一:Block Manager 运行实例观察,从Application启动角度观察BlockManager

1. 在Application启动的时候会在SparkEnv中注册BlockManagerMaster以及MapOutputTracker，其中：
   1. BlockManagerMaster：对整个集群的Block数据进行管理的；
   2. MapOutputTrackerMaster：跟踪所有的Mapper输出的；
2. BlockManagerMasterEndpoint本身是一个消息体，会负责通过远程消息通讯的方式去管理所有节点的BlockManager；（BlockManagerMaster在构造的时候，内部有一个BlockManagerMasterEndpoint，BlockManagerMaster执行通讯的时候通过BlockManagerMasterEndpoint来执行的）；
3. 每启动一个ExecutorBackend都会实例化BlockManager并通过远程通讯的方式注册给BlockManagerMaster：实际上是Executor中的BlockManager在启动的时候注册给了Dirver上的BlockManagerMasterEndpoint；
4. MemoryStore是BlockManager中专门负责内存数据存储和读写的类；
5. DiskStore：BlockManager中专门负责基于磁盘的数据读写的类；
6. DiskBlockManager：管理Logical Block与Disk上的physical Block之间的映射关系并负责磁盘的文件的创建，读写等；

二：从Job运行的角度来观察BlockManager

1. 首先通过MemoryStore来存储广播变量；
2. 在Driver中是通过BlockManagerInfo来管理集群中每个ExecutorBackend中的BlockManager中的元数据信息的；
3. 当改变了具体的ExecutorBackend上的Block信息就必须发消息给Driver中的BlockManagerMaster来更新相应的BlockManagerInfo；
4. 当执行第二个stage的时候，第二个stage会向Dirver中的MapOutputTrackerEndpoint发消息请求上一个stage中相应的输出，此时MapOutputTrackerMaster会把上一个stage的输出数据的元数据信息发送给当前请求的stage；