

系统设计

MapReduce

（九章网站下载最新课件）

本节主讲人：北丐老师

程不允许录像，否则将追究法律责任，赔偿损失

硬盘上有 1T 的数据需要排序，下面的哪种办法看上去比较靠谱？

A: 将数据导入到内存中进行排序，然后再输出到硬盘

B: 按照数据的范围进行拆分，使得每个范围内的数据足够放在内存中，然后分别排序后汇总

C: 按照数据 hash 之后的结果进行拆分，使得拆分之后的每个部分能够放在内存中，然后分别排序再归并排序结果

D: 将数据按照实际存储顺序进行拆分，如前 1G 拆分为一个部分，第 2 个 G 拆分到第二个部分，然后分别排序再归并排序结果。

已回答

我不会

北丐

答错了，好可惜。正确答案是 D，有 44% 的同学答对了，要加油了。

1T 的数据很大，通常很难找到内存超过 1T 的电脑。但是内存超过 1G 的还是很容易的。因此 A 肯定是不对的。如果我们能找到一个算法比较均匀的拆分 1T 的数据到 1024 个 1G 的文件的话，每个文件都可以导入内存中进行排序，最后我们再归并排序后的结果即可。这个拆分排序再归并的算法就是外排序算法。

- 选项 B，按照数据的范围进行拆分，会导致分配不均匀，比如大部分的数据都在一个很小的范围内。而且不是所有的数据都有可数范围，如字符串是很难划定范围的。
- 选项 C，按照 hash 之后的结果进行拆分，也会导致数据拆分不均匀，因而使得某些部分可能依然无法导入内存。
- 选项 D，按照实际存储位置进行拆分，这样才能够确保每个部分可以导入内存。

使用 hash 的方法拆分数据的问题是？

如果我们要将一些数据根据他们 hash 之后的结果进行归类，比如数据是整数，hash function 是 $x \% N$ （x 是数据本身，N 是需要拆分出的类别总数）。

A: hash 运算速度较慢，会影响整体运算效率

B: hash 存在 collision 会将两个不同的数据放到一个类下面

C: 数据的分配可能会不均匀

已回答

我不会

北丐

答错了！正确答案是 C，有 40% 的同学答对了，加油赶上他们！

假如 $N = 3$ ，数据是 [1, 4, 7, 10, 13, 16, 19, 22, 25]，那相当于所有数据还是拆分在一个类里。

- Map Reduce Problems
 - 多台机器并行处理数据
 - Count Word Frequency
 - Build Inverted Index

Map Reduce

Why Map Reduce?

Distributed System is built for fast computing

大数据职位面试敲门砖

学会MapReduce可以找大数据工作

Interviewer: Count the word frequency of a web page?

Google 面试真题

<http://www.lintcode.com/en/problem/word-count/>

<http://www.jiuzhang.com/solutions/word-count/>

常见土方法一 For循环

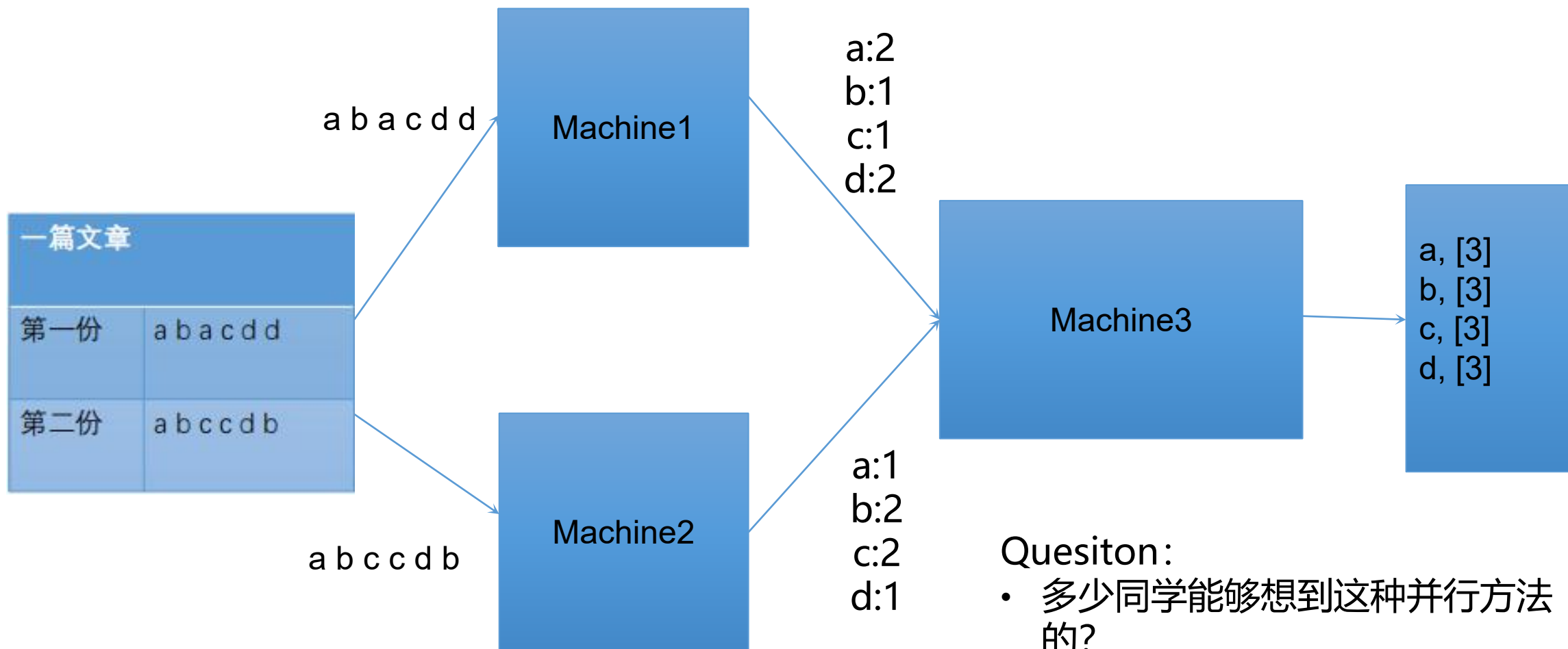
伪代码

- `HashMap<String,int> wordcount;`
- for each word in webpage :
 - `wordcount[word]++`



- Question?
 - 多少同学能够想到这种方法?
 - 有什么优缺点?
 - 优点：简单 缺点：只有一台机器——慢、内存大小受限
 - 如果你有多台机器呢?

常见土方法二 多台机器For循环



Question:

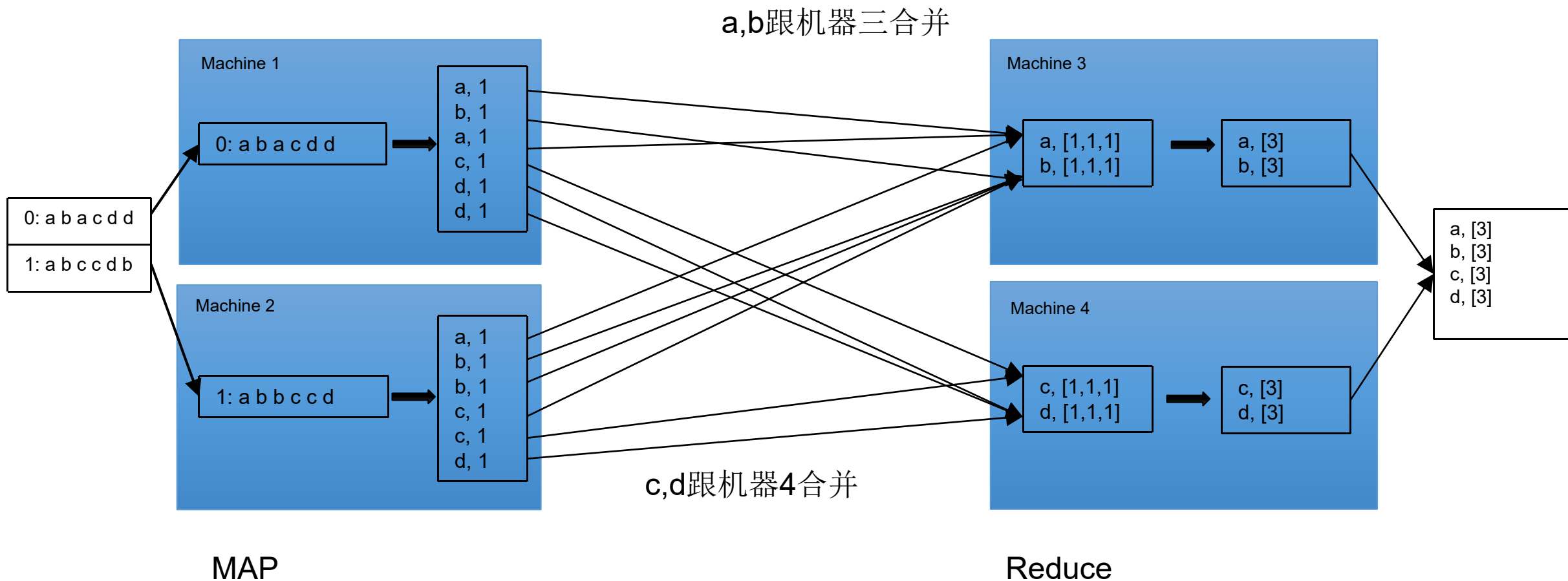
- 多少同学能够想到这种并行方法的?
- 大家觉得这种方法有啥问题?

合并的时候是Bottle Neck

合并是否也可以并行?
以什么标准来划分?
机器 or key?



方法三 多台机器Map Reduce



Map

- 机器1, 2 只负责把文章拆分为一个一个的单词

Reduce

- 机器3, 4各负责一部分word的合并

Map Reduce

Map

把文章拆分单词的过程

Reduce

把单词次数合并在一起的过程

存在的问题

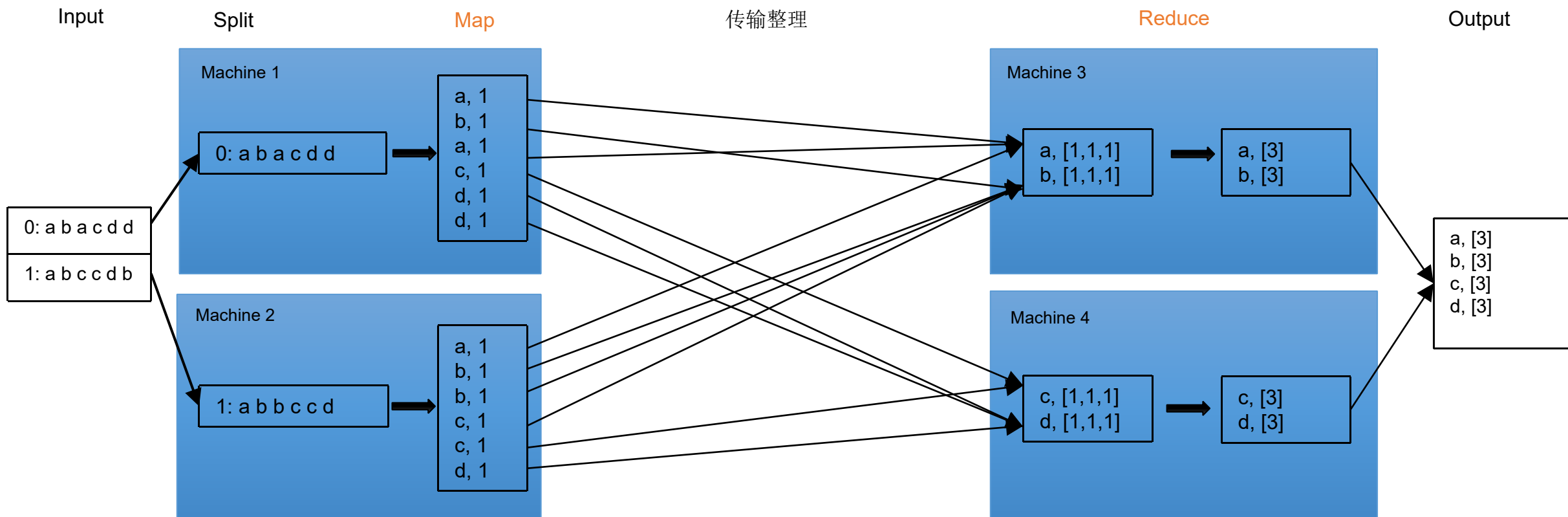
谁来负责把文章拆分为一小段一小段？

中间传输整理谁来负责？比如怎么知道把a放在机器3还是机器4？

依靠Map Reduce的框架实现

Map Reduce Steps

- Map Reduce 是一套实现分布式运算的框架
- Step1 Input
- Step2 Split
- Step3 Map
- Step4 传输整理
- Step5 Reduce
- Step6 Output



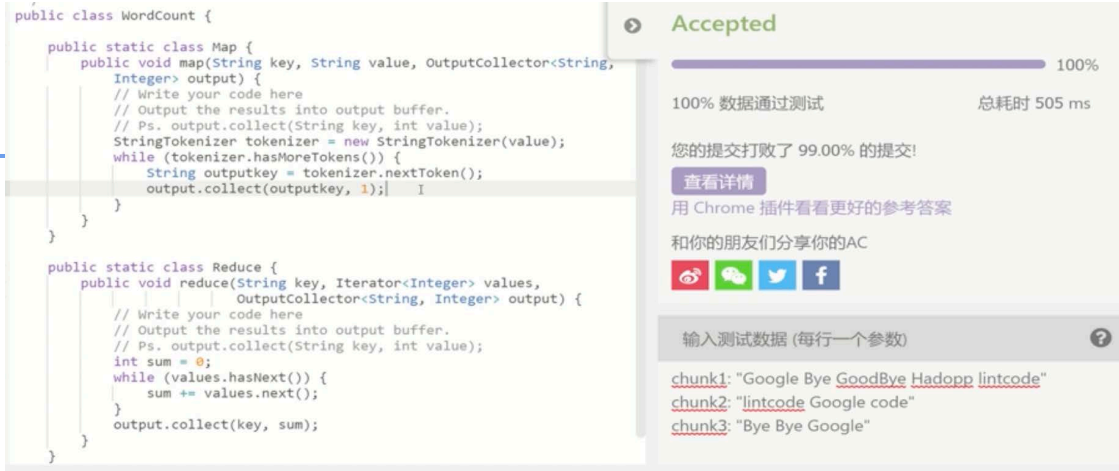
Map为什么不做Aggregation?



我们要实现什么呢？

Map 函数 和 Reduce 函数

- Map Reduce 是一套实现分布式运算的框架
 - Step1 Input
 - Step2 Split
 - Step3 Map 实现怎么把文章切分成单词
 - Step4 传输整理
 - Step5 Reduce 实现怎么把单词统一在一起
 - Step6 Output
- 所以MapReduce帮我们把框架大部分实现好，我们只用实现Map Reduce解决逻辑计算的问题。



Map Reduce 函数接口是什么？

他们的输入和输出必须是Key Value 形式

Map 输入: key:文章存储地址, Value: 文章内容

Reduce 输入: key:map输出的key, value: map输出的value

Google面试真题实战

<http://www.lintcode.com/en/problem/word-count/>

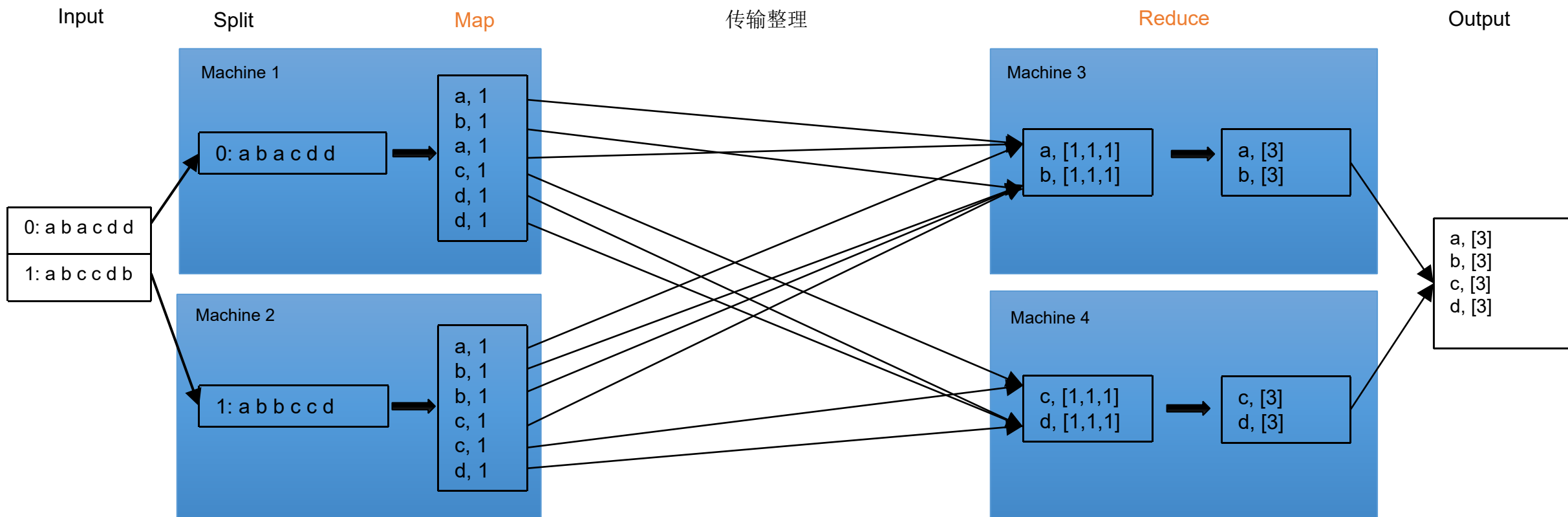
<http://www.jiuzhang.com/solutions/word-count/>

Map Reduce Steps

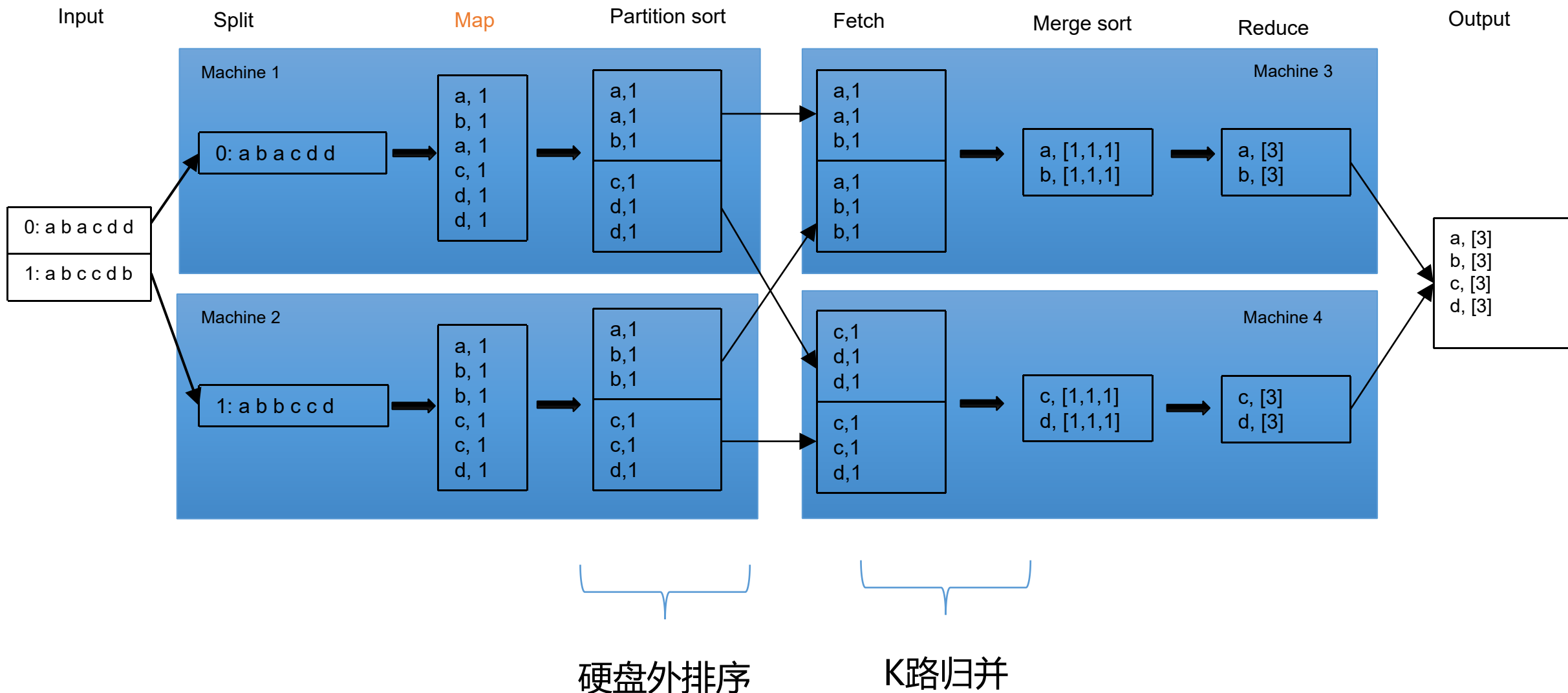
- Map Reduce 是一套实现分布式运算的**框架**
- Step1 Input 设定好输入文件
- Step2 Split 系统帮我们把文件尽量平分到每个机器
- **Step3 Map 实现代码**
- Step4 传输整理 系统帮我们整理
- **Step5 Reduce 实现代码**
- Step6 Output 设定输出文件



“传输整理”详细操作



要你设计这一步你会怎么设计？



- Map Reduce 是一套实现分布式运算的**框架**
 - Step1 Input
 - Step2 Split
 - **Step3 Map** 实现怎么把文章切分成单词
 - Step4 Partition sort
 - Step5 Fetch + Merge Sort
 - **Step6 Reduce** 实现怎么把单词统一在一起
 - Step7 Output
-
- 所以**MapReduce**帮我们把框架大部分实现好，我们只用实现**Map Reduce**解决逻辑计算的问题。

- Question1?
- Map 多少台机器? Reduce 多少台机器?
 - 全由自己决定。一般1000map, 1000reduce规模
- Question2? (加分)
- 机器越多就越好么?
 - Advantage:
 - 机器越多, 那么每台机器处理的就越少, 总处理数据就越快
 - Disadvantage:
 - 启动机器的时间相应也变长了。
- Question3? (加分)
 - 如果不考虑启动时间, Reduce 的机器是越多就一定越快么?
 - Key的数目就是reduce的上限



相同的单词在reduce这一步一定会分到同一台机器吗?

通常是如此, 但是也有特殊情况, 比如某个key 的数据非常多, 就会造成某一台机器要处理的数据非常多, 而其他的机器相对比较空闲。这时候我们要自定义 key 的拆分规则, 来保证数据很多的 key 分到不同的机器上。

锅包又 学员

好奇machine之间是如何通信的?

助教-江畔 助教

@锅包又

好奇machine之间是如何通信的?

Worker 之间是用 RPC 来进行通信的。

Break

Apple Interviewer: Build inverted index with MapReduce?

<http://www.lintcode.com/en/problem/inverted-index-map-reduce/#>

<http://www.jiuzhang.com/solutions/inverted-index-map-reduce/>

Read More:
Novice/Expert, <http://url.cn/fsZ927>

Input

0: Deer Bear River
1: Car River
2: Deer Car Bear



Output

Bear: 0,2
Car: 1,2
Deer: 0,2
River: 0,1

```
public class InvertedIndex {  
  
    public static class Map {  
        public void map(String __, Document value,  
            OutputCollector<String, Integer> output) {  
            // Write your code here  
            // Output the results into output buffer.  
            // Ps. output.collect(String key, int value);  
            StringTokenizer tokenizer = new StringTokenizer(value.content  
            );  
            while (tokenizer.hasMoreTokens()) {  
                String word = tokenizer.nextToken();  
                output.collect(word, value.id);  
            }  
        }  
    }  
  
    public static class Reduce {  
        public void reduce(String key, Iterator<Integer> values,  
            OutputCollector<String, List<Integer>> output) {  
            // Write your code here  
            // Output the results into output buffer.  
            // Ps. output.collect(String key, List<Integer> value);  
            List<Integer> results = new ArrayList<>();  
            int left = -1;  
            while (values.hasNext()) {  
                int now = values.next();  
                if (left != now) {  
                    results.add(now);  
                }  
                left = now;  
            }  
        }  
    }  
}
```

描述 控制台 笔记

Accepted

100%

100% 数据通过测试 总耗时 1904 ms

您的提交打败了 83.60% 的提交!

查看详情

用 Chrome 插件看看更好的参考答案

和你的朋友分享你的AC

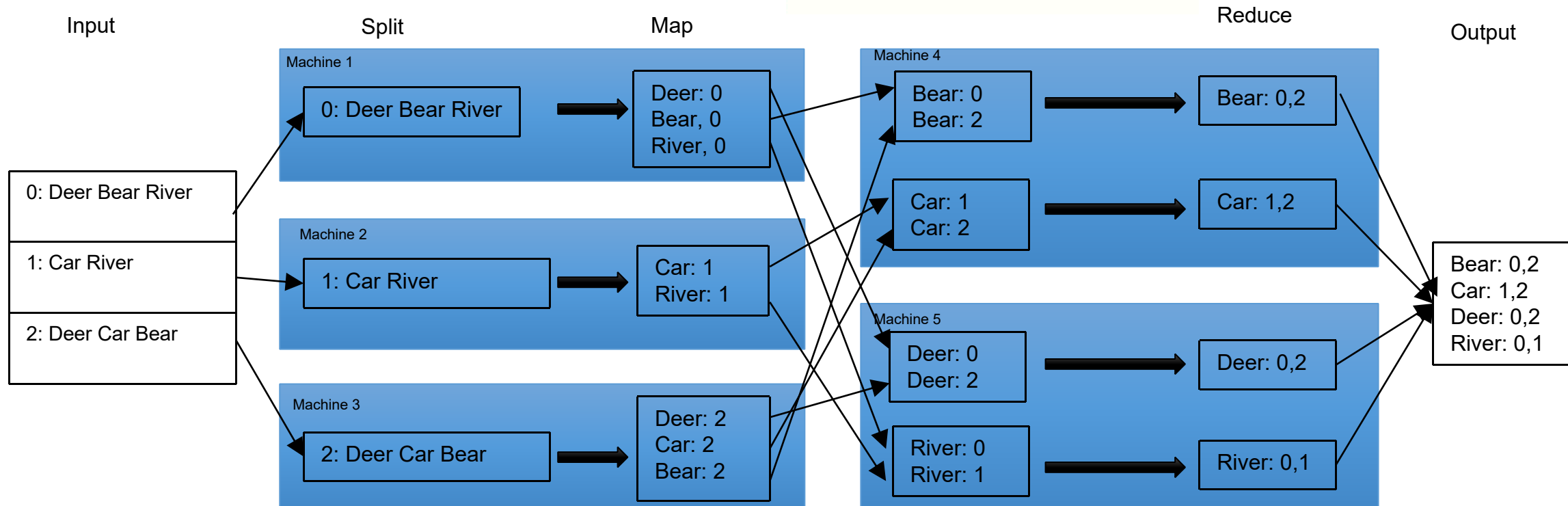


输入测试数据 (每行一个参数)

[[{"id":1,"content":"This is the content of document1"}, {"id":2,"content":"This is the content of document2"}]]

快捷键: Ctrl + Er

Build inverted index with MapReduce?



```
//key: the id of a doc
//value: the content of the line
Map( string key, string value)
  for each word in value:
    Output( word, key);
```

```
//key: the name of a word
//valueList: the appearances of this word in documents
Reduce( string key, list valueList )
  List sumList;
  for value in valueList:
    sumList.append(value);
  OutputFinal( key, sumList );
```




Apple Interviewer: Anagram - Map Reduce

<http://www.lintcode.com/en/problem/anagram-map-reduce/>

<http://www.jiuzhang.com/solutions/anagram-map-reduce/>

```
8 public class Anagram {
9
10 public static class Map {
11     public void map(String key, String value,
12                     OutputCollector<String, String> output) {
13         // Write your code here
14         // Output the results into output buffer.
15         // Ps. output.collect(String key, String value);
16         StringTokenizer tokenizer = new StringTokenizer(value);
17         while (tokenizer.hasMoreTokens()) {
18             String word = tokenizer.nextToken();
19             char[] sc = word.toCharArray();
20             Arrays.sort(sc);
21             output.collect(new String(sc), word);
22         }
23     }
24 }
25
26 public static class Reduce {
27     public void reduce(String key, Iterator<String> values,
28                       OutputCollector<String, List<String>> output) {
29         // Write your code here
30         // Output the results into output buffer.
31         // Ps. output.collect(String key, List<String> value);
32         List<String> results = new ArrayList<>();
33         while (values.hasNext()) {
34             results.add(values.next());
35         }
36         output.collect(key, results);
37     }
38 }
```

Accepted

100% 数据通过测试 总耗时 563 ms

您的提交打败了 87.00% 的提交!

[查看详情](#)

用 Chrome 插件看看更好的参考答案

和你的朋友们分享你的AC

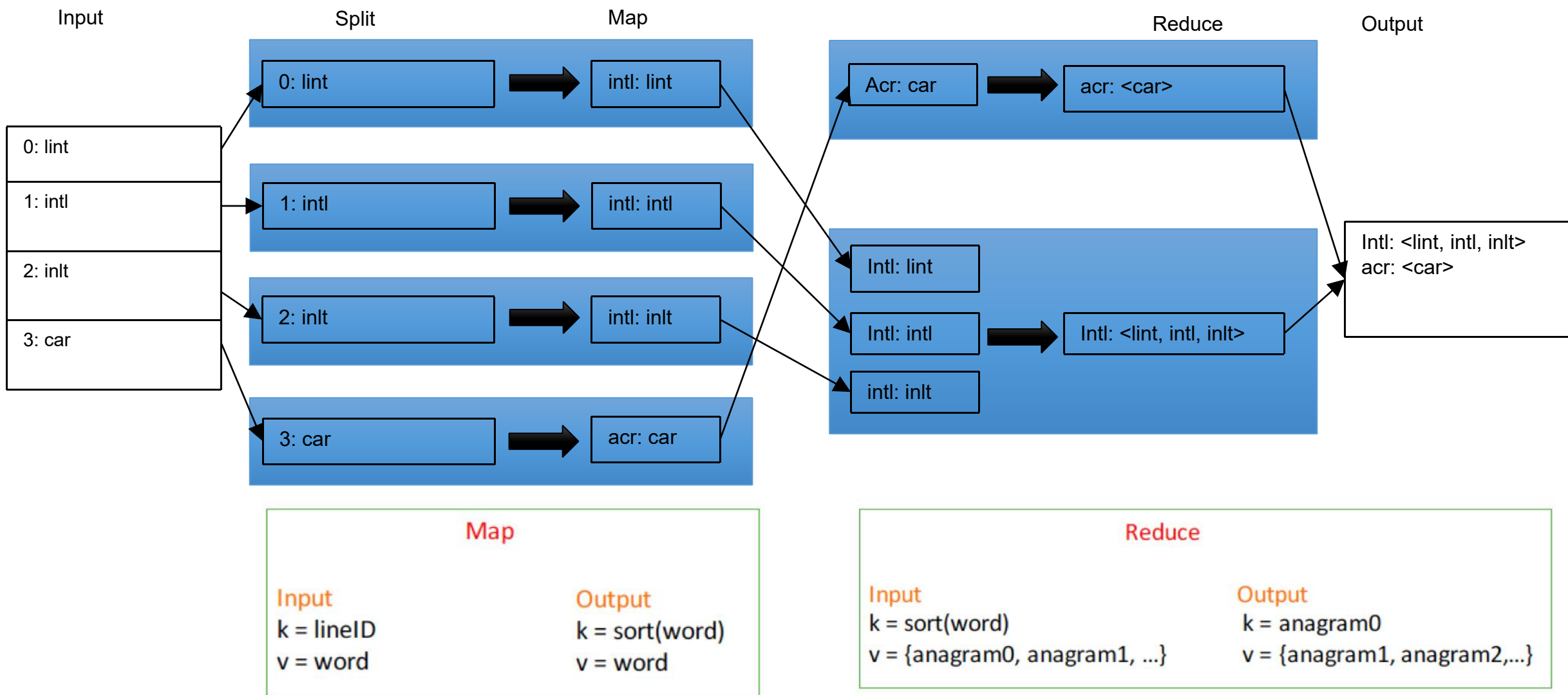
[微博](#) [微信](#) [Twitter](#) [Facebook](#)

输入测试数据 (每行一个参数) ?

chunk1: "lint lint Init In"
chunk2: "abc ltrn code deco"
cunnk3: "ab ba cba"

快捷键: Ctrl + Enter

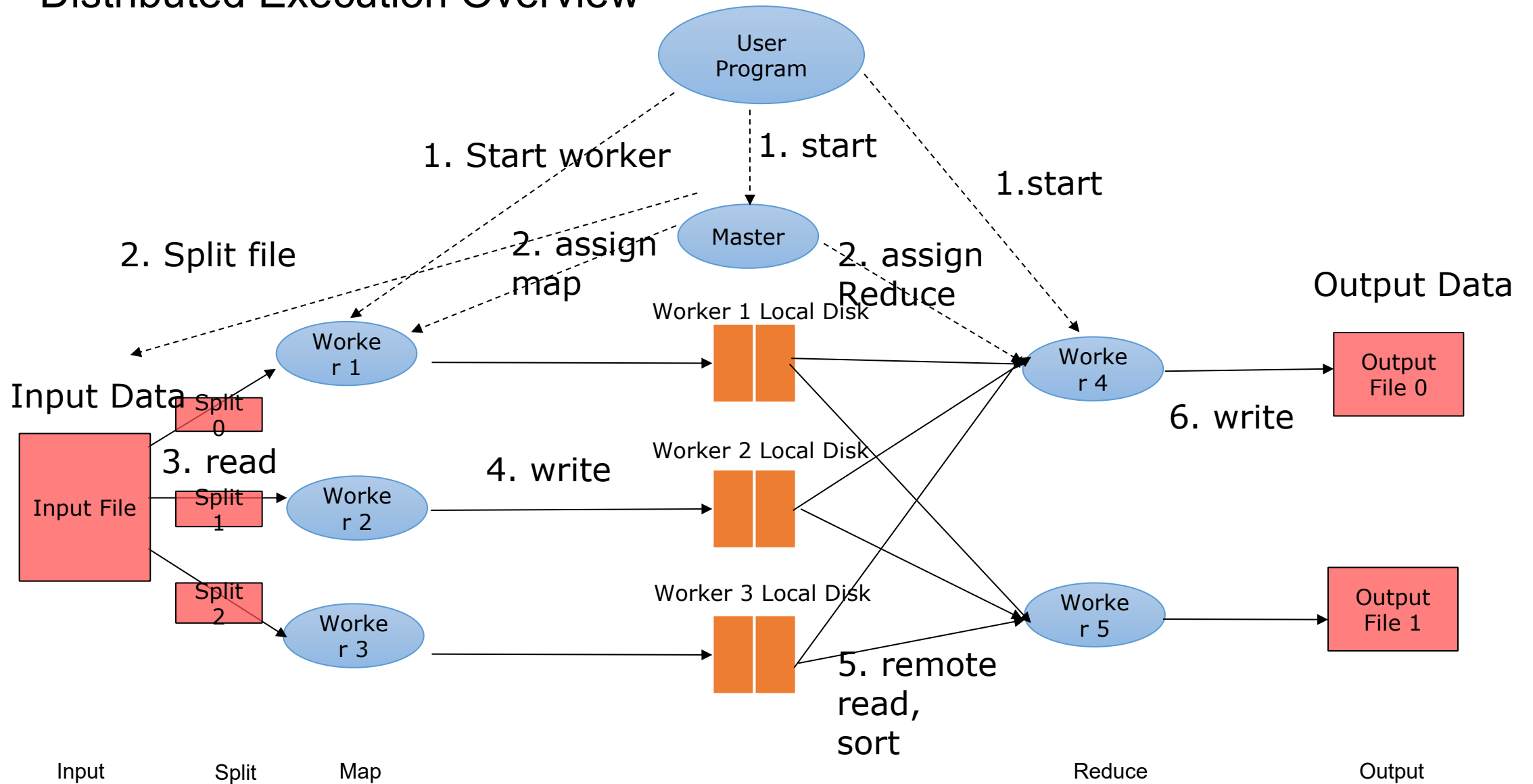
Anagram - Map Reduce



Interviewer: Design a MapReduce system



Distributed Execution Overview



1. Mapper 和 Reducer是同时工作还是先Mapper 工作还是 Reducer工作的么?

Mapper要结束了后Reducer才能运行

2. 运行过程中一个Mapper或者Reducer挂了怎么办?

重新分配一台机器做

3. Reducer一个机器Key特别大怎么办?

加一个random后缀, 类似Shard Key

$f b_1$

$f b_2$

$f b_3$



4. Input 和 Output 存放在哪?

存放在GFS里面

5. Local Disk 上面的data有木有必要保存在GFS上面? 要是丢了怎么办?

不需要, 丢了重做就好

6. Mapper 和 Reducer 可以放在同一台机器么?

这样设计并不是特别好, Mapper 和Reducer之前都有很多需要预处理的工作。两台机器可以并行的预处理。

1. (Start) User program start master and worker.
2. (Assign Task) Master assign task to the map worker and reduce worker. (Assign Map and Reduce code)
3. (Split) Master Split the input data.
4. (Map Read) Each map worker read the split input data.
5. (Map) Each map worker do the “Map” job on their machine.
6. (Map output) Each map worker output the file in the local disk of its worker.
7. (Reduce Fetch) Each reduce worker fetch the data from the map worker.
8. (Reduce) Each Reducer worker do the “Reduce” job on their machine.
9. (Reduce output) Reduce worker output the final output data.

MapReduce Framework

- Map Reduce Solve Problem
 - Words Count
 - Inverted index
 - Anagrams
 - Top K Frequency (<http://bit.ly/25D8Q7I>)
 - PageRank (<http://bit.ly/1TOwoyV>)
- Map Reduce Step
 - Step1 Input
 - Step2 Split
 - Step3 Map
 - Step4 传输
 - Step5 Reduce
 - Step6 Output
- Map Reduce System
 - Master and Worker
- More
 - 大数据班敬请期待.....

再补充几个常见的 QA:

Q: Reduce 之后各个key还是可能会在不同地方，那么怎么再把这些 reducer 的结果 sort 并放在一起呢？

A: Reducer 的结果在全局是不 sort 的。因为很多计算场景下计算结果不需要 sort。如果有 sort 的需求，可以使用外排序算法（External Sorting）进行排序即可。

Q: 系统设计中 map reduce 的问题会以什么形式问？

A: 90% 的概率会问使用 map reduce 来解决比较重的计算问题。10% 的概率会问 map reduce 的原理是怎么样的。所以好好做今天的编程题作业非常重要！

Q: Reduce 的过程全部都在内存里么？是否会装不下？

A: 不是的。Reduce 的过程，key 是在内存里的，value list 通常在代码中是一个 iterator 的形式，也就意味着，有可能是从文件里读进来的。很显然全部放在内存肯定是放不下的，特别是对一些很 hot 的 key。

到此为止呢，Google 的三驾马车已经学完了，不知道你是否收获呢？三驾马车里，GFS 是其他两个系统的基础，是重中之重需要掌握的。Big Table 的设计原理更是直接被当做面试题在多家公司的面经中出现过。Map Reduce 如果你不是面试 Big Data Engineer 的岗位的话，直接问到的概率不是特别大，但是在算法面试中，特别是 Google 的算法面试中，很多时候会出现可以用 Map Reduce 来解决的问题，特别是 Top K Frequent Elements 这个高频算法面试题。如果你能在这类面试中使用 Map Reduce 来解决问题，一定会拿到 Strong Hire !

Map 的步骤和 Reduce 的步骤的顺序是怎么样的

A: Map 的机器必须先全部执行完，Reduce 的机器才能开始工作

B: Map 的机器可以不全部执行完，Reduce 的机器就开始工作

C: Map 的机器还没有全部执行完的时候，Reduce 的机器可以做一些前期准备工作，但是不能执行真正的 Reduce 的部分

已回答

我不会

技巧

正确答案是 C，有58%的同学做对了这道题目哦，继续努力！

Map 的机器如果没有全部执行完，任何一台 Reduce 的机器所负责的数据段都有可能还有更新。因此 Reduce 的部分是不可以开始的。但是机器可以先启动执行一些程序的初始化操作是可以的。

- Novice, <http://langyu.iteye.com/blog/992916>
- Expert, <http://data.qq.com/article?id=543>
- Expert, <https://www.jianshu.com/p/0ddf3ae19b49>
- Expert/Master, <http://novoland.github.io/%E5%B7%A5%E4%BD%9C/2014/09/04/MapReduce%20Algorithms.html>
- Expert/Master, https://www.slideshare.net/romain_jacotin/the-google-mapreduce
- Master, <http://www.cnblogs.com/yepei/p/6292440.html>

在 Map Reduce 中，Master 机器的作用有？

A: 存储中间计算结果的数据

B: 监控 map 和 reduce 机器的健康状况，如果有机器挂了，从“备胎池”中挑选一台机器补上并重新执行任务

C: 分发任务，如当 map 机器执行结束以后，告诉 reduce 的机器可以开始工作了

D: 管理所有被计算数据的 metadata

E: 分配哪些机器是 map，哪些机器是 reduce，并分配相应代码

F: 等分输入文件的数据给 map

已回答

我不会

正确答案是 B C E F，有20%的同学超过了你，但是千万不要气馁。

Map Reduce 过程中，哪些数据会保存在 GFS 里？

A: 输入数据

B: 中间计算结果

C: 输出数据

已回答

我不会

此题

正确答案是 A C，有57%的同学做对了这道题目哦，继续努力！