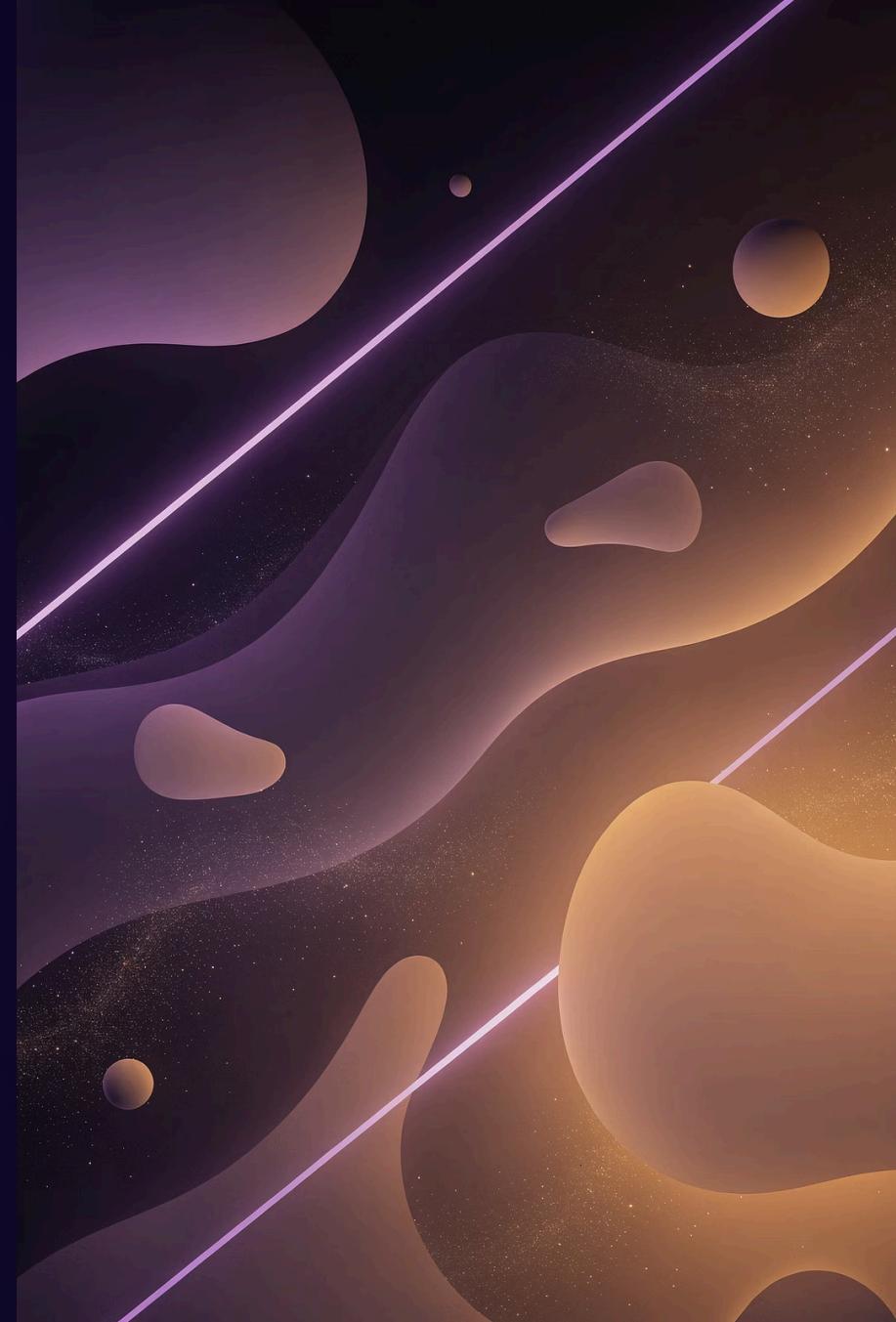
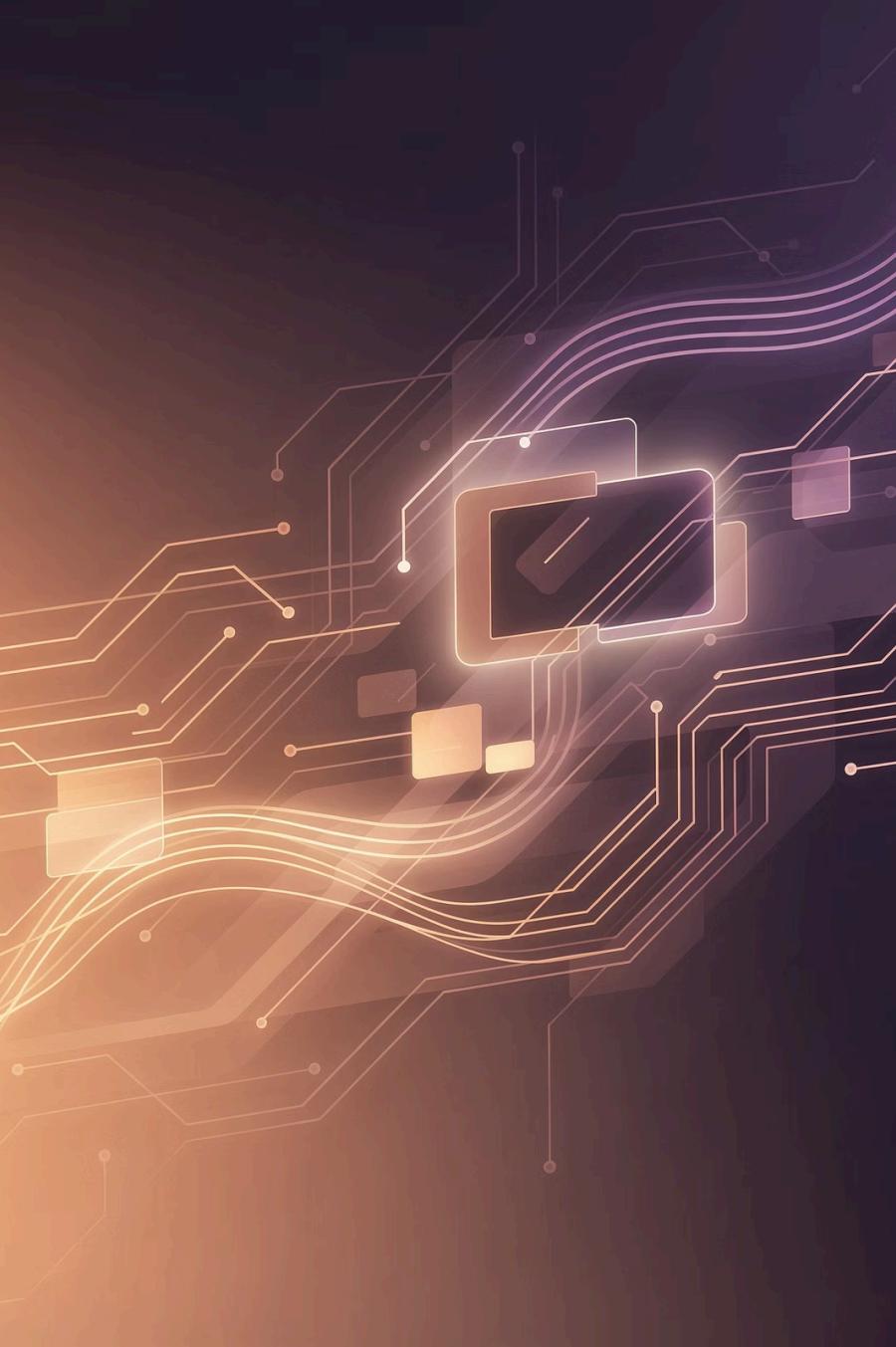


# Advanced Spark Architecture & Features





# Spark Core Internals

Apache Spark Core is the foundation of Spark.

It provides:

## Core Capabilities

- Task scheduling
- Memory management
- Fault tolerance
- DAG execution engine
- Interaction with storage systems

## Core Components

- Driver Program
- SparkContext
- DAGScheduler
- TaskScheduler
- Executors

The Driver builds the execution plan.  
Executors execute tasks on worker nodes.

# DAGs, Stages, and Tasks

When an action is triggered, Spark creates a:

## Directed Acyclic Graph (DAG)

- Logical execution plan
- Represents sequence of transformations
- Optimized before execution

## Stages

A stage is a group of operations that can run without shuffle.

- Narrow transformations → same stage
- Wide transformations (shuffle) → new stage

Example wide operations:

- `groupByKey`
- `join`
- `reduceByKey`

## Tasks

- Smallest unit of execution
- One task per partition
- Tasks run in parallel across executors

More partitions → more parallel tasks

# Cluster Managers in Spark

Spark can run on different cluster managers:

**Standalone**

Built-in

**YARN**

**Apache Mesos**

**Kubernetes**

**Local mode**

## Role of Cluster Manager

- Allocates CPU and memory
- Launches executors
- Manages resources across nodes

Spark focuses on computation. Cluster manager focuses on resource allocation.



# New Features in Spark 4.x

Spark 4.x improves performance and runtime optimization.

Enhancements include:



**Improved Adaptive Query Execution**



**Better shuffle optimization**



**Enhanced physical plan tuning**



**Runtime-based decision improvements**

- Many intelligent optimizations are enabled by default.

# Adaptive Query Execution (AQE)

AQE is a runtime optimizer. It adjusts the execution plan during query execution.

## What AQE Does

- Coalesces shuffle partitions
- Converts sort-merge joins to broadcast joins
- Handles data skew dynamically
- Optimizes based on actual runtime statistics

## Why AQE Matters

- Reduces need for manual tuning
- Improves join performance
- Optimizes shuffle behavior automatically

**AQE modifies the physical plan at runtime.**

# Dynamic Partition Pruning (DPP)

Dynamic Partition Pruning reduces unnecessary data scanning.

## How It Works

- **Uses join filters at runtime**
- **Prunes irrelevant partition directories**
- **Reduces disk I/O**

## Best used for:

- Partitioned tables
- Star schema joins
- Large ETL queries

# Benefit



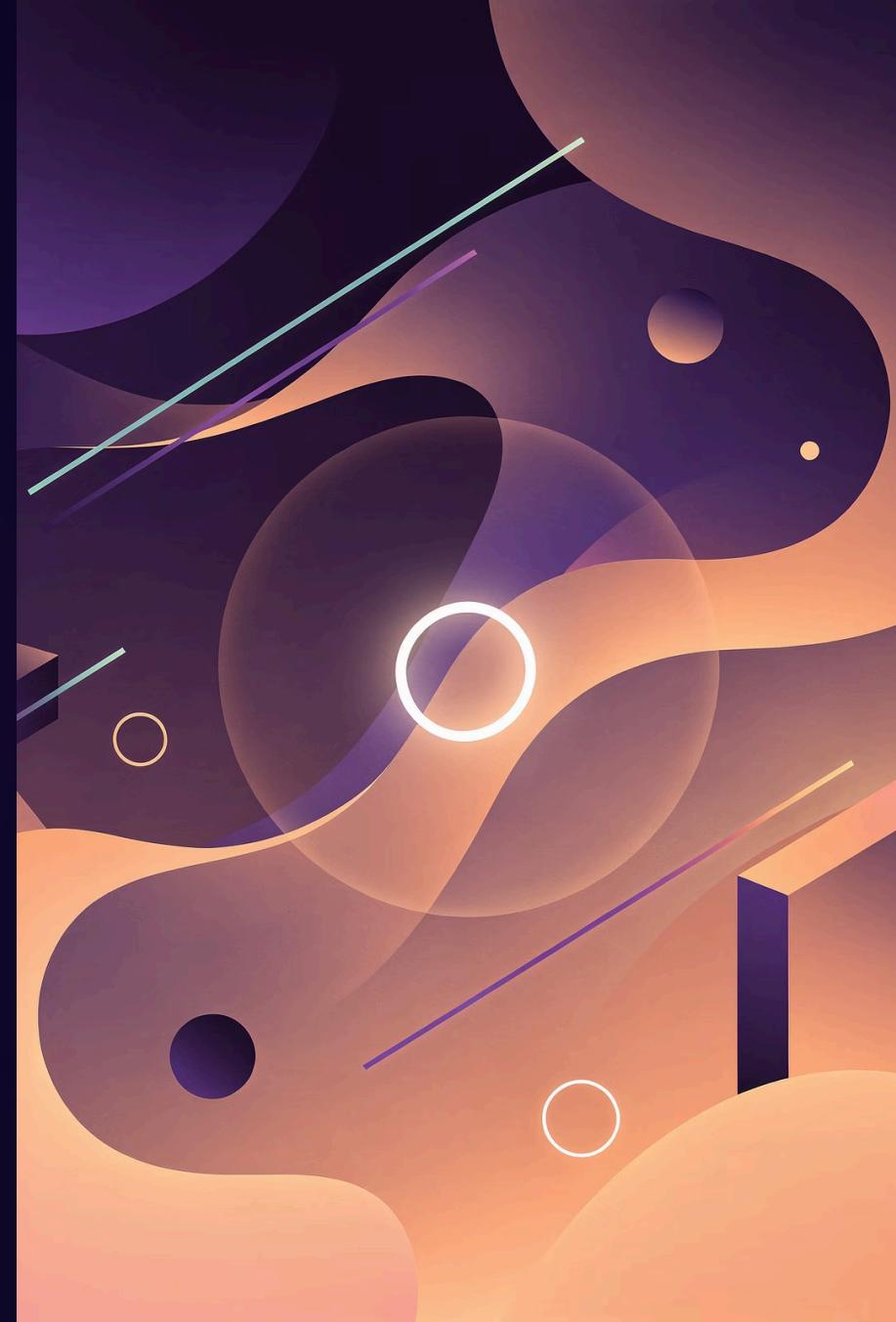
Faster queries



Lower I/O cost



Improved  
performance on  
partitioned datasets



# Spark Execution Modes

Spark supports multiple execution modes.

## Local Mode

- Runs on single machine
- Used for development and testing
- No cluster required

## Client Mode

- Driver runs on submission machine
- Executors run in cluster
- Interactive workloads

## Cluster Mode

- Driver runs inside cluster
- More reliable for production
- Submission client can disconnect

# Summary

01

**Spark Core manages scheduling and execution.**

02

**DAG is divided into stages and tasks.**

03

**Cluster managers allocate resources.**

04

**Spark 4.x improves runtime optimization.**

05

**AQE adjusts execution plans dynamically.**

06

**DPP prunes unnecessary partitions.**

07

**Execution mode determines driver location.**

- Understanding these concepts helps debug performance and optimize real-world workloads.