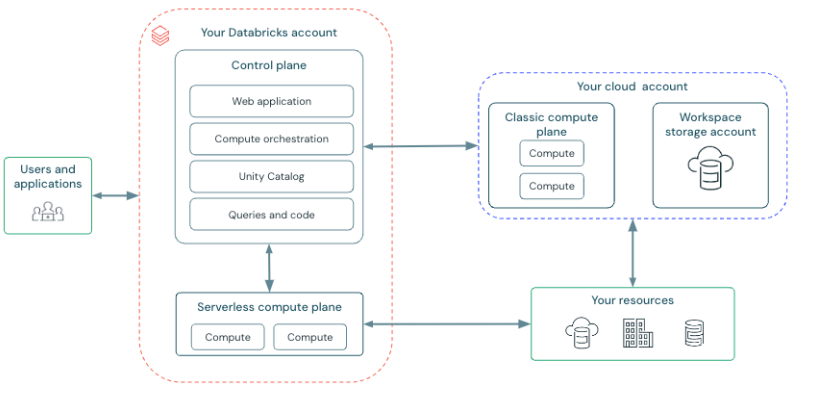
ADB – Architecture

👉Classic Workspace Architecture

Reference: https://learn.microsoft.com/en-us/azure/databricks/getting-started/high-level-architecture



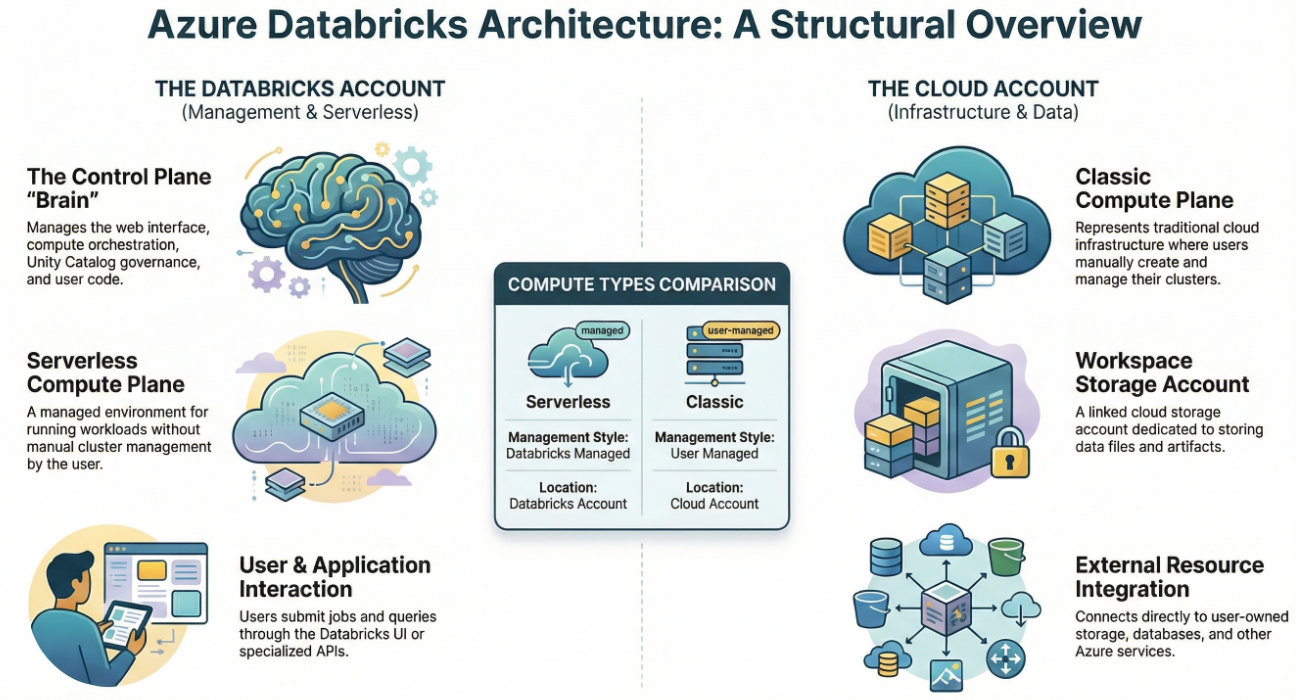
👉Quick Summary

Azure Databricks uses a two-plane architecture: a Databricks-managed control plane for orchestration and governance, and a compute plane where Spark clusters run inside either serverless infrastructure or the customer’s cloud subscription. Data always remains in cloud storage and is governed centrally via Unity Catalog

Control plane -> A managed layer which consist UI, compute, unity catalog and Queries and Code

👉Serverless Compute Plane: Which create a compute layer within workspace rather in a different server, user can connect to the same account workspace for the compute.

👉 Mind map



ADB - Cluster

An Azure Databricks cluster is a Spark compute environment consisting of a driver and workers that runs data engineering, analytics, and ML workloads, with auto-scaling and job-based lifecycle management

👉Cluster Components –

Driver Node (to Orchestrate jobs and maintains spark context)

Worker Nodes (to Execute tasks and transformation)

Auto scaling (to Adds/removes workers based on workload)

👉Types

All purpose (Manually created using UI or CLI or rest API; Used with notebooks; interactive development; long running)

Job Cluster (Automatically Created per job run, terminated after job completed, cost efficient)

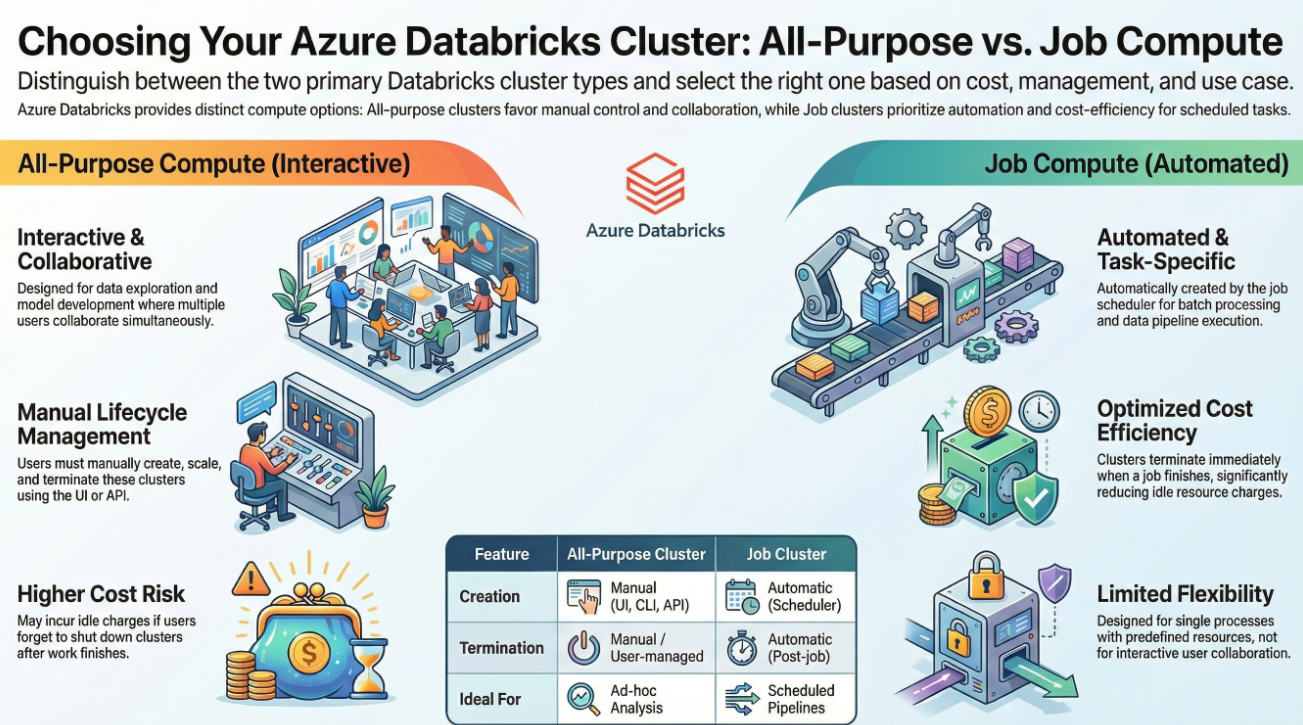
👉 Modes

Standard (Shared by Multiple users);

Single Node (for light weight testing);

High concurrency (for many users and to run queries in parallel)

👉 Mind map

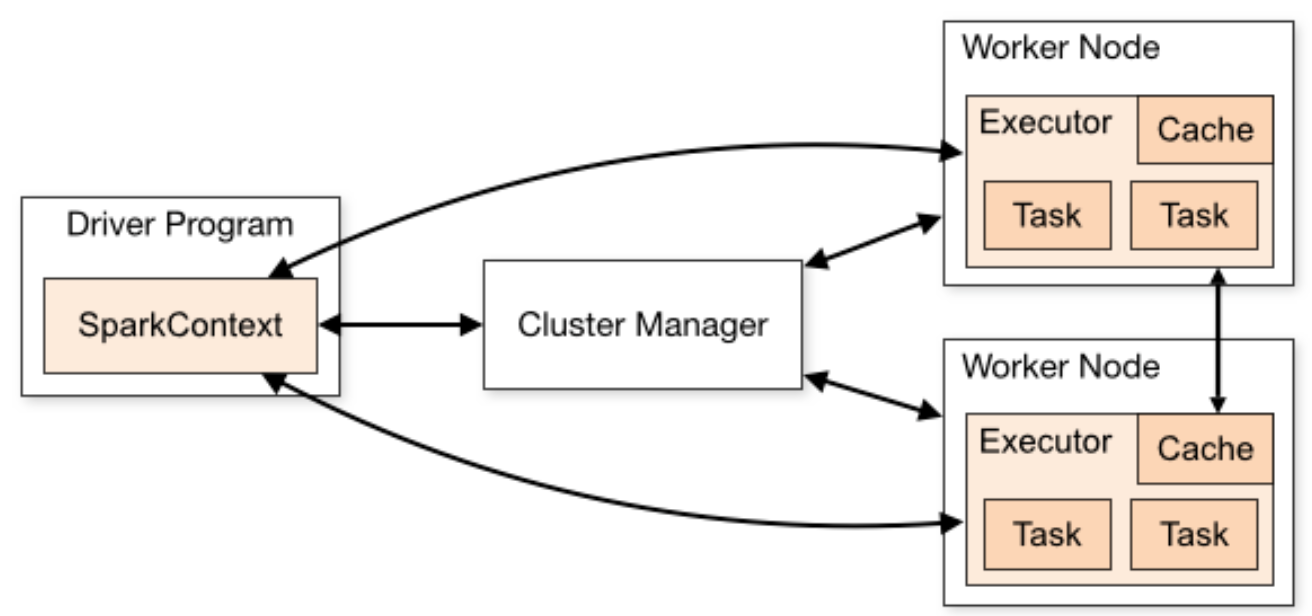


Apache Spark Architecture

👉Quick Summary:

In Azure Databricks, Apache Spark runs on clusters composed of a driver node that orchestrates execution and multiple worker nodes running executors that process tasks in parallel. Spark converts user code into DAGs, breaks them into stages and tasks, and executes them across the cluster while reading and writing cloud storage.”

👉Core Architecture -



* Driver Node (Brain) - Creating SparkSession / SparkContext; Converting code → execution plan; Splitting jobs into stages & tasks; Scheduling work to executors; Collecting results
* **Worker Nodes (Executors = Muscle) -** Execute tasks in parallel; Cache data in memory; Perform shuffles, joins, aggregations; Each executor has - CPU cores,Memory,Task slots

👉How This Fits Inside Azure Databricks

| **Spark Concept** | **In Databricks** |
| --- | --- |
| Driver | Master VM in cluster |
| Executors | Worker VMs |
| Cluster | Databricks compute resource |
| Storage | ADLS / cloud object storage |
| Metadata | Unity Catalog |

👉Data Bricks Enhancements

Optimized Spark Runtime (faster than open source);   
✅ Auto-scaling executors  
✅ Job clusters (ephemeral Spark clusters)  
✅ Delta Lake integration  
✅ Intelligent caching  
✅ Serverless Spark compute

👉**Spark + Databricks Security Layer**

Spark runs only in compute plane; Data accessed through governed storage; Unity Catalog controls Spark table access; Network isolation in Classic compute

👉 Mind map



👉 **Common Interview Questions (with crisp answers)**

* **Where does Spark run in Databricks?**

Inside Databricks clusters (compute plane)

* **What is the driver?**

Node that builds DAGs and schedules tasks

* **What are executors?**

Processes on worker nodes that execute tasks in parallel

* **Why is Spark fast in Databricks?**

In-memory processing + optimized runtime + autoscaling