Main component of Hadoop 2.x are

HDFS

YARN

MapReduce

The most important aspect of Hadoop is that both HDFS and MapReduce are designed with each other in mind and each are co-deployed such that there is a single cluster and thus pro­vides the ability to move computation to the data not the other way around. Thus, the storage system is not physically separate from a processing system.

HDFS:-

HDFS is a distributed file system that provides high-throughput access to data. It provides a limited interface for managing the file system to allow it to scale and provide high throughput. HDFS creates multiple replicas of each data block and distributes them on computers throughout a cluster to enable reliable and rapid access. A file consists of many blocks (large blocks of 64MB and above).The main components of HDFS are as described below:

NameNode :-is the master of the system. It maintains the name system (directories and files) and manages the blocks which are present on the DataNodes.

DataNodes :-are the slaves which are deployed on each machine and provide the actual stor­age. They are responsible for serving read and write requests for the clients.

Secondary NameNode :is responsible for performing periodic checkpoints. In the event of NameNode failure, you can restart the NameNode using the checkpoint.

MapReduce:-

MapReduce is a framework for performing distributed data processing using the MapReduce programming paradigm. In the MapReduce paradigm, each job has a user-defined map phase (which is a parallel, share-nothing processing of input; followed by a user-defined reduce phase where the output of the map phase is aggregated). Typically, HDFS is the storage system for both input and output of the MapReduce jobs.

The main components of MapReduce are as described below:

JobTracker :is the master of the system which manages the jobs and resources in the clus­ter (TaskTrackers). The JobTracker tries to schedule each map as close to the actual data being processed i.e. on the TaskTracker which is running on the same DataNode as the underlying block.

TaskTrackers: are the slaves which are deployed on each machine. They are responsible for running the map and reduce tasks as instructed by the JobTracker.

JobHistoryServer: is a daemon that serves historical information about completed applications. Typically, JobHistory server can be co-deployed with Job­Tracker, but we recommend to run it as a separate daemon.

YARN

YARN is bring into Hadoop 2.x as a resource manager that was created by separating the processing engine and resource management capabilities of MapReduce as it was implemented in Hadoop

YARN is often called the operating system of Hadoop because it is responsible for managing and monitoring workloads, maintaining a multi-tenant environment, implementing security controls, and managing high availability features of Hadoop.

At a high level, Spring YARN provides three different components,

YarnClient, YarnAppmaster and YarnContainer which together can be called a Spring YARN Application.

YarnClient :- It is used to communicate with YARN’s Resource Manager. This provides management actions like submitting new application instances, listing applications and killing running applications.

YarnContainer: It is a worker that does the heavy lifting of what a YARN application will actually do.

YarnAppmaster : It is a communicating with YARN Resource Manager and starts and stops YarnContainers accordingly.