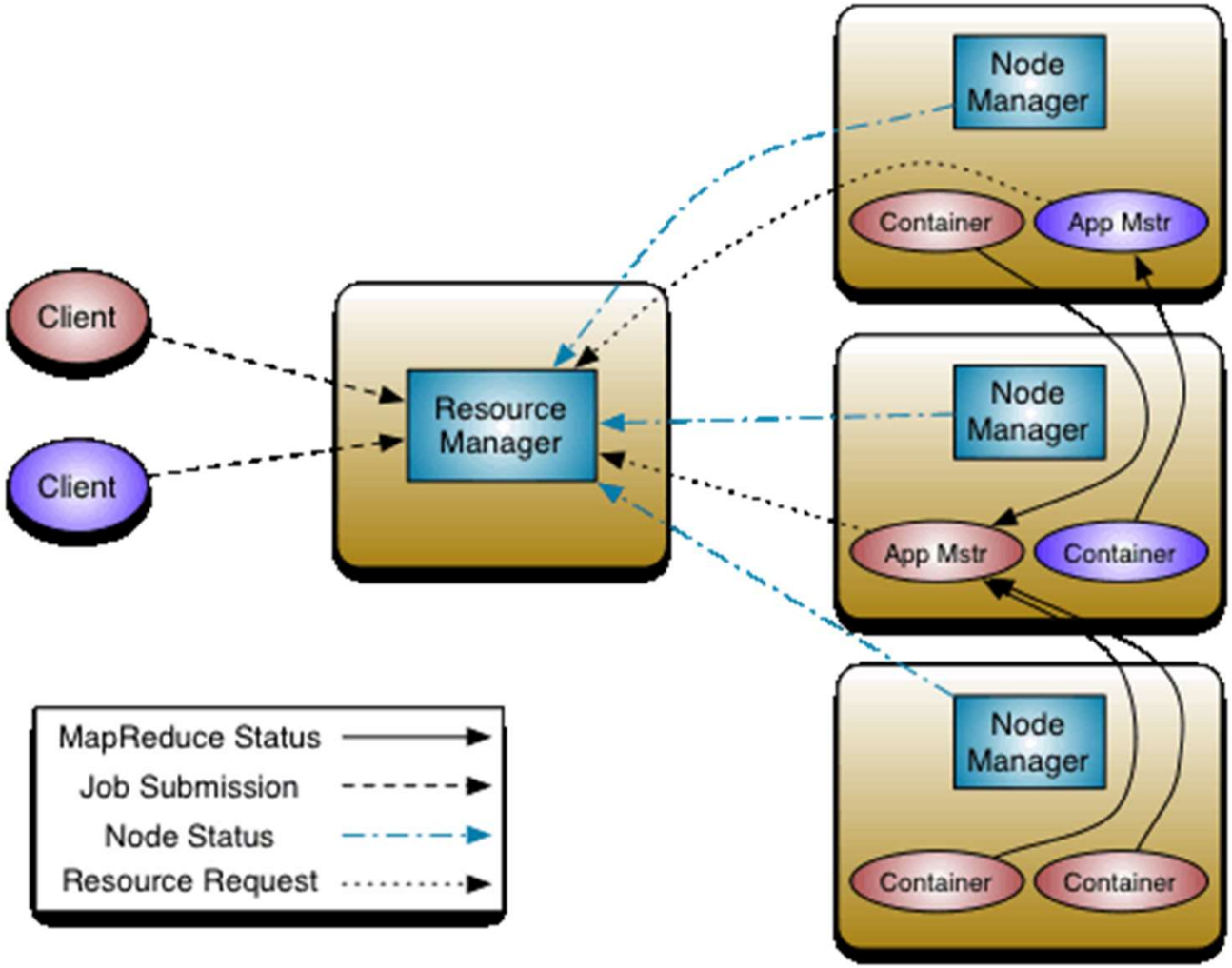
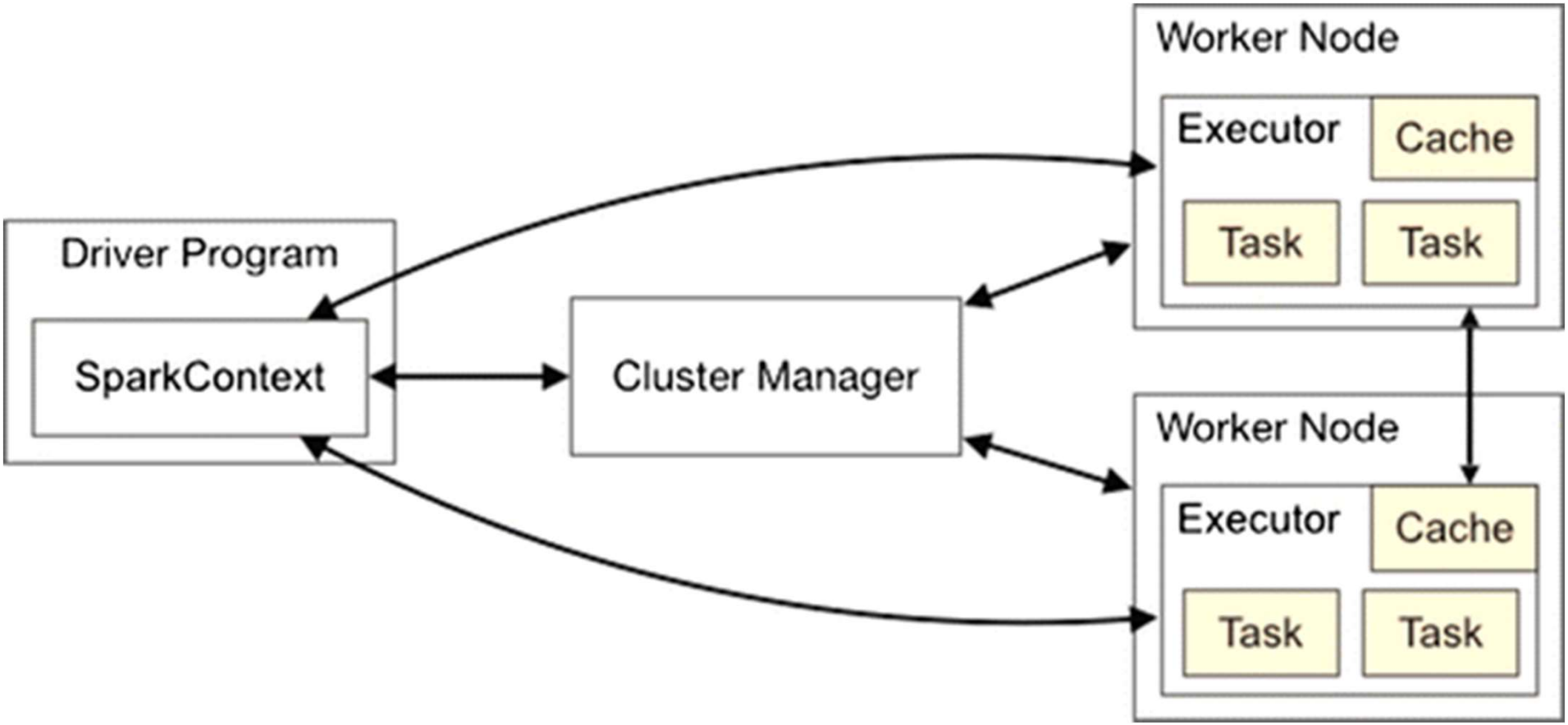
An Introduction to Apache Spark

Spark Architecture

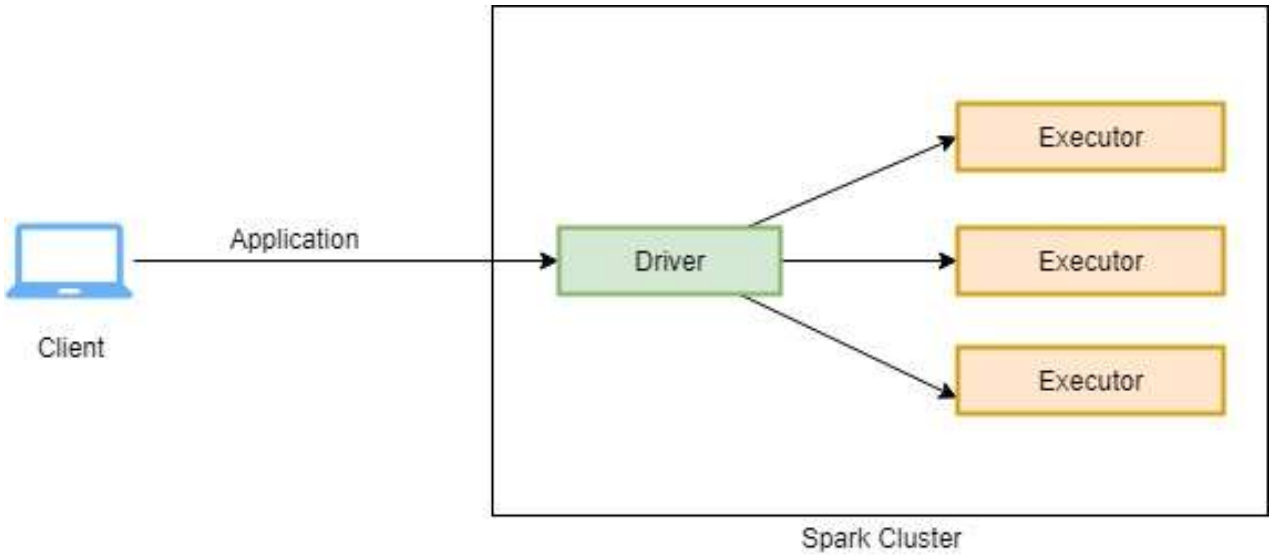


Spark Architecture

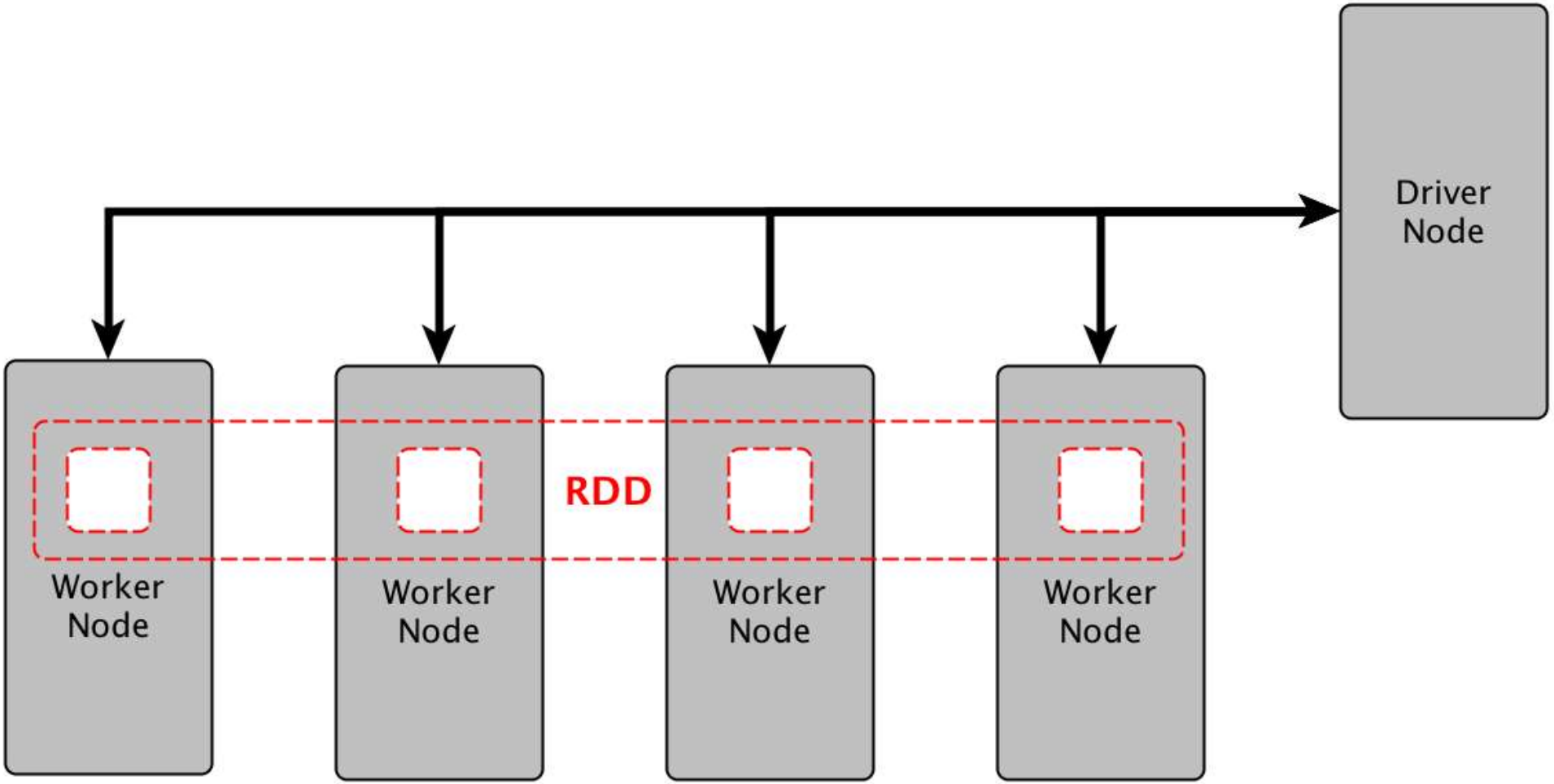


Apache Spark Execution

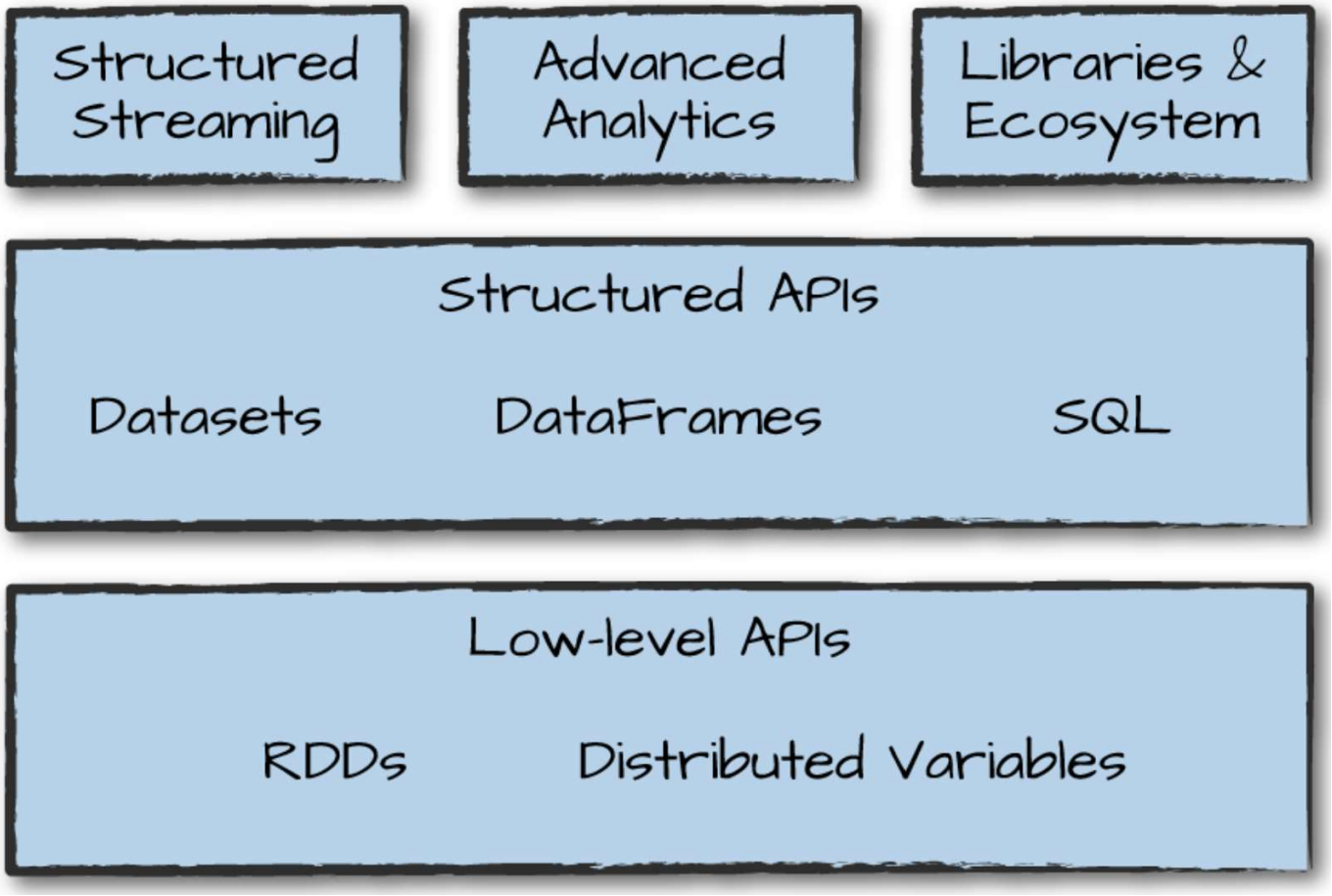
- For every application submitted on spark cluster, spark creates a dedicated Driver process and bunch of Executor processes.
- Driver process is responsible for analyzing, distributing, scheduling and monitoring of executor processes.
- Whereas the executor process is only responsible for running the task they were assigned by drivers and reporting the status back to the driver.



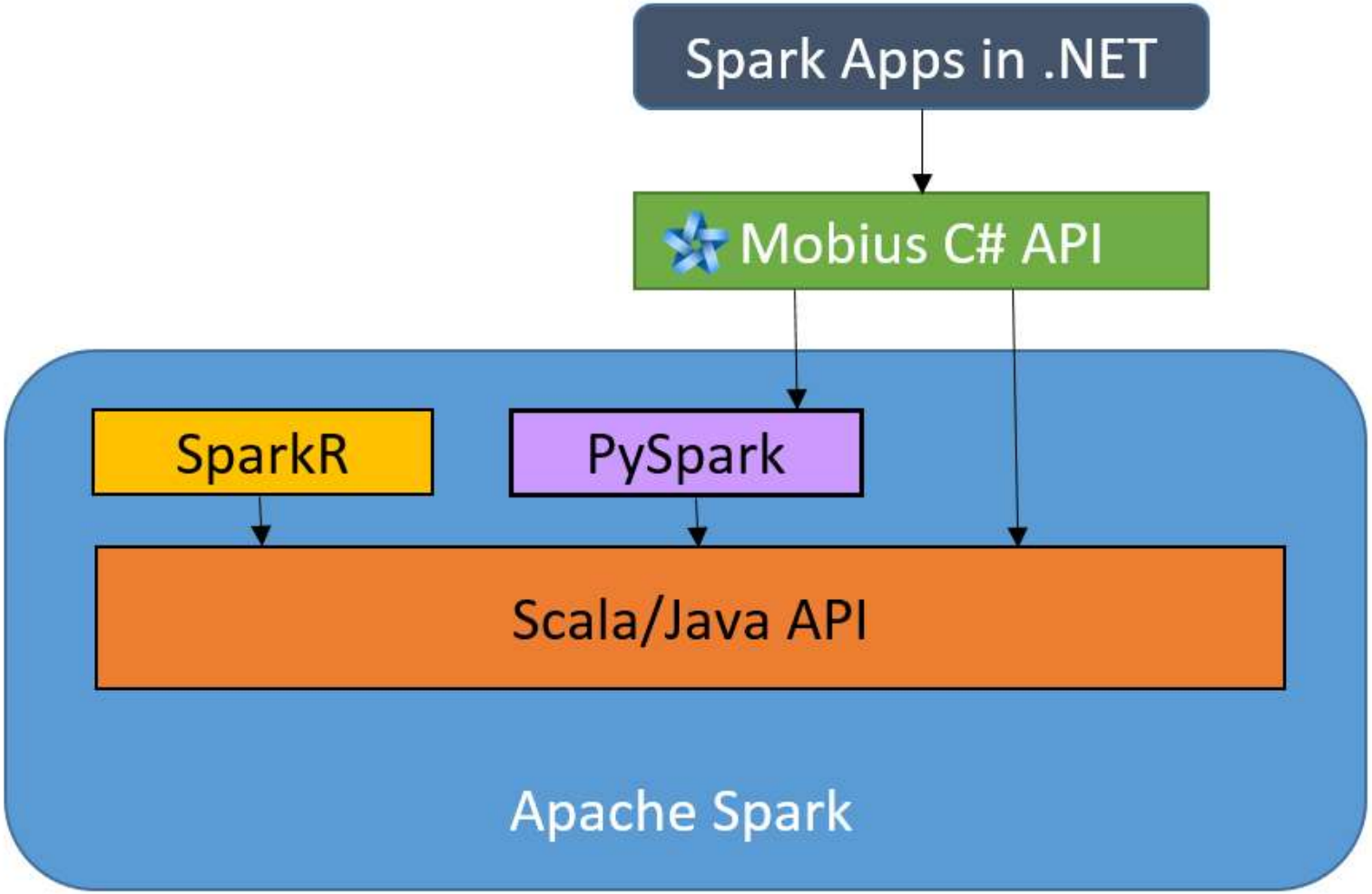
Apache Spark Execution



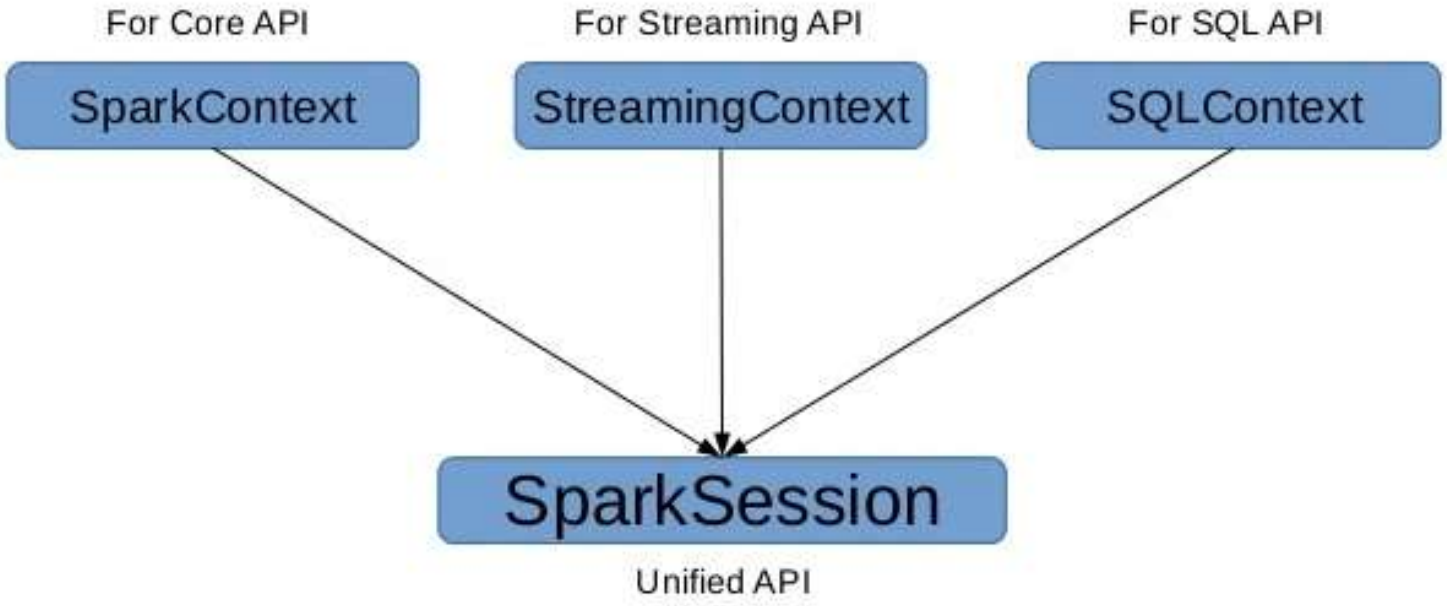
Spark's Language APIs



Spark's Language APIs



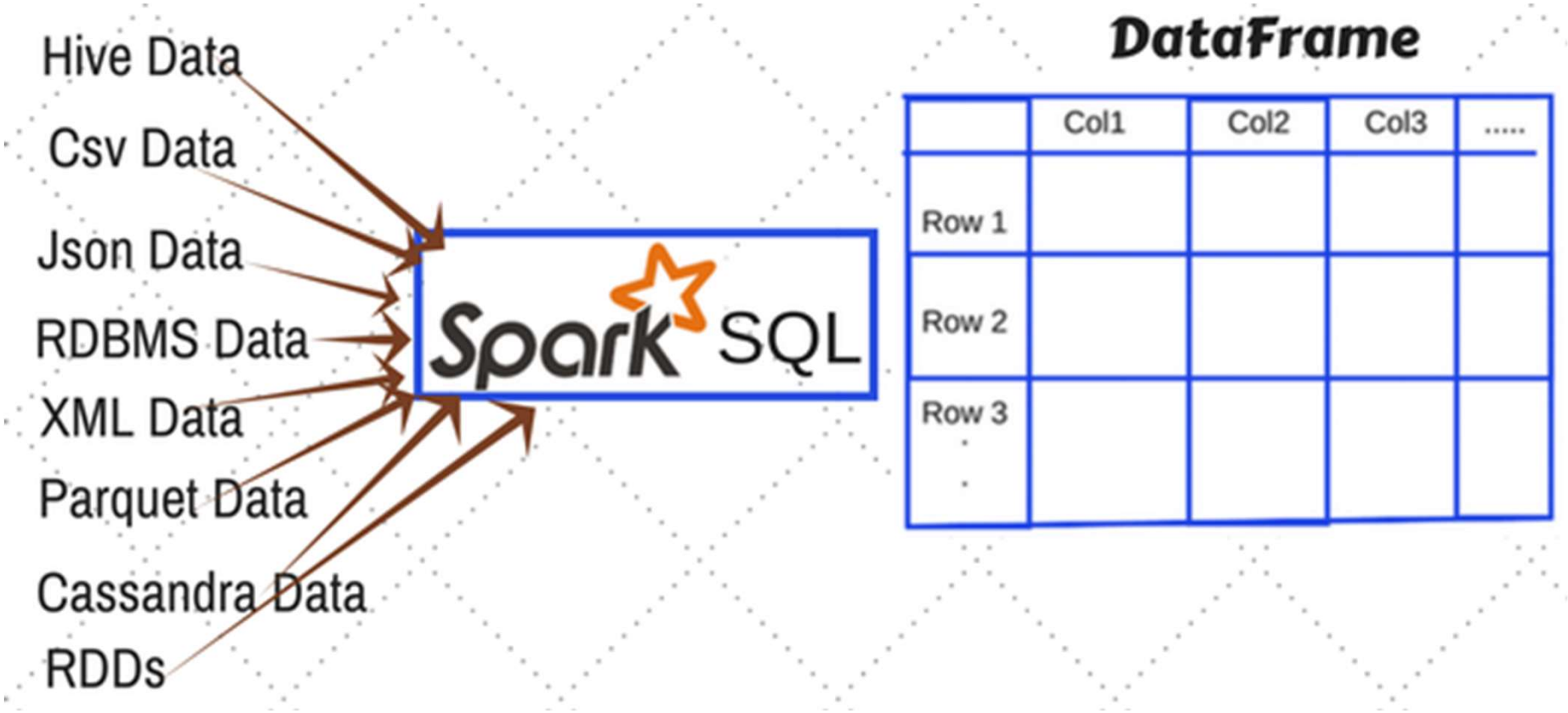
The SparkSession

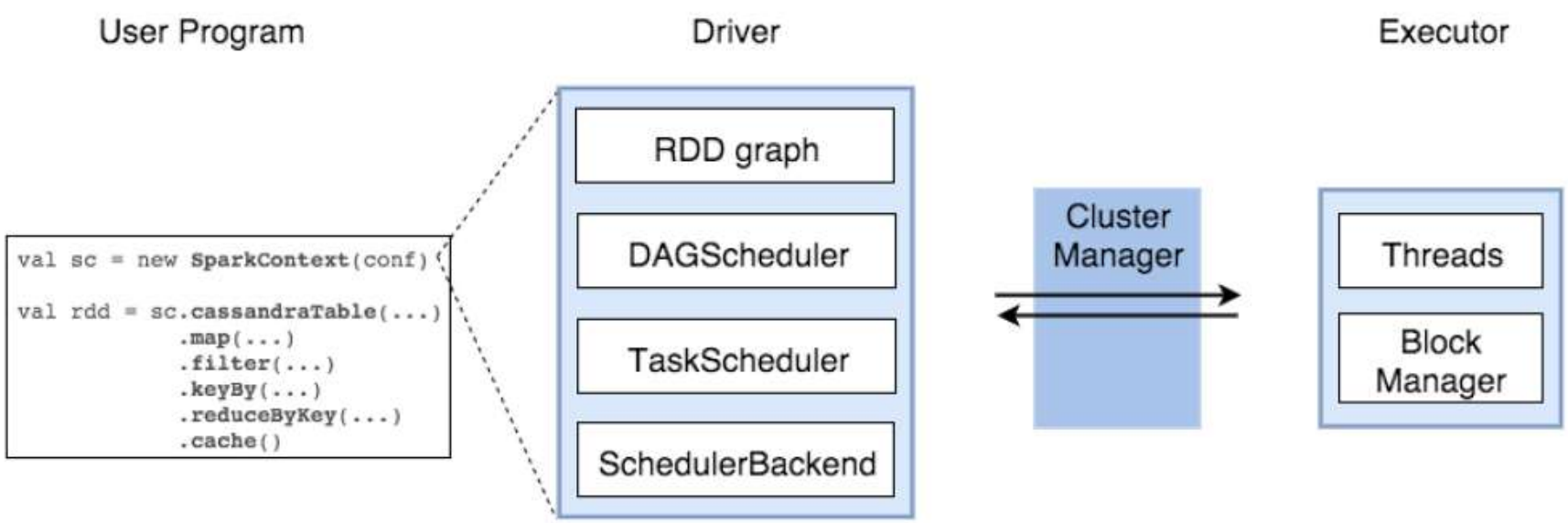


DataFrames

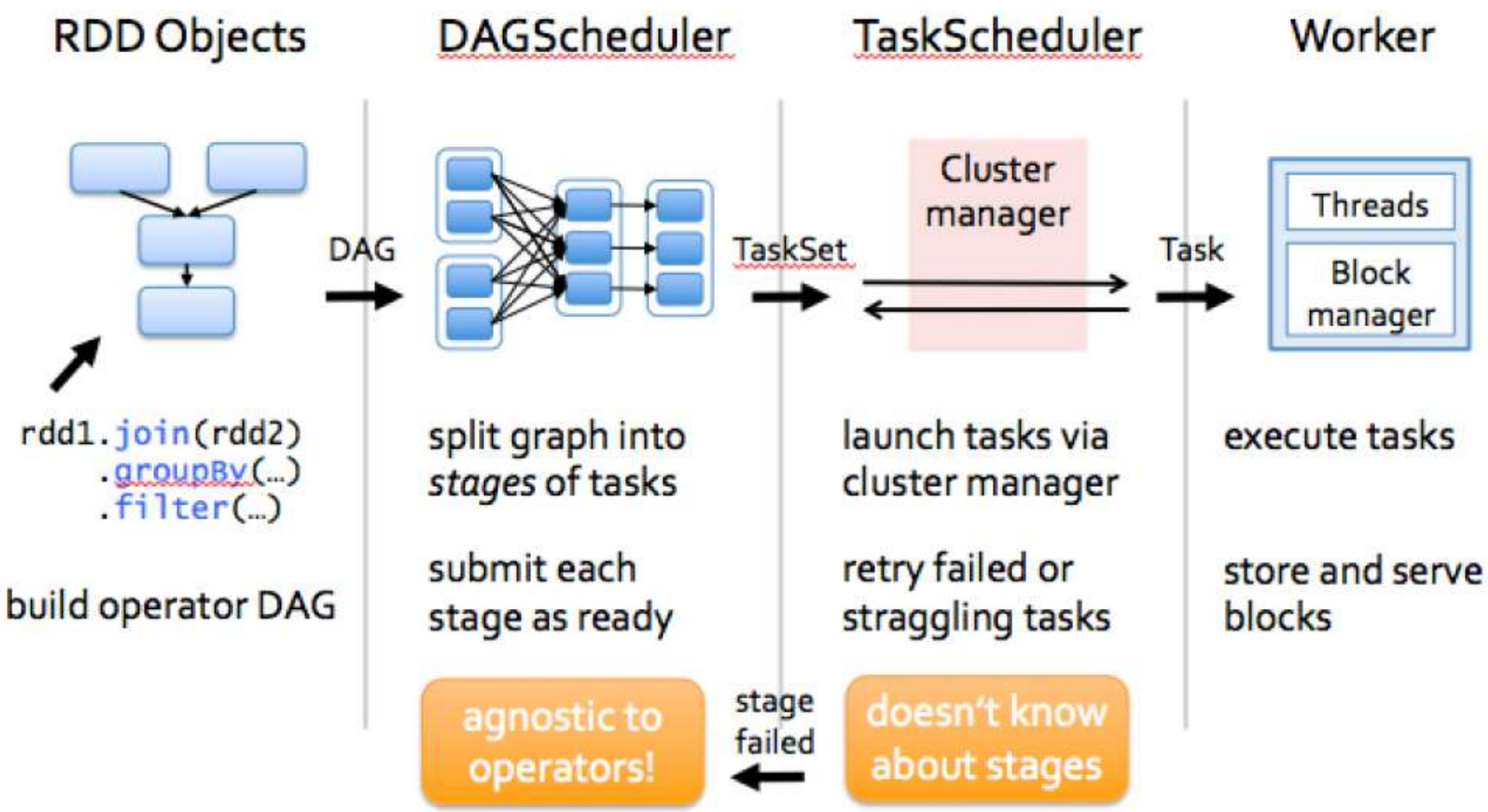
- In Apache Spark, a DataFrame is a distributed collection of rows
- It has below characteristics:
  - Immutable in nature
    - We can create DataFrame RDD once but can’t change it.
  - Lazy Evaluations
    - Which means that a task is not executed until an action is performed.
  - Distributed

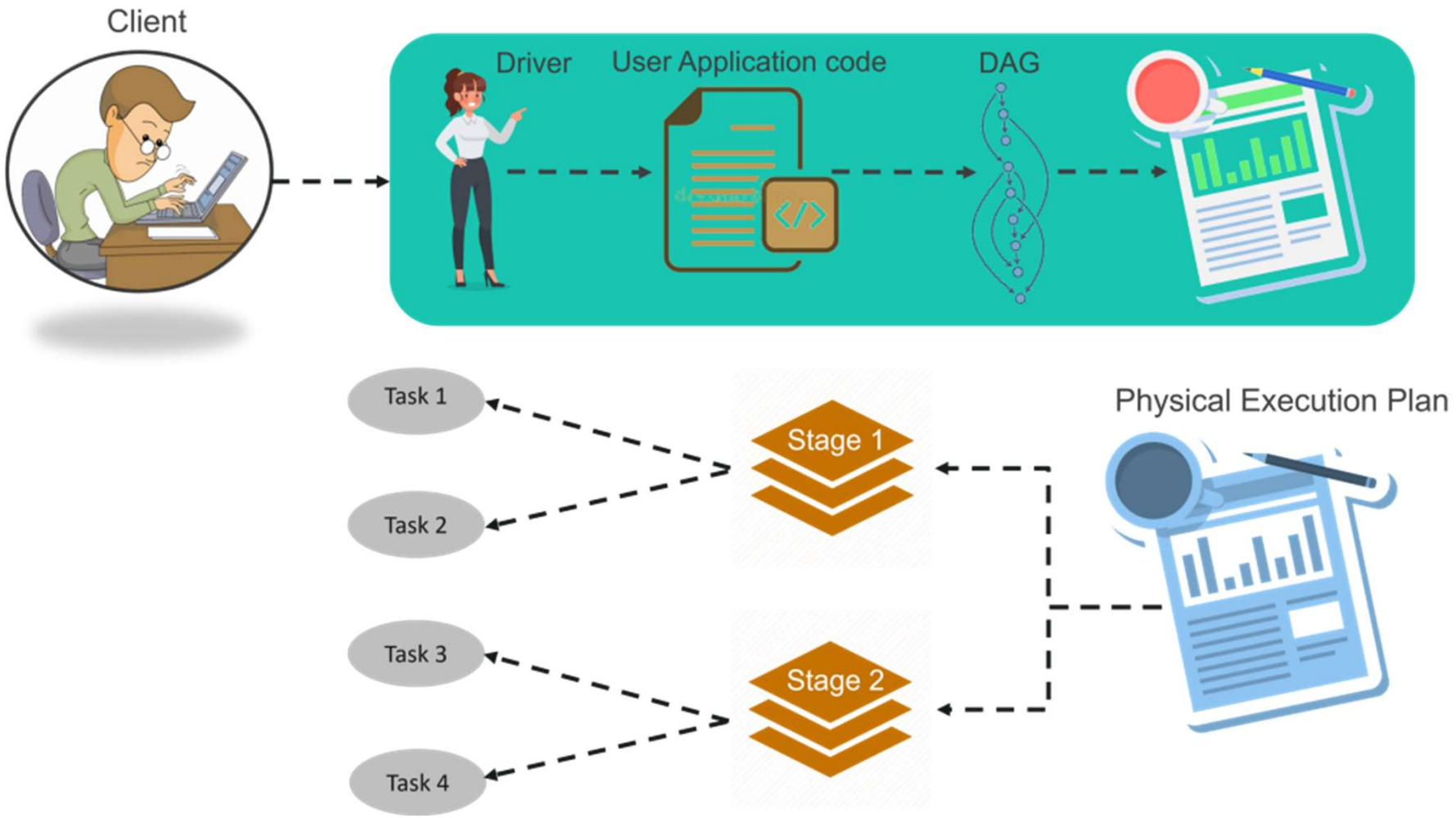
Ways to create DataFrame in Spark




User Program
Driver
Executor
val sc = new SparkContext(conf)
val rdd = sc.cassandraTable(...)
.map(...)
.filter(...)
.keyBy(...)
.reduceByKey(...)
.cache()
RDD graph
DAGScheduler
TaskScheduler
SchedulerBackend
Cluster Manager
Threads
Block Manager

How Sparks work?




Client
Driver
User Application code
DAG
Physical Execution Plan
Task 1
Task 2
Task 3
Task 4
Stage 1
Stage 2

Catalyst Optimizer

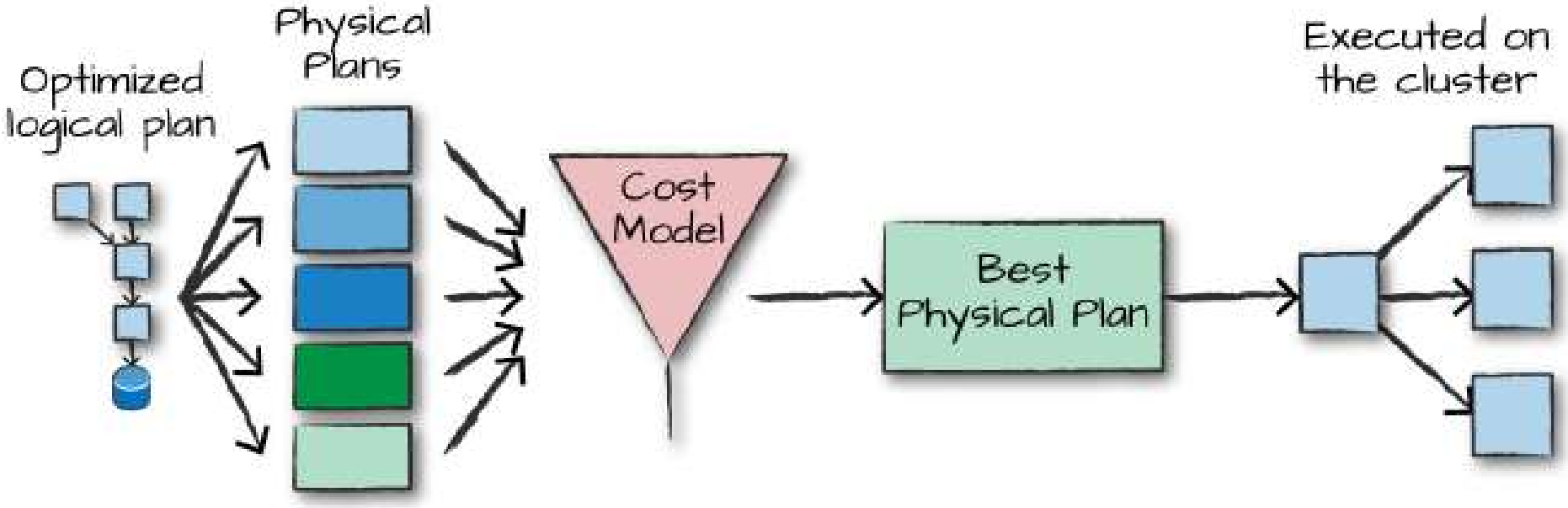


Figure 4-3. The physical planning process

Catalyst Optimizer

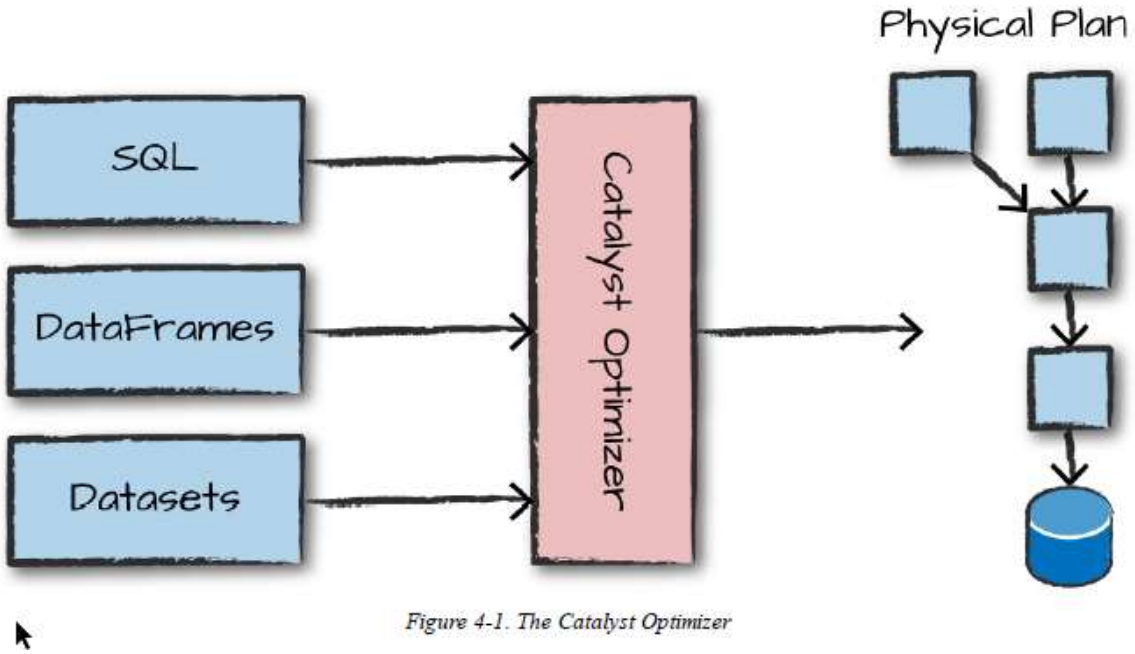
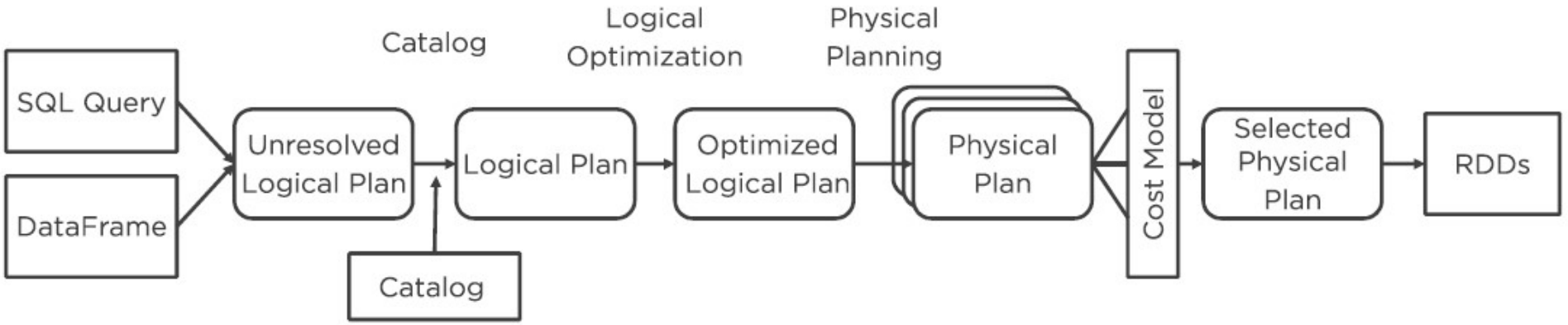


Figure 4-1. The Catalyst Optimizer

Execution


SQL Query
DataFrame
Unresolved Logical Plan
Catalog
Catalog
Logical Plan
Logical Optimization
Optimized Logical Plan
Physical Planning
Physical Plan
Cost Model
Selected Physical Plan
RDDs

Thanks