

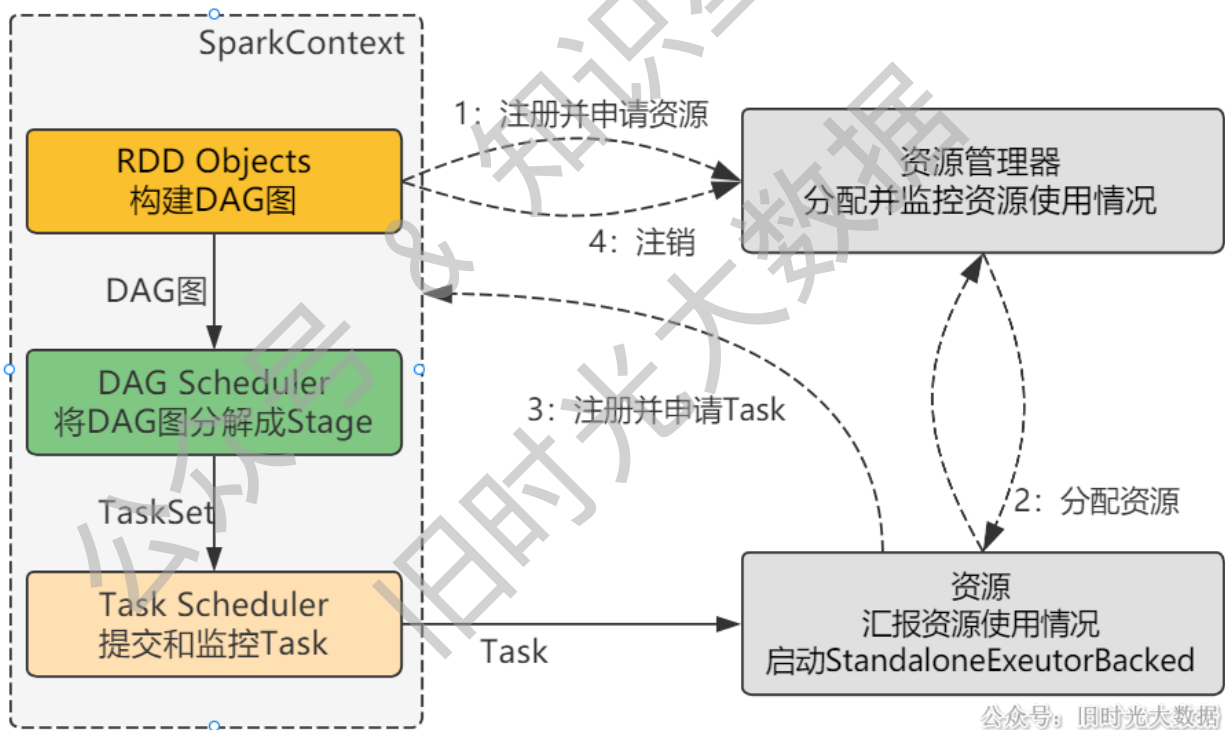
Spark的任务执行流程

可回答：1) Spark的工作流程？ 2) Spark的调度流程； 3) Spark的任务调度原理； 4) Spark的任务提交和执行流程； 5) Spark任务调度到yarn上面，流程； 6) Spark job提交过程讲一下？ 7) Spark On YARN流程，Client与Cluster模式； 8) spark执行机制

参考答案：

运行流程以SparkContext为程序运行的总入口，在SparkContext的初始化过程中，Spark会分别创建DAGScheduler作业调度和TaskScheduler任务调度两级调度模块。

- **作业调度模块**是基于任务阶段的高层调度模块，它为每个Spark作业计算具有依赖关系的多个调度阶段（通常根据shuffle来划分），然后为每个阶段构建出一组具体的任务（通常会考虑数据的本地性等），然后以TaskSets（任务组）的形式提交给任务调度模块来具体执行。
- **任务调度模块**则负责具体启动任务、监控和汇报任务运行情况。



详细的运行流程：

1) 程序提交后，Driver端的SparkSubmit进程和Master进行通信，构建Application的运行环境，创建并启动SparkContext，SparkContext向资源管理器（可以是Standalone、Mesos或YARN）注册并申请运行Executor资源；

```
1 val conf = new SparkConf()
2 conf.setAppName("test")
3 conf.setMaster("local")
4 val sc = new SparkContext(conf)
```

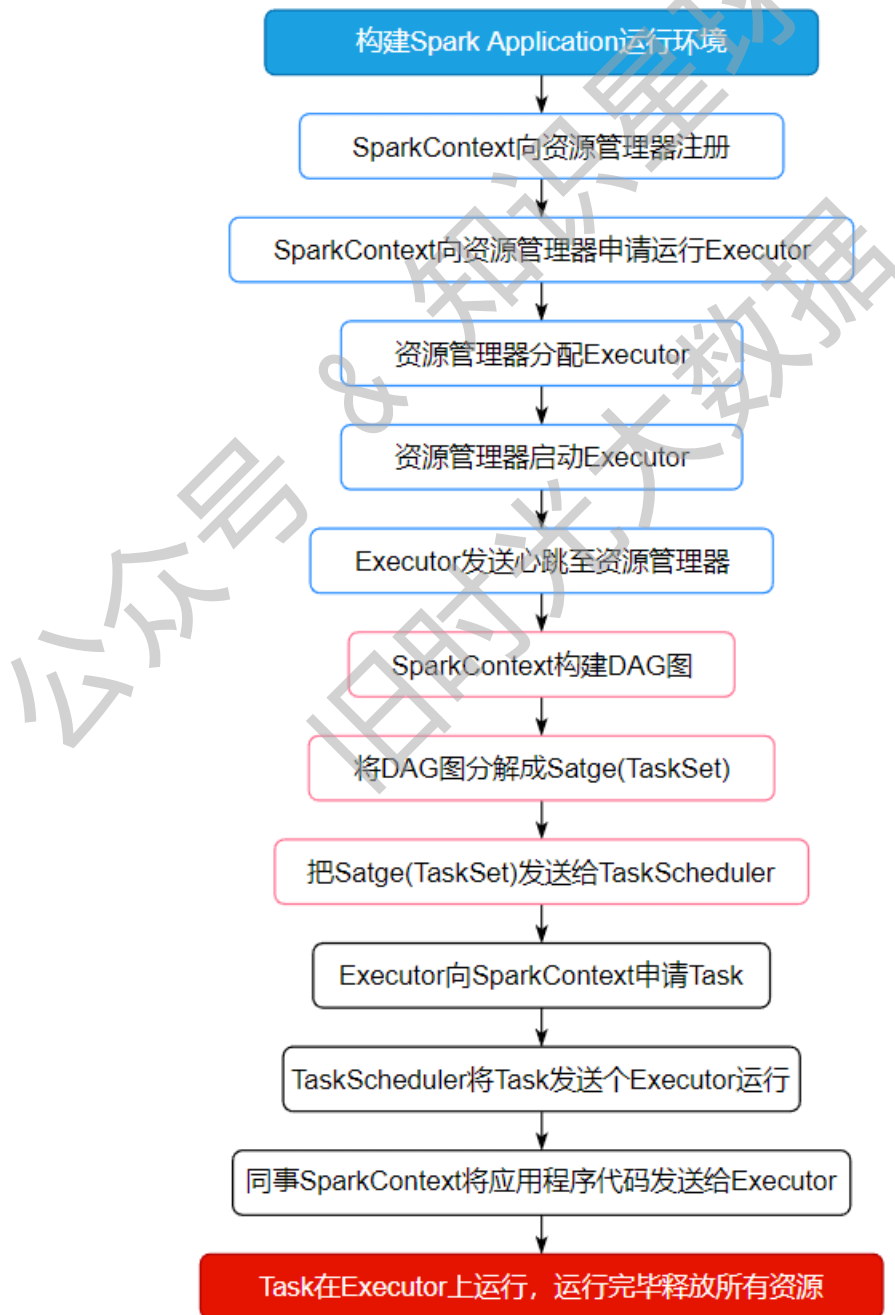
2) 资源管理器分配Executor资源并启动StandaloneExecutorBackend（Standalone模式下），Executor运行情况将随着心跳发送到资源管理器上；

3) SparkContext构建DAG图，将DAG图分解成Stage，并把Taskset发送给TaskScheduler。Executor向SparkContext申请Task，Task Scheduler将Task发放给Executor运行，同时SparkContext将应用程序代码发放给Executor；（Driver开始执行main函数，Spark查询为懒执行，当执行到Action算子时开始反向推算，根据宽依赖进行Stage的划分，随后每一个Stage对应一个Taskset，Taskset中有多个Task，查找可用资源Executor进行调度）

4) Task在Executor上运行，运行完释放所有资源。

综上：

- Spark应用被提交->SparkContext向资源管理器注册并申请资源->启动Executor
- RDD->构建DAG->DAGScheduler划分Stage形成TaskSet->TaskScheduler提交Task->Worker上的Executor执行Task



Spark运行架构特点:

- 1) 每个Application获取专属的Executor进程, 该进程在Application期间一直驻留, 并以多线程方式运行Tasks。这种Application隔离机制有其优势的, 无论是从调度角度看 (每个Driver调度它自己的任务), 还是从运行角度看 (来自不同Application的Task运行在不同的JVM中)。当然, 这也意味着Spark Application不能跨应用程序共享数据, 除非将数据写入到外部存储系统。
- 2) Spark与资源管理器无关, 只要能够获取Executor进程, 并能保持相互通信就可以了。
- 3) 提交SparkContext的Client应该靠近Worker节点 (运行Executor的节点), 最好是在同一个Rack里, 因为Spark Application运行过程中SparkContext和Executor之间有大量的信息交换; 如果想在远程集群中运行, 最好使用RPC将SparkContext提交给集群, 不要远离Worker运行SparkContext。
- 4) Task采用了数据本地性和推测执行的优化机制。

欢迎加入知识星球, 获取《大数据面试题 V4.0》以及更多大数据开发学习资料

