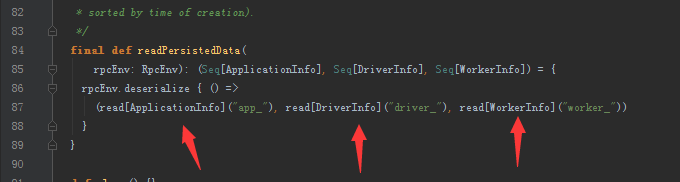
一:Master HA

1,生产环境下一般采用Zookeeper做HA,且建议为3台Master，Zookeeper会自动化管理Masters的切换

2,采用Zookeeper做HA的时候，Zookerper会负责保存整个Spark集群运行时候的元数据：Workers,Drivers,Application,Executors.



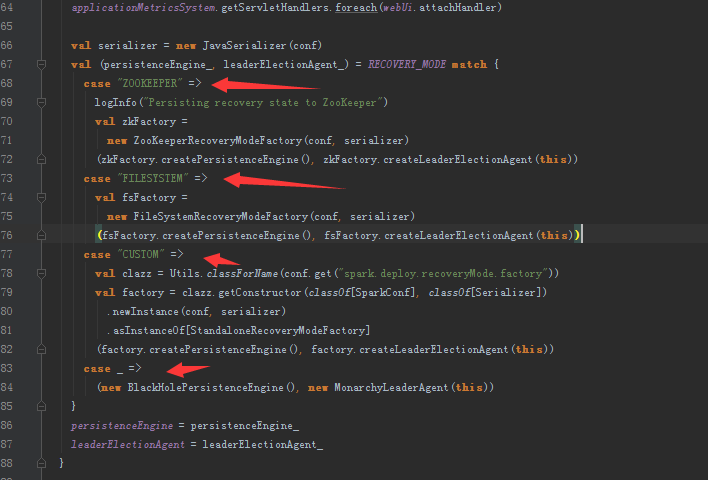
3，Zookeeper遇到当前Active级别的Master出现故障的时候会从StandByMaster中选取一台作为ActiveMaster,但是要注意，被选举后到真正成为真正的ActiveMaster之间需要从Zookeeper中获取集群当前运行状态的元数据信息并进行回复；

4，在Master切换的过程中，所有的已经在运行的程序正常运行！因为SparkApplication在运行前就已经通过Cluster Manager获得了计算资源，所以在运行时Job本身的调度和处理和Master是没有任何关系的！

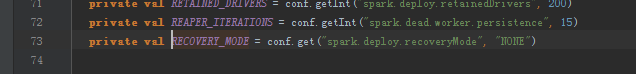
5，在Master的切换过程中唯一的影响是不能提交新的Job：一方面不能够提交新的应用程序给集群，因为只有Active Master才能节后新的程序的提交请求；另外一方面，已经运行的程序也不能够在因为Action操作触发新的Job的提交请求。

二：MasterHA的四大方式：

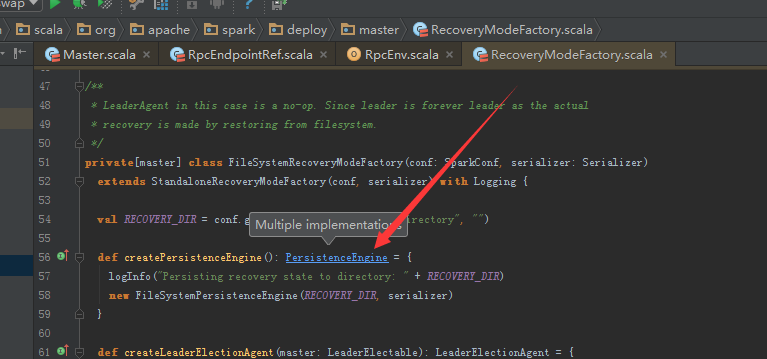
1. Master HA的四大方式分别是：ZOOKEEPER，FILESYSTEM,CUSTOM,NONE.

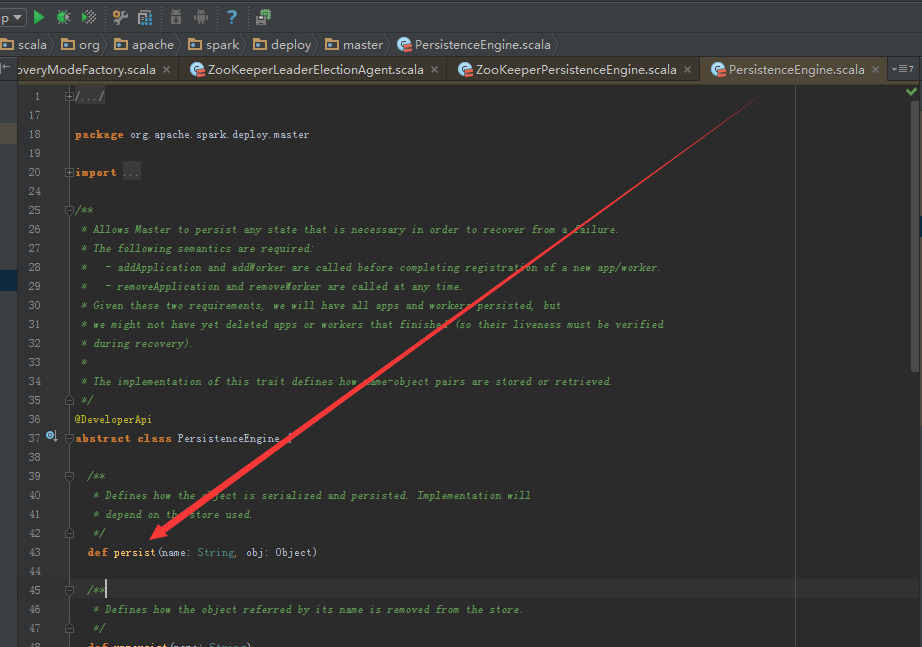


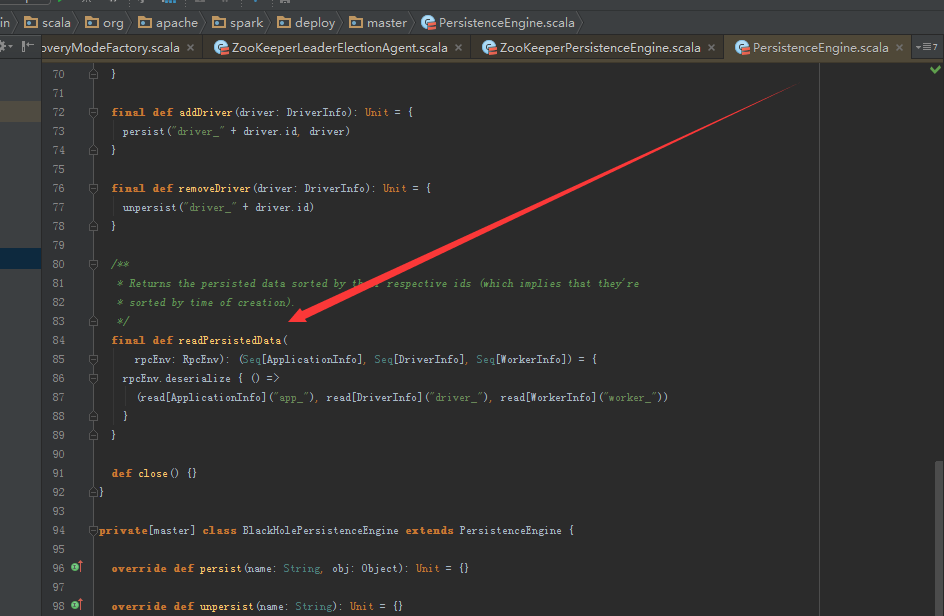
1. 需要说明的是：
   1. Zookeeper是自动管理Master
   2. FILESYSTEM的方式在Master出现故障后需要手动重新启动机器，机器启动后会立即成为Active级别的Master来对外提供服务（接受应用程序提交的请求，接受新的Job运行的请求）
   3. CUSTOM的方式允许用户自定义MasterHA的实现
   4. NONE，这是默认的情况，该方式不会持久化集群的数据。

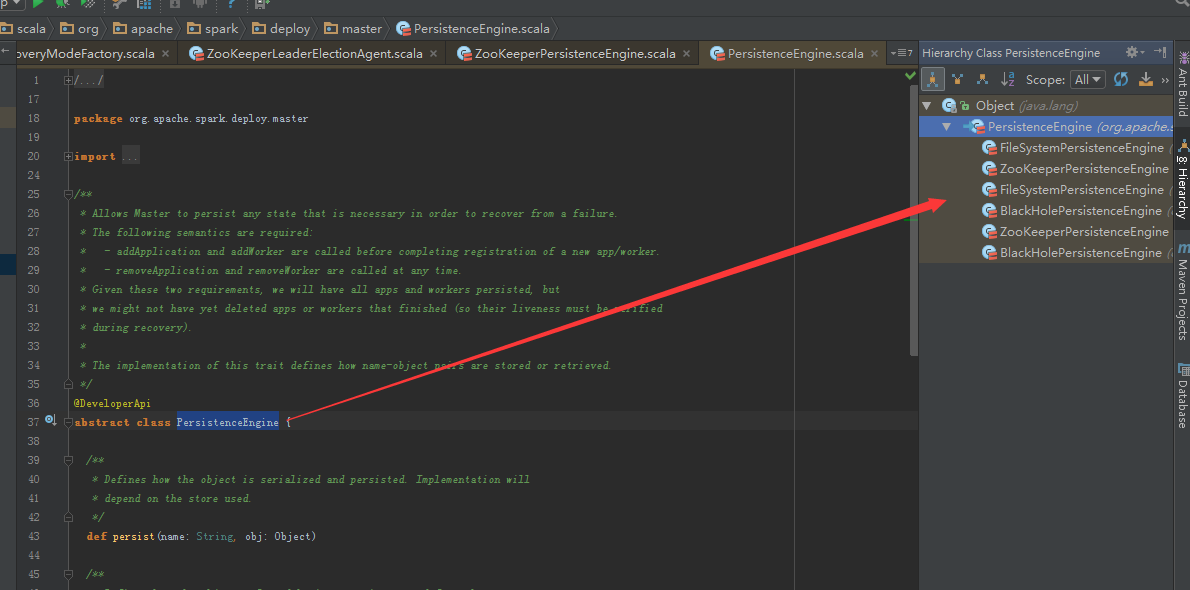


1. persistenceEngine中有一个至关重要的方法persist来实现数据持久化，readPersistData来恢复急群众的元数据。

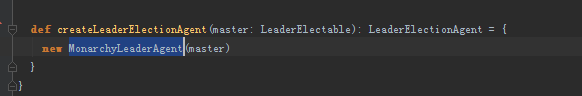




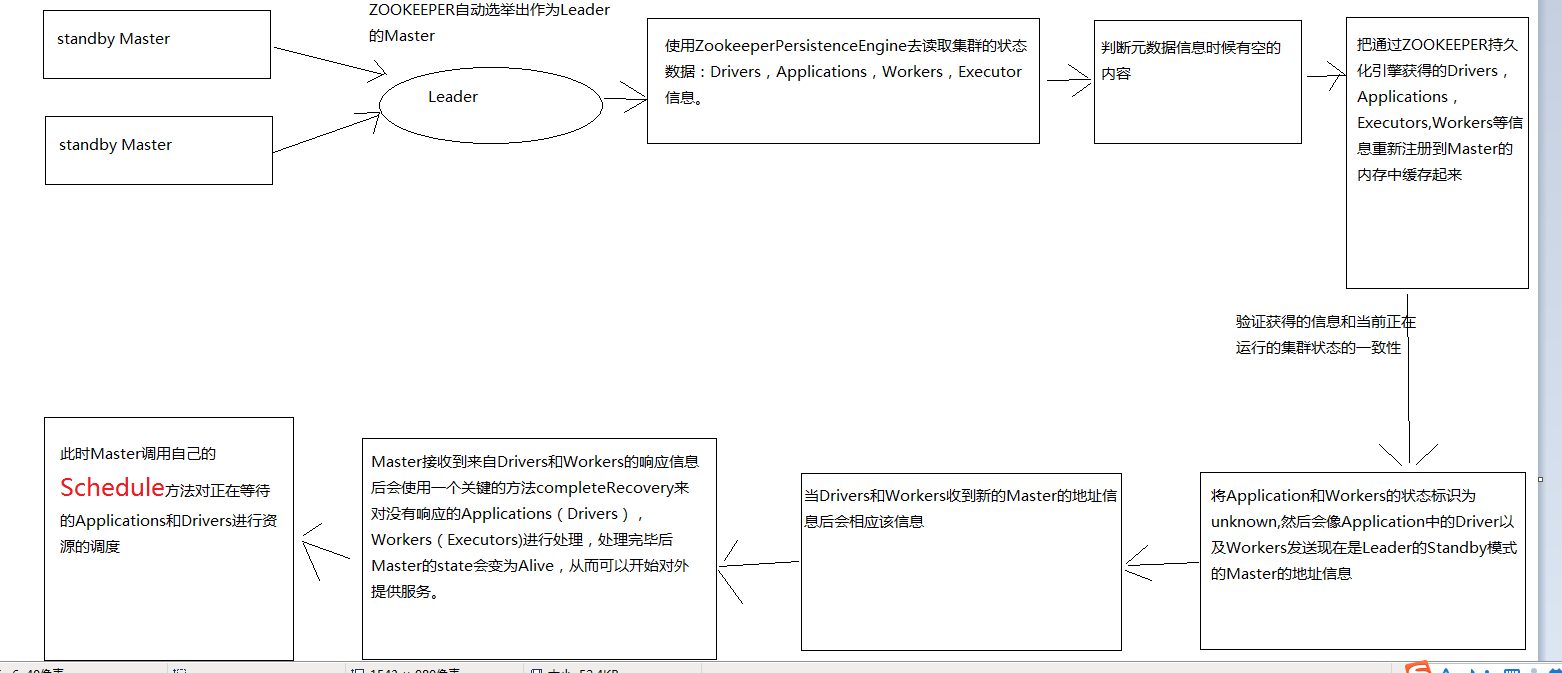




1. FIFLsYSTEM和NONE的方式均是采用MomarchyElectionAgent的方式来完成Leader选举，其实际实现是直接传入的Master设置为Leader。



三，图解（ZOOKEEPER）：



[Schedule](Schedule.docx)方法负责集群资源的调度