ACD\_BDD2.3\_Session\_4\_Assignment\_3

Apache Hadoop 2.0 represents a generational shift in the architecture of Apache Hadoop.

* Hadoop 2.x supports two Name Nodes at a time one node is active and another is standby node
* Active Name Node handles the client operations in the cluster
* StandBy Name Node manages metadata same as Secondary Name Node in Hadoop 1.x
* When Active Name Node is down, Standby Name Node takes over and will handle the client operations then after
* MapReduce 2.x Daemons (YARN)

*YARN is a re-architecture of Hadoop that allows multiple applications to run on the same platform. With YARN, applications run “in” Hadoop, instead of “on” Hadoop:*

*The fundamental idea of YARN is to split up the two major responsibilities of the JobTracker and TaskTracker into separate entities.*

*In Hadoop 2.0, the JobTracker and TaskTracker no longer exist and have been replaced by three components:*

* **ResourceManager**:
  + This daemon process runs on master node (may run on the same machine as name node for smaller clusters)
  + It is responsible for getting job submitted from client and schedule it on cluster, monitoring running jobs on cluster and allocating proper resources on the slave node
  + It communicates with Node Manager daemon process on the slave node to track the resource utilization
  + It uses two other processes named *Application Manager*and *Scheduler*for MapReduce task and resource management

.

* **NodeManager:**
  + This daemon process runs on slave nodes (normally on HDFS Data node machines)
  + It is responsible for coordinating with Resource Manager for task scheduling and tracking the resource utilization on the slave node
  + It also reports the resource utilization back to the Resource Manager
  + It uses other daemon process like Application Master and Container for MapReduce task scheduling and execution on the slave node

.

* **ApplicationMaster:**

An ApplicationMaster runs a specific YARN job and is responsible for negotiating resources from the ResourceManager and also working with the NodeManager to execute and monitor Containers.

The actual data processing occurs within the Containers executed by the ApplicationMaster.

A Container grants rights to an application to use a specific amount of resources (memory, cpu etc.) on a specific host.