Execution modes for a Spark application

Whenever we submit a Spark application to the cluster, the Driver or the Spark App Master should get started. And the Driver will be starting N number of workers. Spark driver will be managing spark context object to share the data and coordinates with the workers and cluster manager across the cluster. Cluster Manager can be **Spark Standalone** or **Hadoop YARN** or **Mesos**. Workers will be assigned a task and it will consolidate and collect the result back to the driver. A spark application gets executed within the cluster in two different modes – one is cluster mode and the second is client mode.

Cluster Mode

In the cluster mode, the Spark driver or spark application master will get started in any of the worker machines. So, the client who is submitting the application can submit the application and the client can go away after initiating the application or can continue with some other work. So, it works with the concept of Fire and Forgets.

Client Mode

In the client mode, the client who is submitting the spark application will start the driver and it will maintain the spark context. So, till the particular job execution gets over, the management of the task will be done by the driver. Also, the client should be in touch with the cluster. The client will have to be online until that particular job gets completed.

Comparison:

Client	Cluster
Driver is launched in the same process that submitted job.	Driver is launched in a worker or Application Master in the cluster.
Need to wait to get the result when job finishes	Can quit without waiting for getting job results
Used for interactive job (shell)Good for debugging, testing	Cannot get job result back to client Useful for running long job