# Apache Spark

1. Apache Spark is processing engine
2. It's alternative to Map-reduce
3. General purpose engine
4. Basic unit which holds the data in Spark is RDD(Resilient Distributed Dataset)

## Disadvantages Of Mapreduce:

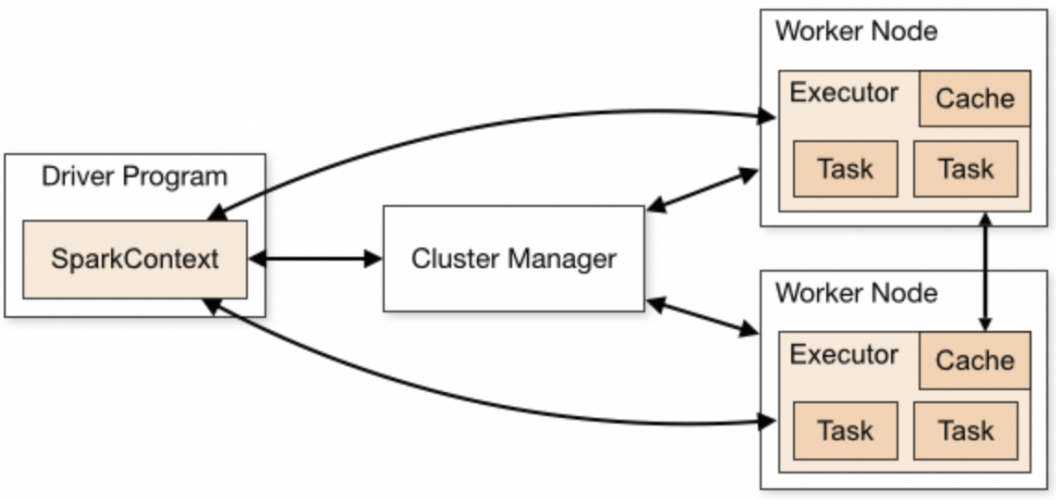
1. Mapreduce programs are written in Java
2. Development in Mapreduce was lengthy
3. Mapreduce has high latency cause it involves more DISK read-write operations than spark
4. Provides Batch-processing. No real time processing
5. Doesn't support Caching Data

## Spark Components & API :

**Components:** SparkSQL , SparkMLlib , Spark Structured Streaming, GraphX

**API's:** Scala,Python,Java,R,SQL

## Spark Architecture:



Cluster Manager : Responsible for acquiring resources on a cluster
Driver Program : It requests for resources to the cluster manager
Cluster Manager launches executors on worker nodes as requested by driver program

==========

- Spark uses a master-slave architecture
- Main work is to distribute the data across the cluster & process data in parallel over nodes
- Computes the data **in-memory(**ram**)**
- Spark provides low-latency performance because it involves less DISK read-write than mapreduce

Spark Application consists of driver program and executors

- Driver program initiates the execution of program
- Runs the main() of the application
- Creates SparkContext.
    - SparkContext is an entrypoint to spark
    - Rdd is created using SparkContext

Execution Mode: Cluster mode & Client mode

Cluster Mode : Driver is launched inside the cluster .
Client Mode : Driver is launched in the client machine (not in the cluster); so when machine goes down , the program ends

Executors : It is Java process. Launched in Worker node. It registers itself with driver program in the beginning. The executors are dynamically added / removed during task execution

Task : It is chunk of data that sent to executor.
Job is a process of parallel computation. It involves execution of multiple tasks.

❖ **RDD** - Resilient Distributed Dataset

➢ Resilient : Fault-tolerance i.e. Ability to recover failure

➢ Distributed : partitioned across worker nodes

➢ Dataset : Collection of records are stored in form of csv / json/ text etc.

- Rdd is a Data structure in Apache Spark
- Using SparkContext we can create RDD
- RDD is immutable
- Rdd is partitioned across worker nodes

In spark , there are 2 types of operations(are applied on RDD):

- Transformations
- Actions

**Transformations :**

- When we create Rdd , after that we can apply many transformations on it as per requirements
- Transformations - Loading the file / Operations on RDDs
- Transformations are Lazy i.e when we perform transformations; No actual computations has happened . Only Diagrams are created in backend, named DAG(Directed Acyclic Graph)
- Operations on Rdd - map() , filter() , reduceByKey()
- When we apply transformation on RDD , it will return a new RDD. existing RDD will remain same as RDD is immutable

**Actions:**

- Actions are not lazy i.e. as soon as an action is called , everything starts to execute
- collect()