

Yarn is the Platform which supports map reduce and other application like messaging etc;

Yarn mainly consists of Resource Manager(allocation of resources) and Application Master(managing and monitoring Map reduce and other application)

It is responsible for allocating data nodes to a job and for running application

ResourceManager has two main components: Scheduler and Applications Master.

Scheduler

The Scheduler is responsible for allocating resources to the various running applications subject to familiar constraints of capacities, queues etc. The Scheduler is pure scheduler in the sense that it performs no monitoring or tracking of status for the application. Also, it offers no guarantees about restarting failed tasks either due to application failure or hardware failures. The Scheduler performs its scheduling function based the resource requirements of the applications; it does so based on the abstract notion of a resource *Container* which incorporates elements such as memory, cpu, disk, network etc. In the first version, only memory is supported.

Application Master

The Application Master is responsible for the execution of a single application. It asks for containers to the Resource Scheduler (Resource Manager) and executes specific programs (e.g., the main of a Java class) on the obtained containers. The Application Master knows the application logic and thus it is framework-specific. The MapReduce framework provides its own implementation of an Application Master.

**Node Manager** (many per cluster)

It is the slave of the infrastructure. When it starts, it announces himself to the Resource Manager. Periodically, it sends an heartbeat to the Resource Manager. Each Node Manager offers some resources to the clusterThe NodeManager (NM) is YARN’s per-node agent, and takes care of the individual compute nodes in a Hadoop cluster. This includes keeping up-to date with the ResourceManager (RM), overseeing containers’ life-cycle management; monitoring resource usage (memory, CPU) of individual containers, tracking node-health, log’s management and auxiliary services which may be exploited by different YARN applications

Container

It represents a **collection of physical resources**. Also could mean CPU cores, disk along with RAM.

When an application is about to get submitted into the YARN platform, the YarnClient allocates a container from the ResourceManager, where its ApplicationMaster will run