account. As of the current release, serverless compute is supported for use with Databricks SQL, Delta Live Tables, Workflows, and Notebooks.

In [Chapter 6](#), we explore the details of instant compute and serverless for the platform.



The Databricks Data Intelligence Platform workloads

The Databricks Data Intelligence Platform architecture supports different workloads such as data warehousing, data engineering, data streaming, data science and machine learning on one simple, open and multicloud data platform.

Data warehousing

Databricks SQL is a serverless data warehouse built on lakehouse architecture that lets you run all your BI and ETL workloads at scale with up to 12x better price/performance, a unified governance model, open formats and APIs, and your tools of choice — no lock-in.

In [Chapter 7](#), we go into the details of data warehousing on the lakehouse.

Data engineering

Easily ingest and transform batch and streaming data on the Databricks Data Intelligence Platform. Orchestrate reliable production workflows while Databricks automatically manages your infrastructure at scale. Increase the productivity of your teams with built-in data quality testing and support for software development best practices.

In [Chapter 8](#), we go into the details of data engineering on the lakehouse.

Data streaming

The Databricks Data Intelligence Platform dramatically simplifies data streaming to deliver real-time analytics, machine learning and applications on one platform.

Enable your data teams to build streaming data workloads with the languages and tools they already know. Simplify development and operations by automating the production aspects associated with building and maintaining real-time data workloads. Eliminate data silos with a single platform for streaming and batch data.

In [Chapter 9](#), we go into the details of data streaming on the lakehouse.

Data science, AI and machine learning

Streamline the end-to-end data science workflow — from data prep to modeling to sharing insights — with a collaborative and unified data science environment built on an open lakehouse foundation. Get quick access to clean and reliable data, preconfigured compute resources, IDE integration, multi-language support, and built-in advanced visualization tools for maximum flexibility for data analytics teams.

In [Chapter 10](#), we explore the details of data science, AI and machine learning on the data intelligence platform.

Databricks Data Intelligence Platform and your modern data stack

[Partner Connect](#) makes it easy for you to discover data, analytics and AI tools directly within the Databricks platform — and quickly integrate the tools you already use today. With Partner Connect, you can simplify tool integration to just a few clicks and rapidly expand the capabilities of your lakehouse.

In [Chapter 11](#), we go into the details of our technology partners and the modern data stack.

Global adoption of the Databricks Data Intelligence Platform

Today, Databricks has over 10,000+ **customers**, from Fortune 500 to unicorns across industries doing transformational work. Organizations around the globe are driving change and delivering a new generation of data, analytics and AI applications. We believe that the unfulfilled promise of data and AI can finally be fulfilled with one platform for data analytics, data science and machine learning with the Databricks Data Intelligence Platform.



[Learn more](#)

[Databricks Data Intelligence Platform](#)

[Databricks Data Intelligence Platform Demo Center](#)

[Databricks Data Intelligence Platform Customer Stories](#)

[Databricks Data Intelligence Platform Documentation](#)

[Databricks Data Intelligence Platform Training and Certification](#)

[Databricks Data Intelligence Platform Resources](#)

CHAPTER

03

Data reliability and performance

To bring openness, reliability and lifecycle management to data lakes, the Databricks lakehouse architecture is built on the foundation of Delta Lake. Delta Lake solves challenges around unstructured/structured data ingestion, the application of data quality, difficulties with deleting data for compliance or issues with modifying data for data capture.

Although data lakes are great solutions for holding large quantities of raw data, they lack important attributes for data reliability and quality and often don't offer good performance when compared to data warehouses.

Problems with today's data lakes

When it comes to data reliability and quality, examples of these missing attributes include:

- **Lack of ACID transactions:** Makes it impossible to mix updates, appends and reads
- **Lack of schema enforcement:** Creates inconsistent and low-quality data. For example, rejecting writes that don't match a table's schema.
- **Lack of integration with data catalog:** Results in dark data and no single source of truth

Even just the absence of these three attributes can cause a lot of extra work for data engineers as they strive to ensure consistent high-quality data in the pipelines they create.

As for performance, data lakes use object storage, so data is mostly kept in immutable files leading to the following problems:

- **Ineffective partitioning:** In many cases, data engineers resort to "poor man's" indexing practices in the form of partitioning that leads to hundreds of dev hours spent tuning file sizes to improve read/write performance. Often, partitioning proves to be ineffective over time if the wrong field was selected for partitioning or due to high cardinality columns.
- **Too many small files:** With no support for transactions, appending new data takes the form of adding more and more files, leading to "small file problems," a known root cause of query performance degradation.

These challenges are solved with two key technologies that are at the foundation of the lakehouse: Delta Lake and Photon.

What is Delta Lake?

Delta Lake is the only open format storage layer that can automatically and instantly translate across open formats. Delta Lake unifies all data types for transactional, analytical and AI use cases out of the box, with support for streaming and batch operations. Delta Lake offers industry-leading performance and is the foundation of a cost-effective, highly scalable lakehouse.

With Delta Lake Universal Format (UniForm), you will be able to use your favorite Iceberg or Hudi client to read your Delta tables through the Unity Catalog endpoint. Delta Lake 3.0 simplifies the connector ecosystem. Delta Kernel offers a stable library API, making it easier for connectors to incorporate new Delta features without code changes.

Delta Lake features

ACID guarantees

Delta Lake ensures that all data changes written to storage are committed for durability and made visible to readers atomically. In other words, no more partial or corrupted files.

Scalable data and metadata handling

Since Delta Lake is built on data lakes, all reads and writes using Spark or other distributed processing engines are inherently scalable to petabyte-scale. However, unlike most other storage formats and query engines, Delta Lake leverages Spark to scale out all the metadata processing, thus efficiently handling metadata of billions of files for petabyte-scale tables.

Audit history and time travel

The Delta Lake transaction log records details about every change made to data, providing a full audit trail of the changes. These data snapshots allow developers to access and revert to earlier versions of data for audits, rollbacks or to reproduce experiments.

Schema enforcement and schema evolution

Delta Lake automatically prevents the insertion of data with an incorrect schema, i.e., not matching the table schema. And when needed, it allows the table schema to be explicitly and safely evolved to accommodate ever-changing data.

Support for deletes, updates and merges

Most distributed processing frameworks do not support atomic data modification operations on data lakes. Delta Lake supports merge, update and delete operations to enable complex use cases including but not limited to change data capture (CDC), slowly changing dimension (SCD) operations and streaming upserts.

Streaming and batch unification

A Delta Lake table can work both in batch and as a streaming source and sink. The ability to work across a wide variety of latencies, ranging from streaming data ingestion to batch historic backfill, to interactive queries all work out of the box.

The Delta Lake transaction log

A key to understanding how Delta Lake provides all these capabilities is the transaction log. The Delta Lake transaction log is the common thread that runs through many of Delta Lake's most notable features, including ACID transactions, scalable metadata handling, time travel and more. The Delta Lake transaction log is an ordered record of every transaction that has ever been performed on a Delta Lake table since its inception.

Delta Lake is built on top of Spark to allow multiple readers and writers of a given table to work on a table at the same time. To always show users correct views of the data, the transaction log serves as a single source of truth: the central repository that tracks all changes that users make to the table.

When a user reads a Delta Lake table for the first time or runs a new query on an open table that has been modified since the last time it was read, Spark checks the transaction log to see what new transactions are posted to the table. Then, Spark updates the table with those recent changes. This ensures that a user's version of a table is always synchronized with the master record as of the most recent query, and that users cannot make divergent, conflicting changes to a table.

Flexibility and broad industry support

Delta Lake is an open source project, with an engaged community of contributors building and growing the Delta Lake ecosystem atop a set of open APIs and is part of the Linux Foundation. With the growing adoption of Delta Lake as an open storage standard in different environments and use cases, comes a broad set of integration with industry-leading tools, technologies and formats.

Organizations leveraging Delta Lake on the Databricks Data Intelligence Platform gain flexibility in how they ingest, store and query data. They are not limited in storing data in a single cloud provider and can implement a true multicloud approach to data storage.

Connectors to tools, such as Fivetran, allow you to leverage Databricks' ecosystem of partner solutions, so organizations have full control of building the right ingestion pipelines for their use cases. Finally, consuming data via queries for exploration or business intelligence (BI) is also flexible and open.

Delta Lake integrates with all major analytics tools

Eliminates unnecessary data movement and duplication



In addition to a wide ecosystem of tools and technologies, Delta Lake supports a broad set of data formats for structured, semi-structured and unstructured data. These formats include image binary data that can be stored in Delta Tables, graph data format, geospatial data types and key-value stores.

Learn more

[Delta Lake on the Databricks Data Intelligence Platform](#)

[Data Intelligence Platform Documentation](#)

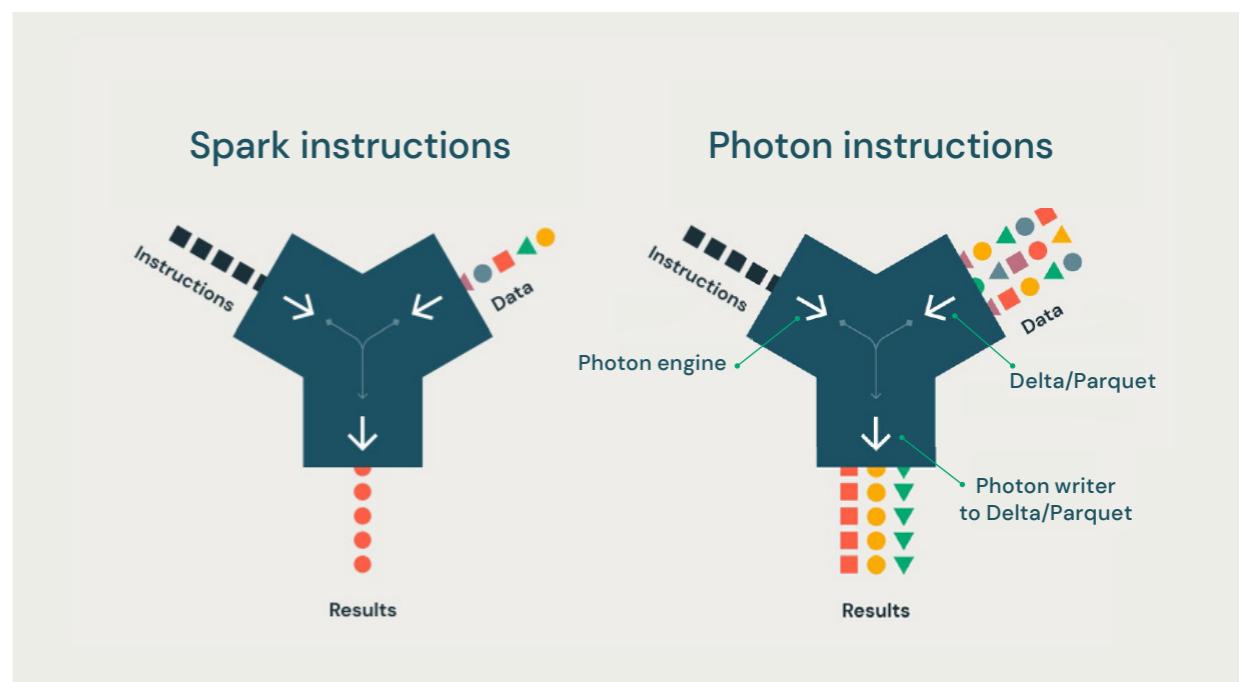
[Delta Lake Open Source Project](#)

[eBooks: The Delta Lake Series](#)

What is Photon?

As many organizations standardize on the lakehouse paradigm, this new architecture poses challenges with the underlying query execution engine for accessing and processing structured and unstructured data. The execution engine needs to provide the performance of a data warehouse and the scalability of data lakes.

Photon is the next-generation query engine on the Databricks Data Intelligence Platform that provides dramatic infrastructure cost savings and speedups for all use cases — from data ingestion, ETL, streaming, data science and interactive queries — directly on your data lake. Photon is compatible with Spark APIs and implements a more general execution framework that allows efficient processing of data with support of the Spark API. This means getting started is as easy as turning it on — no code change and no lock-in. With Photon, typical customers are seeing up to 80% TCO savings over traditional Databricks Runtime (Spark) and up to 85% reduction in VM compute hours.

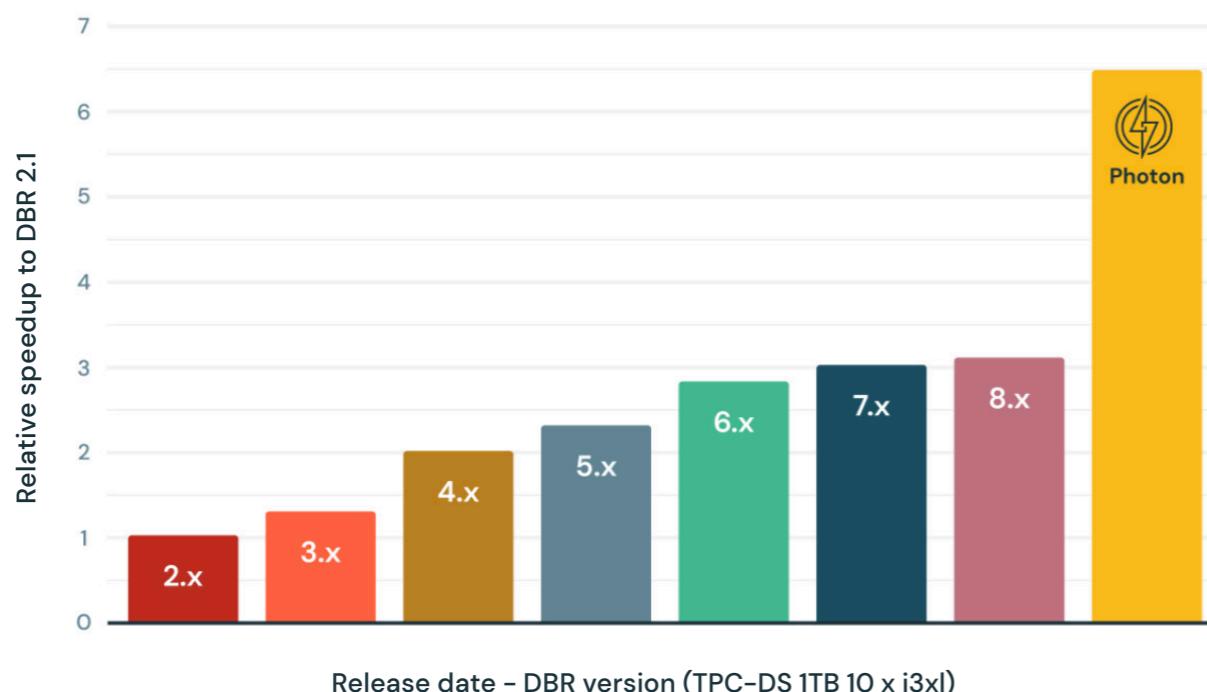


Why process queries with Photon?

Query performance on Databricks has steadily increased over the years, powered by Spark and thousands of optimizations packaged as part of the Databricks Runtime (DBR). Photon provides an additional 2x speedup per the TPC-DS 1TB benchmark compared to the latest DBR versions.

Relative speedup to DBR 2.1 by DBR version

Higher is better



Customers have observed significant speedups using Photon on workloads such as:

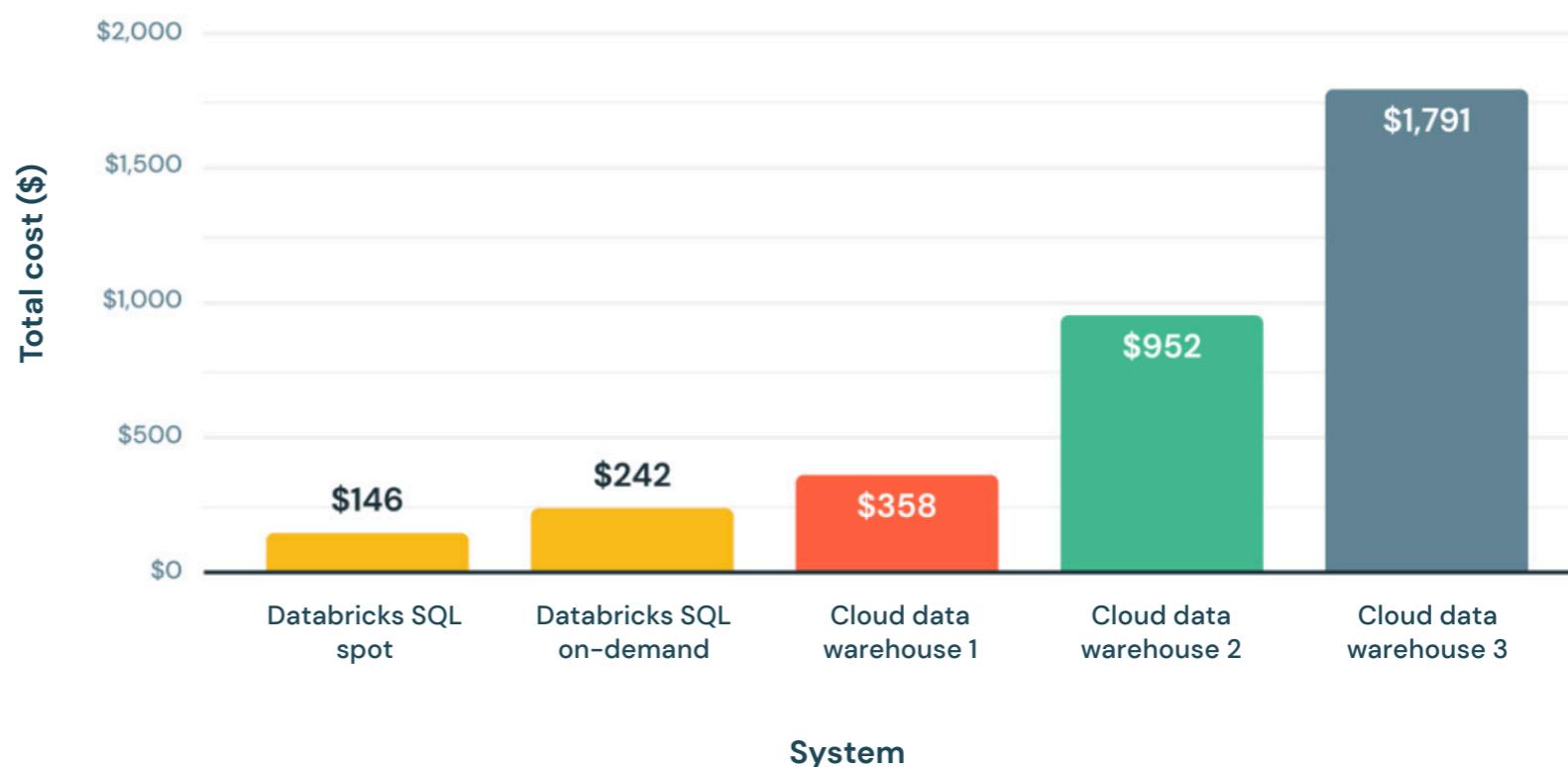
- **SQL-based jobs:** Accelerate large-scale production jobs on SQL and Spark DataFrames
- **IoT use cases:** Faster time-series analysis using Photon compared to Spark and traditional Databricks Runtime
- **Data privacy and compliance:** Query petabytes-scale data sets to identify and delete records without duplicating data with Delta Lake, production jobs and Photon
- **Loading data into Delta and Parquet:** Vectorized I/O speeds up data loads for Delta and Parquet tables, lowering overall runtime and costs of data engineering jobs

Best price/performance for analytics in the cloud

Written from the ground up in C++, Photon takes advantage of modern hardware for faster queries, providing up to 12x better price/performance compared to other cloud data warehouses — all natively on your data lake.

100TB TPC-DS price/performance

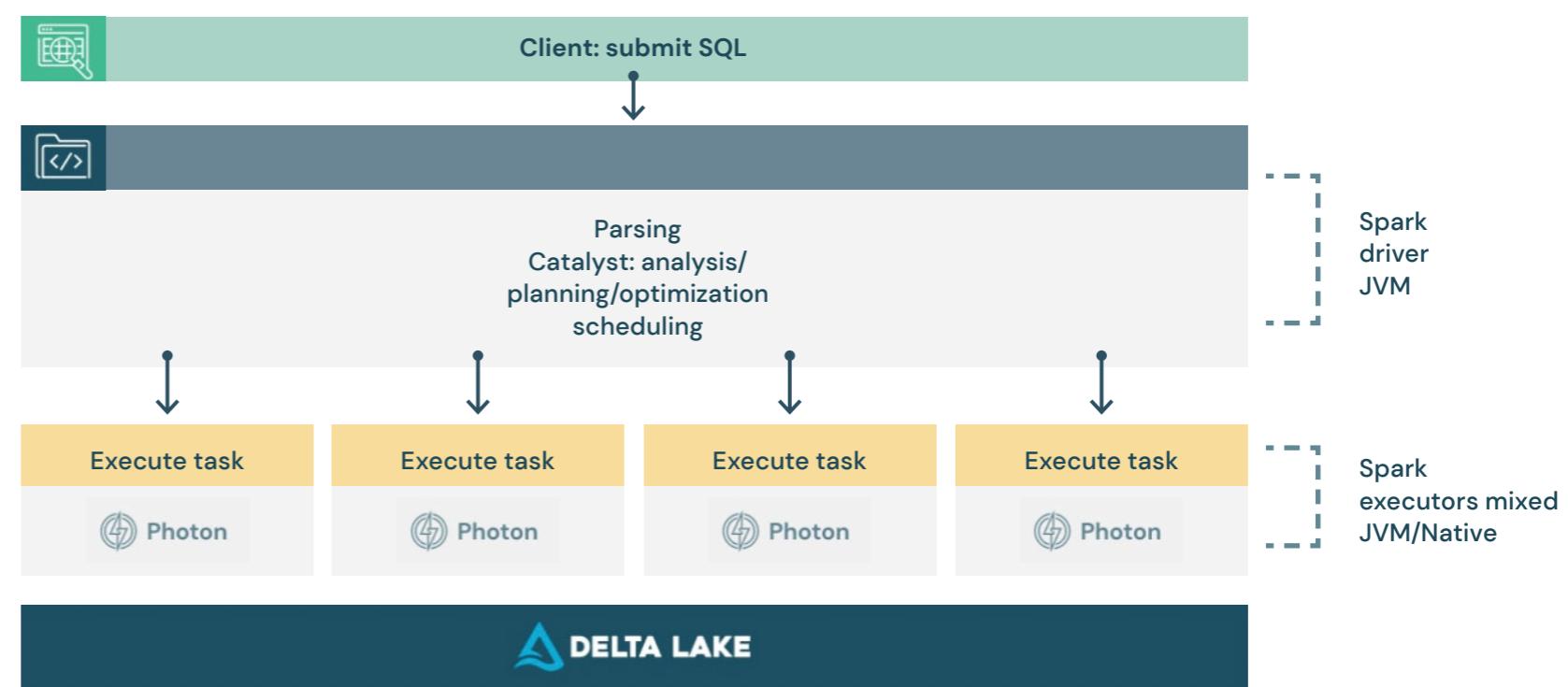
Lower is better



Works with your existing code and avoids vendor lock-in

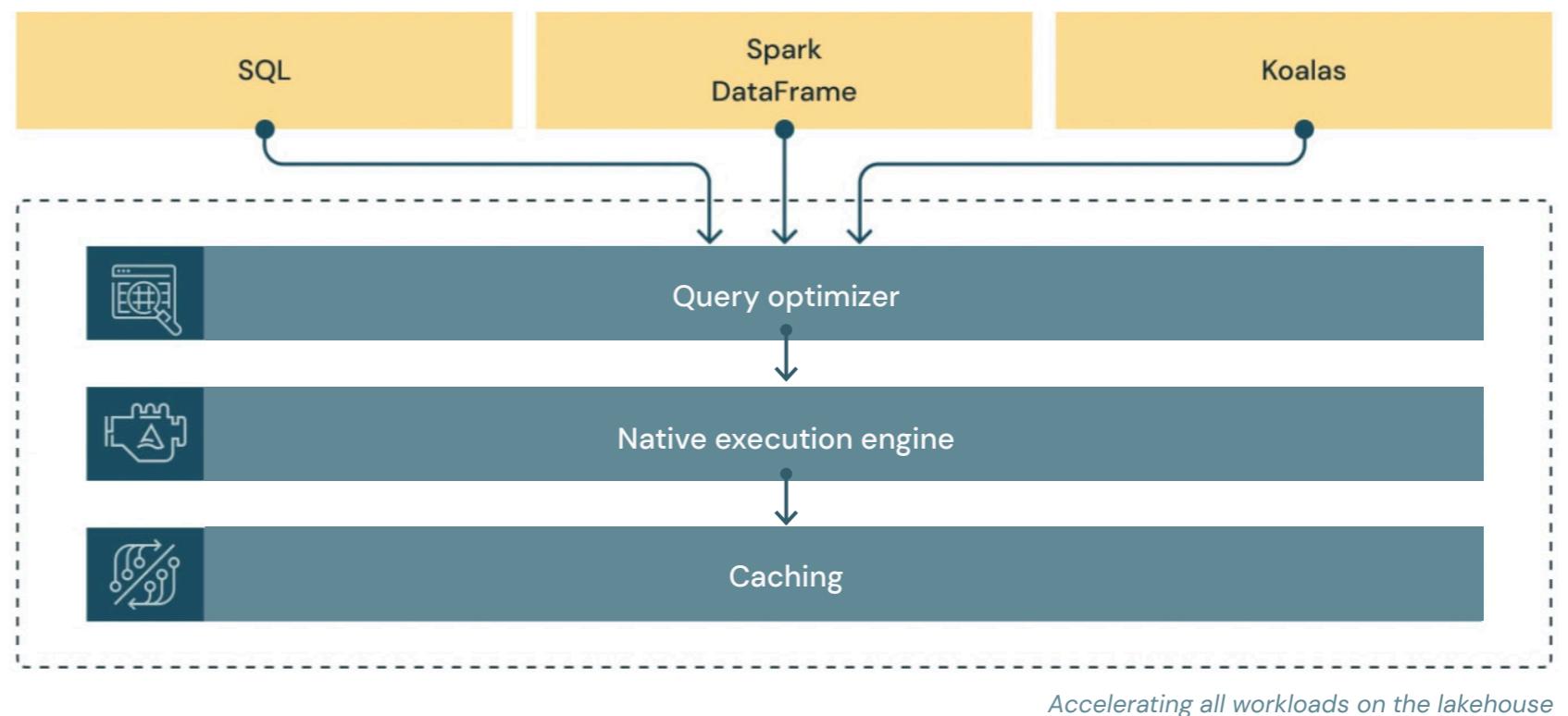
Photon is designed to be compatible with the Apache Spark DataFrame and SQL APIs to ensure workloads run seamlessly without code changes. All you do is turn it on. Photon will seamlessly coordinate work and resources and transparently accelerate portions of your SQL and Spark queries. No tuning or user intervention required.

Photon in the Databricks Data Intelligence Platform



Optimizing for all data use cases and workloads

Photon is the first purpose-built lakehouse engine designed to accelerate all data and analytics workloads: data ingestion, ETL, streaming, data science, and interactive queries. While we started Photon primarily focused on SQL to provide customers with world-class data warehousing performance on their data lakes, we've significantly increased the scope of ingestion sources, formats, APIs and methods supported by Photon since then. As a result, customers have seen dramatic infrastructure cost savings and speedups on Photon across all their modern Spark (e.g., Spark SQL and DataFrame) workloads.



Learn more

[Announcing Photon Public Preview: The Next-Generation Query Engine on the Databricks Data Intelligence Platform](#)

[Databricks Sets Official Data Warehousing Performance Record](#)

CHAPTER

04

Unified governance and sharing for data, analytics and AI

Today, more and more organizations recognize the importance of making high-quality data readily available to data teams across business units, or externally with customers and partners to drive actionable insights and business value. At the same time, organizations also understand the risks of data breaches which negatively impact brand value and inevitably lead to erosion of customer trust. Governance is one of the most critical components of a lakehouse architecture; it helps ensure that data assets are securely managed throughout the enterprise. However, many companies are using different incompatible governance models leading to complex and expensive solutions.

Key challenges with data and AI governance

Diversity of data and AI assets

The increased use of data and the added complexity of the data landscape have left organizations with a difficult time managing and governing all types of their data-related assets. No longer is data stored in files or tables. Data assets today take many forms, including dashboards, machine learning models and unstructured data like video and images that legacy data governance solutions simply are not built to govern and manage.

Two disparate and incompatible data platforms

Organizations today use two different platforms for their data analytics and AI efforts — data warehouses for BI and data lakes for AI. This results in data replication across two platforms, presenting a major governance challenge. With no unified view of the data landscape, it is difficult to see where data is stored, who has access to what data, and consistently define and enforce data access policies across the two platforms with different governance models.

Rising multicloud adoption

More and more organizations now leverage a multicloud strategy to optimize costs, avoid vendor lock-in, and meet compliance and privacy regulations. With nonstandard, cloud-specific governance models, data governance across clouds is complex and requires familiarity with cloud-specific security and governance concepts, such as identity and access management (IAM).

Disjointed tools for data governance on the lakehouse

Today, data teams must deal with a myriad of fragmented tools and services for their data governance requirements, such as data discovery, cataloging, auditing, sharing, access controls, etc. This inevitably leads to operational inefficiencies and poor performance due to multiple integration points and network latency between the services.

One security and governance approach

Lakehouse systems provide a uniform way to manage access control, data quality and compliance across all of an organization's data using standard interfaces similar to those in data warehouses by adding a management interface on top of data lake storage.

Modern lakehouse systems support fine-grained (row, column and view level) access control via SQL, query auditing, attribute-based access control, data versioning and data quality constraints and monitoring. These features are generally provided using standard interfaces familiar to database administrators (for example, SQL GRANT commands) to allow existing personnel to manage all the data in an organization in a uniform way. Centralizing all the data in a lakehouse system with a single management interface also reduces the administrative burden and potential for error that comes with managing multiple separate systems.

What is Unity Catalog?

Databricks Unity Catalog offers a unified governance layer for data and AI within the Databricks Data Intelligence Platform. Unity Catalog simplifies governance by empowering data teams with a common governance model based on ANSI-SQL to define and enforce fine-grained access controls. With attribute-based access controls, data administrators can enable fine-grained access controls on rows and columns using tags (attributes). Built-in data search and discovery allows data teams to quickly find and reference relevant data for any use case. Unity Catalog offers automated data lineage for all workloads in SQL, R, Scala and Python, to build a better understanding of the data and its flow in the lakehouse. Unity Catalog also allows data sharing across or within organizations and seamless integrations with your existing data governance tools.

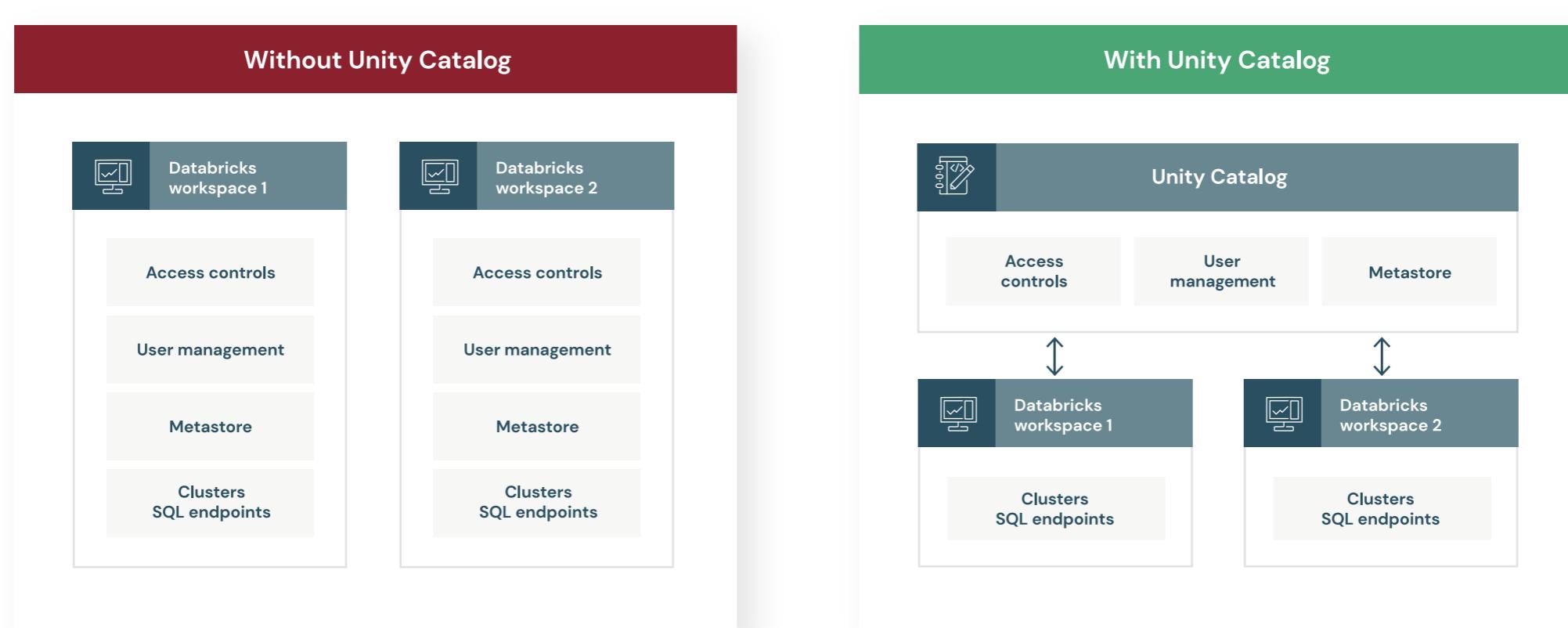
With Unity Catalog, data teams can simplify governance for all data and AI assets with one consistent model to discover, access and share data, giving you much better native performance, management and security across clouds.

Key benefits

Catalog, secure and audit access to all data assets on any cloud

Unity Catalog provides centralized metadata, enabling data teams to create a single source of truth for all data assets ranging from files, tables, dashboards to machine learning models in one place.

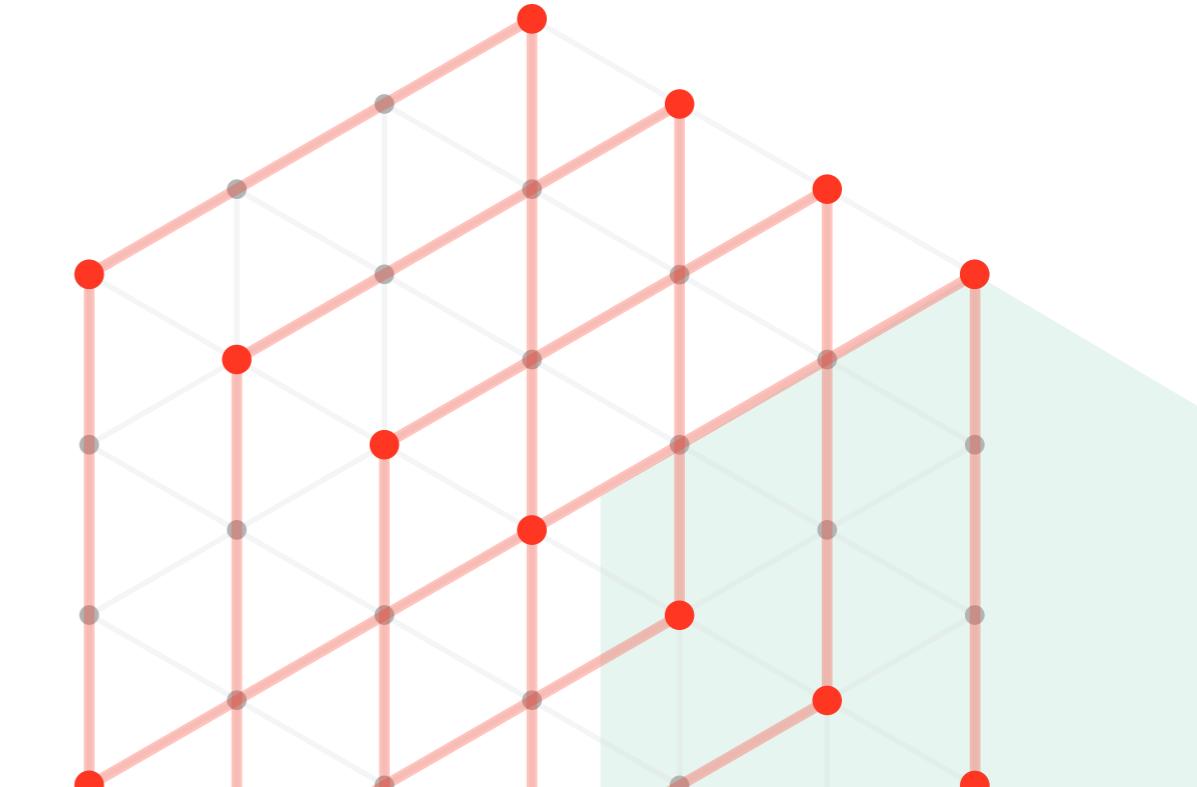
The common metadata layer for cross-workspace metadata is at the account level and eases collaboration by allowing different workspaces to access Unity Catalog metadata through a common interface and break down data silos. Further, the data permissions in Unity Catalog are applied to account-level identities, rather than identities that are local to a workspace, allowing a consistent view of users and groups across all workspaces.



Unity Catalog offers a unified data access layer that provides a simple and streamlined way to define and connect to your data through managed tables, external tables, or files, while managing their access controls. Unity Catalog centralizes access controls for files, tables and views.

It allows fine-grained access controls for restricting access to certain rows and columns to the users and groups who are authorized to query them. With Attribute-Based Access Controls (ABAC), you can control access to multiple data items at once based on user and data attributes, further simplifying governance at scale. For example, you will be able to tag multiple columns as personally identifiable information (PII) and manage access to all columns tagged as PII in a single rule.

Today, organizations are dealing with an increased burden of regulatory compliance, and data access auditing is a critical component to ensure your organization is set up for success while meeting compliance requirements. Unity Catalog also provides centralized fine-grained auditing by capturing an audit log of operations such as create, read, update and delete (CRUD) that have been performed against the data. This allows a fine-grained audit trail showing who accessed a given data set and helps you meet your compliance and business requirements.



Built-in data search and discovery

Data discovery is a critical component to break down data silos and democratize data across your organization to make data-driven decisions. Unity Catalog provides a rich user interface for data search and discovery, enabling data teams to quickly search relevant data assets across the data landscape and reference them for all use cases — BI, analytics and machine learning — accelerating time-to-value and boosting productivity.

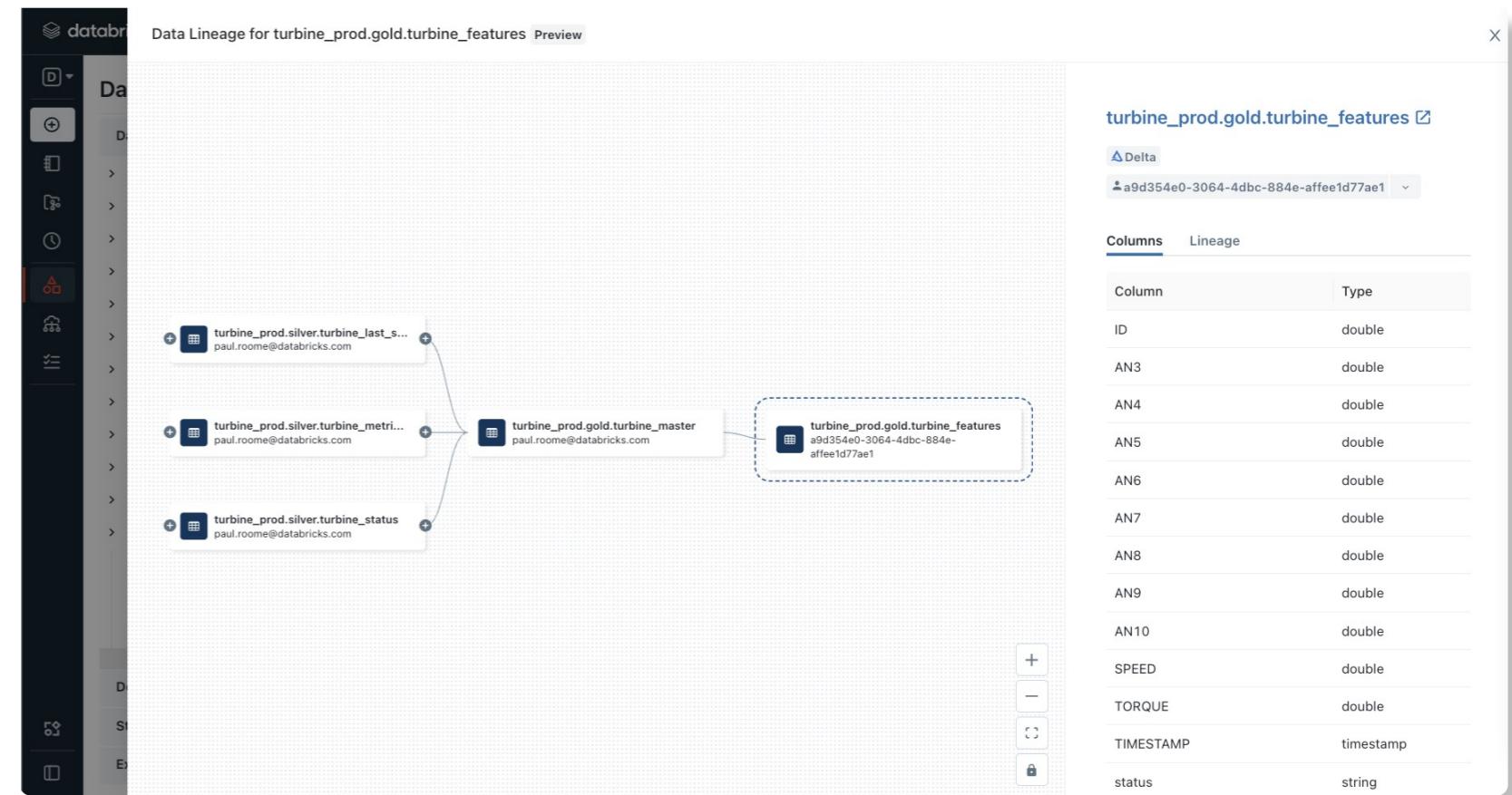
The screenshot shows the Databricks Data Explorer interface. On the left, there's a sidebar with various icons for navigation. The main area is titled "Data Explorer" and shows a tree view of datasets under "main". One dataset, "nyc_taxi_trips", is expanded, revealing its sub-tables: "main.default", "main.mohan_uc", and "sample_datasets.mohan". Another dataset, "flights", is also visible. A search modal is open in the center, with a search bar containing "taxi". Below the search bar are dropdown menus for "Catalog" (set to "Catalog") and "Database" (set to "Database"). The modal lists several results matching the search term:

- nyc_taxi_trips (main.default)
- nyc_taxi_trips (main.mohan_uc)
- nyc_taxi_trips (sample_datasets.mohan)
- flights (deltasharing.airlinedata)
- flights (as_deltasharing.airlinedata)

Automated data lineage for all workloads

Data lineage describes the transformations and refinements of data from source to insight. Lineage includes capturing all the relevant metadata and events associated with the data in its lifecycle, including the source of the data set, what other data sets were used to create it, who created it and when, what transformations were performed, which other data sets leverage it, and many other events and attributes. Unity Catalog offers automated data lineage down to table and column level, enabling data teams to get an end-to-end view of where data is coming from, what transformations were performed on the data and how data is consumed by end applications such as notebooks, workflows, dashboards, machine learning models, etc.

With automated data lineage for all workloads — SQL, R, Python and Scala, data teams can quickly identify and perform root cause analysis of any errors in the data pipelines or end applications. Second, data teams can perform impact analysis to see dependencies of any data changes on downstream consumers and notify them about the potential impact. Finally, data lineage also empowers data teams with increased understanding of their data and reduces tribal knowledge. Unity Catalog can also capture lineage associated with non-data entities, such as notebooks, workflows and dashboards. Lineage can be



Data lineage with Unity Catalog

retrieved via REST APIs to support integrations with other catalogs.

Integrated with your existing tools

Unity Catalog helps you to future-proof your data and AI governance with the flexibility to leverage your existing data catalogs and governance solutions — Collibra, Alation, Immuta, Privacera, Microsoft Purview and AWS Lakeformation.



Resources

[Learn more about Unity Catalog](#)

[AWS Documentation](#)

[Azure Documentation](#)

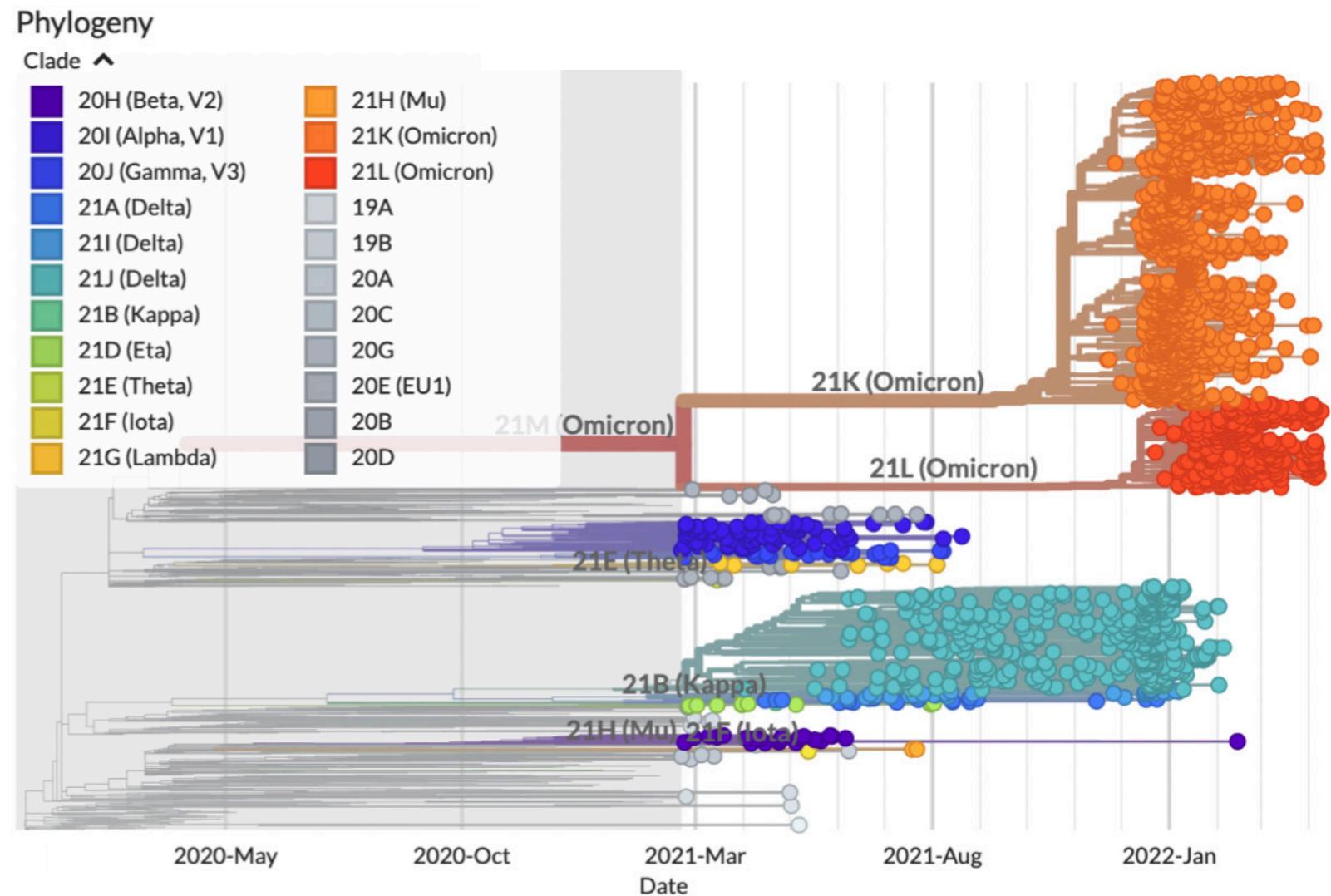
Open data sharing and collaboration

Data sharing has become important in the digital economy as enterprises wish to exchange data easily and securely with their customers, partners, suppliers and internal lines of business to better collaborate and unlock value from that data. But to date, a lack of standards-based data sharing protocol has resulted in data sharing solutions tied to a single vendor or commercial product, introducing vendor lock-in risks. What the industry deserves is an open approach to data sharing.

Why data sharing is hard

Data sharing has evolved from an optional feature of a few data platforms to a business necessity and success factor for organizations. Our solution architects encounter daily the classic scenarios of a retailer looking to publish sales data to their suppliers in real time or a supplier that wants to share real-time inventory.

As a reminder, data sharing recently triggered the most impressive scientific development that humankind has ever seen. On January 5, 2021, the first sample of the genome of the coronavirus was uploaded to the internet. It wasn't a lung biopsy



from a patient in Wuhan, but a shared digital genomic data set that triggered the development of the first batch of COVID vaccines worldwide.

Since then, coronavirus experts have daily exchanged public data sets, looking for better treatments, tests and tracking mutations as they

are passed down through a lineage, a branch of the coronavirus family tree. The above graphic shows such a [publicly shared mutation data set](#).

Sharing data, as well as consuming data from external sources, allows you to collaborate with partners, establish new partnerships, enable research and can generate new revenue streams with data monetization.

Despite those promising examples, existing data sharing technologies come with several limitations:

- Traditional data sharing technologies, such as Secure File Transfer Protocol (SFTP), do not scale well and only serve files offloaded to a server
- Cloud object stores operate on an object level and are cloud-specific
- Commercial data sharing offerings baked into vendor products often share tables instead of files, but scaling them is expensive and they are not open and, therefore, do not permit data sharing with a different platform

The following table compares proprietary vendor solutions with SFTP, cloud object stores and Delta Sharing.

	Proprietary vendor solutions	SFTP	Cloud object store	Delta Sharing
Secure	✓	✓	✓	✓
Cheap		✓	✓	✓
Vendor agnostic		✓		✓
Multicloud		✓		✓
Open source		✓		✓
Table/DataFrame abstraction	✓			✓
Live data	✓			✓
Predicate pushdown	✓			✓
Object store bandwidth			✓	✓
Zero compute cost			✓	✓
Scalability			✓	✓

Open source data sharing and Databricks

To address the limitations of existing data sharing solutions, Databricks developed [Delta Sharing](#), with various contributions from the OSS community, and donated it to the Linux Foundation.

An open source-based solution, such as Delta Sharing, eliminates the lock-in of commercial solutions and brings a number of additional benefits such as community-developed integrations with popular, open source data processing frameworks. In addition, open protocols allow the easy integration of commercial clients, such as BI tools.

What is Databricks Delta Sharing?

Databricks Delta Sharing provides an open solution to securely share live data from your lakehouse to any computing platform. Recipients don't have to be on the Databricks Platform or on the same cloud or a cloud at all. Data providers can share live data, without replicating or moving it to another system. Recipients benefit from always having access to the latest version of data and can quickly query shared data using tools of their choice for BI, analytics and machine learning, reducing time-to-value. Data providers can centrally manage, govern, audit and track usage of the shared data on one platform.

Unity Catalog natively supports [Delta Sharing](#), the world's first open protocol for data sharing, enabling organizations to share live, large-scale data without replication and make data easily and quickly accessible from tools of your choice, with enterprise-grade security.

Key benefits

Open cross-platform sharing

Easily share existing data in Delta Lake and Apache Parquet formats between different vendors. Consumers don't have to be on the Databricks Platform, same cloud or a cloud at all. Native integration with Power BI, Tableau, Spark, pandas and Java allow recipients to consume shared data directly from the tools of their choice. Delta Sharing eliminates the need to set up a new ingestion process to consume data. Data recipients can directly access the fresh data and query it using tools of their choice. Recipients can also enrich data with data sets from popular data providers.

Sharing live data without copying it

Share live ready-to-query data, without replicating or moving it to another system. Most enterprise data today is stored in cloud data lakes. Any of the existing data sets on the provider's data lake can easily be shared across clouds, regions or data platforms without any data replication or physical movement of data. Data providers can update their data sets reliably in real time and provide a fresh and consistent view of their data to recipients.

Centralized administration and governance

You can centrally govern, track and audit access to the shared data from a single point of enforcement to meet compliance requirements. Detailed user-access audit logs are kept to know who is accessing the data and monitor usage of the shared data down to table, partition and version level.

An open Marketplace for data solutions

The demand for third-party data to make data-driven innovations is greater than ever, and data marketplaces act as a bridge between data providers and data consumers to help facilitate the discovery and distribution of data sets.

Databricks Marketplace provides an open marketplace for exchanging data products such as data sets, notebooks, dashboards and machine learning models. To accelerate insights, data consumers can discover, evaluate and access more data products from third-party vendors than ever before. Providers can now commercialize new offerings and shorten sales cycles by providing value-added services on top of their data.

Databricks Marketplace is powered by Delta Sharing, allowing consumers to access data products without having to be on the Databricks Platform. This open approach allows data providers to broaden their addressable market without forcing consumers into vendor lock-in.

Databricks Marketplace
Powered by DELTA SHARING

Discover, evaluate, and gain access to data sets, notebooks, ML models, and more.

Provider console

Search the Databricks Marketplace for data sets, notebooks, ML models, and more.

All providers | All categories | All types

Collaborators

- Axiom**
- Dun & Bradstreet**
- IQVIA**
- Nasdaq Data Link**
- S&P Global**

Featured data products

- Nasdaq Data Link**
Consolidated Quotes and Trades
Provides investors with easy and flexible access to high quality and reliable real time, streaming level 1 data, including the best bid, best ask, and best price for...
- Dun & Bradstreet**
Global Worldbase Layouts
The Global Worldbase Data Layouts can contains firmographic information on companies around the world including their D-U-N-S number, location, industry...
- AccuWeather**
Actionable Weather Forecasts
Hyper-localized daily and hourly forecasts focused impact to people and businesses, so they can make the best weather-impacted decisions.
- Carto**
Spatial Features
Spatial Features is a dataset curated by CARTO providing access to a set of location-based features with global coverage that have been unified in...

Databricks Marketplace

Privacy-safe data cleanrooms

Powered by open source Delta Sharing, the Databricks Data Intelligence Platform provides a flexible data cleanroom solution allowing businesses to easily collaborate with their customers and partners on any cloud in a privacy-safe way. Participants in the data cleanrooms can share and join their existing data, and run complex workloads in any language — Python, R, SQL, Java and Scala — on the data while maintaining data privacy. Additionally, data cleanroom participants don't have to do cost-intensive data replication across clouds or regions with other participants, which simplifies data operations and reduces cost.



Data cleanrooms with the Databricks Data Intelligence Platform

How it works

Delta Sharing is designed to be simple, scalable, non-proprietary and cost-effective for organizations that are serious about getting more from their data. Delta Sharing is natively integrated with Unity Catalog, which allows customers to add fine-grained governance and security controls, making it easy and safe to share data internally or externally.

Delta Sharing is a simple REST protocol that securely shares access to part of a cloud data set. It leverages modern cloud storage systems — such as AWS S3, Azure ADLS or Google's GCS — to reliably transfer large data sets. Here's how it works for data providers and data recipients.



The data provider shares existing tables or parts thereof (such as specific table versions or partitions) stored on the cloud data lake in Delta Lake format. The provider decides what data they want to share and runs a sharing server in front of it that implements the Delta Sharing protocol and manages access for recipients. To manage shares and recipients, you can use SQL commands or the Unity Catalog CLI or the intuitive user interface.

The data recipient only needs one of the many Delta Sharing clients that supports the protocol. Databricks has released open source connectors for pandas, Apache Spark, Java and Python, and is working with partners on many more.

The Delta Sharing data exchange follows three efficient steps:

1. The recipient's client authenticates to the sharing server and asks to query a specific table. The client can also provide filters on the data (for example, "country=US") as a hint to read just a subset of the data.
2. The server verifies whether the client is allowed to access the data, logs the request, and then determines which data to send back. This will be a subset of the data objects in cloud storage systems that make up the table.
3. To transfer the data, the server generates short-lived presigned URLs that allow the client to read these Parquet files directly from the cloud provider, so that the transfer can happen in parallel at massive bandwidth, without streaming through the sharing server.



[Learn more](#)

[Try Delta Sharing](#)

[Delta Sharing Demo](#)

[Introducing Delta Sharing: An Open Protocol for Secure Data Sharing](#)

[Introducing Data Cleanrooms for the Lakehouse](#)

[Introducing Databricks Marketplace](#)

[Delta Sharing ODSC Webinar](#)

CHAPTER

05

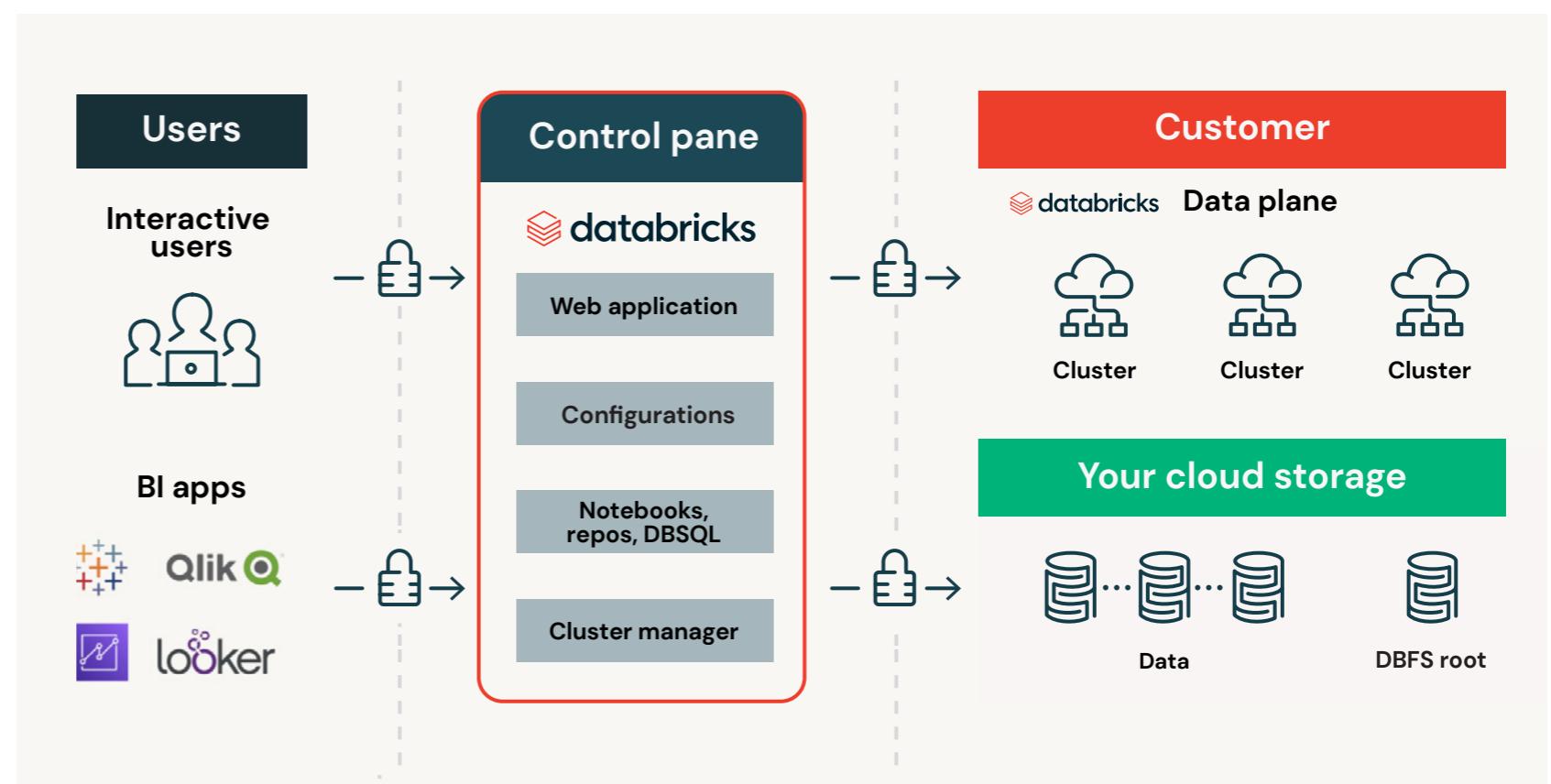
Security

Organizations that operate in multicloud environments need a unified, reliable and consistent approach to secure data. We've learned from our customers that a simple and unified approach to data security for the lakehouse is one of the most critical requirements for modern data solutions. Databricks is trusted by the world's largest organizations to provide a powerful lakehouse architecture with high security and scalability. In fact, thousands of customers trust Databricks with their most sensitive data to analyze and build data products using machine learning (ML). With significant investment in building a highly secure and scalable platform, Databricks delivers end-to-end platform security for data and users.

Platform architecture reduces risk

The Databricks lakehouse architecture is split into two separate planes to simplify your permissions, avoid data duplication and reduce risk. The control plane is the management plane where Databricks runs the workspace application and manages notebooks, configuration and clusters. Unless you choose to use [serverless compute](#), the data plane runs inside your cloud service provider account, processing your data without taking it out of your account. You can embed Databricks in your data exfiltration protection architecture using features like customer-managed VPCs/VNets and admin console options that disable export.

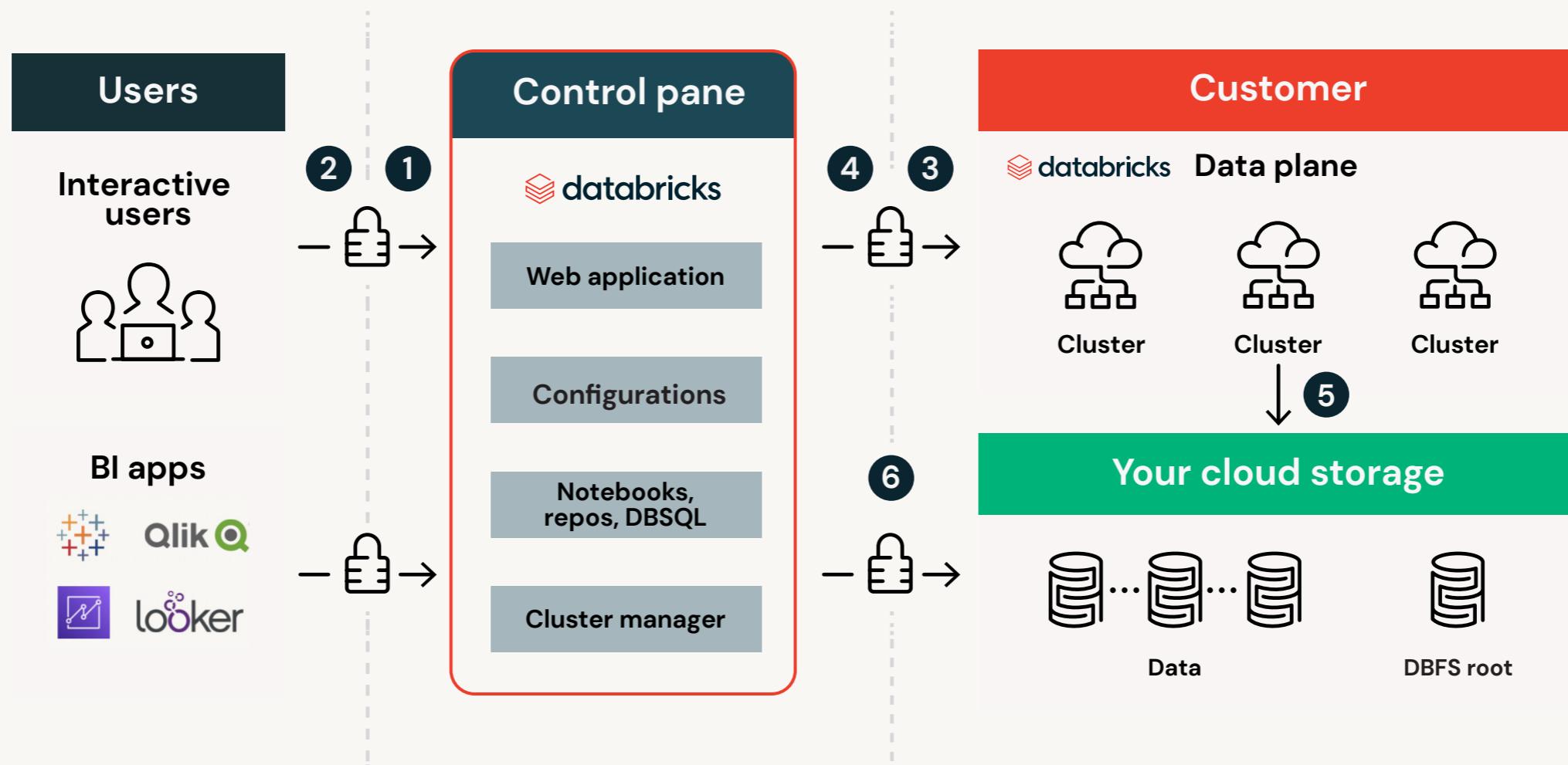
While certain data, such as your notebooks, configurations, logs, and user information, is present within the control plane, that information is encrypted at rest, and communication to and from the control plane is encrypted in transit.



You also have choices for where certain data lives: You can host your own store of metadata about your data tables (Hive metastore), or store query

results in your cloud service provider account and decide whether to use the [Databricks Secrets API](#).

Step-by-step example



Suppose you have a data engineer that signs in to Databricks and writes a notebook that transforms raw data in Kafka to a normalized data set sent to storage such as Amazon S3 or Azure Data Lake Storage. Six steps make that happen:

1. The data engineer seamlessly authenticates, via your single sign-on if desired, to the Databricks web UI in the control plane, hosted in the Databricks account.
2. As the data engineer writes code, their web browser sends it to the control plane. JDBC/ODBC requests also follow the same path, authenticating with a token.
3. When ready, the control plane uses Cloud Service Provider APIs to create a Databricks cluster, made of new instances in the data plane, in your CSP account. Administrators can apply cluster policies to enforce security profiles.
4. Once the instances launch, the cluster manager sends the data engineer's code to the cluster.
5. The cluster pulls from Kafka in your account, transforms the data in your account and writes it to a storage in your account.
6. The cluster reports status and any outputs back to the cluster manager.

The data engineer does not need to worry about many of the details — simply write the code and Databricks runs it.

Network and server security

Here is how Databricks interacts with your cloud service provider account to manage network and server security

Networking

Regardless of where you choose to host the data plane, Databricks networking is straightforward. If you host it yourself, Databricks by default will still configure networking for you, but you can also control data plane networking with your own managed VPC or VNet.

The serverless data plane network infrastructure is managed by Databricks in a Databricks cloud service provider account and shared among customers, with additional network boundaries between workspaces and between clusters.

Databricks does not rewrite or change your data structure in your storage, nor does it change or modify any of your security and governance policies. Local firewalls complement security groups and subnet firewall policies to block unexpected inbound connections.

Customers at the enterprise tier can also use the IP access list feature on the control plane to limit which IP addresses can connect to the web UI or REST API — for example, to allow only VPN or office IPs.

Servers

In the data plane, Databricks clusters automatically run the latest hardened system image. Users cannot choose older (less secure) images or code. For AWS and Azure deployments, images are typically updated every two-to-four weeks. GCP is responsible for its system image.

Databricks runs scans for every release, including:

- System image scanning for vulnerabilities
- Container OS and library scanning
- Static and dynamic code scanning

Databricks code is peer reviewed by developers who have security training. Significant design documents go through comprehensive security reviews. Scans run fully authenticated, with all checks enabled, and issues are tracked against the timeline shown in this table.

Note that Databricks clusters are typically short-lived (often terminated after a job completes) and do not persist data after they terminate. Clusters typically share the same permission level (excluding high concurrency or Databricks SQL clusters, where more robust security controls are in place). Your code is launched in an unprivileged container to maintain system stability. This security design provides protection against persistent attackers and privilege escalation.

Severity	Remediation time
Critical	< 14 days
High	< 30 days
Medium	< 60 days
Low	When appropriate

Databricks access

Databricks access to your environment is limited to cloud service provider APIs for our automation and support access. Automated access allows the Databricks control plane to configure resources in your environment using the cloud service provider APIs. The specific APIs vary based on the cloud. For instance, an AWS cross-account IAM role, or Azure-owned automation or GKE automation do not grant access to your data sets (see the next section).

Databricks has a custom-built system that allows staff to fix issues or handle support requests — for example, when you open a support request and check the box authorizing access to your workspace. Access requires either a support ticket or engineering ticket tied expressly to your workspace and is limited to a subset of employees and for limited time periods. Additionally, if you have configured audit log delivery, the audit logs show the initial access event and the staff's actions.

Identity and access

Databricks supports robust ACLs and SCIM. AWS customers can configure SAML 2.0 and block non-SSO logins. Azure Databricks and Databricks on GCP automatically integrate with Microsoft Entra ID or GCP identity.

Databricks supports a variety of ways to enable users to access their data.

Examples include:

- The Table ACLs feature uses traditional SQL-based statements to manage access to data and enable fine-grained view-based access
- IAM instance profiles enable AWS clusters to assume an IAM role, so users of that cluster automatically access allowed resources without explicit credentials
- External storage can be mounted or accessed using a securely stored access key
- The Secrets API separates credentials from code when accessing external resources

Data security

Databricks provides encryption, isolation and auditing.

Databricks encryption capabilities are in place both at rest and in motion

For data-at-rest encryption:

- Control plane is encrypted
- Data plane supports local encryption
- Customers can use encrypted storage buckets
- Customers at some tiers can configure customer-managed keys for managed services

For data-in-motion encryption:

- Control plane \leftrightarrow data plane is encrypted
- Offers optional intra-cluster encryption
- Customer code can be written to avoid unencrypted services (e.g., FTP)

Customers can isolate users at multiple levels:

- **Workspace level:** Each team or department can use a separate workspace
- **Cluster level:** Cluster ACLs can restrict the users who can attach notebooks to a given cluster
- **High concurrency clusters:** Process isolation, JVM whitelisting and limited languages (SQL, Python) allow for the safe coexistence of users of different privilege levels, and is used with Table ACLs
- **Single-user cluster:** Users can create a private dedicated cluster

Activities of Databricks users are logged and can be delivered automatically to a cloud storage bucket. Customers can also monitor provisioning activities by monitoring cloud audit logs.

Compliance

Databricks supports the following compliance standards on our multi-tenant platform:

- SOC 2 Type II
- ISO 27001
- ISO 27017
- ISO 27018
- ISO 27701

Certain clouds support Databricks deployment options for FedRAMP High, HITRUST, HIPAA and PCI. Databricks Inc. and the Databricks Platform are also GDPR and CCPA ready.



[Learn more](#)

To learn more about Databricks security, visit the [Security and Trust Center](#)

CHAPTER

06

Instant compute
and serverless

What is serverless compute?

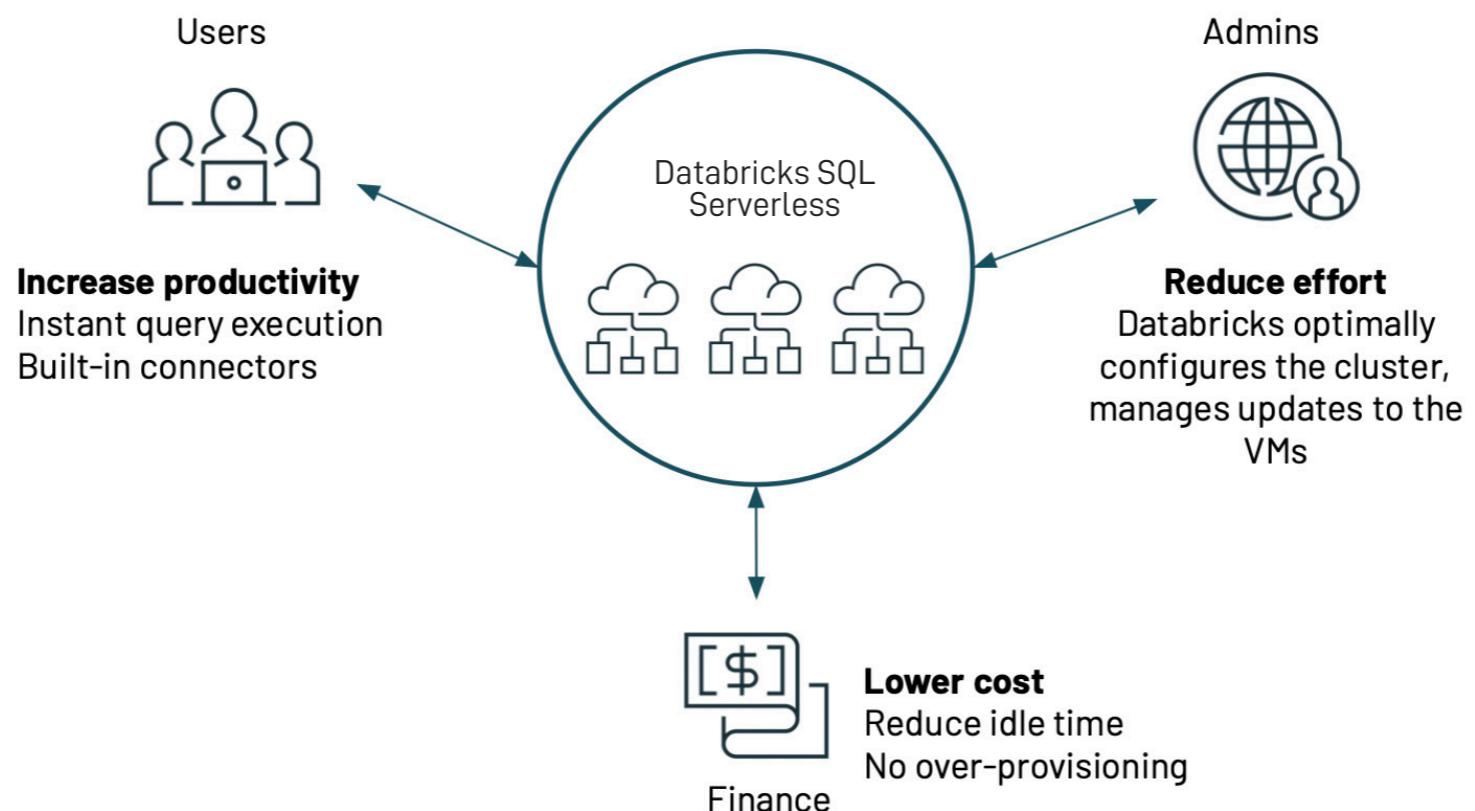
Serverless compute is a fully managed service where Databricks provisions and manages the compute layer on behalf of the customer in the Databricks cloud account instead of the customer account. As of the current release, serverless compute is supported for use with Databricks SQL, Delta Live Tables, and Notebooks.

Databricks SQL Serverless

For example, this is how serverless compute supports the capabilities for Databricks SQL, providing instant compute to users for their BI and SQL workloads, with minimal management required and capacity optimizations that can lower overall cost by 20%-40% on average. This makes it even easier for organizations to expand adoption of the lakehouse for business analysts who are looking to access the rich, real-time data sets of the lakehouse with simple and performant solution.

Databricks SQL Serverless is much easier to administer with Databricks taking on the responsibility of deploying, configuring and managing your cluster VMs. Databricks can transfer compute capacity to user queries typically in about 15 seconds — so you no longer need to wait for clusters to start up or scale out to run your queries.

Databricks SQL Serverless also has built-in connectors to your favorite tools such as Tableau, Power BI, Qlik, etc. These connectors use optimized JDBC/ODBC drivers for easy authentication support and high performance. And finally, you save on cost because you do not need to overprovision or pay for the idle capacity.



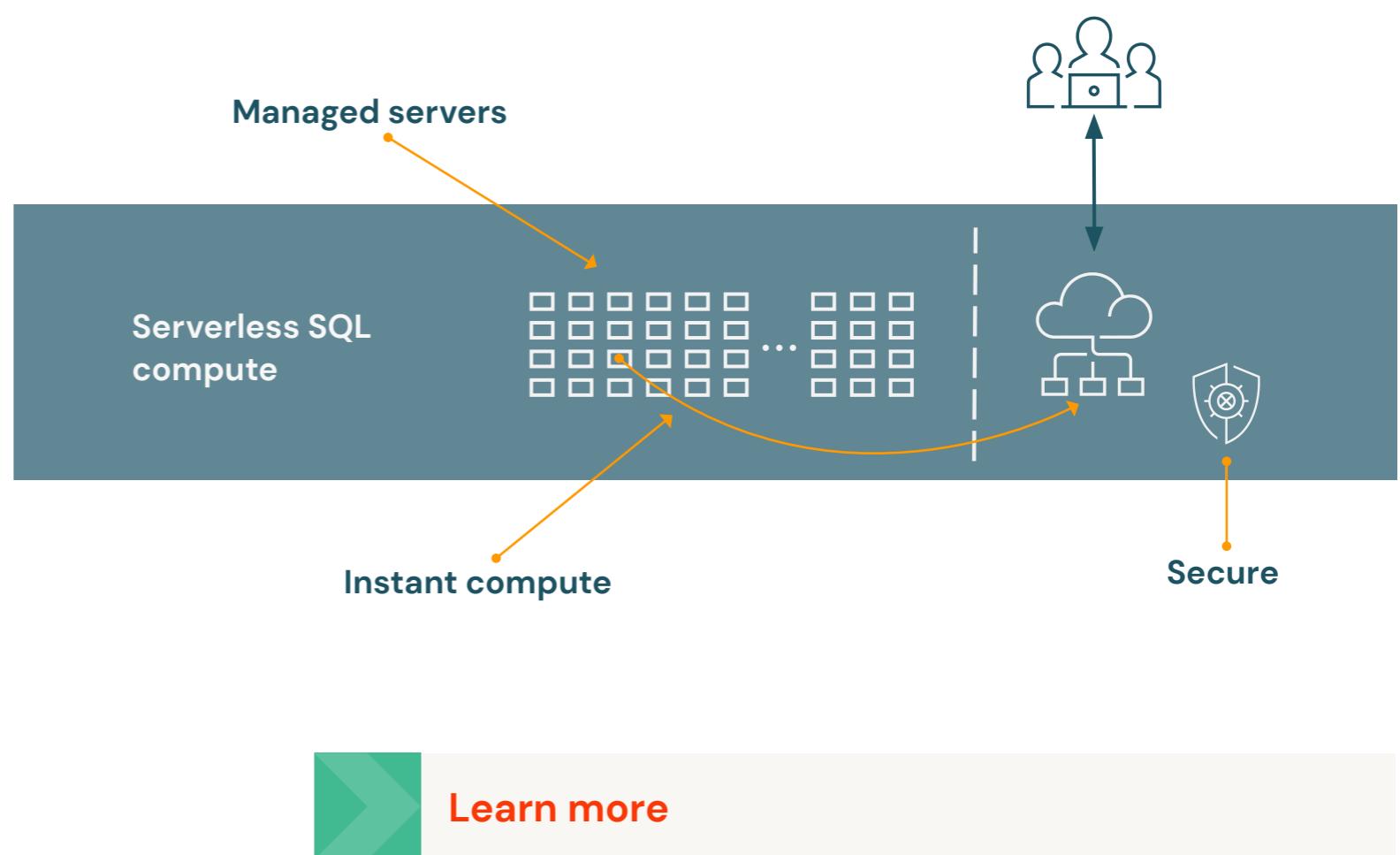
Inside Serverless

At the core of Serverless SQL is a compute platform that operates a pool of servers located in a Databricks' account, running Kubernetes containers that can be assigned to a user within seconds.

When many users are running reports or queries at the same time, the compute platform adds more servers to the cluster (again, within seconds) to handle the concurrent load. Databricks manages the entire configuration of the server and automatically performs the patching and upgrades as needed.

Each server is running a secure configuration and all processing is secured by three layers of isolation: The Kubernetes container hosting the runtime; the virtual machine (VM) hosting the container; and the virtual network for the workspace. Each layer is isolated to one workspace with no sharing or cross-network traffic allowed. The containers use hardened configurations, VMs are shut down and not reused, and network traffic is restricted to nodes in the same cluster.

Databricks Serverless



To learn more about Databricks Serverless, visit the [documentation page](#)

CHAPTER

07

Data warehousing

Data warehouses are not keeping up with today's world. The explosion of languages other than SQL and unstructured data, machine learning, IoT and streaming analytics are forcing organizations to adopt a bifurcated architecture of disjointed systems: Data warehouses for BI and data lakes for ML. While SQL is ubiquitous and known by millions of professionals, it has never been treated as a first-class citizen on data lakes, until the lakehouse.

The best data warehouse is a lakehouse

The Databricks Data Intelligence Platform provides a simplified multicloud and serverless architecture for your data warehousing workloads. Data warehousing on the lakehouse allows SQL analytics and BI at scale with a common governance model. Now you can ingest, transform and query all your data in-place — using your SQL and BI tools of choice — to deliver real-time business insights at the best price/performance. Built on open standards and APIs, the lakehouse provides the reliability, quality and performance that data lakes natively lack, and integrations with the ecosystem for maximum flexibility — no lock-in.

With data warehousing on the lakehouse, organizations can unify all analytics and simplify their architecture to enable their business with real-time business insights at the best price/performance.

Key benefits

Best price/performance

Lower costs, get the best price/performance and eliminate resource management overhead

On-premises data warehouses have reached their limits — they physically cannot scale to handle the growing volumes of data, and don't provide the elasticity customers need to respond to ever-changing business needs. Cloud data warehouses are a great alternative to on-premises data warehouses, providing greater scale and elasticity, but cloud costs for proprietary cloud data warehouses typically yield to an exponential cost increase following the growth of data volume.

The Databricks Data Intelligence Platform provides instant, elastic SQL serverless compute — decoupled from storage on cheap cloud object stores — and thousands of performance optimizations that can lower overall infrastructure costs by [an average of 40%](#). Databricks automatically determines instance types and configuration for the best price/performance — [up to 12x better than traditional cloud data warehouses](#) — and scale for high concurrency use cases.

Built-in governance

One source of truth and one unified governance layer across all data teams

Underpinned by Delta Lake, the Databricks Data Intelligence Platform simplifies your architecture by allowing you to establish one single copy of all your data for in-place analytics and ETL/ELT on your existing data lakes — no more data movements and copies in disjointed systems. Then, seamless integration with Databricks Unity Catalog lets you easily discover, secure and manage all your data with fine-grained governance, data lineage, and standard SQL.

Rich ecosystem

Ingest, transform and query all your data in-place with your favorite tools

Very few tools exist to conduct BI on data lakes. Generally, doing so has required data analysts to submit Spark jobs or use a developer interface. While these tools are common for data scientists, they require knowledge of languages and interfaces that are not traditionally part of a data analyst's tool set. As a result, the learning curve for an analyst to make use of a data lake is too high when well-established tools and methods already exist for data warehouses.

The Databricks Data Intelligence Platform works with your preferred tools like dbt, Fivetran, Power BI or Tableau, allowing analysts and analytical engineers to easily ingest, transform and query the most recent and complete data, without having to move it into a separate data warehouse. Additionally, it empowers every analyst across your organization to quickly and collaboratively find and share new insights with a built-in SQL editor, visualizations and dashboards.

Break down silos

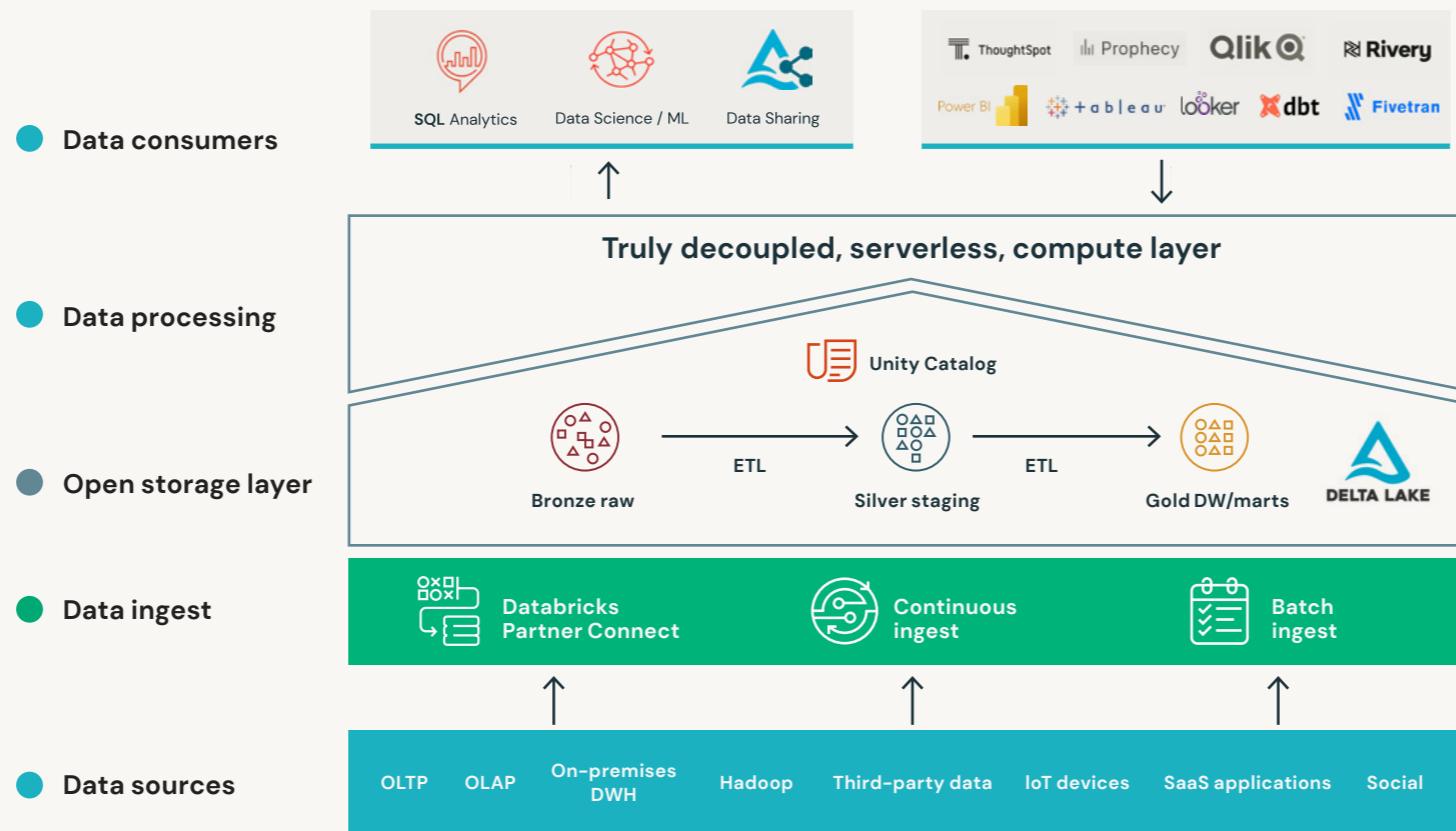
Accelerate time from raw to actionable data and go effortlessly from BI to ML

It is challenging for data engineering teams to enable analysts at the speed that the business requires. Data warehouses need data to be ingested and processed ahead of time before analysts can access and query it using BI tools. Because traditional data warehouses lack real-time processing and do not scale well for large ETL jobs, they create new data movements and bottlenecks for the data engineering team, and make it slow for analysts to access the latest data. And for advanced analytics (ML) applications, organizations

will need to manage an entirely different system than their SQL-only data warehouse, slowing down collaboration and innovation.

The Databricks Data Intelligence Platform provides the most complete end-to-end data warehousing solution for all your modern analytics needs, and more. Now you can empower data teams and business users to access the latest data faster for downstream real-time analytics and go effortlessly from BI to ML. Speed up the time from raw to actionable data at any scale — in batch and streaming. And go from descriptive to advanced analytics effortlessly to uncover new insights.

Data warehousing on Databricks



[Learn more](#)

[Try Databricks SQL for free](#)

[Databricks SQL Demo](#)

[Databricks SQL Data Warehousing Admin Demo](#)

[On-demand Webinar: Learn Databricks SQL From the Experts](#)

[eBook: Inner Workings of the Lakehouse for Analytics and BI](#)

CHAPTER

08

Data engineering

Organizations realize the value data plays as a strategic asset for growing revenues, improving the customer experience, operating efficiently or improving a product or service. Data is really the driver of all these initiatives. Nowadays, data is often streamed and ingested from hundreds of different data sources, sometimes acquired from a data exchange, cleaned in various ways with different orchestrated steps, versioned and shared for analytics and AI. And increasingly, data is being monetized.

Data teams rely on getting the right data at the right time for analytics, data science and machine learning, but often are faced with challenges meeting the needs of their initiatives for data engineering.

Why data engineering is hard

One of the biggest challenges is accessing and managing the increasingly complex data that lives across the organization. Most of the complexity arises with the explosion of data volumes and data types, with organizations amassing an estimated **80% of data that is unstructured and semi-structured**.

With this volume, managing data pipelines to transform and process data is slow and difficult, and increasingly expensive. And to top off the complexity, most businesses are putting an increased emphasis on multicloud environments which can be even more difficult to maintain.

Zhamak Dehghani, a principal technology consultant at Thoughtworks, wrote that data itself has become a product, and the challenging goal of the data engineer is to build and run the machinery that creates this high-fidelity data product all the way from ingestion to monetization.

Despite current technological advances data engineering remains difficult for several reasons:

Complex data ingestion methods

Data ingestion means retrieving batch and streaming data from various sources and in various formats. Ingesting data is hard and complex since you either need to use an always-running streaming platform like Apache Kafka or you need to be able to keep track of which files haven't been ingested yet. Data engineers are required to spend a lot of time hand-coding repetitive and error-prone data ingestion tasks.

Data engineering principles

These days, large operations teams are often just a memory of the past. Modern data engineering principles are based on agile software development methodologies. They apply the well-known "you build it, you run it" paradigm, use isolated development and production environments, CI/CD, and version control transformations that are pushed to production after validation. Tooling needs to support these principles.

Third-party tools

Data engineers are often required to run additional third-party tools for orchestration to automate tasks such as ELT/ETL or customer code in notebooks. Running third-party tools increases the operational overhead and decreases the reliability of the system.

Performance tuning

Finally, with all pipelines and workflows written, data engineers need to constantly focus on performance, tuning pipelines and architectures to meet SLAs. Tuning such architectures requires in-depth knowledge of the underlying architecture and constantly observing throughput parameters.

Most organizations are dealing with a complex landscape of data warehouses and data lakes these days. Each of those platforms has its own limitations, workloads, development languages and governance model.

Data engineering is an integrated part of the Data Intelligence Platform that empowers data practitioners everywhere to ingest, transform, and orchestrate data pipelines. Databricks allows both technical and non-technical users to easily deliver the right data, at the right time, at any scale.

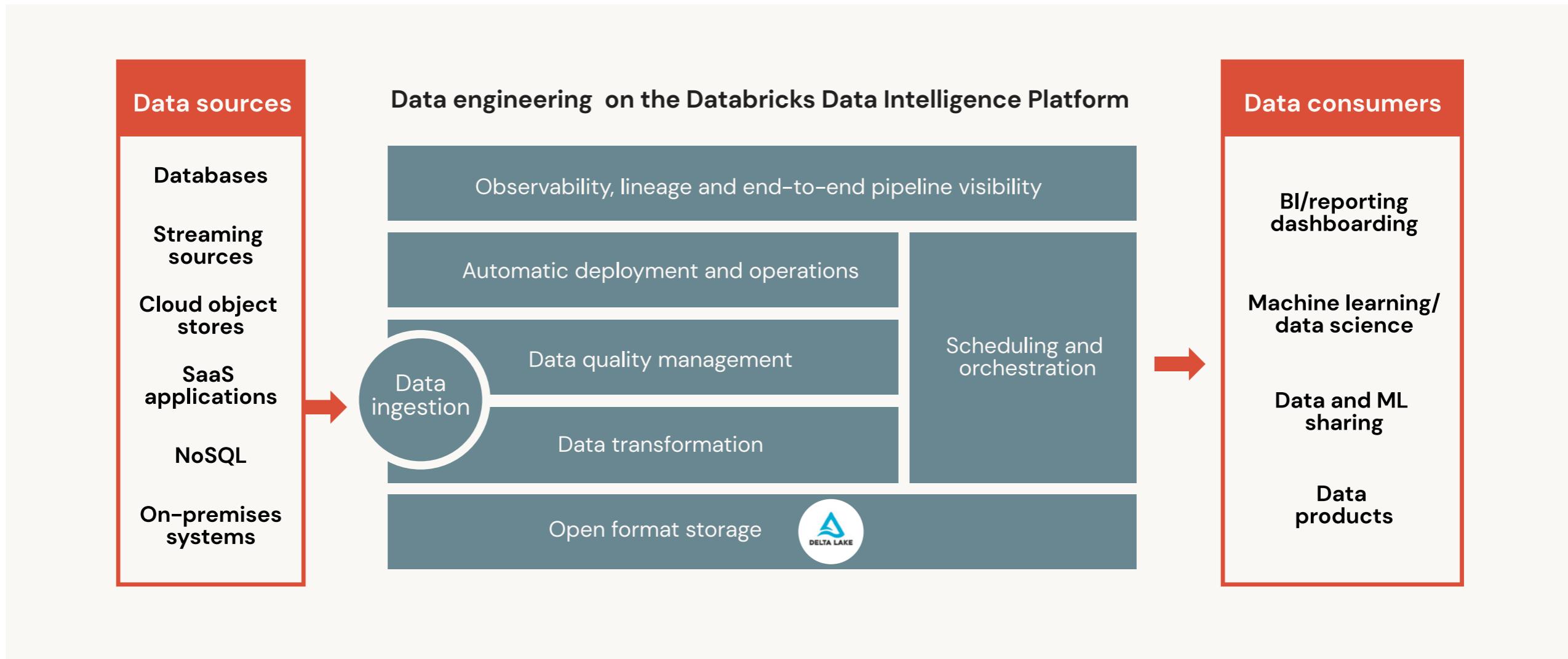
With the Databricks Data Intelligence Platform, data engineers have access to an end-to-end data engineering solution for ingesting, transforming, processing, scheduling and delivering data. The lakehouse architecture automates the complexity of building and maintaining pipelines and running ETL workloads directly on a data lake so data engineers can focus on quality and reliability to drive valuable insights.

Data engineering on the Databricks Data Intelligence Platform serves as a robust foundation for a broad spectrum of downstream use cases, including cutting-edge applications such as Gen AI and Large Language Models (LLMs). Leveraging Databricks Assistant, AI-assistant combining natural language processing (NLP) and intelligent automation, simplifies data engineer's workflows by performing the desired actions within the platform.

Databricks makes modern data engineering simple

There is no industry-wide definition of modern data engineering.
This should come close:

A **unified data platform** with **managed data ingestion**, schema detection, enforcement, and evolution, paired with **declarative, auto-scaling data flow** integrated with a **lakehouse native orchestrator** that supports all kinds of workflows.



Benefits of data engineering on the lakehouse

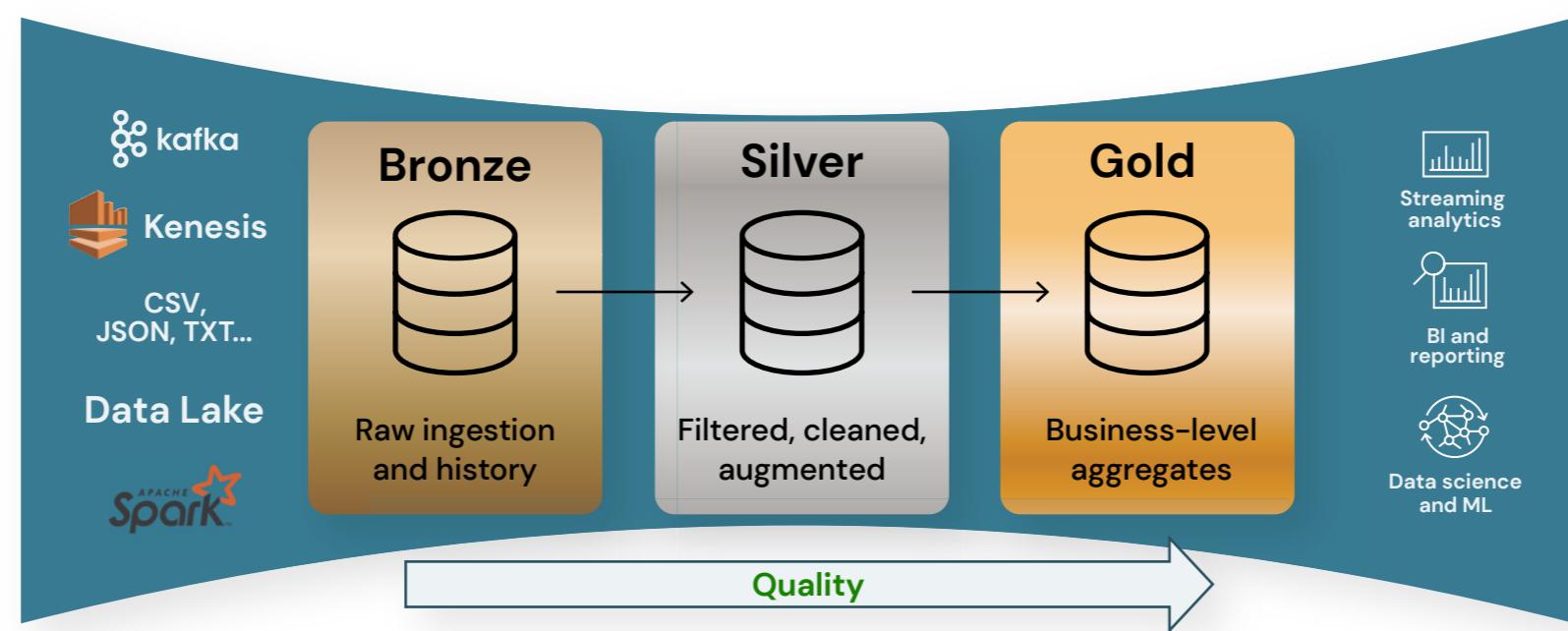
By simplifying and modernizing with the lakehouse architecture, data engineers gain an enterprise-grade and enterprise-ready approach to building data pipelines. The following are nine key differentiating capabilities that a data engineering solution team can enable with the Databricks Data Intelligence Platform:

- **Easy data ingestion:** With the ability to ingest petabytes of data, data engineers can enable fast, reliable, scalable and automatic data ingestion for analytics, data science or machine learning.
- **Automated ETL pipelines:** Data engineers can reduce development time and effort and focus on implementing business logic and data quality checks within the data pipeline using SQL or Python.
- **Data quality checks:** Improve data reliability throughout the data lakehouse so data teams can confidently trust the information for downstream initiatives with the ability to define data quality and automatically address errors.
- **Batch and streaming:** Allow data engineers to set tunable data latency with cost controls without having to know complex stream processing and implement recovery logic.
- **Automatic recovery:** Handle transient errors and use automatic recovery for most common error conditions that can occur during the operation of a pipeline with fast, scalable fault-tolerance.
- **Data pipeline observability:** Monitor overall data pipeline estate status from a dataflow graph dashboard and visually track end-to-end pipeline health for performance, quality, status and latency.
- **Simplified operations:** Ensure reliable and predictable delivery of data for analytics and machine learning use cases by enabling easy and automatic data pipeline deployments into production or roll back pipelines and minimize downtime.
- **Scheduling and orchestration:** Simple, clear and reliable orchestration of data processing tasks for data and machine learning pipelines with the ability to run multiple non-interactive tasks as a directed acyclic graph (DAG) on a Databricks compute cluster.
- **Natural Language Assistant:** Databricks assistant makes data engineers more productive with Text2SQL, document summary, code completion and generation, code explanation, and debugging.

Data engineering is all about data quality

The goal of modern data engineering is to distill data with a quality that is fit for downstream analytics and AI. Within the lakehouse, data quality is achieved on three different levels.

1. On a **technical level**, data quality is guaranteed by enforcing and evolving schemas for data storage and ingestion.
2. On an **architectural level**, data quality is often achieved by implementing the medallion architecture. A medallion architecture is a data design pattern used to logically organize data in a **lakehouse** with the goal of incrementally and progressively improving the structure and quality of data as it flows through each layer of the architecture, e.g., from Bronze to Silver to Gold layer tables.
3. The **Databricks Unity Catalog** comes with robust data quality management with built-in quality controls, testing, monitoring and enforcement to ensure accurate and useful data is available for downstream BI, analytics and machine learning workloads.



Data ingestion

With the Databricks Data Intelligence Platform, data engineers can build robust hyper-scale ingestion pipelines in streaming and batch mode. They can incrementally process new files as they land on cloud storage — with no need to manage state information — in scheduled or continuous jobs.

Data engineers can efficiently track new files (with the ability to scale to billions of files) without having to list them in a directory. Databricks automatically infers the schema from the source data and evolves it as the data loads into the Delta Lake lakehouse. Efforts continue with enhancing and supporting Auto Loader, our powerful data ingestion tool for the lakehouse.

What is Auto Loader?

Have you ever imagined that ingesting data could become as easy as dropping a file into a folder? Welcome to Databricks Auto Loader.

[Auto Loader](#) is an optimized data ingestion tool that incrementally and efficiently processes new data files as they arrive in the cloud storage built into the Databricks Data Intelligence Platform. Auto Loader can detect and enforce the schema of your data and, therefore, guarantee data quality. New files or files that have been changed since the last time new data was processed are identified automatically and ingested. Non-compliant data sets are quarantined into rescue data columns. You can use the [trigger once] option with Auto Loader to turn it into a job that turns itself off.

Ingestion for data analysts: COPY INTO

Ingestion also got much easier for data analysts and analytics engineers working with Databricks SQL. [COPY INTO](#) is a simple SQL command that follows the lake-first approach and loads data from a folder location into a Delta Lake table. COPY INTO can be scheduled and called by a job repeatedly. When run, only new files from the source location will be processed.

Data transformation

Turning SQL queries into production ETL pipelines typically involves a lot of tedious, complicated operational work. Even at a small scale, the majority of a data practitioner's time is spent on tooling and managing infrastructure.

Although the medallion architecture is an established and reliable pattern for improving data quality, the implementation of this pattern is challenging for many data engineering teams.

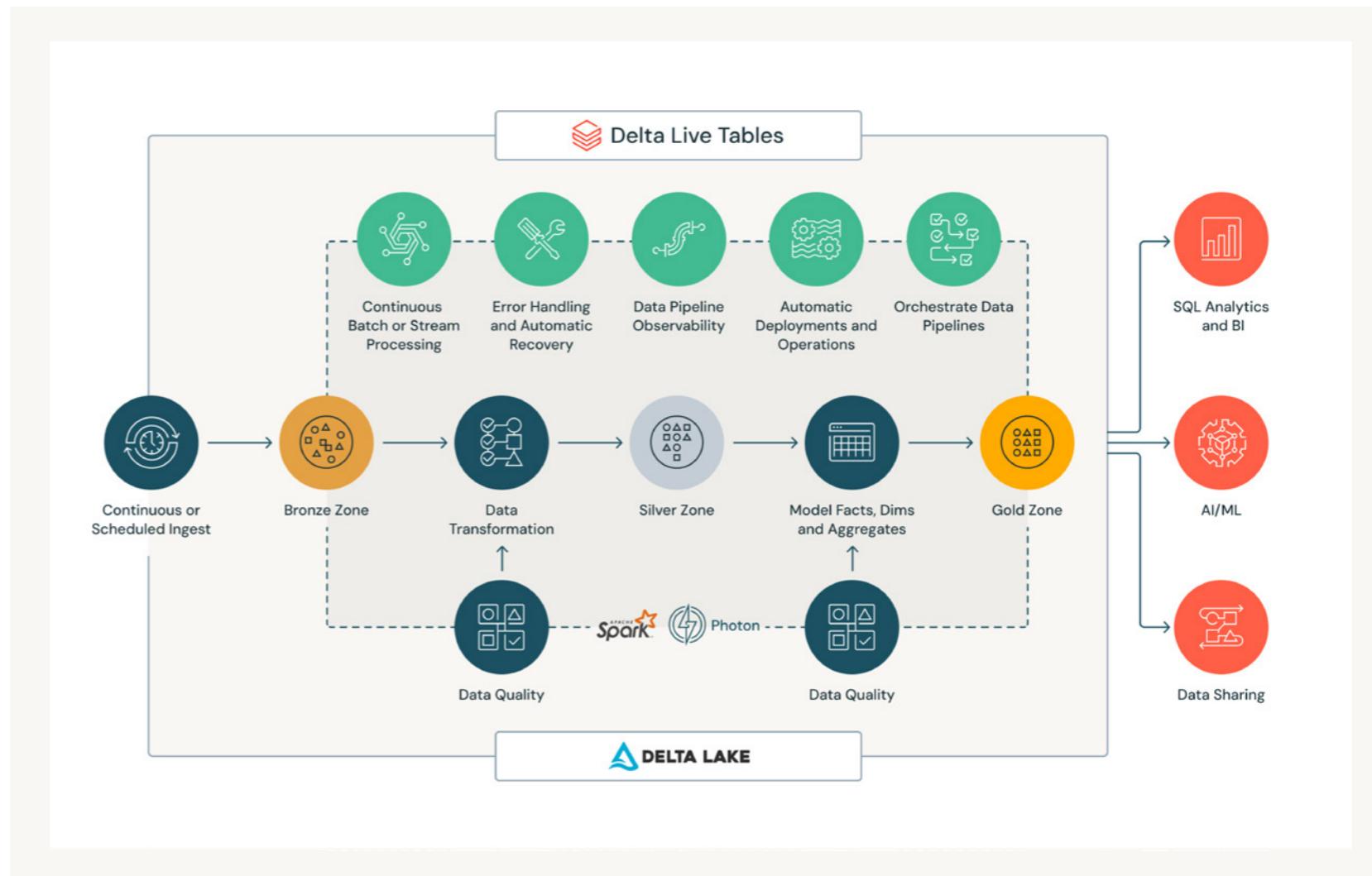
While hand-coding the medallion architecture was hard for data engineers, creating data pipelines was outright impossible for data analysts not being able to code with Spark Structured Streaming in Scala or Python.

Even at a small scale, most data engineering time is spent on tooling and managing infrastructure rather than transformation. Auto-scaling, observability and governance are difficult to implement and, as a result, often left out of the solution entirely.

What is Delta Live Tables?

Delta Live Tables (DLT) is a declarative ETL framework that helps data teams simplify and make ETL cost-effective in streaming and batch. Simply define the transformations you want to perform on your data and let DLT pipelines automatically handle task orchestration, cluster management, monitoring, and data quality and error management. Engineers are able to **treat their data as code** and apply modern software engineering best practices like testing, error-handling, monitoring and documentation to deploy reliable pipelines at scale. DLT fully supports both Python and SQL and is tailored to work with both streaming and batch workloads.

With DLT you write a Delta Live Table in a SQL notebook, create a pipeline under Workflows and simply click [Start].



DLT reduces the implementation time by accelerating development and automating complex operational tasks. Since DLT can use plain SQL, it also enables data analysts to create production pipelines and turns them into the often discussed “analytics engineer.” At runtime, DLT speeds up pipeline execution applied with Photon.

Software engineering principles are applied for data engineering to foster the idea of treating your data as code. Your data is the sole source of truth for what is going on inside your business.

Beyond just the transformations, there are many things that should be included in the code that define your data. Declaratively express entire data flows in SQL or Python. Natively enable modern software engineering best practices like separate development and production environments, the ability to easily test before deploying, deploy and manage environments using parameterization, unit testing and documentation.

DLT also automatically scales compute, providing the option to set the minimum and maximum number of instances and let DLT size up the cluster according to cluster utilization. In addition, tasks like orchestration, error handling and recovery, and performance optimization are all handled automatically.

```
CREATE LIVE TABLE raw_data as SELECT * FROM json.`
CREATE LIVE TABLE clean_data as SELECT ... FROM LIVE.raw_data
```



Repos



Unity Catalog*



Databricks Workflows

Delta **LIVE** Tables

Full refresh



Dependency management



Expectations



Incremental computation*



Checkpointing and retries

*Coming soon

Expectations in the code help prevent bad data from flowing into tables, track data quality over time, and provide tools to troubleshoot bad data with granular pipeline observability. This enables a high-fidelity lineage diagram of your pipeline to track dependencies and aggregate data quality metrics across all your pipelines.

Unlike other products that force you to deal with streaming and batch workloads separately, DLT supports any type of data workload with a single API so data engineers and analysts alike can build cloud-scale data pipelines faster without the need for advanced data engineering skills.

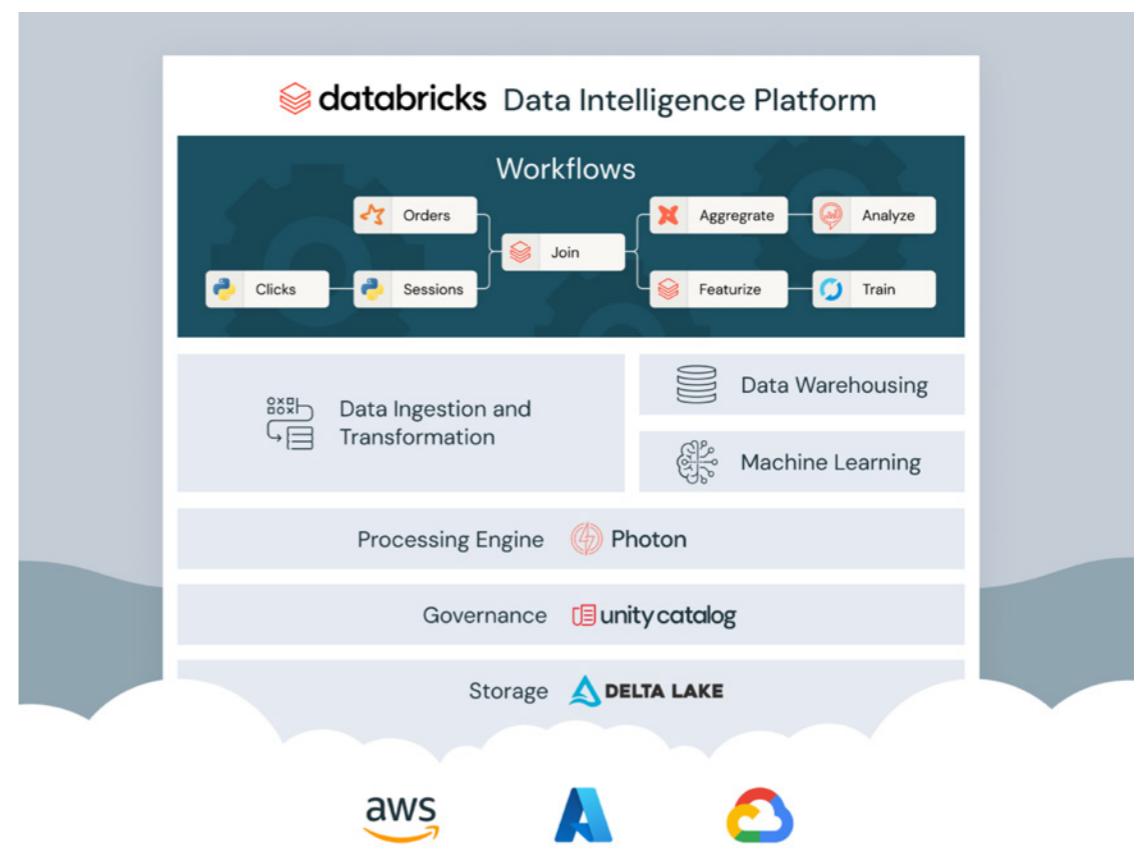
Data orchestration

The lakehouse makes it much easier for businesses to undertake ambitious data and machine learning (ML) initiatives. However, orchestrating and managing end-to-end production workflows remains a bottleneck for most organizations, relying on external tools or cloud-specific solutions that are not part of their lakehouse architecture. Tools that decouple task orchestration from the underlying data processing platform reduce the overall reliability of their production workloads, limit observability, and increase complexity for end users.

What is Databricks Workflows?

Databricks **Workflows** is the first fully managed and integrated lakehouse **orchestration** service that allows data teams to build reliable workflows on any cloud.

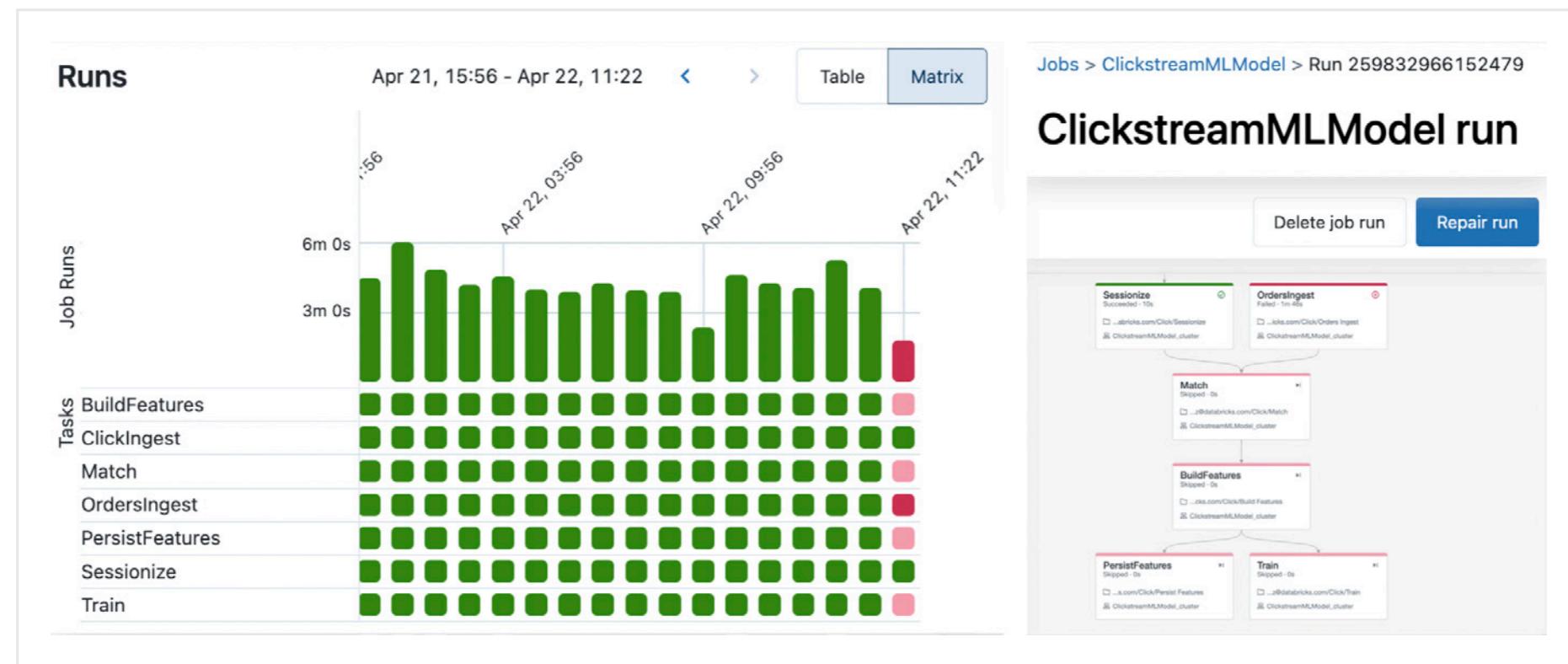
Workflows lets you orchestrate data flow pipelines (written in DLT or dbt), as well as machine learning pipelines, or any other tasks such as notebooks or Python wheels. Since Databricks Workflows has serverless compute, fully managed and reliable it eliminates operational overhead for data engineers, enabling them to focus on your workflows not on managing your infrastructure. It provides an easy point-and-click authoring experience for all your data teams, not just those with specialized skills. Deep integration with the underlying lakehouse architecture ensures you will create and run reliable production workloads on any cloud while providing deep and centralized monitoring with simplicity for end users.



Serverless Workflows provides a hands-off approach, fast start up times (<60s), automatic failover, and cost optimized mode.

Databricks Workflows' deep integration with the Databricks Data Intelligence Platform can best be seen with its monitoring and observability features. The matrix view in the following graphic shows a history of runs for a job. Failed tasks are marked in red. A failed job can be repaired and rerun with the click of a button. Rerunning a failed task detects and triggers the execution of all dependent tasks.

You can create workflows with the UI, but also through the Databricks Workflows API, or with external orchestrators such as Apache Airflow. Even if you are using an external orchestrator, Databricks Workflows' monitoring acts as a single pane of glass that includes externally triggered workflows.

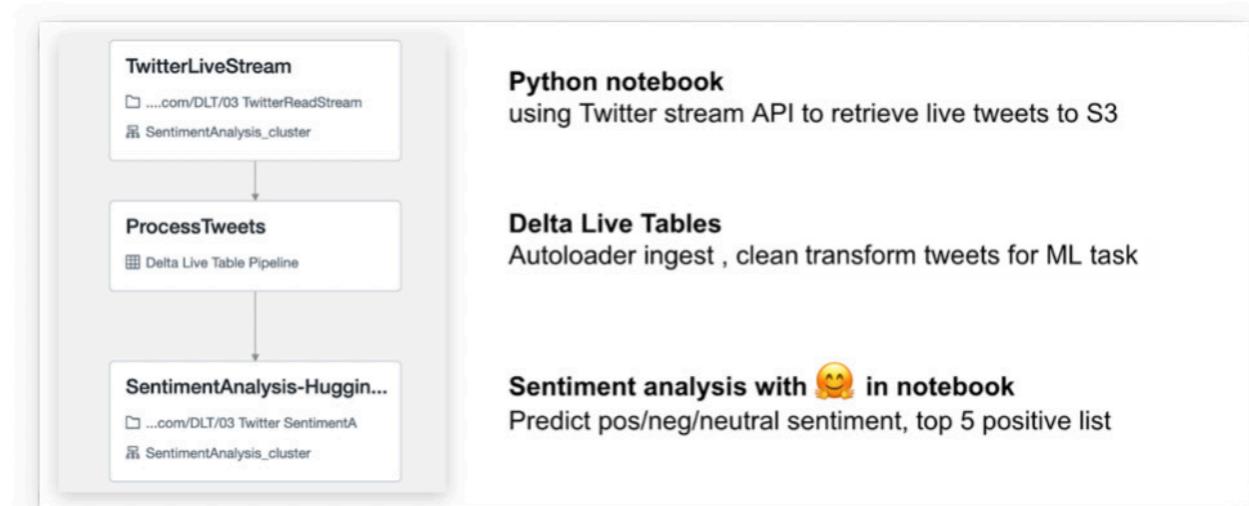


Orchestrate anything

Remember that DLT is one of many task types for Databricks Workflows. This is where the managed data flow pipelines with DLT tie together with the easy point-and-click authoring experience of Databricks Workflows.

In the following example, you can see an end-to-end workflow built with customers in a workshop: Data is streamed from Twitter according to search terms, then ingested with Auto Loader using automatic schema detection and enforcement. In the next step, the data is cleaned and transformed with Delta Live table pipelines written in SQL, and finally run through a pre-trained BERT language model from Hugging Face for sentiment analysis of the tweets. Different task types for ingest, cleanse/transform and ML are combined in a single workflow.

Using Workflows, these tasks can be scheduled to provide a daily overview of social media coverage and customer sentiment for a business. After streaming tweets with filtering for keywords such as “data engineering,” “lakehouse” and “Delta Lake,” we curated a list of those tweets that were classified as positive with the highest probability score.



[Learn more](#)

[Data Engineering on the Databricks Data Intelligence Platform](#)

[Big Book of Data Engineering](#)

[Delta Live Tables](#)

[Databricks Workflows](#)

CHAPTER

09

Data streaming?

There are two types of data processing: batch processing and streaming processing.

Batch processing refers to the discontinuous, periodic processing of data that has been stored for a period of time. For example, an organization may need to run weekly reports on a set of predictable transaction data. There is no need for this data to be streaming — it can be processed on a weekly basis.

Streaming processing, on the other hand, refers to unbounded processing of data as it arrives.

In a wide variety of cases, an organization might find it useful to leverage streaming data. Here are some common examples:

- **Retail:** Real-time inventory updates help support business activities, such as inventory and pricing optimization and optimization of the supply chain, logistics and just-in-time delivery.
- **Smart energy:** Smart meter monitoring in real time allows for smart electricity pricing models and connection with renewable energy sources to optimize power generation and distribution.
- **Preventative maintenance:** By reducing unplanned outages and unnecessary site and maintenance visits, real-time streaming analytics can lower operational and equipment costs.
- **Industrial automation:** Manufacturers can use streaming and predictive analytics to improve production processes and product quality, including setting up automated alerts.
- **Healthcare:** To optimize care recommendations, real-time data allows for the integration of various smart sensors to monitor patient condition, medication levels and even recovery speed.
- **Financial institutions:** Firms can conduct real-time analysis of transactions to detect fraudulent transactions and send alerts. They can use fraud analytics to identify patterns and feed data into machine learning algorithms.

Regardless of specific use cases, the central tenet of streaming data is that it gives organizations the opportunity to leverage the freshest possible insights for better decision-making and more optimized customer experiences.

Data Streaming Challenges

However, getting value from streaming data can be a tricky practice. While most data today can be considered streaming data, organizations are overwhelmed by the need to access, process and analyze the volume, speed and variety of this data moving through their platforms. To keep pace with innovation, they must quickly make sense of data streams decisively, consistently and in real time.

Three common technical challenges organizations experience with implementing real-time data streaming include:

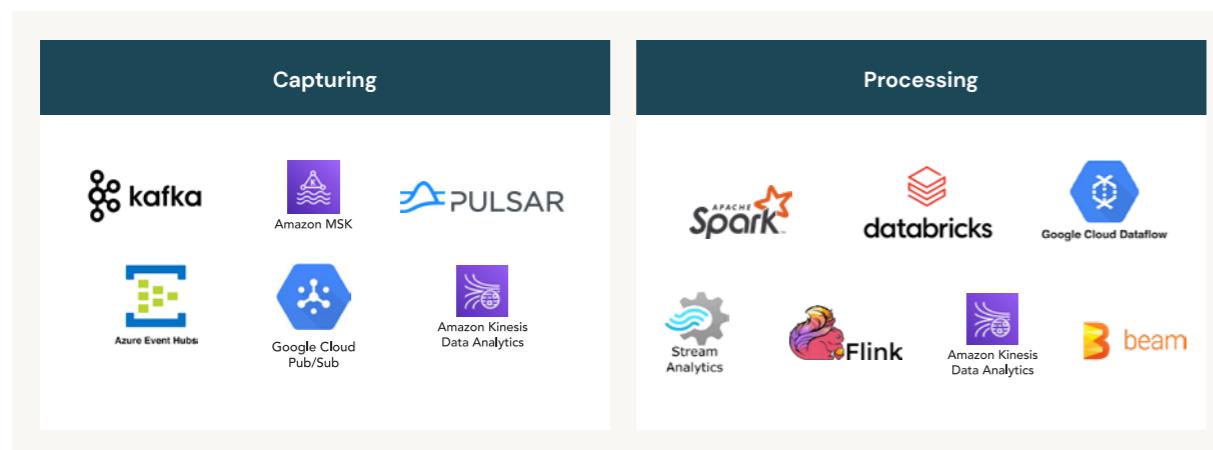
- **Specialized APIs and language skills:** Data practitioners encounter barriers to adopting streaming skillsets because there are new languages, APIs and tools to learn.
- **Operational complexity:** To implement data streaming at scale, data teams need to integrate and manage streaming-specific tools with their other cloud services. They also have to manually build complex operational tooling to help these systems recover from failure, restart workloads without reprocessing data, optimize performance, scale the underlying infrastructure, and so on.
- **Incompatible governance models:** Different governance and security models across real-time and historical data platforms makes it difficult to provide the right access to the right users, see the end-to-end data lineage, and/or meet compliance requirements.

Data streaming architecture

Before addressing these challenges head-on, it may help to take a step back and discuss the ingredients of a streaming data pipeline. Then, we will explain how the Databricks Data Intelligence Platform operates within this context to address the aforementioned challenges.

Every application of streaming data requires a pipeline that brings the data from its origin point — whether sensors, IoT devices or database transactions — to its final destination.

In building this pipeline, streaming architectures typically employ two layers. First, streaming capture systems **capture** and temporarily store streaming data for processing. Sometimes these systems are also called messaging systems or messaging buses. These systems are optimized for small payloads and high frequency inputs/outputs. Second, streaming **processing** systems continuously process data from streaming capture systems and other storage systems.



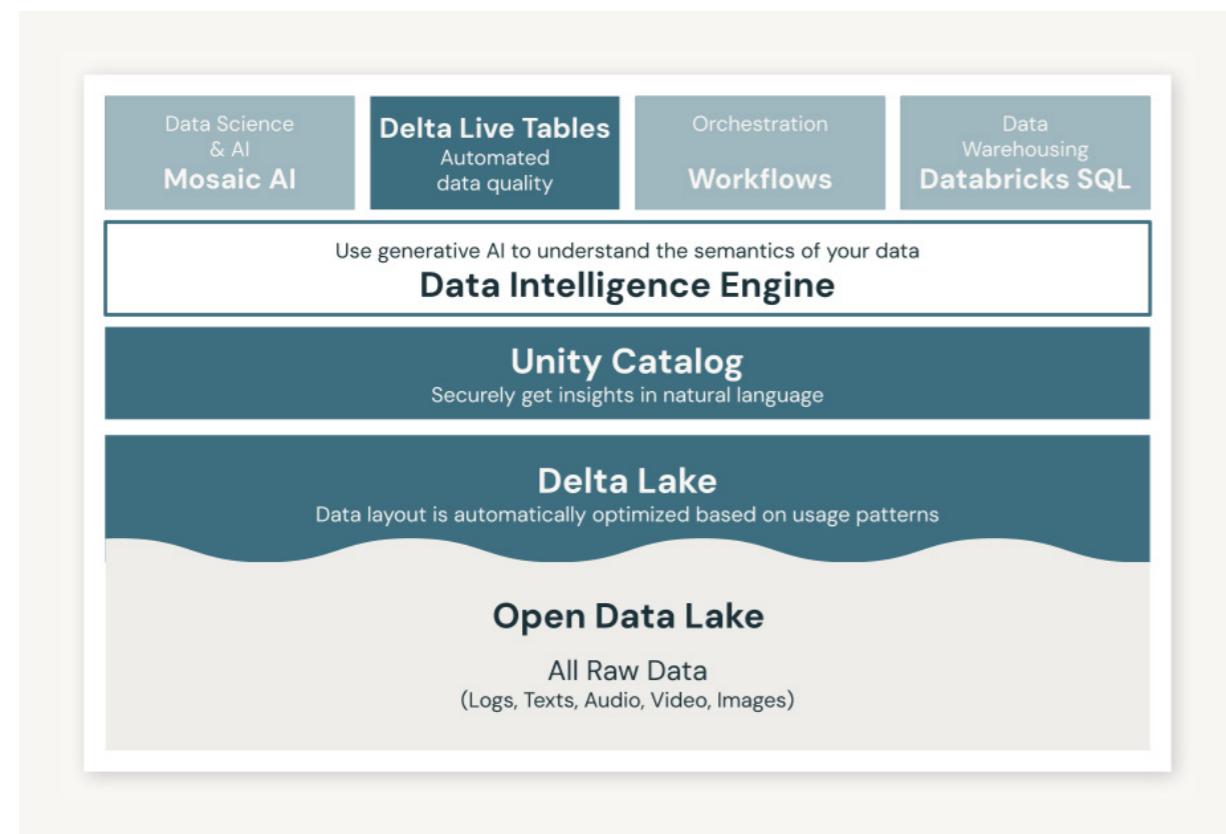
It may help to think of a simplified streaming pipeline according to the following seven phases:

1. Data is continuously generated at origin points
2. The generated data is captured from those origin points by a capture system like Apache Kafka (with limited retention)
3. **The captured data is extracted and incrementally ingested to a processing platform like Databricks; data is ingested exactly once and stored permanently, even if this step is rerun**
4. **The ingested data is converted into a workable format**
5. **The formatted data is cleansed, transformed and joined in a number of pipeline steps**
6. **The transformed data is processed downstream through analysis or ML modeling**
7. The resulting analysis or model is used for some sort of practical application, which may be anything from basic reporting to an event-driven software application

You will notice four of the steps in this list are in boldface. This is because the lakehouse architecture is specifically designed to optimize this part of the pipeline. Uniquely, the Databricks Data Intelligence Platform can ingest, transform, analyze and model on streaming data *alongside* batch-processed data. It can accommodate both structured *and* unstructured data. It is here that the value of unifying the best pieces of data lakes and data warehouses really shines for complex enterprise use cases.

Data Streaming on the Databricks Data Intelligence Platform

Now let's zoom in a bit and see how the Databricks Data Intelligence Platform addresses each part of the pipeline mentioned above.



Streaming data ingestion and transformation begins with continuously and incrementally collecting raw data from streaming sources through a feature called Auto Loader. Once the data is ingested, it can be transformed from raw, messy data into clean, fresh, reliable data appropriate for downstream analytics, ML or applications. **Delta Live Tables (DLT)** makes it easy to build and manage these data pipelines while automatically taking care of infrastructure management and scaling, data quality, error testing and other administrative tasks. DLT is a high-level abstraction built on Spark Structured Streaming, a scalable and fault-tolerant stream processing engine.

Real-time analytics refers to the downstream analytical application of streaming data. With fresher data streaming into SQL analytics or BI reporting, more actionable insights can be achieved, resulting in better business outcomes.

Real-time ML involves deploying ML models in a streaming mode. This deployment is supported with structured streaming for continuous inference from a live data stream. Like real-time analytics, real-time ML is a downstream impact of streaming data, but for different business use cases (i.e., AI instead of BI). Real-time modeling has many benefits, including more accurate predictions about the future.

Real-time applications process data directly from streaming pipelines and trigger programmatic actions, such as displaying a relevant ad, updating the price on a pricing page, stopping a fraudulent transaction, etc. There typically is no human-in-the-loop for such applications.

Databricks Data Intelligence Platform differentiators

Understanding what the lakehouse architecture provides is one thing, but it is useful to understand how Databricks uniquely approaches the common challenges mentioned earlier around working with streaming data.

Databricks empowers unified data teams. Data engineers, data scientists and analysts can easily build streaming data workloads with the languages and tools they already know and the APIs they already use.

Databricks simplifies development and operations. Organizations can focus on getting value from data by reducing complexity and automating much of the production aspects associated with building and maintaining real-time data workloads.

Databricks is one platform for streaming and batch data. Organizations can eliminate data silos, centralize security and governance models, and provide complete support for all their real-time use cases under one roof — the roof of the lakehouse.

Finally — and perhaps most important — Delta Lake, the core of the [Databricks Data Intelligence Platform](#), was built for streaming from the ground up. Delta Lake is deeply integrated with Spark Structured Streaming and overcomes many of the limitations typically associated with streaming systems and files.

In summary, the Databricks Data Intelligence Platform dramatically simplifies data streaming to deliver real-time analytics, machine learning and applications on one platform. And, that platform is built on a foundation with streaming at its core. This means organizations of all sizes can use their data in motion and make more informed decisions faster than ever.

See why customers love streaming on the Databricks Data Intelligence Platform with these resources.



Learn more

[Data Streaming Webpage](#)

[Project Lightspeed: Faster and Simpler Stream Processing With Apache Spark](#)

[Structured Streaming Documentation](#)

[Streaming — Getting Started With Apache Spark on Databricks](#)

CHAPTER

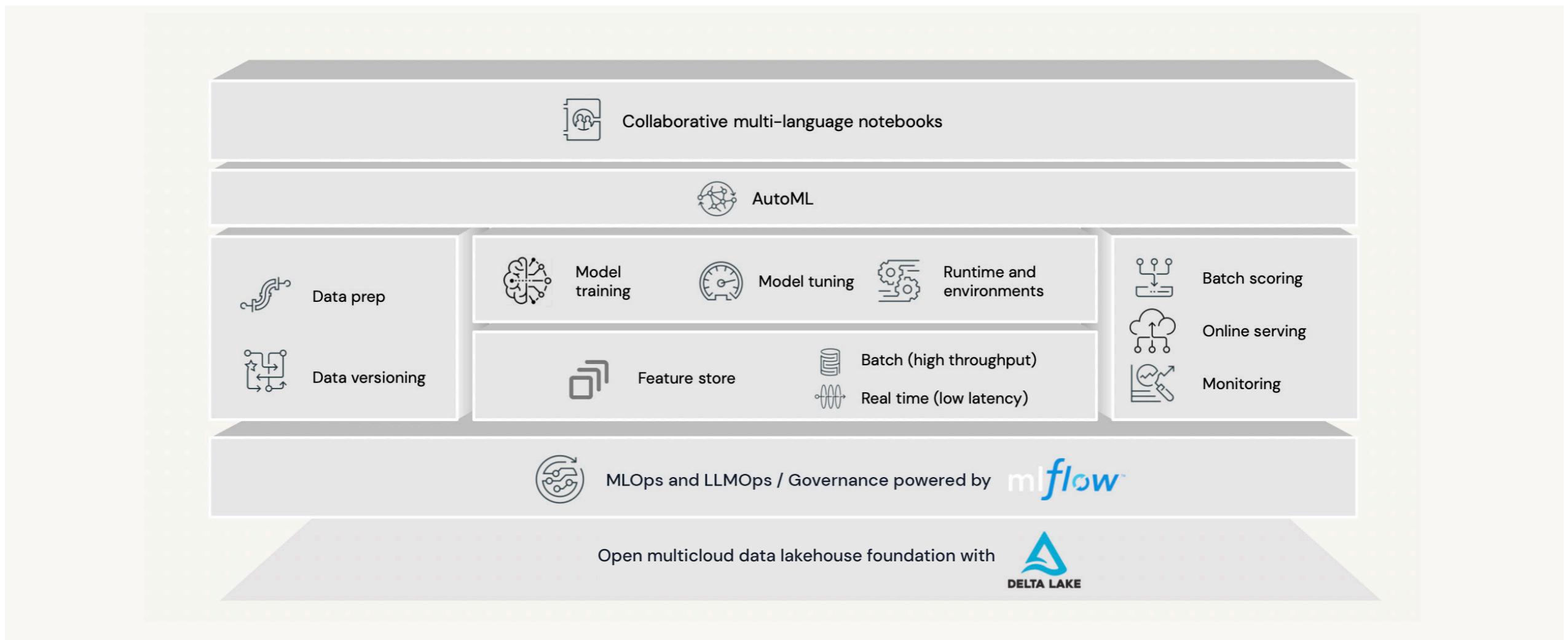
10

Data science, AI and machine learning

While most companies are aware of the potential benefits of applying machine learning and AI, realizing these potentials can often be quite challenging for those brave enough to take the leap. Some of the largest hurdles come from siloed/disparate data systems, complex experimentation environments, and getting models served in a production setting.

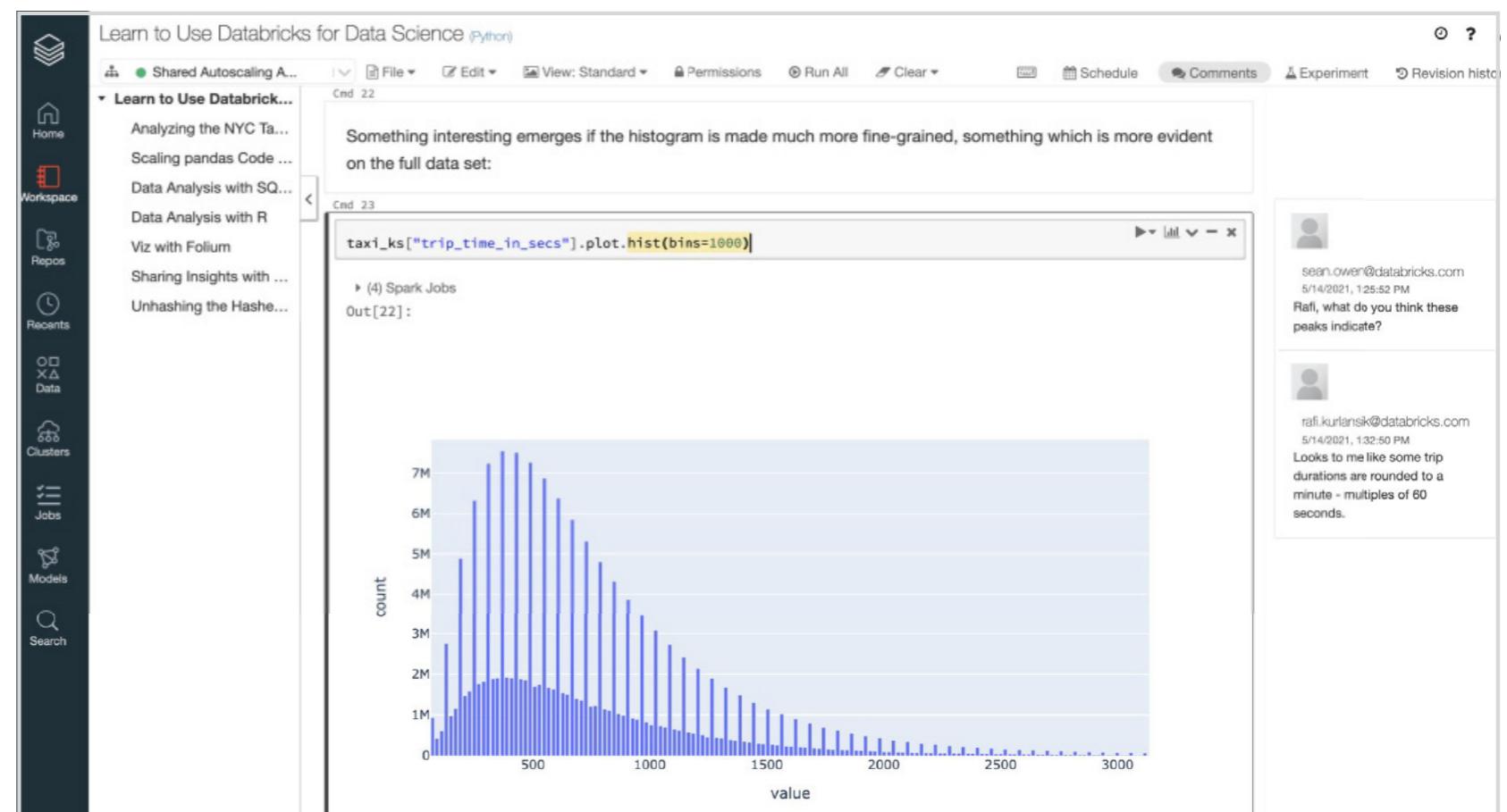
Fortunately, the Databricks Data Intelligence Platform provides a helping hand and lets you use data to derive innovative insights, build powerful predictive and generative AI models, and enable data scientists, ML engineers, and developers of all kinds to create within the space of machine learning and AI.

Databricks Machine Learning



Exploratory data analysis

With all the data in one place, data is easily explored and visualized from within the notebook-style experience that provides support for various languages (R, SQL, Python and Scala) as well as built-in visualizations and dashboards. Confidently and securely share code with co-authoring, commenting, automatic versioning, Git integrations and role-based access controls. The platform provides laptop-like simplicity at production-ready scale.



Model creation and management

From data ingestion to ML model and LLM training and tuning, all the way through to production model serving and versioning, Mosaic AI brings the tools needed to simplify those tasks.

Get right into experimenting with the Databricks ML runtimes, optimized and preconfigured to include most popular libraries like scikit-learn, XGBoost and more. Also, experiment with the foundation models on Databricks like DBRX, Llama 2, Mistral, or foundation models hosted elsewhere like GPT-4, Claude 3, Cohere, and Stable Diffusion. Massively scale thanks to built-in support for distributed training and hardware acceleration with GPUs.

From within the runtimes, you can track model training sessions, package and reuse models easily with [MLflow](#), an open source machine learning platform created by Databricks and included as a managed service within Mosaic AI. It provides a centralized location from which to manage models and package code in an easily reusable way.

Training these models often involves the use of features housed in a centralized feature store.

Fortunately, Databricks has a built-in feature store that allows you to create new features, explore and re-use existing features, select features for training and scoring machine learning models, and publish features to low-latency online stores for real-time inference.

If you are looking to get a head start, [AutoML](#) allows for low to no-code experimentation by pointing to your data set and automatically training models and tuning hyperparameters to save both novice

and advanced users precious time in the machine learning process.

AutoML will also report back metrics related to the model training results as well as the code needed to repeat the training already custom-tailored to your data set. This glass box approach ensures that you are never trapped or suffer from vendor lock-in.

In that regard, Mosaic AI supports the industry's widest range of data tools, development environments, and a thriving ISV ecosystem so you can make your workspace your own and put out your best work.

Compute platform

Any ML workload optimized and accelerated



Databricks Machine Learning Runtime

- Optimized and preconfigured ML frameworks
- Turnkey distribution ML
- Built-in AutoML
- GPU support out of the box

Built-in ML frameworks and model explainability



Built-in support for AutoML and hyperparameter tuning



Built-in support for distributed training



Built-in support for hardware accelerators

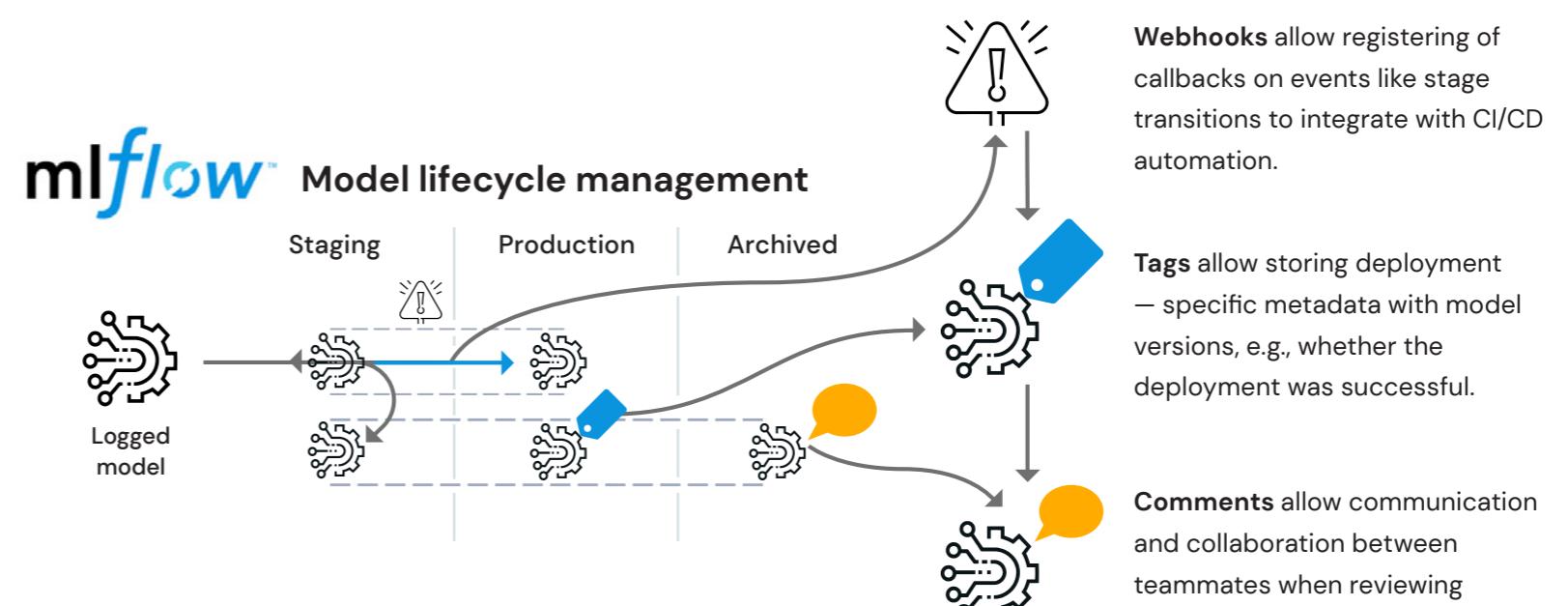


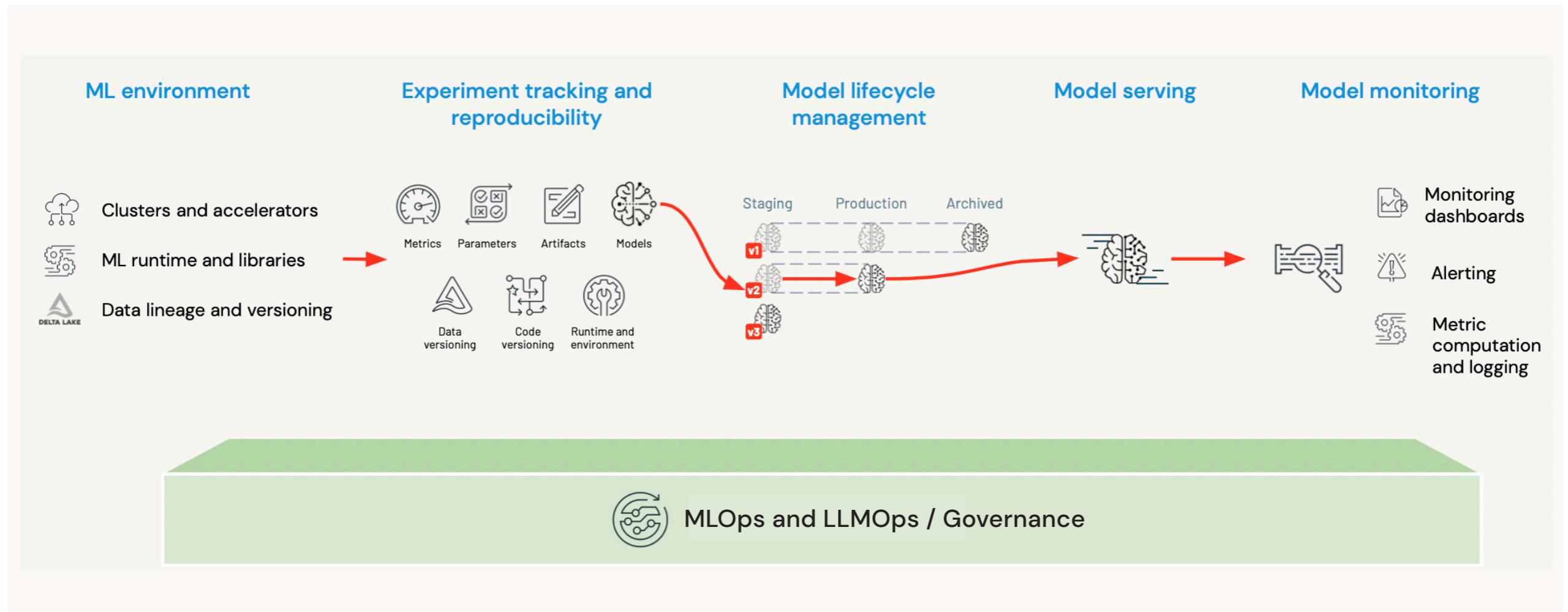
Deploy your models to production

Exploring and creating your machine learning and large language models typically represent only part of the task. Once the models exist and perform well, they must become part of a pipeline that keeps models updated, monitored and available for use by others.

Databricks can help here by providing a world-class experience for model versioning, monitoring and serving within the same platform that you can use to generate the models themselves. This means you can make all your ML and GenAI pipelines in the same place, monitor them for drift, retrain them with new data, and promote and serve them easily and at scale.

Throughout the ML lifecycle, rest assured knowing that lineage and governance are being tracked the entire way. This means regulatory compliance and security woes are significantly reduced, potentially saving costly issues down the road.



 Learn more[Databricks Machine Learning](#)[Databricks Data Science](#)[Databricks ML Runtime Documentation](#)

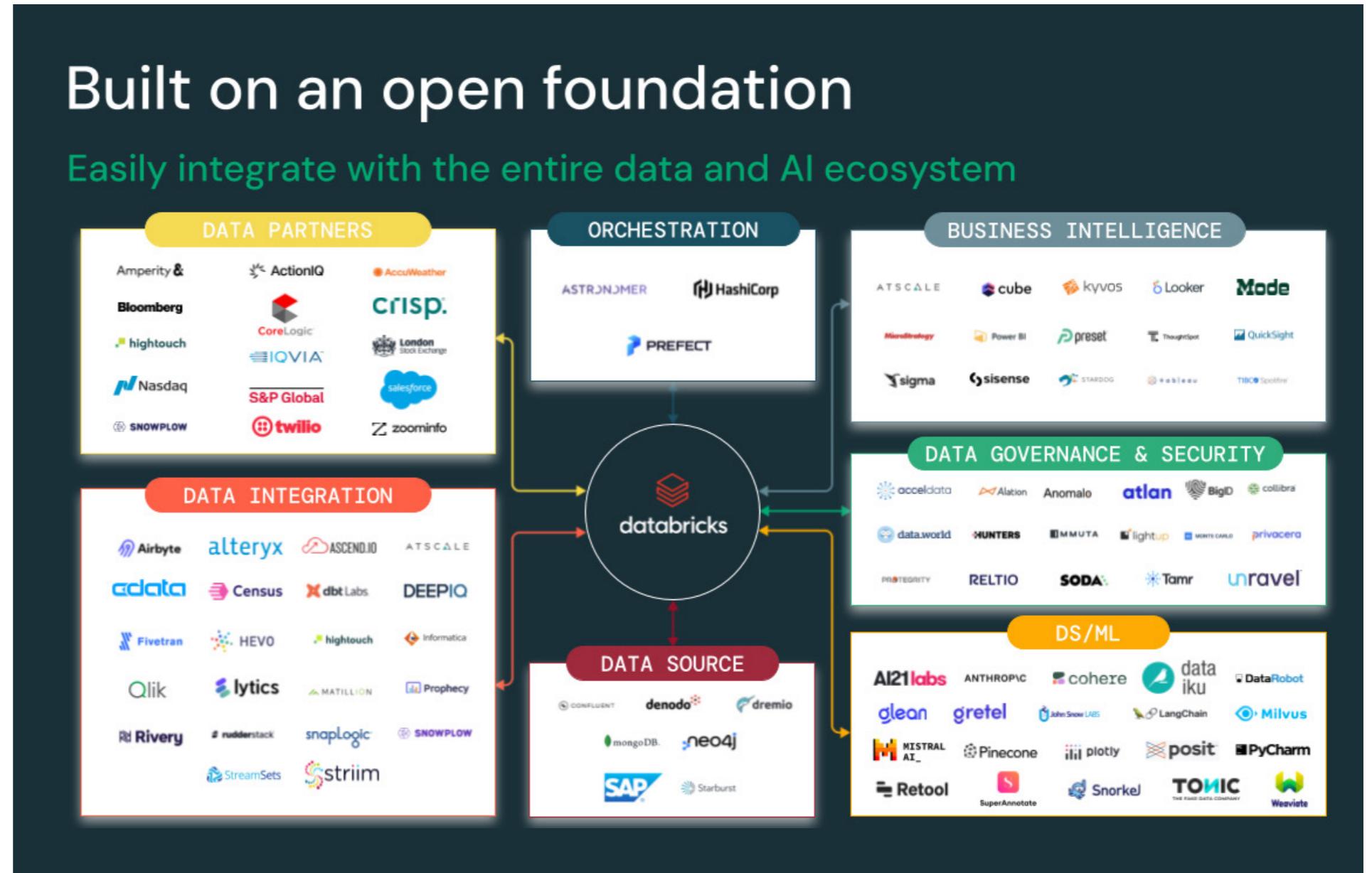
CHAPTER

11

Databricks Technology Partners and the modern data stack

Databricks Technology Partners integrate their solutions with Databricks to provide complementary capabilities for ETL, data ingestion, business intelligence, machine learning and governance. These integrations allow customers to leverage the Databricks Data Intelligence Platform's reliability and scalability to innovate faster while deriving valuable data insights. Use preferred analytical tools with optimized connectors for fast performance, low latency and high user concurrency to your data lake.

With **Partner Connect**, you can bring together all your data, analytics and AI tools on one open platform. Databricks provides a fast and easy way to connect your existing tools to your lakehouse using validated integrations and helps you discover and try new solutions.



[Learn more](#)

CHAPTER

12

Get started with
the Databricks Data
Intelligence Platform

Databricks Trial

Get a collaborative environment for data teams to build solutions together with interactive notebooks to use Apache Spark™, SQL, Python, Scala, Delta Lake, MLflow, TensorFlow, Keras, scikit-learn and more.

- Available as a 14-day full trial in your own cloud or as a lightweight trial hosted by Databricks

[Try Databricks for free](#)

[Databricks documentation](#)

Get detailed documentation to get started with the Databricks Data Intelligence Platform on your cloud of choice: [Databricks on AWS](#), [Azure Databricks](#) and [Databricks on Google Cloud](#).

[Databricks Demo Hub](#)

Get a firsthand look at Databricks from the practitioner's perspective with these simple on-demand videos. Each demo is paired with related materials — including notebooks, videos and eBooks — so that you can try it out for yourself on Databricks.

[Databricks Academy](#)

Whether you are new to the data lake or building on an existing skill set, you can find a curriculum tailored to your role or interest. With training and certification through Databricks Academy, you will learn to master the Databricks Data Intelligence Platform for all your big data analytics projects.

[Databricks Community](#)

Get answers, network with peers and solve the world's toughest problems, together.

[Databricks Labs](#)

Databricks Labs are projects created by the field to help customers get their use cases into production faster.

[Databricks customers](#)

Discover how innovative companies across every industry are leveraging the Databricks Data Intelligence Platform.

About Databricks

Databricks is the data and AI company. More than 10,000 organizations worldwide — including Comcast, Condé Nast, Grammarly and over 50% of the Fortune 500 — rely on the Databricks Data Intelligence Platform to unify and democratize data, analytics and AI. Databricks is headquartered in San Francisco, with offices around the globe, and was founded by the original creators of Lakehouse, Apache Spark™, Delta Lake and MLflow. To learn more, follow Databricks on [LinkedIn](#), [X](#) and [Youtube](#).