

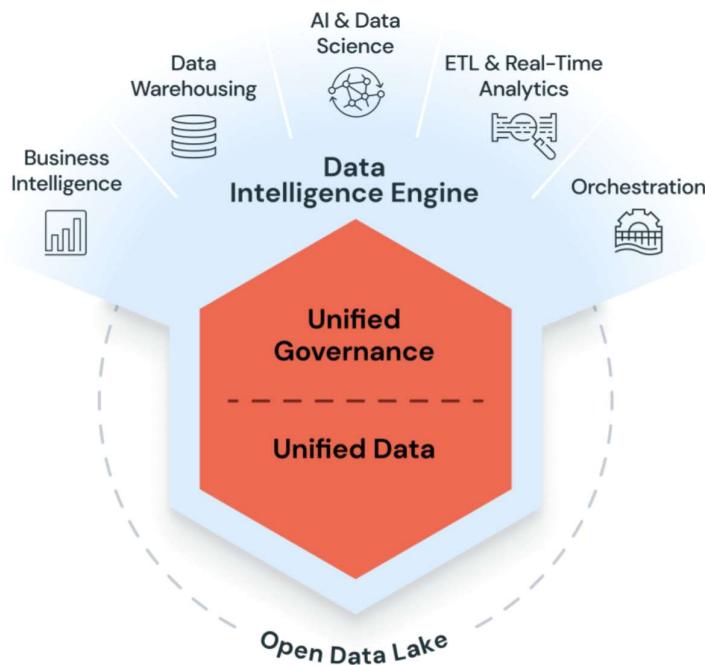
## 7\_What is Data Bricks?

Wednesday, February 11, 2026 12:38 PM

Databricks is a unified, open analytics platform for building, deploying, sharing, and maintaining enterprise-grade data, analytics, and AI solutions at scale. The Databricks Data Intelligence Platform integrates with cloud storage and security in your cloud account, and manages and deploys cloud infrastructure for you

What Databricks Is — In Simple Terms

- Databricks is a cloud-based platform that helps companies store, process, analyze, and use large amounts of data for analytics and AI.
- Think of it as a single workspace where data engineers, data scientists, and analysts can work together without worrying about infrastructure.
- At its core, Databricks is built on Apache Spark, a powerful engine for big data processing. It was created by the original Spark creators.



### **1. Unified Analytics Platform**

Databricks provides one place to:

- Ingest data
- Clean and transform data
- Run large-scale analytics
- Build machine learning models
- Deploy AI applications

It removes the need for multiple disconnected tools.

### **2. The Lakehouse Architecture**

Databricks popularized the **data lakehouse**, which combines:

- The flexibility of a **data lake** (store any data cheaply)
- The reliability and performance of a **data warehouse**

This lets you run BI, ML, and streaming workloads on the same data.

### **3. Delta Lake**

Databricks created **Delta Lake**, an open-source technology that adds:

- ACID transactions

- Schema enforcement
- Time travel
- Faster performance

...on top of data lakes. This makes big data more reliable and easier to manage.

#### 4. Collaboration Workspace

Databricks gives teams:

- Notebooks (Python, SQL, Scala, R)
- Version control
- Job scheduling
- Cluster management

All in one place, so teams can work together efficiently.

#### 5. AI + Natural Language Features

Modern Databricks includes:

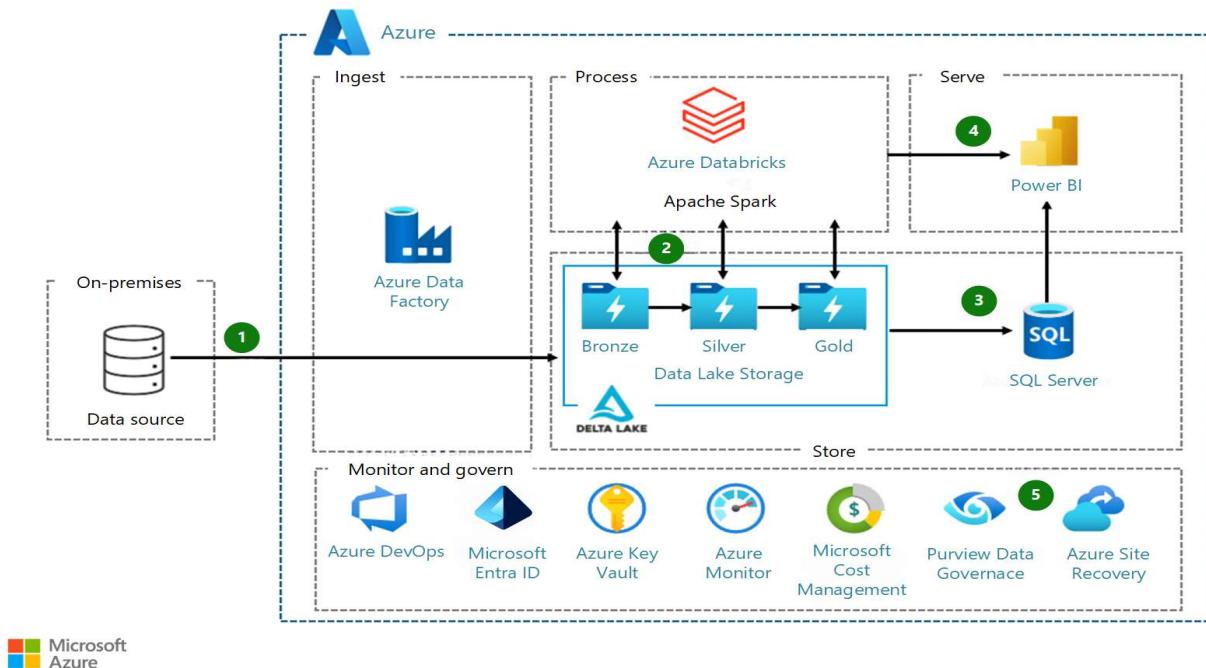
- Natural language search
- AI-assisted coding
- Automated optimization

This helps users interact with data using plain English.

#### Why Companies Use Databricks

- Handles **huge datasets** easily
- Reduces infrastructure complexity
- Supports **ETL, analytics, ML, and AI** in one platform
- Integrates with all major cloud providers
- Scales automatically

It's widely used in finance, healthcare, retail, manufacturing, and tech.



## Azure Databricks has clear advantages over other cloud service providers

- In today's data-driven world, organizations are seeking analytics platforms that simplify management, offer seamless scalability, and deliver consistent performance. While Databricks is available across major cloud service providers (CSPs), not all implementations are equal.
- [Azure Databricks](#) is a first party Microsoft offering co-engineered by Microsoft and Databricks, which stands out for its superior integration, performance, and governance capabilities. It not only delivers strong performance for workloads like decision support systems (DSSs), but it also seamlessly integrates with the Microsoft ecosystem, including solutions such as [Azure AI Foundry](#), [Microsoft Power BI](#), [Microsoft Purview](#), [Microsoft Power Platform](#), [Microsoft Copilot Studio](#), [Microsoft Entra ID](#), [Microsoft Fabric](#), and much more.
- Choosing Azure Databricks can streamline your entire data lifecycle—from data engineering and Extract Transform

Load (ETL) workloads to machine learning (ML), AI, and business intelligence (BI)—within a single, scalable environment.

## ★ The Core Idea

**Azure Databricks is not just “connected” to Azure — it is *built together with Azure*.**

Because of this, it uses Azure’s internal systems directly, instead of going through public APIs.

This is what gives Azure Databricks tighter security, better performance, and deeper integration than the AWS or GCP versions.

## ★ API-Based Integration vs Native Integration

Think of two ways to connect a service to a cloud:

### 1. API-Based Integration (AWS/GCP Databricks)

This is like plugging a device into a wall socket.

- Databricks calls AWS/GCP services using **public APIs**
- Authentication happens through **IAM tokens**
- Storage access happens through **S3/GCS APIs**
- Monitoring uses **CloudWatch/Stackdriver APIs**

It works well, but it’s still an **external connection**.

### 2. Native Integration (Azure Databricks)

This is like wiring the device *inside the wall* during construction.

Azure Databricks uses Azure’s internal systems directly:

- **Azure Active Directory** for identity
- **ADLS Gen2** with the **ABFS driver** (built jointly by Microsoft + Databricks)
- **Azure networking stack** (VNET, Private Link)
- **Azure Monitor** for logs
- **Azure Key Vault** for secrets
- **Azure Data Factory** for pipelines

These are not API calls — they are **internal, platform-level integrations**.

This is why Azure Databricks behaves like a **first-party Azure service**, even though it’s Databricks technology.

## ★ Why Native Integration Is Better

Here’s what native integration gives you that API-based integration cannot:

### 1. Tighter Security

Because Azure Databricks uses Azure’s internal identity and networking:

- No need for access keys
- No need for cross-service API tokens
- No need for custom IAM roles
- No need for S3/GCS credential passthrough hacks

Everything flows through **AAD + Managed Identities**, which is the same system used by Azure SQL, Synapse, Key Vault, etc.

### 2. Better Performance

Because Databricks and Microsoft co-engineered:

- The **ABFS driver** (optimized for ADLS Gen2)
- High-throughput data paths
- Photon engine tuned for Azure hardware
- Cluster bootstrap optimized for Azure VM families

This level of optimization is **not possible** on AWS or GCP because they use different storage systems (S3, GCS).

### 3. Seamless Experience

Azure Databricks feels like part of Azure because:

- You create it from the **Azure Portal**
- You monitor it in **Azure Monitor**
- You secure it with **Azure RBAC**
- You automate it with **Azure DevOps**
- You orchestrate it with **ADF**

On AWS/GCP, Databricks is managed mostly through the **Databricks console**, not the cloud provider’s portal.

## ★ The Simplest Analogy

Here’s the easiest way to understand it:

**AWS/GCP Databricks = A guest living in the house**

They can use the house, but only through the front door (APIs).

**Azure Databricks = A family member who has keys to every room**

**It uses the house’s internal systems directly.**

**Azure Databricks is co-engineered with Microsoft, so it integrates with Azure at a platform level — not just through APIs — giving it tighter security, better performance, and a more seamless experience than AWS or GCP Databricks.**

### **Cross-cloud governance: One platform, multiple clouds**

Azure Databricks now supports [cross-cloud data governance](#), allowing direct access and management of AWS S3 data via Unity Catalog—without the need for data migration or duplication. This unified approach means you can standardize policies, access controls, and auditing across both Azure and AWS, simplifying operations and enhancing security in hybrid and multicloud environments.