**The Components of Hadoop 2.x**

Hadoop 2.x has the following three Major Components:

* HDFS
* YARN
* MapReduce

**HDFS :** HDFS comprises of 3 important components-NameNode, DataNode and Secondary NameNode. HDFS operates on a Master-Slave architecture model where the NameNode acts as the master node for keeping a track of the storage cluster and the DataNode acts as a slave node summing up to the various systems within a Hadoop cluster.

**Name Node:**

* NameNode is the centerpiece of HDFS.
* NameNode is also known as the Master
* NameNode only stores the metadata of HDFS – the directory tree of all files in the file system, and tracks the files across the cluster.
* NameNode does not store the actual data or the dataset. The data itself is actually stored in the DataNodes.
* NameNode knows the list of the blocks and its location for any given file in HDFS. With this information NameNode knows how to construct the file from blocks.
* NameNode is so critical to HDFS and when the NameNode is down, HDFS/Hadoop cluster is inaccessible and considered down.
* NameNode is a single point of failure in Hadoop cluster.
* NameNode is usually configured with a lot of memory (RAM). Because the block locations are help in main memory.

**DataNode:**

* DataNode is responsible for storing the actual data in HDFS.
* DataNode is also known as the Slave
* NameNode and DataNode are in constant communication.
* When a DataNode starts up it announce itself to the NameNode along with the list of blocks it is responsible for.
* When a DataNode is down, it does not affect the availability of data or the cluster. NameNode will arrange for replication for the blocks managed by the DataNode that is not available.
* DataNode is usually configured with a lot of hard disk space. Because the actual data is stored in the DataNode.

**Secondary Name Node:**

NameNode is the single point of failure in HDFS, if NameNode fails entire HDFS file system is lost. So in order to overcome this, Hadoop implemented Secondary NameNode whose main function is to store a copy of FsImage file and edits log file.

**Secondary NameNode Functions:**

1. Stores a copy of FsImage file and edits log.

2. Periodically applies edits log records to FsImage file and refreshes the edits log and sends this updated FsImage file to NameNode so that NameNode doesn’t need to re-apply the EditLog records during its start up process. Thus Secondary NameNode makes NameNode start up process fast.

3. If NameNode is failed, File System metadata can be recovered from the last saved FsImage on the Secondary NameNode but Secondary NameNode can’t take the primary NameNode’s functionality.

4. Check pointing of File system metadata is performed.

**YARN :**

YARN has total three major components

ResourceManager

NodeManager

ApplicationMaster

**1) Resourse Manager**

This daemon process resides on the Master Node (not necessarily on NameNode of Hadoop)

Responsible for, Managing resources scheduling for different compute applications in an optimum way

Coordinating with two process on master node, Scheduler and Application Manager.

**2) NodeManager**

This daemon process resides on the slave nodes (runs along with DataNode daemon)

Responsible for, Managing and executing containers Monitoring resource usage (i.e. usage of memory, cpu, network etc..) and reporting it back to ResourceManager daemon, Periodically sending heart-bits to ResourceManager for its health status update

**3) ApplicationMaster**

This daemon process runs on the slave node (along with the NodeManager daemon)

It is per application specific library works with NodeManager to execute the task

The instance of this daemon is per application, which means in case of multiple jobs submitted on cluster, it may have more than one instances of ApplicationMaster on slave nodes

Responsible for,

Negotiating suitable resource containers on slave node from ResourceManager

Working with one or multiple NodeManagers to monitor task execution on slave nodes