Apache Spark Core Components

> Apache Spark is an in-memory cluster **computing framework** designed for big data workloads.

Spark is designed to handle a wide range of big data workloads

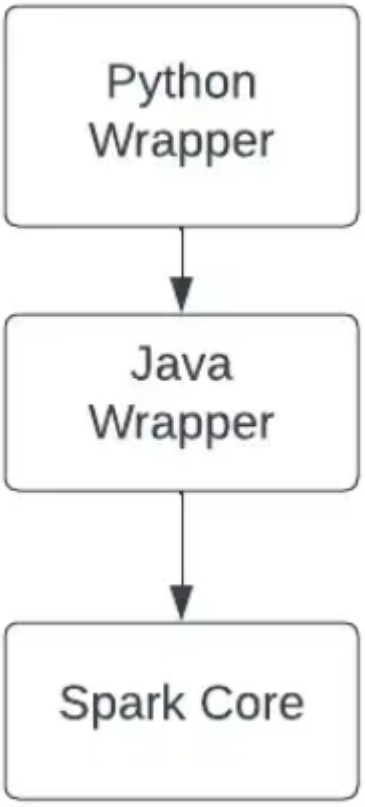
1. Data Integration and ETL (Extract Transform and Load)
2. High Performance Batch Computation
3. Machine Learning Analytics
4. Real-time streaming processing
5. Graph Computation

Important Points

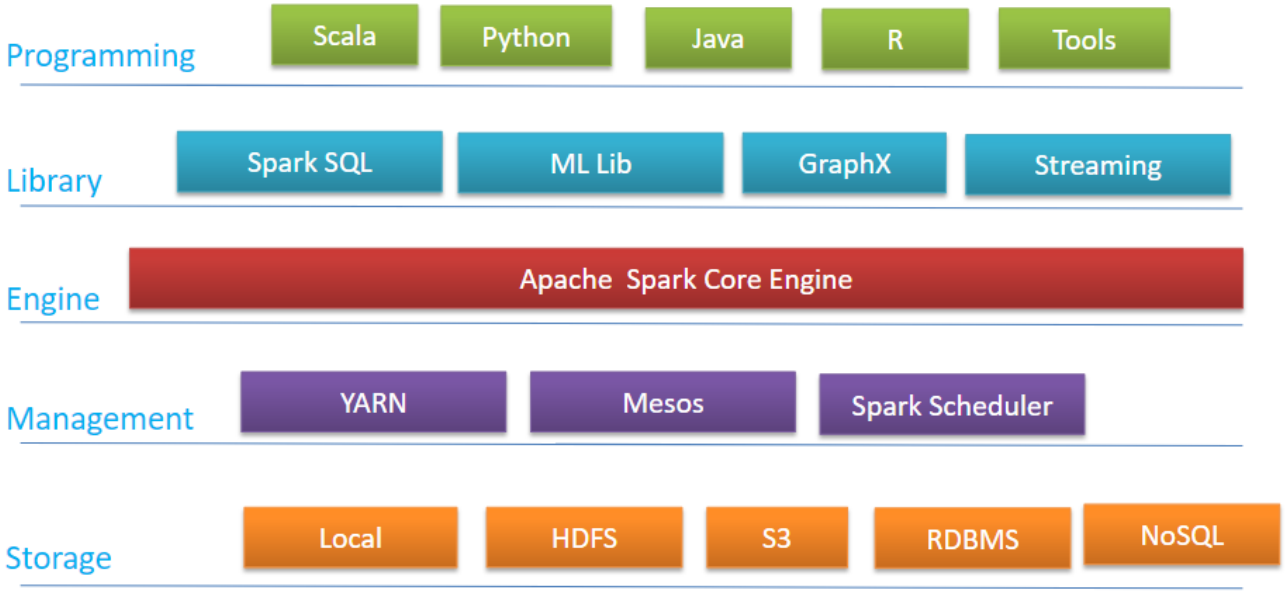
- Apache Spark is written using Scala

What is PySpark

> PySpark is the Python API for Apache Spark



Ecosystem



Spark Interactive Shell

2 main interactive shells

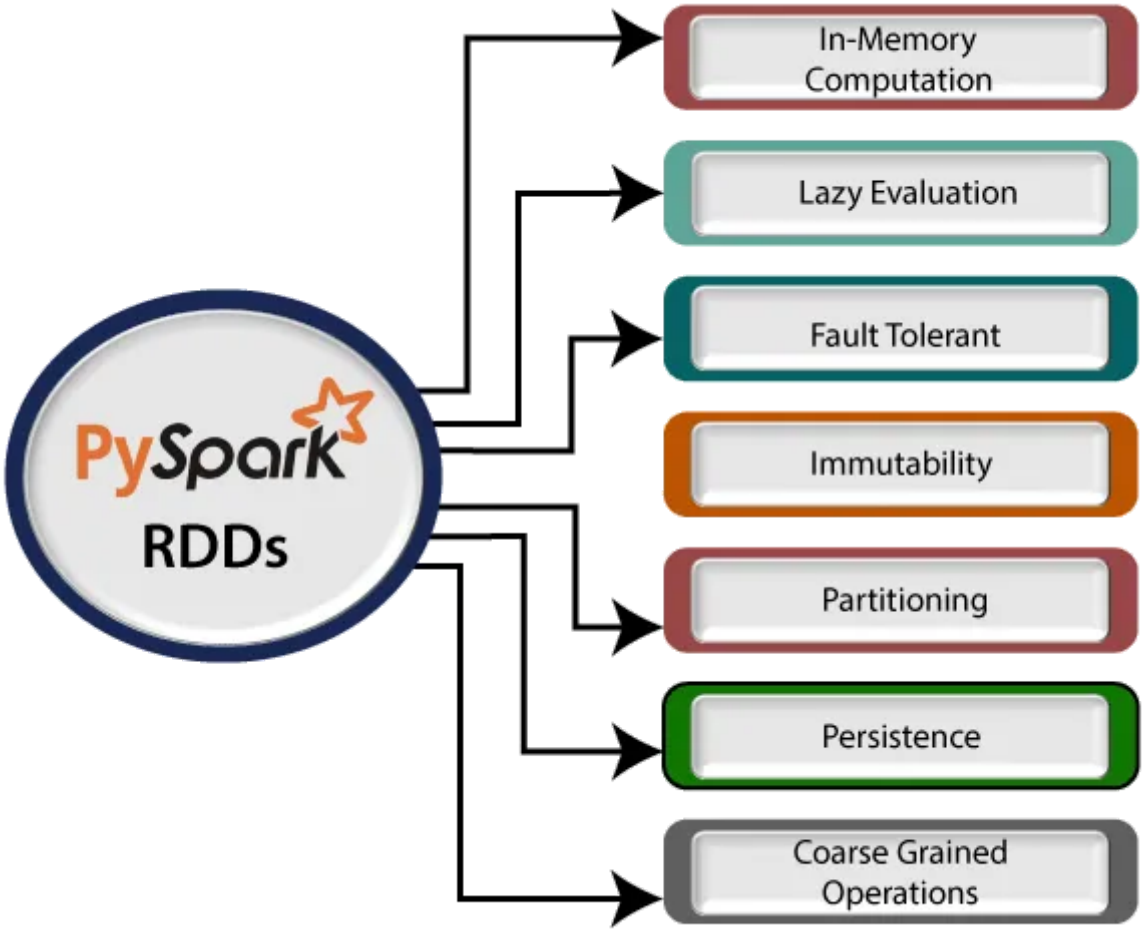
1. Spark Shell (Scala)
2. PySpark Shell (Python)

Two Levels of API

1. Low Level API : RDD's (Deprecated)
2. High Level API : Spark SQL -> DataFrame API, SQL

Spark Core API's (RDD's)

> Resilient Distributed Dataset, are the building blocks of any spark application

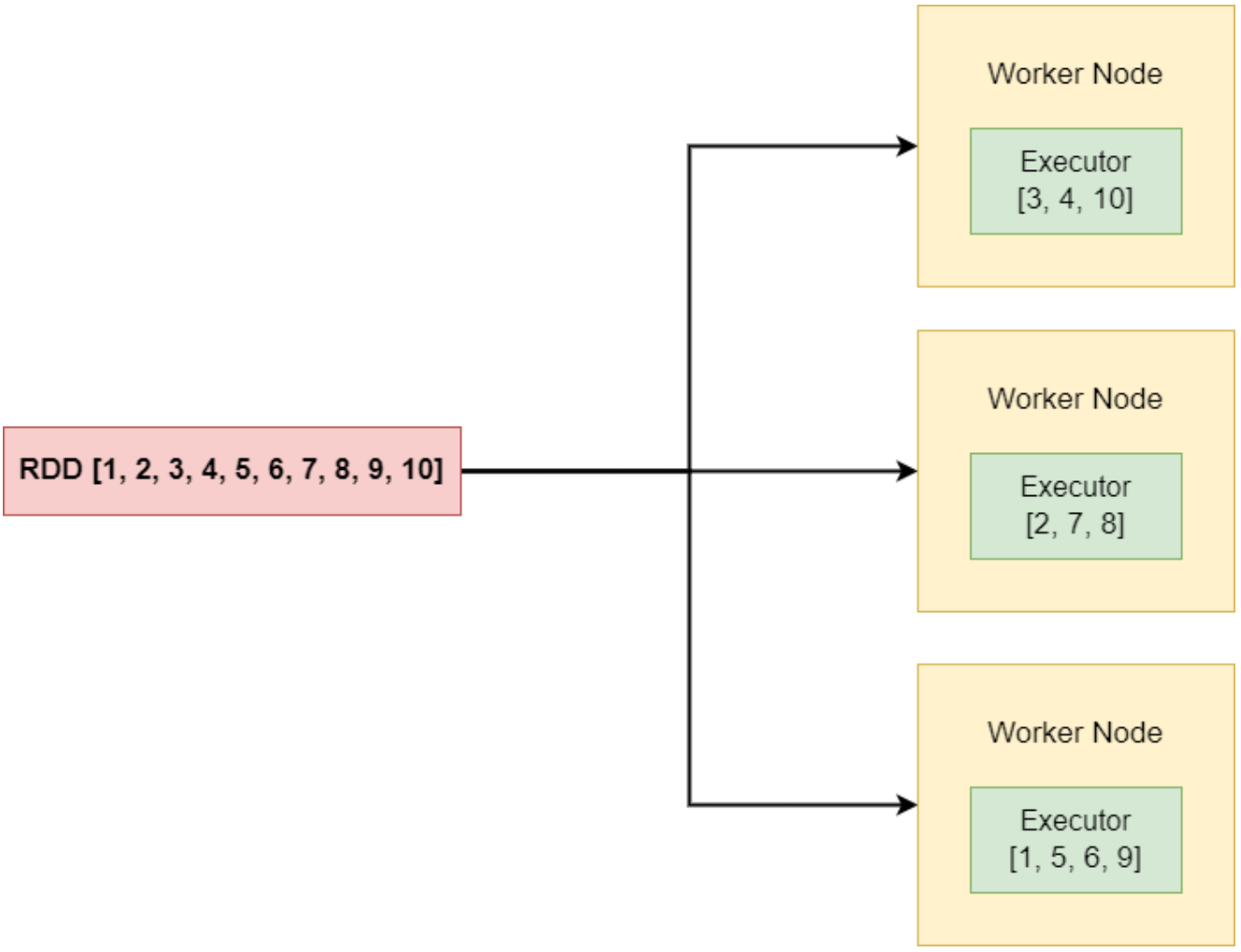


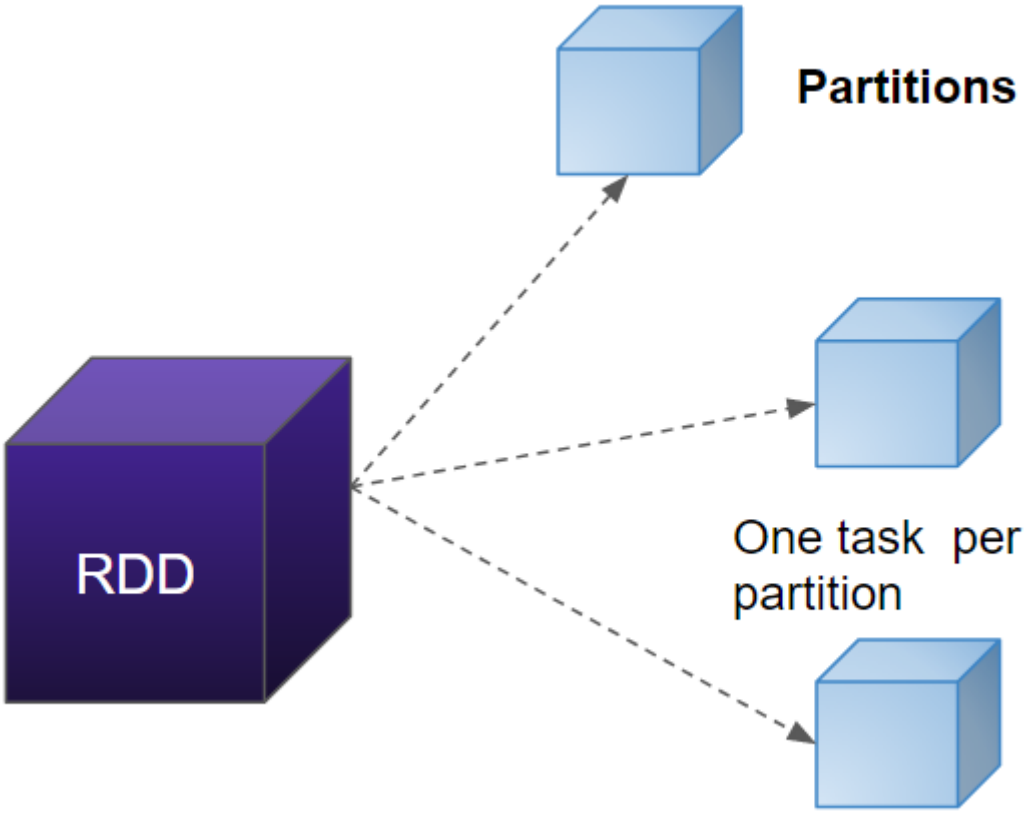
- In the world of apache spark its all about RDD, Create RDD --> Transformation --> Store Results

Partitions

RDD is a collection of objects that is partitioned and distributed across nodes in a cluster

> A partition in spark is a logical division of data stored on a node in the cluster

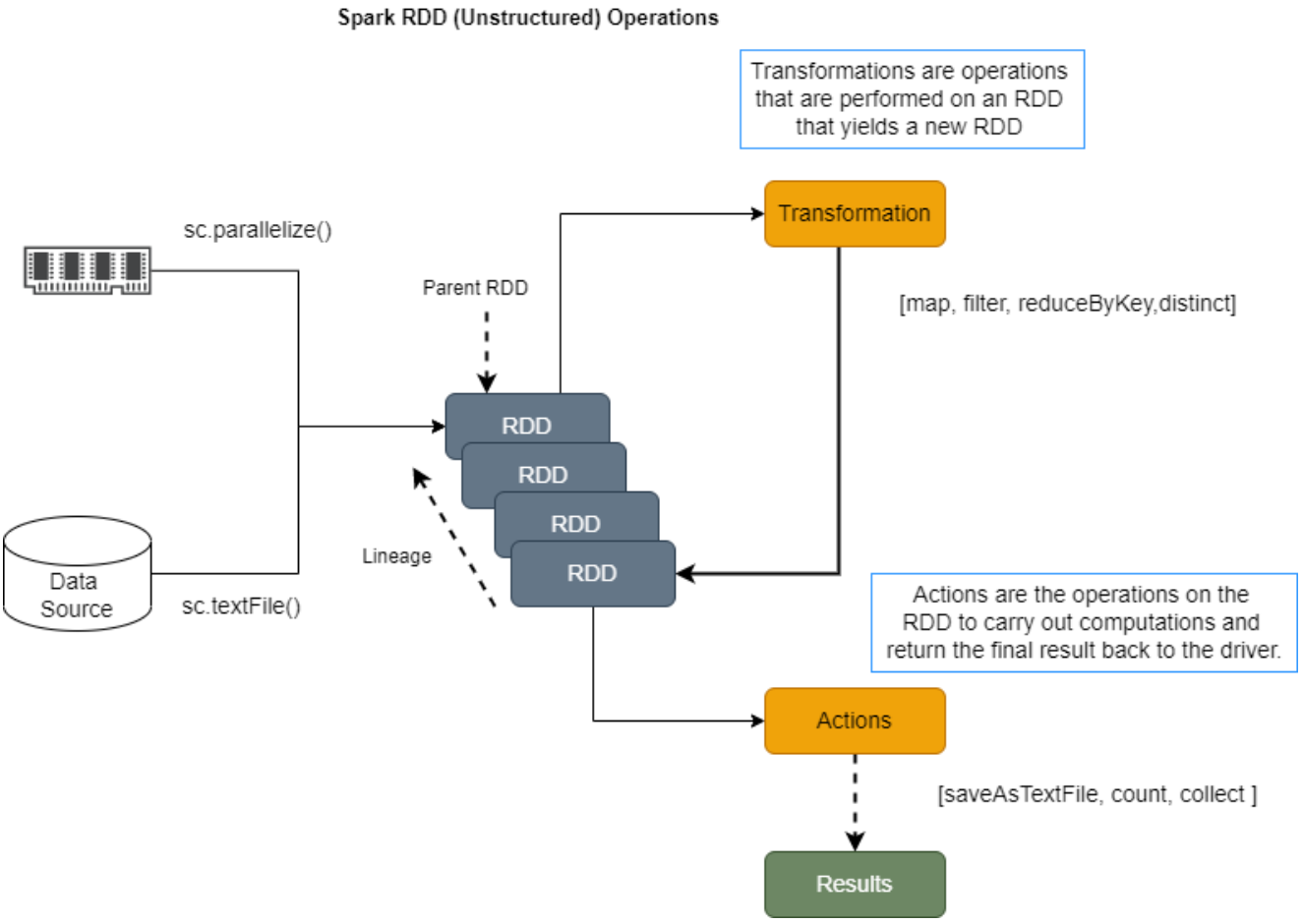

Worker Node
Executor
[3, 4, 10]
Worker Node
Executor
[2, 7, 8]
Worker Node
Executor
[1, 5, 6, 9]
RDD [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]


Partitions
RDD
One task per
partition

RDD Creation and Operation

There are two popular ways to create an RDD

1. Create an RDD from Collections L = [1,2,3,4,5,6]
2. Create an RDD from external source



Operations

Once an RDD is created, we can perform two types of operations

1. Transformations
2. Actions

Transformations

- Transformation creates a new RDD from an existing RDD by applying a certain transformation logic

- E.g map(), filter(), union(), groupByKey(), repartition()

Actions

Actions are the operations in the RDD to carry out final computation

E.g : count(), saveAsTextFile(), .take(), collect()