# **Foundation**

## **HDFS:**

Hadoop Distributed File System is a distributed file system designed to store and manage large volumes of data across multiple commodity hardware nodes in a distributed computing environment. It is part of the Apache Hadoop project and is one of the core components of the Hadoop ecosystem.

## **Map Reduce:**

MapReduce is a programming model and processing framework for parallel and distributed computation of large datasets across clusters of computers. It was popularized by Google and later adopted and implemented in the Apache Hadoop project as a core component of the Hadoop ecosystem.

# **Spark**

Apache Spark is a lightning-fast cluster computing technology, designed for fast computation. It is based on Hadoop MapReduce and it extends the MapReduce model to efficiently use it for more types of computations, which includes interactive queries and stream processing. The main feature of Spark is its in-memory cluster computing that increases the processing speed of an application.

## **Features of Apache Spark:**

**Speed −** Spark helps to run an application in Hadoop cluster, up to 100 times faster in memory, and 10 times faster when running on disk. This is possible by reducing number of read/write operations to disk. It stores the intermediate processing data in memory.

**Supports multiple languages −** Spark provides built-in APIs in Java, Scala, or Python. Therefore, you can write applications in different languages. Spark comes up with 80 high-level operators for interactive querying.

**Advanced Analytics −** Spark not only supports ‘Map’ and ‘reduce’. It also supports SQL queries, Streaming data, Machine learning (ML), and Graph algorithms.

## **Components of Spark or Spark EcoSystem:**

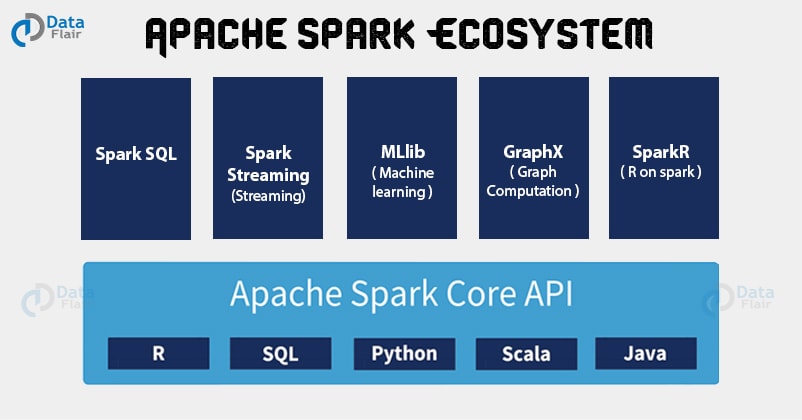
**Apache Spark Core:** Spark Core is the underlying general execution engine for spark platform that all other functionality is built upon. It provides In-Memory computing and referencing datasets in external storage systems.

**Spark SQL:** Spark SQL is a component on top of Spark Core that introduces a new data abstraction called SchemaRDD, which provides support for structured and semi-structured data.

**Spark Streaming**: Spark Streaming leverages Spark Core's fast scheduling capability to perform streaming analytics. It ingests data in mini-batches and performs RDD (Resilient Distributed Datasets) transformations on those mini-batches of data.

**MLlib (Machine Learning Library):** MLlib is a distributed machine learning framework above Spark because of the distributed memory-based Spark architecture. It is, according to benchmarks, done by the MLlib developers against the Alternating Least Squares (ALS) implementations. Spark MLlib is nine times as fast as the Hadoop disk-based version of Apache Mahout (before Mahout gained a Spark interface).

**GraphX:** GraphX is a distributed graph-processing framework on top of Spark. It provides an API for expressing graph computation that can model the user-defined graphs by using Pregel abstraction API. It also provides an optimized runtime for this abstraction.



**Note: Apache Spark does not offer Cluster Management and Storage Management.**

## **Spark Architecture**

The Spark architecture depends upon two abstractions:

* Resilient Distributed Dataset (RDD)
* Directed Acyclic Graph (DAG)

**Resilient Distributed Dataset (RDD)**

# **PySpark**

PySpark is the Python API for Apache Spark. It is an open-source distributed system that is used for big data processing. It also supports Spark’s features like Spark DataFrame, Spark SQL, Spark Streaming, Spark MLlib and Spark Core.