What is Hadoop?

Definition:- Apache Hadoop is a framework designed for the processing of big data sets distributed over large sets of machines with commodity hardware. The basic ideas have been taken from the Google File System (GFS or GoogleFS) and the MapReduce. The main advantage of Apache Hadoop is its design for scalability, i.e it is easy to add new hardware to extend an existing cluster which means of storage and computation power. The hardware available with Hadoop is highly reliable than other solutions. It can manage huge and reliable cluseters without investing in expensive hardware.

Modules of Hadoop: -

1. **Hadoop Common**
2. **Hadoop HDFS**
3. **Hadoop YARN**
4. **Hadoop MapReduce**

Hadoop Common: - This module of Hadoop manages all the information from the other modules associated with different data source.

Hadoop HDFS: - A distributed file system similar to the one developed by Google under the name GFS(Google File System).

Hadoop YARN: This module provides the job scheduling resources used by the MapReduce framework.

Hadoop MapReduce:- A framework designed to process huge amount of data.

Advantages of Hadoop

1. Hadoop framework allows the user to quickly write and test distributed systems. It is efficient, and it automatic distributes the data and work across the machines and in turn, utilizes the underlying parallelism of the CPU cores.
2. Hadoop does not rely on hardware to provide fault-tolerance and high availability **(FTHA)**, rather Hadoop library itself has been designed to detect and handle failures at the application layer.
3. Servers can be added or removed from the cluster dynamically and Hadoop continues to operate without interruption.
4. Another big advantage of **Hadoop** is that apart from being open source, it is compatible on all the platforms since it is **Java** based.

Hadoop Operation Modes

Once you have downloaded Hadoop and configured, you can operate your Hadoop cluster in one of the three supported modes:

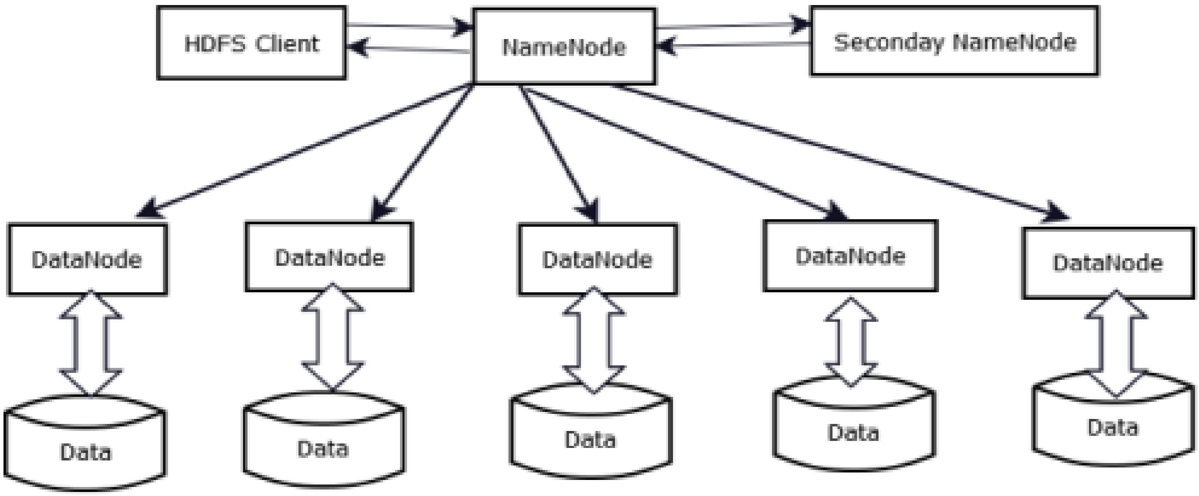
1. **Local/Standalone Mode**: After downloading Hadoop in your system, by default, it is configured in a standalone mode and can be run as a single java process.

2. **Pseudo Distributed Mode**: It is a distributed simulation on single machine. Each Hadoop daemon such as hdfs, yarn, MapReduce etc., will run as a separate java process. This mode is useful for development.

3. **Fully Distributed Mode**: This mode is fully distributed with minimum two or more machines as a cluster.

HDFS ARCHITECTURE

HDFS (Hadoop Distributed File System) is, as the name already states, a distributed file system that runs on commodity hardware. Like other distributed file systems it provides access to files and directories that are stored over different machines on the network transparently to the user application.



HDFS-Architecture

HDFS Architecture consisted by the following parts:-

1. HDFS Client
2. NameNode
3. DataNode
4. Seconday NameNode
5. Storage Data Machine

HDFS Client: - Hadoop client is an interface used to communicate with the Hadoop File System. There are different types of clients available with Hadoop to perform different tasks. The basic file system client HDFS is used to connect to a Hadoop and perform basic file related tasks.

NameNode: - The NameNode serves all metadata operations on the file system like creating, opening, closing or renaming files and directories. Therefore it manages the complete structure of the file system. The fact that the whole cluster has only one NameNode makes the complete architecture very simple but also introduces a single point of failure (SPOF).

DataNode:- A file is broken up into one or more data blocks and these data blocks are stored on one or more DataNodes, hence the client receives the list of data blocks from the NameNode and can later on contact the DataNodes directly in order to read or write the data.

Storage Data Machine: - In Hadoop architecture all NameNode is attached with Data Storage machine which store huge data for big-data analysis.

Secondary NameNode: - It is in Hadoop specially dedicated node in HDFS cluster whose main function is to take checkpoints of the file stystem metadata present on NameNode. It is a helper to the primary NameNode but not replace for primary NameNode.