**MapReduce-1**

**王佳伟，童效凯，周沛，曾天煦**

1-1.定义

Hoop MapReduce是：

①是一个软件框架；

②让某些应用易于编写；

③以并行方式处理大量数据；

④运行于大型集群上；

⑤具有可靠性，容错特点；

1-2.什么是并行运算

它是一种一次可执行多个指令的运算，目的是提高计算速度，以通过扩大问题求解规模，解决大型二复杂的计算问题。

并行运算分类：

1、时间上的并行：即为流水线技术

2、空间上的并行：指用多个处理器并发执行的计算

并行运算基本思想：将被求解问题分成若干个部分，各部分均分由一个独立的处理器并行计算。

并行计算系统即可以是多个处理器的超级计算机，也可以是多个独立的计算机构成的集群。通过并行运算集群处理，再发回用户。

1-3.来源

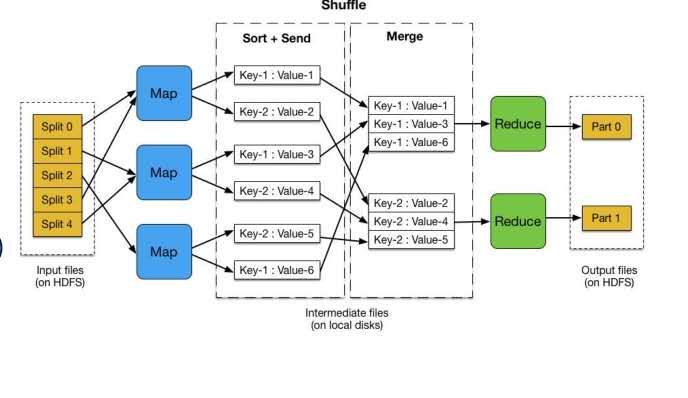
参考Google MapReduce框架，基于HapHoop平台而实现的一个并行计算框架。

1-4与传统并行的对比



**2.Madreduce的工作机制**

\*Map与Reduce函数都是输入<key,value>序列后，经过处理后，输出也是<key’,value’>序列。



MapReduce执行过程:

1将输入的数据切片分给不同的机器;

2执行Map任务的Worker将输入数据解析成<key, value>，用户定义的Map函数把输入的<key, value>转成中间形式的<key,value>;

3按照key值对中间形式的key/value进行排序、聚合;

4把不同的key值和相应的value集分配给不同的机器，完成Reduce运算;

5输出Reduce结果。

**小结：**

1.MapReduce将 复杂的、运行于大规模集群上的并行计算过程高度地抽象到了两个函数:Map和Reduce;

2.编程容易，不需要掌握分布式并行编程细节，也可以很容易把自己的程序运行在分布式系统上，完成海量数据的计算;

3.MapReduce采用“分而治之”策略，-一个存储在分布式文件系统中的大规模数据集，会被切分成许多独立的分片(split) ，这些分片可以被多个Map任务并行处理;

4.MapReduce设计的一个理念就是“计算向数据靠拢”，而不是“数据向计算靠拢”，因为，移动数据需要大量的网络传输开销;

5.Hadoop框 架是用Java实现的，但是，MapReduce应用 程序则不一-定要用Java来写。

**Mapreduce体系结构**

运行过程：

1 运行Mapreduce作业；

2 客户端（Client ）提交Mapreduce作业；

3 JobTracker协调作业的运行；

4 TaskTracker 运行作业划分后的任务；

5 运行Mapreduce作业的步骤包括 提交作业、初始化作业、分配任务、执行任务、更新进度和状态、完成作业等6个步骤。

Hadoop Mapreduce采用Master/Slave 结构

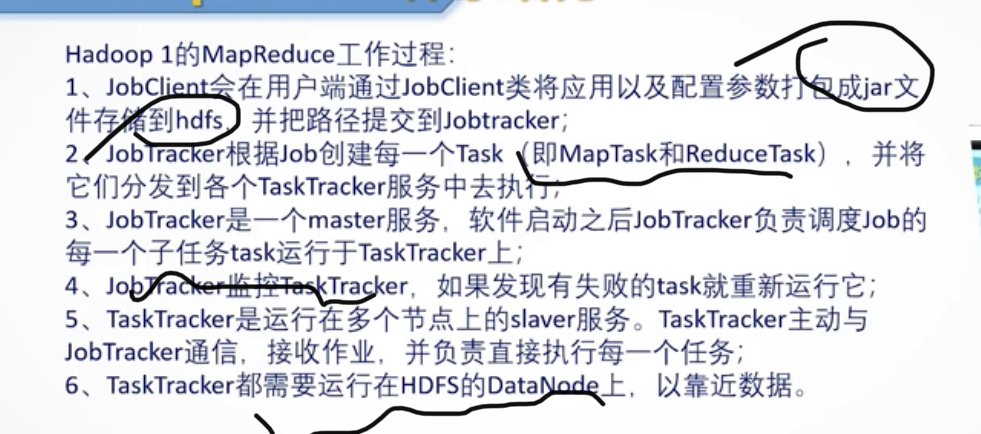
Master：即Mapreduce中的JobTracker 是整个集群的唯一管理者 功能包括：作业管理、状态监控和任务调度。

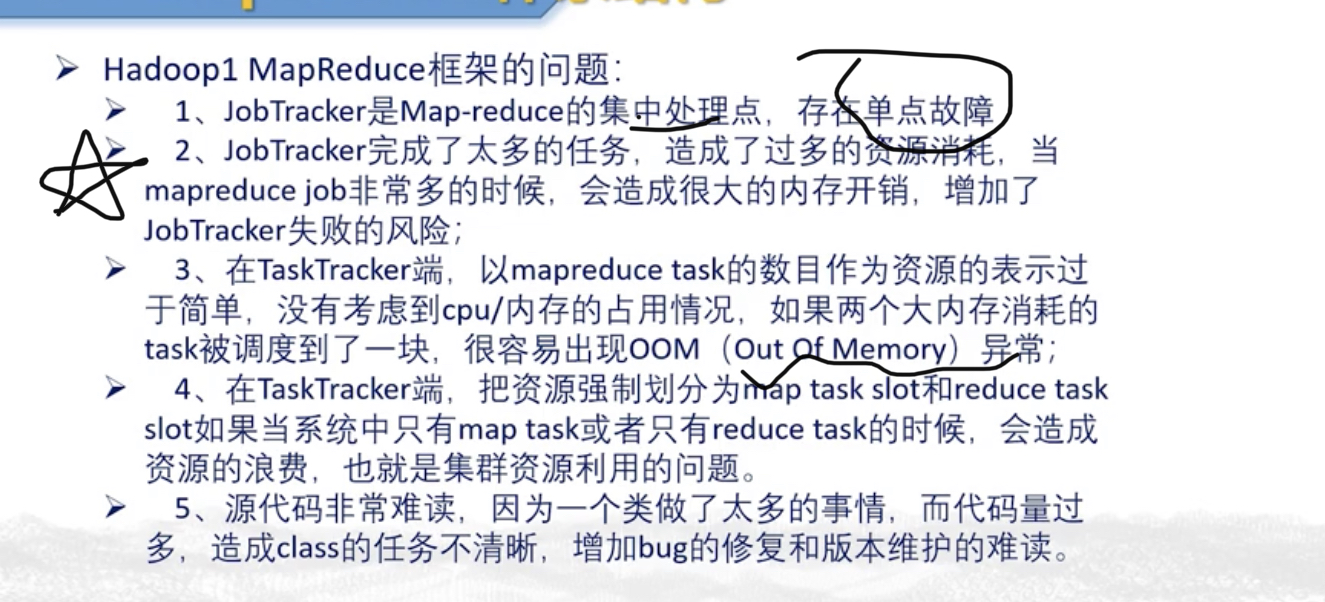
Slave：负责任务的执行和任务状态的回报，即Mapreduce中的TaskTracker。

JobTracker：是一个后台服务进程，启动之后会一直监听并接收来自各个TaskTracker发送的心跳信息，包括资源使用情况和任务运行情况等信息，其功能主要包括：

作业控制、状态监控、资源管理。

Hadoop1的Mapreduce工作过程

Hadoop1 Mapreduce中存在的问题



**MapReduce的具体应用**

基本思路:

➢在map阶段,把关键字作为key输出，并在 value中标记出数据是 来自data1还是data2;

在shuffle阶段会自然按 key分组;

➢reduce阶段，判断每一个value是来自data1还 是data2,在内部分成2 组，做集合的乘积。

