
Azure Databricks
PLURALSIGHT

# 01

## *Introduction to Azure Databricks*

# Overview

## Definition

Azure Databricks is a unified analytics platform for big data and machine learning, built on Apache Spark and optimized for Azure.

## Purpose

It combines collaborative notebooks, enterprise-grade security, and scalability to streamline data engineering and data science workflows.

# Core Features

## Data Engineering & Science

Azure Databricks enables collaborative efforts in data engineering and science, enhancing teamwork and project outputs.

## Advanced Analytics

The platform supports advanced analytics and machine learning to derive actionable insights from big data efficiently.

## Stream Processing

It facilitates real-time stream processing for immediate data processing and decision-making.

# 02

## *Fit in the Azure Ecosystem*

# Integration with Azure Services

01

## Data Lake Integration

Seamlessly integrates with Azure Data Lake, providing a robust environment for large-scale data storage and processing.

02

## Synapse Connectivity

Offers native integration with Azure Synapse, enabling sophisticated analytics and data warehousing capabilities.

03

## Power BI Compatibility

Direct connectivity with Power BI allows for advanced data visualization and reporting to drive insightful business decisions.

# Business Applications

## ETL/ELT Processes

Simplifies ETL and ELT processes, improving efficiency and reliability in data movement and transformation.

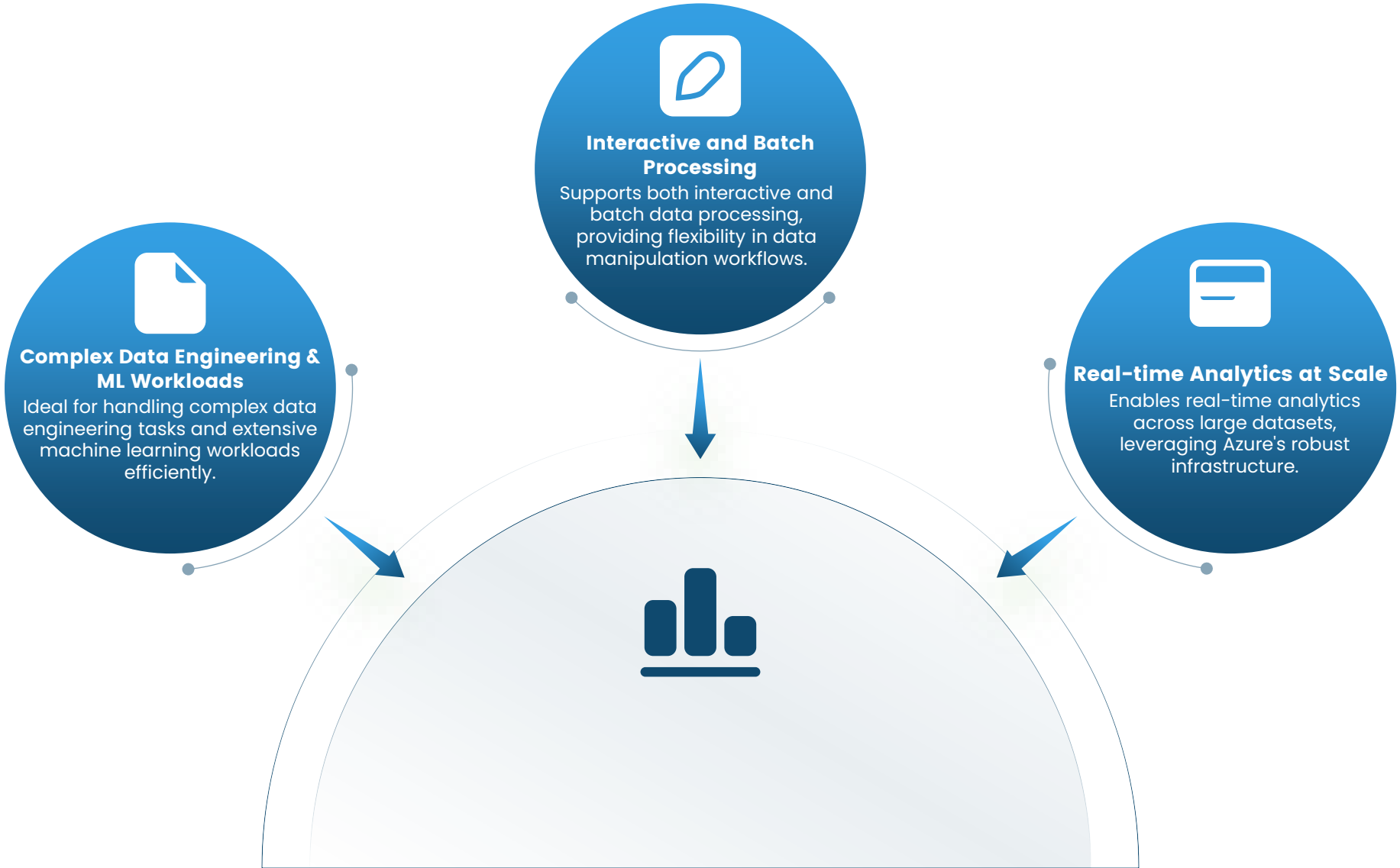
## Machine Learning Projects

Supports various machine learning projects by providing an integrated environment for model development and deployment.

## Streamlining Operations

Optimizes business operations through automated data workflows and real-time analytics.

# When to Use Azure Databricks



# Integration Points

### Azure Data Lake Storage Gen2

Integrates seamlessly with Azure Data Lake Storage Gen2 for scalable and secure data storage.

### Azure Event Hubs / Kafka

Supports integration with Azure Event Hubs and Kafka for real-time data ingestion and streaming analytics.

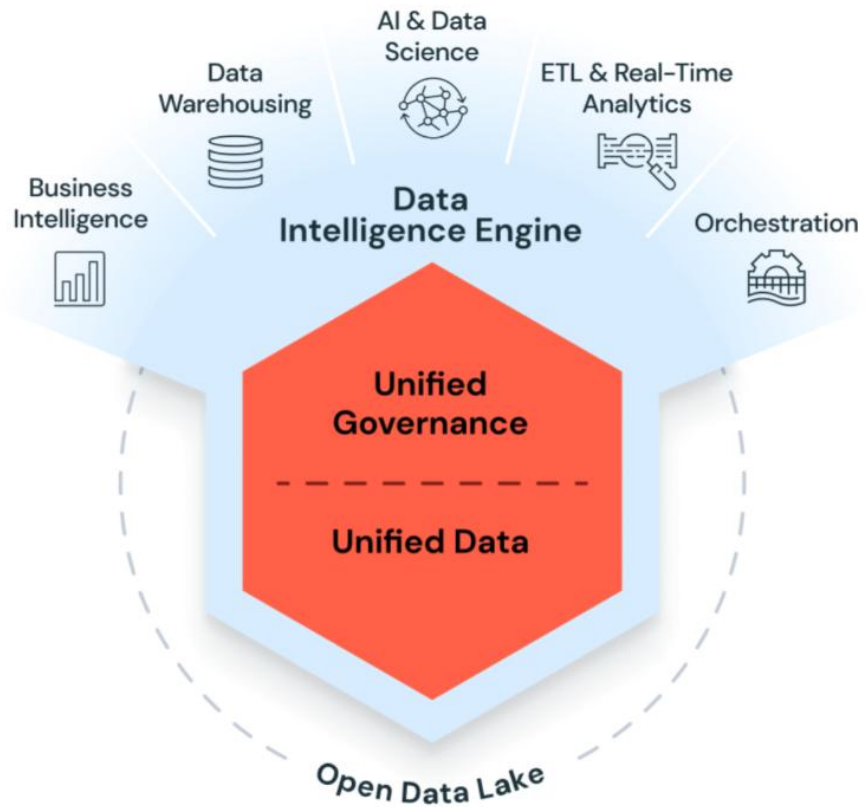
### Power BI for Visualization

Allows easy integration with Power BI, enhancing data visualization and business intelligence capabilities.

# 03

## Architecture Overview

# Governance + Data




https://learn.microsoft.com/en-us/azure/databricks/introduction/

# Underlying Technologies

## Apache Spark

Utilizes Apache Spark as the foundational technology for distributed data processing.

## Delta Lake

Enhances data reliability and performance with Delta Lake, optimizing storage and queries.

## MLflow

Incorporates MLflow for managing the end-to-end machine learning lifecycle, from experimentation to deployment.

# Managing Clusters

01

**Cluster Creation**

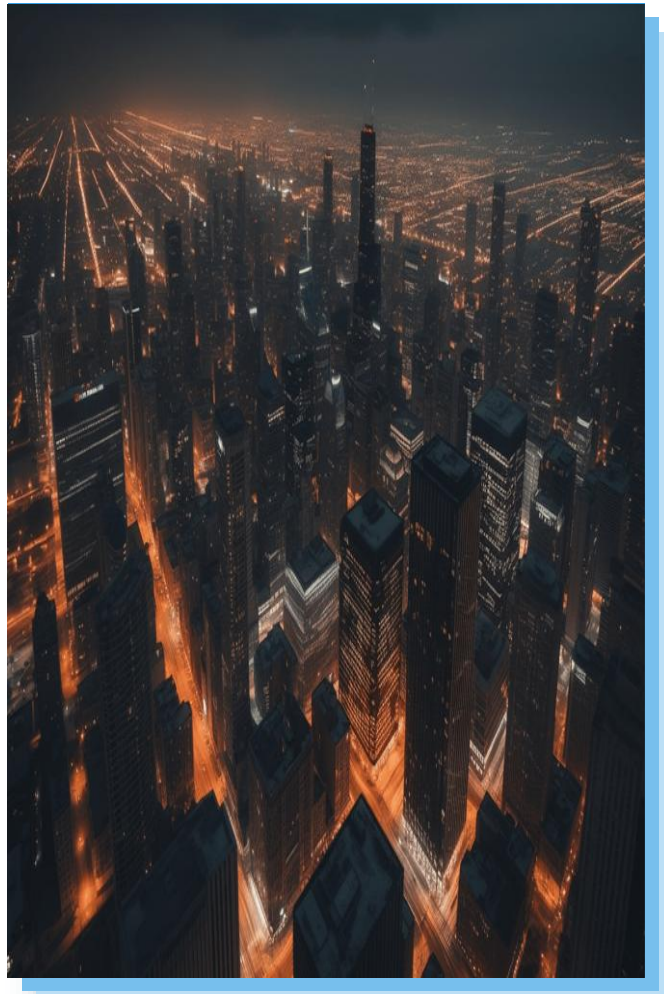
Process of setting up clusters within Databricks, including configuration options.

02

**Scaling Clusters**

Methods to scale clusters up or down based on workload requirements.

# Managing Workspaces



## 01. Workspace Setup

Execute steps for initial workspace configuration and setup in Databricks to take advantage of its capability

## 02. User Management

Leverage the tools provided by Databricks to manage users and permissions effectively – Principles of Least Privilege, Secure by Default, and Defense in Depth

# Platform Components

### Compute Resources

Utilizes scalable cloud resources to handle large-scale data processing and computational tasks.

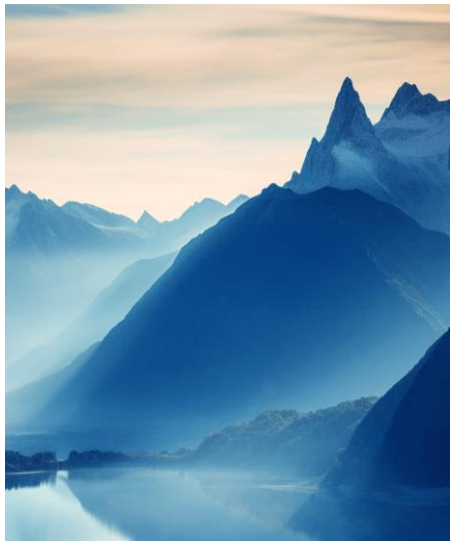
### Data Storage Solutions

Supports various data storage solutions, ensuring accessibility, reliability, and scalability of data.

### Integration Modules

Provides integration modules to connect seamlessly with other Azure services and third-party applications.

# Workflow Management



Enables efficient job scheduling to manage, execute, and monitor data pipelines and analytics tasks.

**Job Scheduling**

Integrates with tools to orchestrate complex workflows involving multiple data processing stages and services.

**Workflow Orchestration**

# Core Components

## Workspace

Helps with the organization and management of resources

## Clusters

Scalable units of compute that power the platform

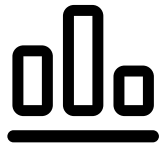
## Jobs

Scheduling and execution of jobs that support automated ETL and ML workflows

# Control Plane

## UI

The user interface managed by Databricks offers links to clusters, notebooks, and various tools.



## Jobs Scheduler

Schedules jobs efficiently within the Databricks environment, ensuring systematic processing.

# Data Plane

## Compute Resources

Compute resources are hosted in your VNet, providing dedicated processing power.

01

## Data Access

Facilitates connection to Azure Data Lake Storage and Blob Storage for seamless data retrieval.

02

# Cluster Types

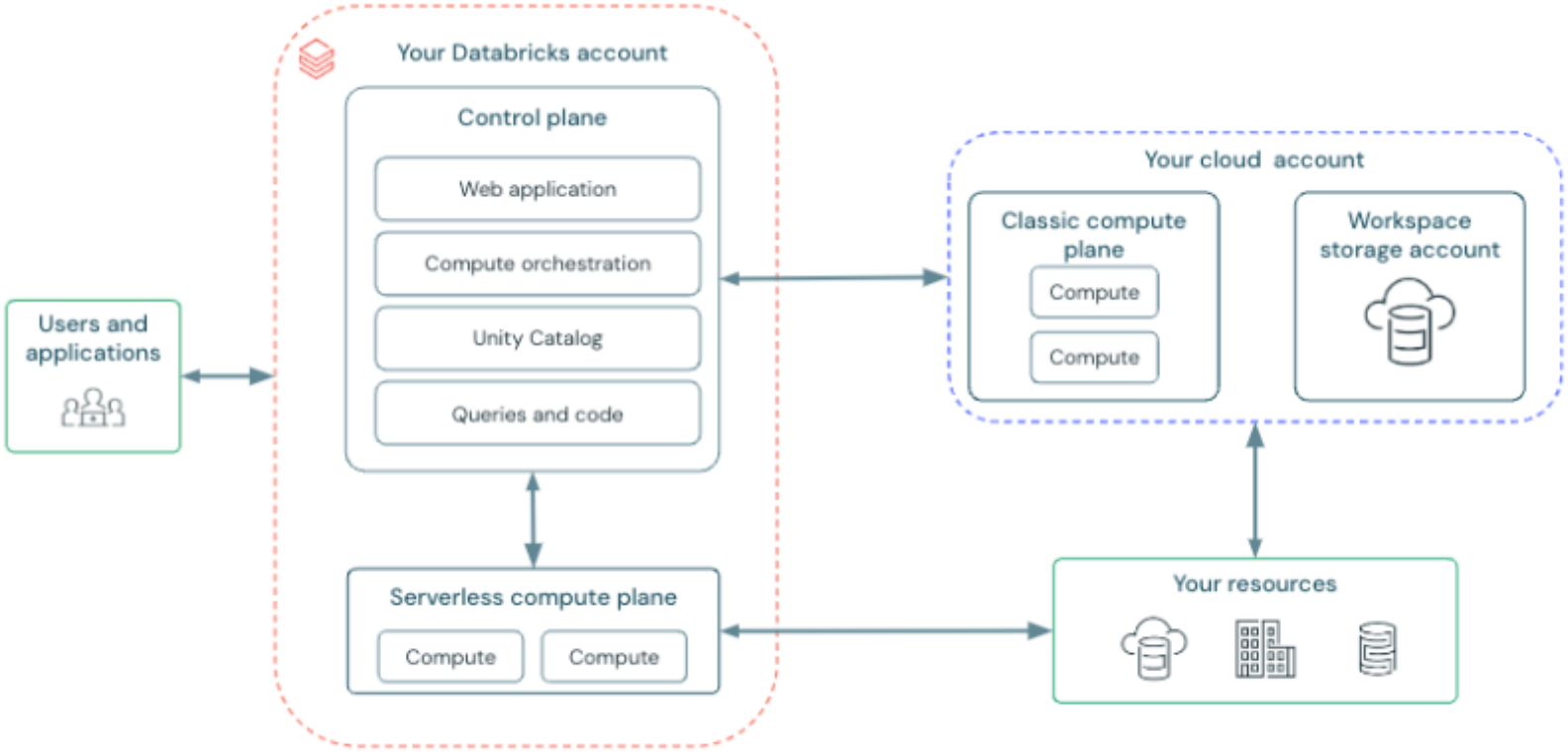
### Interactive Clusters

Designed for interactive analytics, providing immediate query results.

### Job Clusters

Optimized for scheduled jobs, tailored for batch processing tasks.

# Underlying Technologies




https://learn.microsoft.com/en-us/azure/databricks/getting-started/overview

# Security Features

### Authentication & Authorization

Robust authentication and authorization features ensure secure access to data and resources.

### Data Encryption

Data encryption at rest and in transit protects sensitive information from unauthorized access.

### Network Security

Enhanced network security measures prevent unauthorized access and ensure data integrity.

# 04

## *Performance Considerations*

# Optimization Techniques

## Query Tuning

Employ query tuning techniques to enhance the performance and speed of data retrieval operations.

01

02

## Resource Allocation

Dynamic resource allocation ensures optimal performance based on workload requirements and priorities.

# Cluster Configuration

## Worker types and autoscaling

Optimize cluster performance by choosing appropriate worker types and enabling autoscaling to dynamically adjust resources based on workload needs.

## Spot vs. on-demand VMs

Evaluate cost and reliability trade-offs between using spot instances and on-demand VMs for running workloads on Azure Databricks.

# Benchmarking Strategies

### Performance Metrics

Use comprehensive performance metrics to evaluate and benchmark system efficiency and effectiveness.

### Continuous Monitoring

Implement continuous performance monitoring to detect issues and maintain optimal operational efficiency.

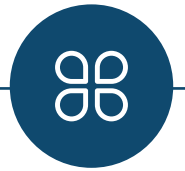
# Caching Strategies

### Delta Cache

Utilize Delta Cache for improving read performance of data stored in Delta Lake.

### Databricks IO Cache

Leverage Databricks IO Cache for faster data retrieval in big data workflows.



### Memory Management

Efficient memory management techniques help improve overall system performance.

# Query Optimization

### Cost-Based Optimizer

Employ a Cost-Based Optimizer to choose the most efficient plan for query execution.

### Adaptive Query Execution

Adaptive Query Execution dynamically optimizes queries during runtime for better performance.

### Vectorized Execution

Vectorized execution model enhances processing speed of large datasets.

# Compute Resources

## Autoscaling Clusters

Set clusters to automatically scale up or down based on workload to optimize resource utilization.

## Spot Instances

Utilize spot instances for cost-effective scaling while ensuring computational efficiency.

## GPU Acceleration

Harness GPU acceleration to improve performance for machine learning and deep learning tasks.

# Delta Lake Performance

### ACID transactions

Ensure data reliability and integrity in Delta Lake by leveraging ACID transaction properties for consistent and isolated data processing.

### Data skipping and Z-order indexing

Enhance query performance through data skipping and advanced indexing techniques like Z-order indexing to reduce I/O operations.

# 05

## *Security and Compliance*

# Security Features

### Authentication Protocols

Employs robust authentication protocols to ensure secure access and protect sensitive data.

### Encryption Standards

Adheres to industry-standard encryption for data at rest and in transit to safeguard against unauthorized access.

# Compliance Standards

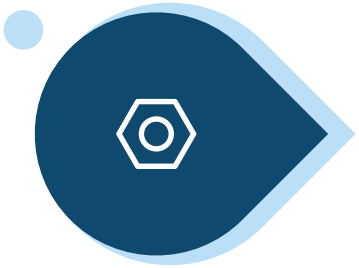
### Regulatory Compliance

Meets various regulatory compliance standards, ensuring the platform is suitable for use in regulated industries.

### Data Governance

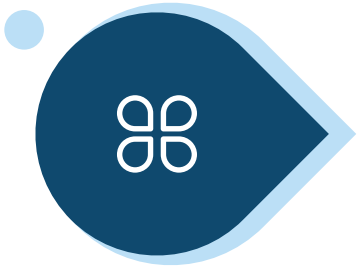
Implements comprehensive data governance policies to ensure data integrity, quality, and security.

# Data Governance



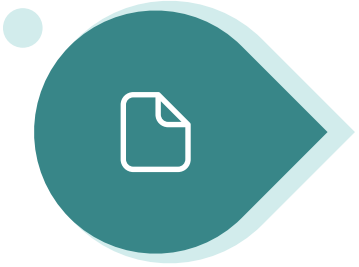
## Data Lineage

Tracks data movement and transformations for better governance and transparency.



## Auditing

Comprehensive auditing capabilities to monitor and log access and modifications.



## Policy Management

Defines and enforces data policies to comply with regulatory requirements.

# Compliance Standards

### GDPR

Ensures compliance with GDPR regulations for processing personal data within the EU.

### HIPAA

Supports HIPAA compliance for managing healthcare-related data securely and efficiently.

### SOC 2

Meets SOC 2 standards for secure data management and privacy.

# Identity Management

### Role-Based Access Control (RBAC)

Implements RBAC to control resource access based on user roles and responsibilities.

### Azure Active Directory Integration

Seamlessly integrates with Azure Active Directory for robust identity and access management.

### Multi-Factor Authentication

Enhances security by requiring multi-factor authentication for accessing critical resources.

# 06

## Use Cases

# Data Engineering

## ETL Pipelines

Builds efficient ETL pipelines to extract, transform, and load large datasets.

## Data Warehousing

Utilizes Databricks for creating scalable and performant data warehouses.

## Data Migration

Efficiently migrates data from legacy systems to modern data platforms.

# Machine Learning

## Model Training

Facilitates large-scale model training with distributed computing capabilities.

## Hyperparameter Tuning

Automates hyperparameter tuning and optimization processes for better model performance.

## Model Deployment

Streamlines the deployment of machine learning models to production environments.

# Real-Time Analytics

01

### Event Processing

Processes real-time streams of events for immediate insights and actions.

02

### Stream Analytics

Enables real-time stream analytics for monitoring critical business operations.

03

### Predictive Maintenance

Uses real-time data to predict and prevent equipment failures and downtime.

# 07

## Best Practices

# Development

## Collaboration

Promotes collaboration between data engineers, scientists, and analysts through shared workspaces.

## Version Control

Implements version control for managing changes and maintaining consistency in code.

## Testing

Ensures thorough testing of data pipelines and models to maintain reliability and accuracy.

# Automation

## Continuous Integration

Applies continuous integration pipelines to streamline and automate workflow deployment.

## Monitoring & Logging

Sets up extensive monitoring and logging for tracking performance and diagnosing issues.

## Infrastructure as Code

Utilizes Infrastructure as Code (IaC) for consistent and reproducible infrastructure setups.

# Cost Management

01

## Budgeting

Sets and monitors budgets to control costs associated with data processing and storage.

## Resource Optimization

Optimizes resource usage to balance performance and cost-effectiveness.

02

03

## Cost Tracking

Implements tools and processes for tracking and analyzing costs in real-time.

# Data Governance

## Use Unity Catalog for fine-grained data governance

Implementing fine-grained data governance, metadata management, and access controls using Unity Catalog to ensure data security and compliance.

# What is Unity Catalog?

## Definition and Overview

Unity Catalog is Databricks' data governance layer providing a unified metastore for tables, views, volumes, files, functions, and ML models.

## Importance in Data Governance

Essential for secure, scalable data access across workspaces and teams, crucial in multi-tenant or enterprise-level environments.

# Core Components

### Metastore

A top-level container for cataloging and governance typically scoped to a region, shared across workspaces.

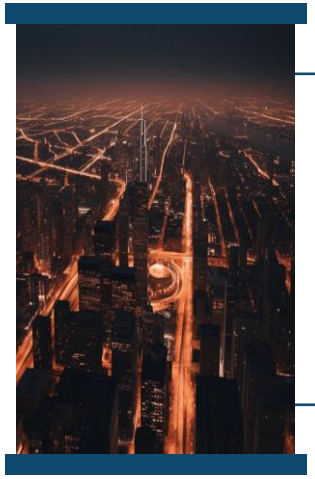
### Schema (Database)

Grouping elements including tables, views, functions, enabling structured data management.

### Catalog

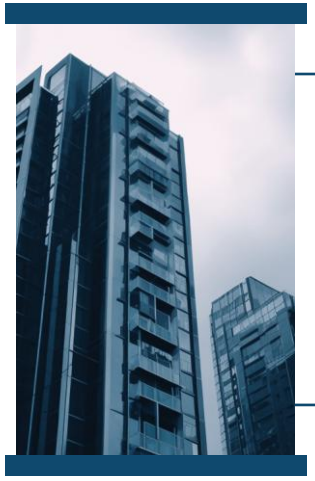
A logical grouping of schemas acting as collections of databases, facilitating organization.

# Access Controls



## Fine-Grained Access Control

Provides table, column, and row-level security, allowing detailed data access management.



## Attribute-Based Access Control (ABAC)

Integration with Microsoft Entra ID enables attribute-based security measures.

# Data Masking and Filtering

**01**

## Data Masking Techniques

Ensures sensitive information is hidden during data access, providing an additional security layer.

**02**

## Row-Level Filtering

Allows dynamic data visibility based on user roles and permissions, enhancing security.

# Integration with Azure Data Lake Storage

## Storage and Compute Decoupling

Multiple workspaces access a single metastore, ensuring a flexible and efficient architecture.

## Configuration Options

Unity Catalog can be configured via Azure Databricks account console, Terraform, or REST APIs.

# User Configuration

### Setting Up Metastore

Defining a metastore in Azure region and attaching it to multiple Databricks workspaces.

### Assigning Roles and Permissions

Creating SQL definitions for tables, views, and assigning precise access permissions.

# Benefits: Centralized Governance

### Single Access Point

One centralized place to manage access policies across workspaces, simplifying administrative tasks.

### Simplified Data Sharing

Securely share data across various workspaces, ensuring consistency and compliance.

# Benefits: Multicloud Support

01.

### Consistent API

Provides a consistent API and governance framework across AWS, Azure, and GCP.

02.

### Lineage Tracking

Automatically captures and visualizes how data is created and used, aiding in transparency.

# Setting Up a Catalog

## Creating a Catalog

Use SQL commands to create a catalog in an Azure environment for structured data organization.

## Creating Schemas and Tables

Define schemas within the catalog and populate them with tables using detailed access controls.

# Setting Up a Catalog

```sql
-- Create a catalog
CREATE CATALOG finance_catalog;

-- Create a schema inside the catalog
CREATE SCHEMA finance_catalog.sales;

-- Create a table
CREATE TABLE finance_catalog.sales.q1_revenue (
  region STRING,
  amount DECIMAL(10, 2)
);

-- Grant SELECT permission to a group
GRANT SELECT ON TABLE finance_catalog.sales.q1_revenue TO `data_analysts`;
```

# Role Assignment

### Data Steward Roles

Assign specific roles responsible for the management and governance of catalog and schemas.

### Access Permissions

Utilize SQL to grant specific permissions to user groups for data access and manipulation.

# Azure Databricks Requirements

### Premium or Enterprise Plans

Ensure workspaces are on Azure Databricks Premium or Enterprise plans for Unity Catalog access.

### Cluster Policies

Implement cluster policies or shared access mode for data access, ensuring compliance with security requirements.

# Workspace Configuration

01

### Enabling Workspaces

Enable workspaces for Unity Catalog usage and ensure configurations align with governance policies.

02

### Data Access Policies

Establish and enforce data access policies to maintain security and operational efficiency.

# 08

## Summary

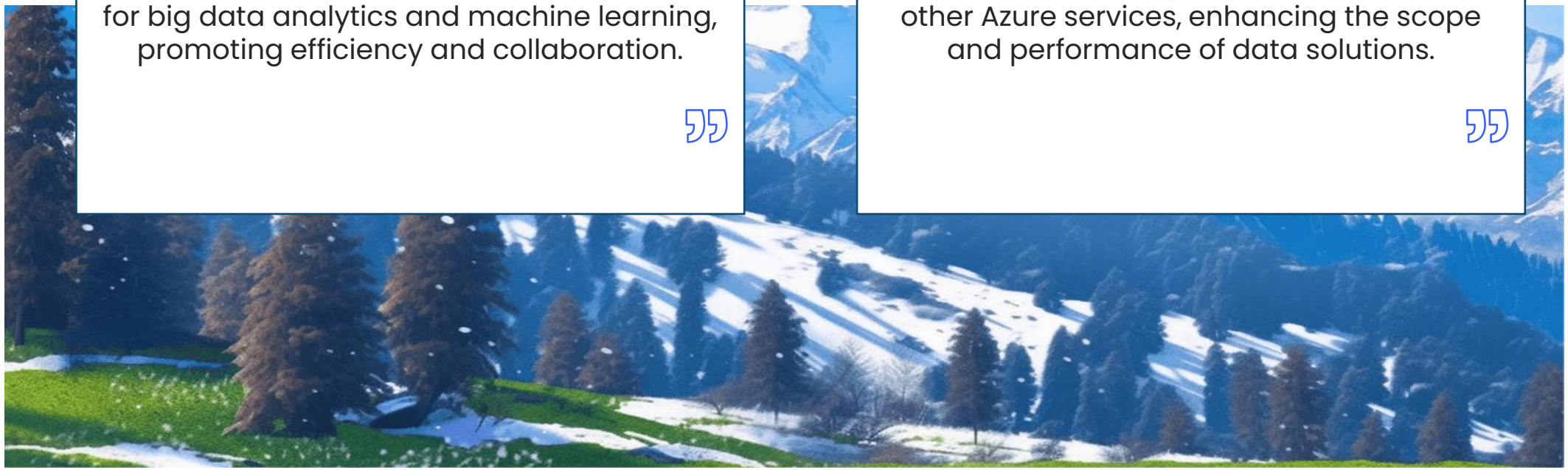
# Key Takeaways

### Unified Platform

Azure Databricks provides a unified platform for big data analytics and machine learning, promoting efficiency and collaboration.

### Integration Capabilities

Offers extensive integration capabilities with other Azure services, enhancing the scope and performance of data solutions.




Thanks