

## Master Node

Master

Driver

```
def main() {  
  val sc = new SparkContext()  
  val rdd = sc.makeRDD()  
  val finalRDD = rdd.transformation()  
  val result = finalRDD.action()  
}
```

dagScheduler

dagSchedulerEvent  
ProcessActor

taskScheduler

registerShuffle()

mapOutputTrackerMaster

sparkDeploySchedulerBackend

DriverActor

sc.runJob()

submitTasks()

reviveOffers()

! LaunchTask(serializedTask)

## Worker Node 1

Worker

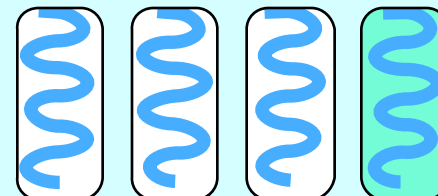
ExecutorRunner

CoarseGrainedExecutorBackend  
(Process & Actor)

executor.launchTask() =>  
executor.threadPool.execute(task)

Executor

task.run()



3 ShuffleMapTasks 1 ResultTask