# ApplicationMaster

ApplicationMaster is a YARN process (daemon, Java process) needed to run Spark jobs. It is responsible for:

* Negotiating resources with the YARN ResourceManager
* Monitoring the job’s progress
* Handling task failures and retries

# Lifecycle of a Spark Job on Yarn

* User runs spark-submit
* The spark-submit client talks to YARN ResourceManager to submit the application
* ResourceManager picks a node (NodeManager) on which to run the ApplicationMaster and allocates a container on it
* The chosen NodeManager and launches the container
* ApplicationMaster:
  + initlializes Spark Driver (Spark Driver and ApplicationMaster run as one process)
  + Sets up SparkContext
  + Prepares the job DAG
  + Parses job logic into stages and tasks
  + Requests containers for Executors from the ResourceManager
* ResourceManager assigns containers to various NodeManagers
* ApplicationMaster gets info from the ResourceManager about which NodeManagers will run containers.
* ApplicationMaster requests NodeManagers to start executor JVM processes inside those containers
* Executors communicate with the Driver to receive and execute tasks
* ApplicationMaster and Executors’ containers are shut down
* Logs and metrics are written