5-3-cn

1 在这节课中，我们讨论用 MapReduce 的批处理计算模型。
2
数据处理系统提供大数据计算处理能力和应用开发平台。从计算架构的角度，将数据处理系统分为数据算法层、计算模型层、计算平台层、计算引擎层等。
与大数据相关的计算算法包括机器学习算法和数据挖掘算法。
计算模型是指不同类型的大数据在不同场景下的处理方式，
包括批处理、流计算、结构化数据的大规模并发处理(MPP)模型、内存计算模型和数据流图模型。
就计算平台和引擎而言，通常具有代表性的大数据处理平台有 Hadoop、Spark、storm、Pregel 等，
本节我们讨论下 MapReduce 为代表得大数据批处理模型。

3 MR 包括两个阶段，Map 和 Reduce， 从 HDFS 中取出的输入数据经过 Map 阶段产生中间结果存到 HDFS 中，然后经过 Reduce 阶段产生 Output，再存储到 HDFS 中。
让我们看一个视频 用玩扑克牌学习 MapReduce ”Learn MapReduce with Playing Cards “ 来轻松、直观理解 MR 的处理原理

4 MapReduce 工作流程概述
从视频中我们了解了 Map Reduce 的基本工作机制。
MR 试图在分布式环境下实现大型计算任务的并行化，以提高效率。
如图所示 HDFS 中的存储单元是数据块，从 HDFS 数据块中检索数据，然后将输入数据组织为很多数据 split，送入 map 任务中。
map 任务的输出进行排序、复制和合并等操作，即 Shuffle 洗牌阶段。
shuffle 后重组的中间结果作为 reduce 任务的输入，经过 reduce 阶段，计算最终结果存到 HDFS 中。
我们在看一个关于 Hadoop 架构工作原理的视频

5 MapReduce 架构
如图所示，在 MapReduce 中，JobTracker 接收 JobClient 提交的 Job，将它们按 InputFormat 的划分以及其他相关配置，生成若干个 Map 和 Reduce 任务。
TaskScheduler， 顾名思义，就是 MapReduce 中的任务调度器。
然后，当一个 TaskTracker 通过心跳告知 JobTracker 自己还有空闲的任务资源 Slot 时，
JobTracker 就会向其分派任务。具体应该分派一些什么样的任务给这台 TaskTracker，这就是 TaskScheduler 所需要考虑的事情。
TaskScheduler 工作在 JobTracker 上。在 JobTracker 启动时，根据配置“mapred.jobtracker.taskScheduler”选定一个 TaskScheduler 的派生类实例作为任务调度器。
任务的分派配由 JobTracker 的调度框架和 TaskScheduler 的具体调度策略配合完成。
简单来说：
1、JobTracker 通过一种 Listener 机制，将 Job 的变化情况同步给 TaskScheduler。
然后 TaskScheduler 就按照自己的策略将需要调度的 Job 管理起来；
2、在 JobTracker 需要向 TaskTracker 分派任务时，调用 TaskScheduler 的 assignTask()方法来获得应当分派的任务；MapReduce 架构主要有四部分:Client、JobTracker、TaskTracker、Task
1)客户端用户编写的 MapReduce 程序通过客户端提交给 JobTracker 用户可以通过 Client 提供的部分界面查看作业的运行状态
2) JobTracker 负责资源监视和作业调度，JobTracker 监视所有 tasktracker 和 Jobs 的运行状况，
如果发现故障，它将把相应的任务转移到其他节点 JobTracker，将跟踪任务执行进度、资源使用情况和其他信息，并通知任务调度器(TaskScheduler)，当资源空闲时，调度器将选择适当的任务来使用这些资

源
3) TaskTracker 会定期通过“心跳”向 JobTracker 报告节点上的资源使用情况和任务的进度，同时接收 JobTracker 发送的命令并执行相应的操作(如启动新任务、终止任务等)。
TaskTracker 使用“slot”来划分这个节点上的资源数量(CPU、内存等)。Task 在获得槽位后有机会运行，Hadoop 调度器的作用是将每个 TaskTracker 上的空闲槽位分配给 Task。
槽位分为 Map 槽位和 Reduce 槽位，分别用于 MapTask 和 Reduce Task。
4) 任务 Task 分为 Map Task 和 Reduce Task，由 TaskTracker 启动任务调度器负责在资源空闲时选择适当的任务来使用这些资源。槽位 Slot 是指资源(CPU、内存等)的数量。其中包括 Map Slot 和 Reduce Slot。Hadoop 调度器是将每个 TaskTracker 上的空闲 Slot 分配给 Task。

6 让我们把 HDFS 和 MapReduce 结合在一起。HDFS 和 MapReduce 应该构建在同一个集群上。
主节点应该是 HDFS 中的 name 节点和 MapReduce 中的 Job tracker，从节点应该同时是 HDFS 中的 datanode 和 MapReduce 中的 tasktracker。
例如，我们构建了一个主节点和三个从节点 HDFS 和 MapReduce 集群。为了减少数据传输开销，我们应该尽量使对应的 Map 任务的输入数据接近，最好是在同一台机器上。

7 HDFS 一次写入多次读取，不支持修改
为什么呢？因为如果读块的同时写一个文本到此块，此块变大，导致后续块的偏移量都不对了，若缩小此块后重新划块，多出的分到下一个块，此操作会导致泛洪，即集群有很多的节点它们的 CPU、内存、网卡会参与到因为一个修改的事情而造成的资源高度使用，网络会被疯狂的传输数据占用，所有的计算机 CPU 都在算此块应该拿出多少，另一个块应该接受多少。。。
此集群 2000 台难道只是实现一个可读、可写、可修改的文件系统么？还是为了更想让在他们上面多跑一些程序做计算，挖掘出数据的价值这才是他的核心思想。
如果想让 HDFS 支持修改，必然会影响到其他的软件的功能，所以这时 HDFS 设计者做了一个折中的方案不支持修改

8 本节我们以 MapReduce 为例学习了批处理计算模型，今天的学习就到这里，谢谢大家