系统设计
MapReduce + Lookup Service
（九章网站下载最新课件）

本节主讲人：北丐

扫描二维码关注微信/微博
获取最新面试题及权威解答

微信: ninechapter
微博: http://www.weibo.com/ninechapter
知乎: http://zhuanlan.zhihu.com/jiuzhang
官网: http://www.jiuzhang.com

今日课程大纲

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

- Design a Lookup Service

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/

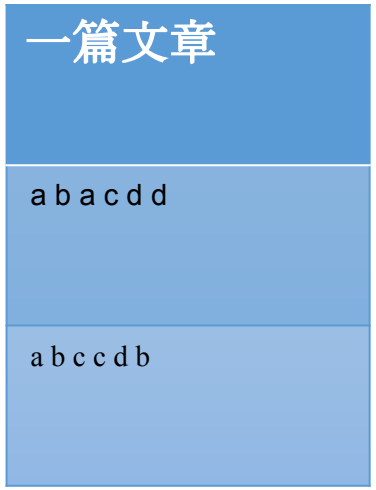
http://www.jiuzhang.com/solutions/word-count/

常见土方法一 For循环

方法一 For循环

伪代码

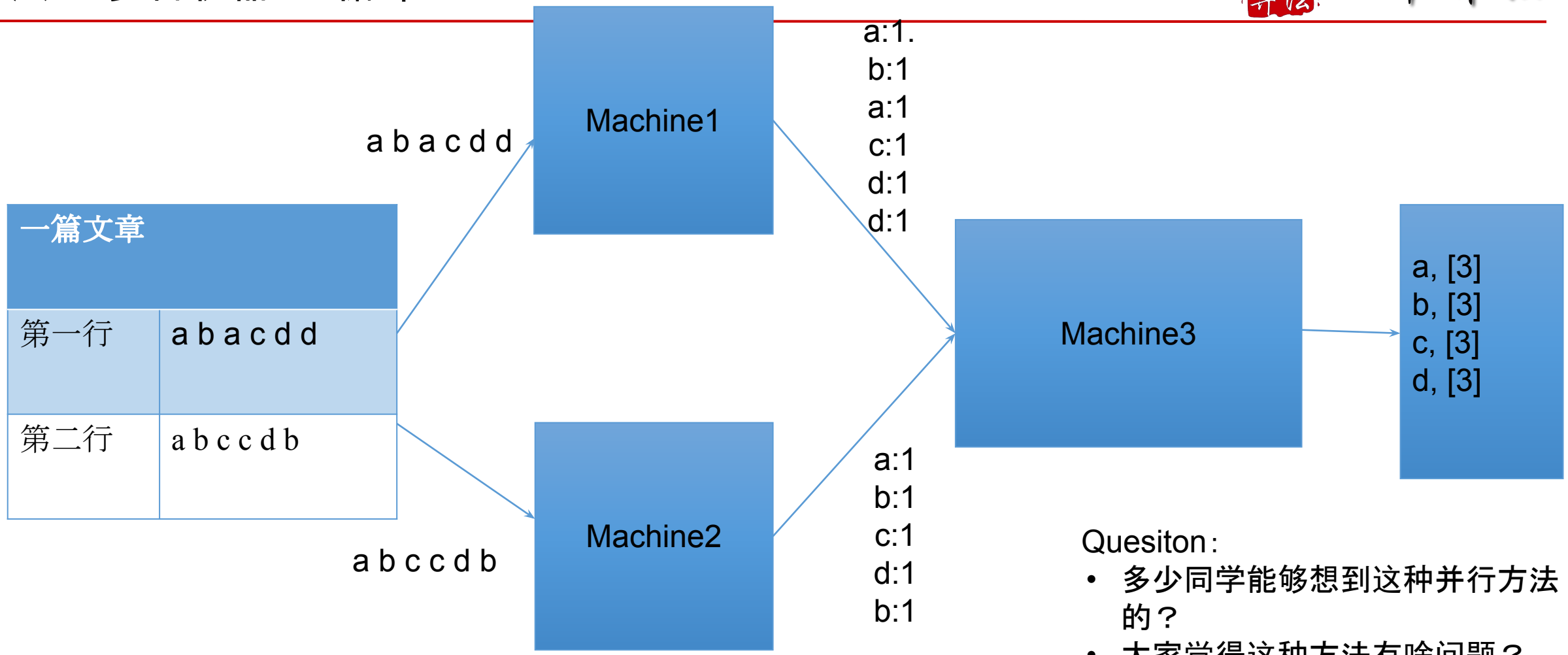
- HashMap<String, Interger> wordcount;
- for each word in webpage :
  - wordcount[word]++



- Question？
  - 多少同学能够想到这种方法？
  - 问题？
    - 慢
  - 如果你有多台机器呢？

常见土方法二 多台机器For循环

方法二 多台机器For循环



Quesiton：

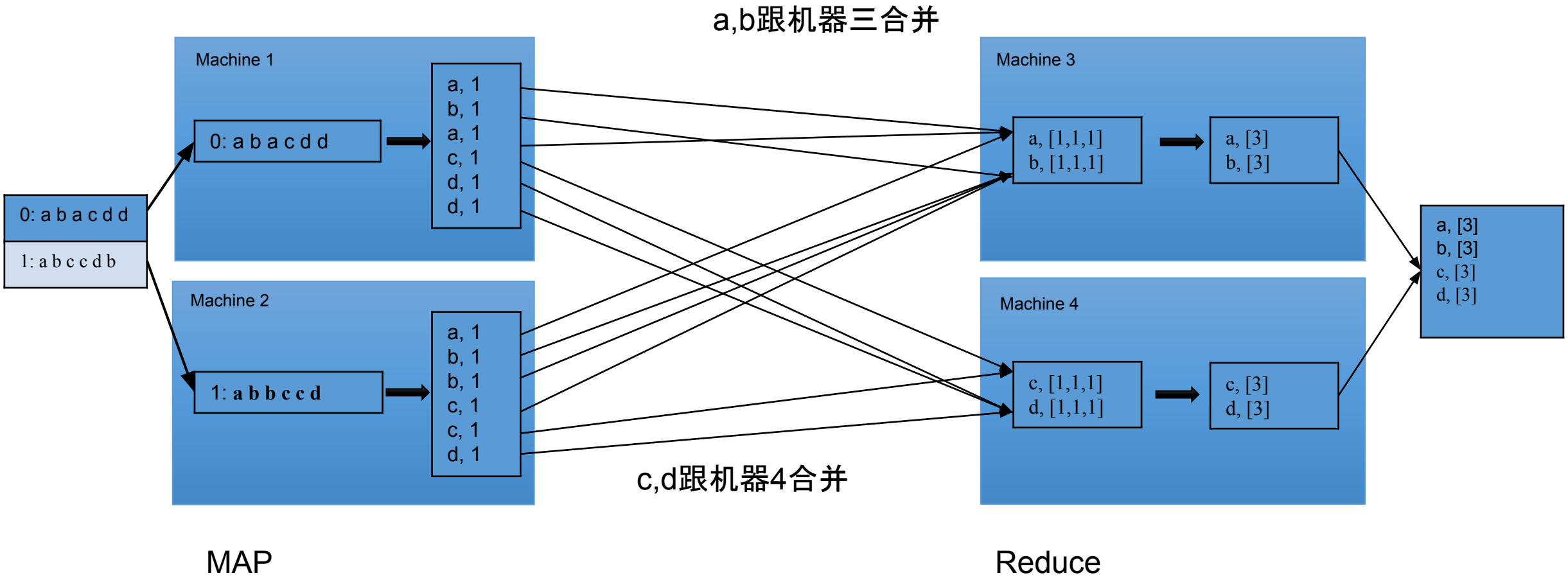
- 多少同学能够想到这种并行方法的？
- 大家觉得这种方法有啥问题？

合并的时候是Bottle Neck

合并是否也可以并行？

方法三 多台机器Map Reduce

方法三：Map Reduce



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？

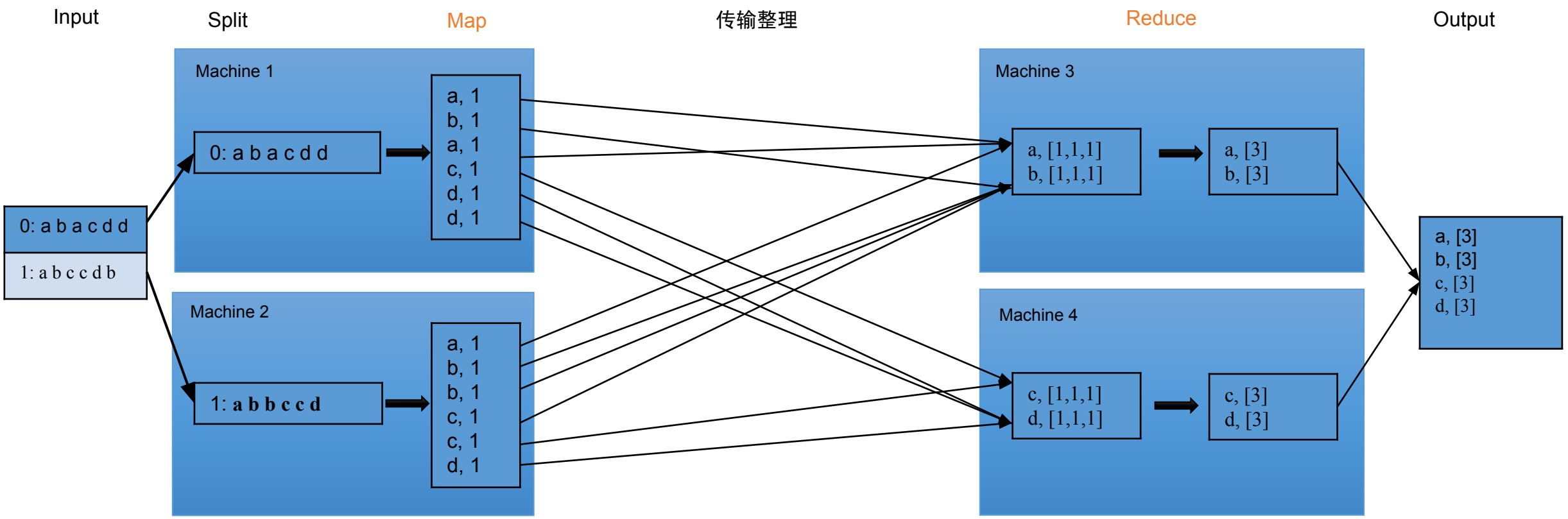
中间传输整理谁来负责？比如怎么知道把a放在机器3还是机器4？

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

Word Count



我们要实现什么代码呢？

我们要实现什么呢？

Map 函数 和 Reduce 函数

Map Reduce Steps

- 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

Map Reduce Steps

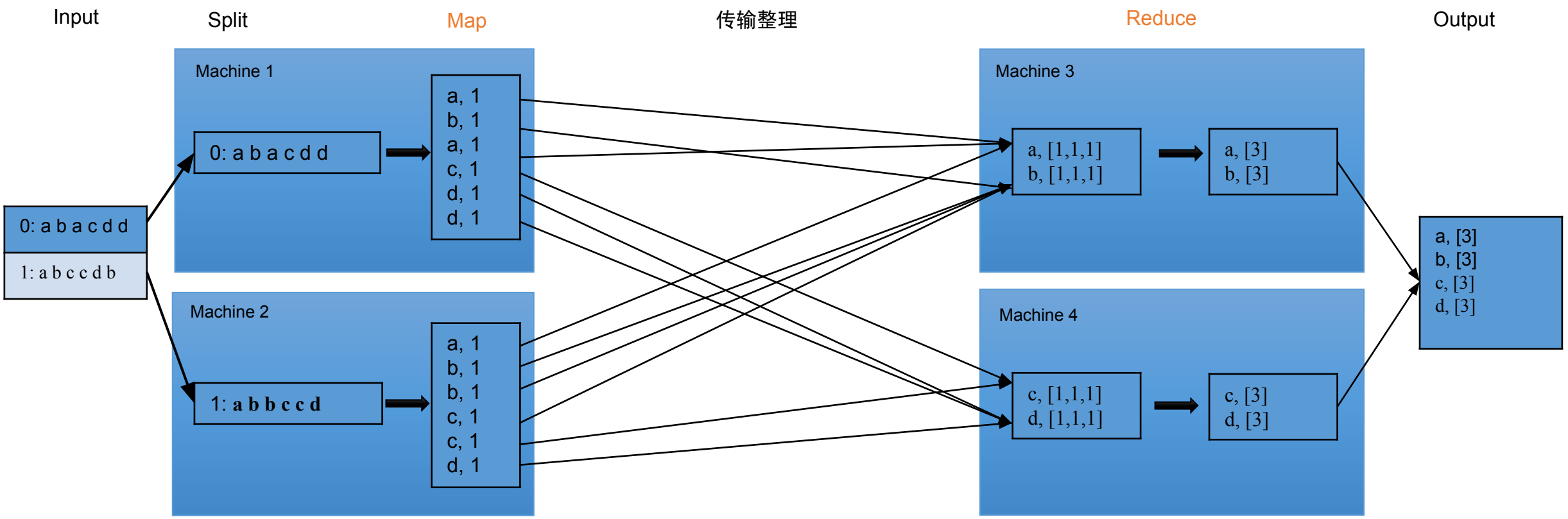
- Question1?
- Map 多少台机器？ Reduce 多少台机器？
  - 全由自己决定。一般1000map，1000reduce规模

- 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

“传输整理”详细操作

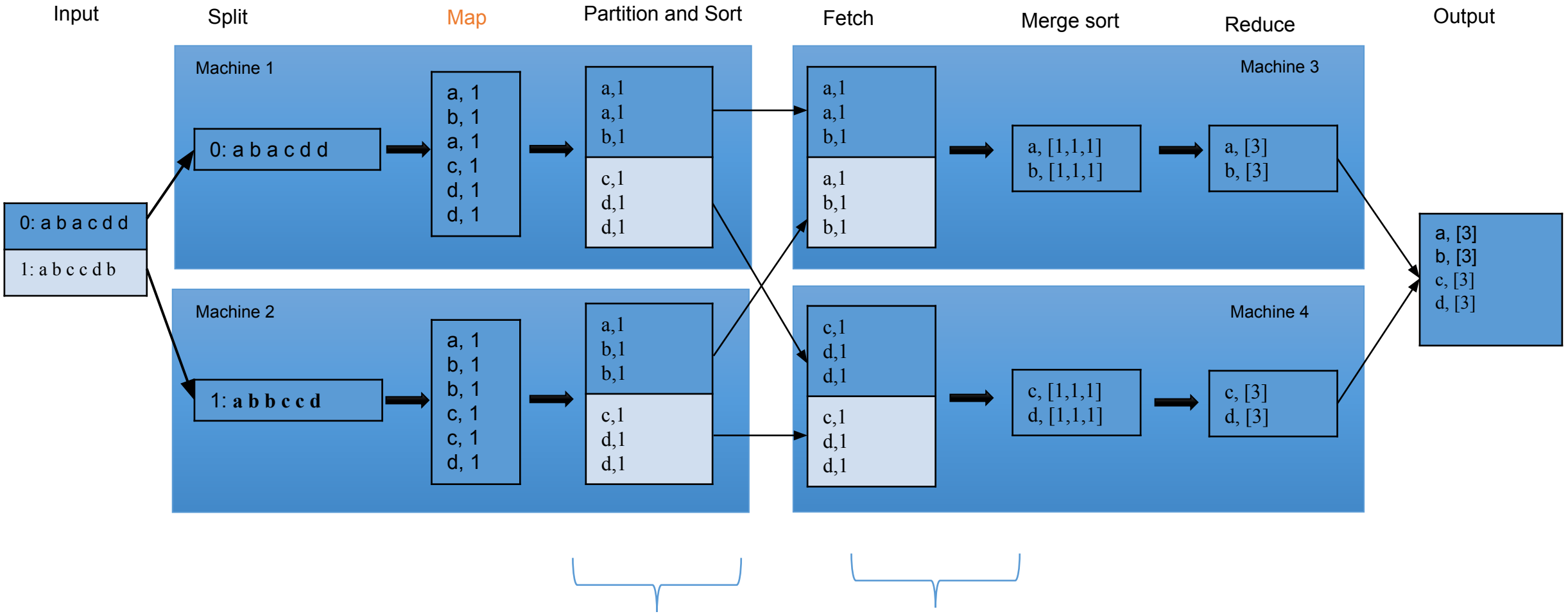
Word Count



要你设计这一步你会怎么设计？

1. Map端用一个HashMap去先做一次合并，把相同的key合并到一起
2. Reducer 端再用一个去把相同的key再排序到一起。怎么排序？快速排序

“传输整理”详细操作



硬盘外排序

K路归并

ReadMore:
外排序:http://bit.ly/1WAXqzw

Map Reduce Steps

- 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。

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

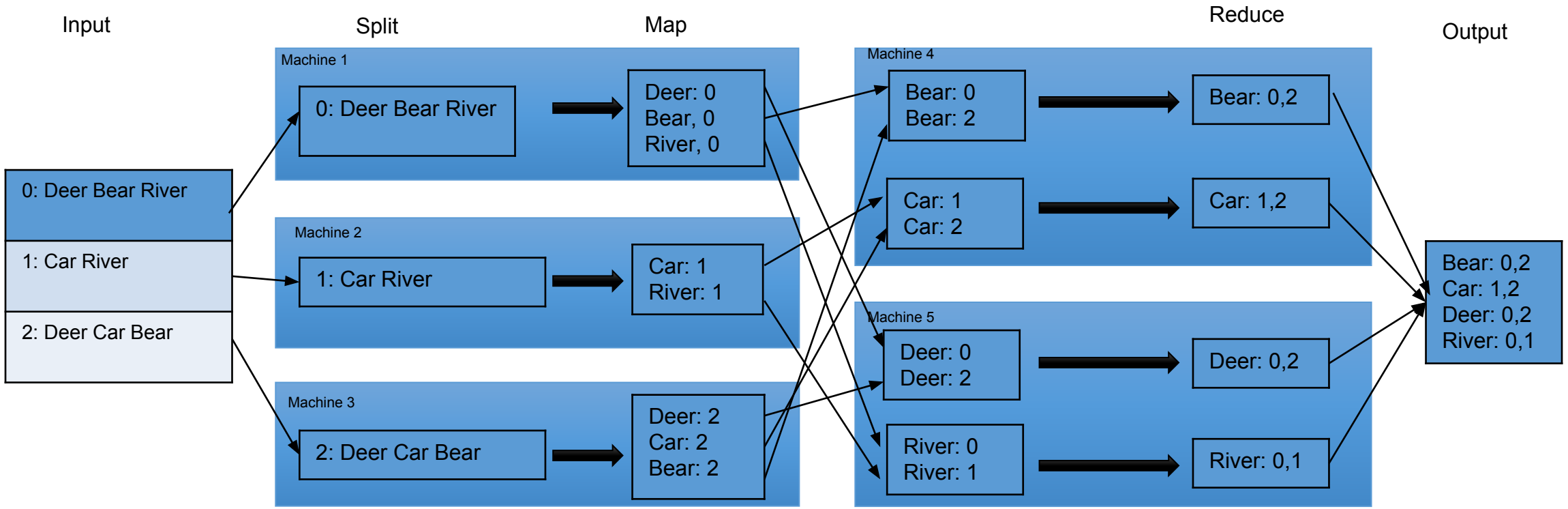
Input

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

Output

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

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);
```

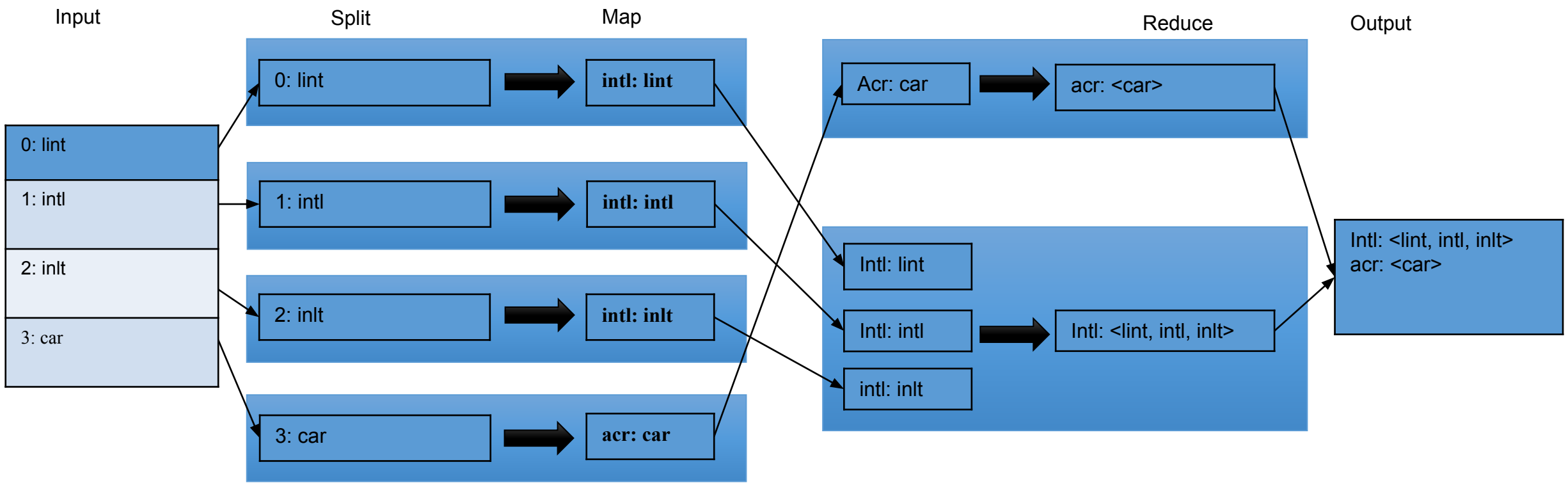
```
//key: the name of a word
//valueList: the appearances of this word in documents
Reduce( string key, list valueList )
    List sumList;
    for value in valueList:
        sumList.append(value);
    OutputFinal( key, sumList );
```

Apple Interviewer: Anagram - Map Reduce

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

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

Anagram - Map Reduce



Map

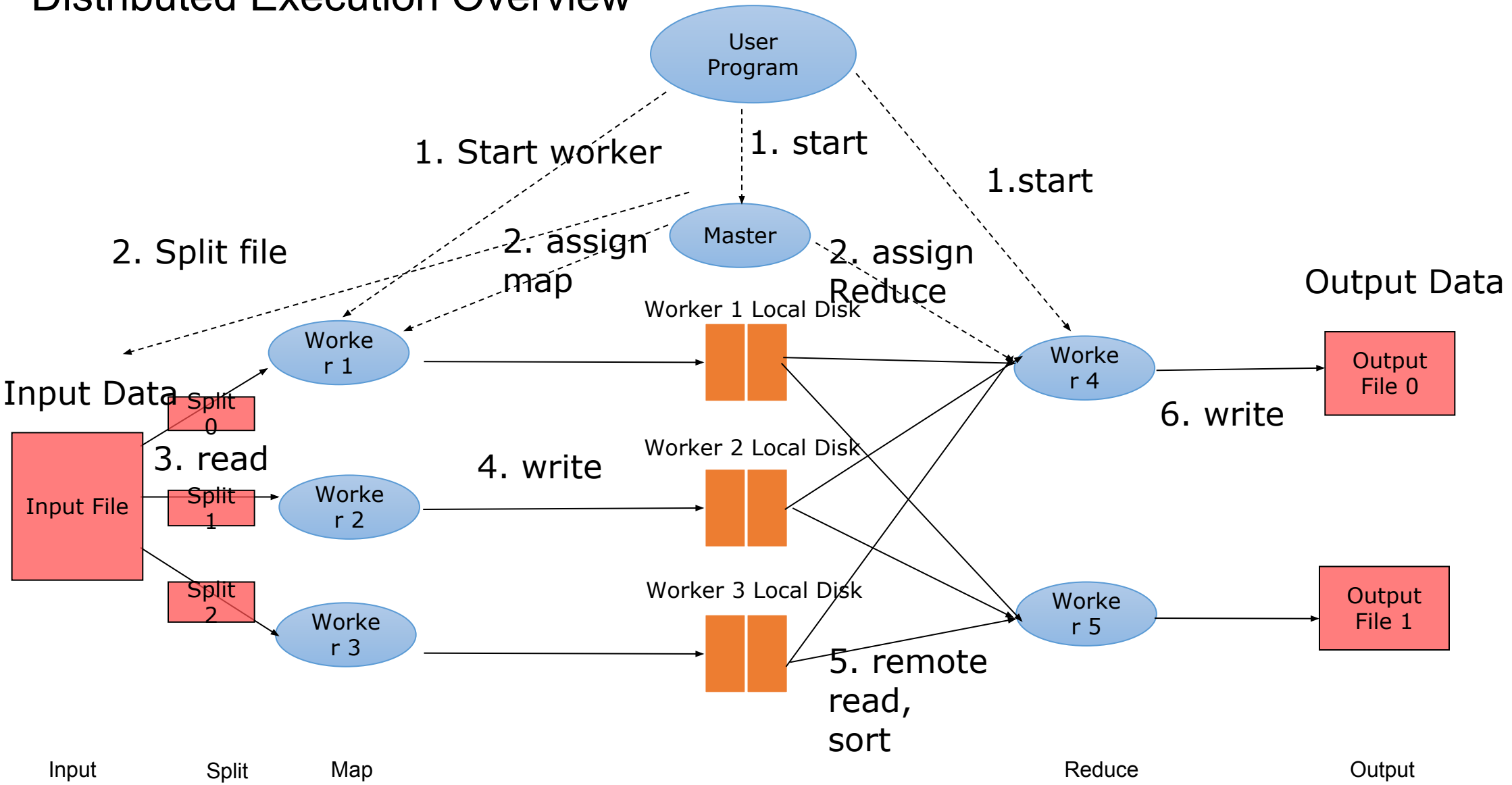
Input	Output
k = lineID	k = sort(word)
v = word	v = word

Reduce

Input	Output
k = sort(word)	k = anagram0
v = {anagram0, anagram1, ...}	v = {anagram1, anagram2,...}

Interviewer:
Design a MapReduce system

Distributed Execution Overview



常见问题答疑

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

4. Input 和 Output 存放在哪？
   存放在GFS里面

5. Local Disk 上面的Mapper output 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。

MapReduce Whole process

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
  - 大数据班尽请期待.....

相关阅读资料

- Novice, http://url.cn/YM1tHI
- Expert, http://url.cn/b41Qzf
- Expert, http://url.cn/1VO6Qa
- Expert, http://url.cn/ccvLOr
- Expert/Master, http://url.cn/SuVoAP
- Expert/Master, http://url.cn/SJCoso
- Master, http://url.cn/Z3OOVZ

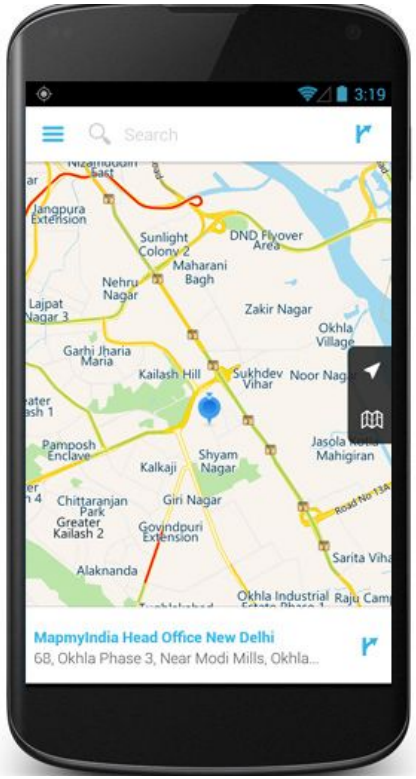
课间休息

Microsoft Interviewer: Design a Lookup Service

Bing Map Look up Service

Design a Lookup Service

- 问题：设计一个只读的lookup service
  - 10 billion key-value pair
  - (not update every day)
  - Key: GeoLocation; Value: image and building name



Scenario

- **S**cenario 比较明确 Lookup
  - How big is the data? How big is the key, and value?
    - Key (Latitude37.4088799, longitude-122.0894253)
      - Each key size < 20 B
      - Total key size = 200GB
    - Value(pic and all the building name on this pic)
      - Each Value size = 100 KB,
      - Total value size = 1PB

- **S**ervice: App client + Web servers + Storage Service

Service

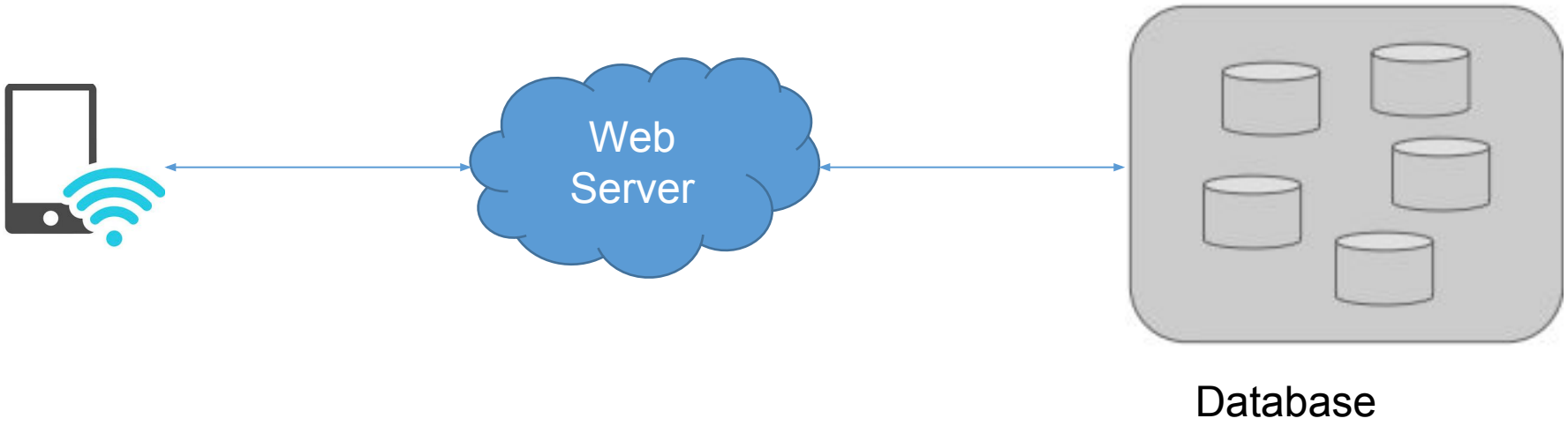
- **S**ervice: App client + Web servers + Storage Service

Storage

- **S**torage:
  - Hash Map
  - Database (Sql, NoSql)
  - GFS

Storage

- **S**torage:
  - Hash Map
  - Database (Sql, NoSql)
  - GFS

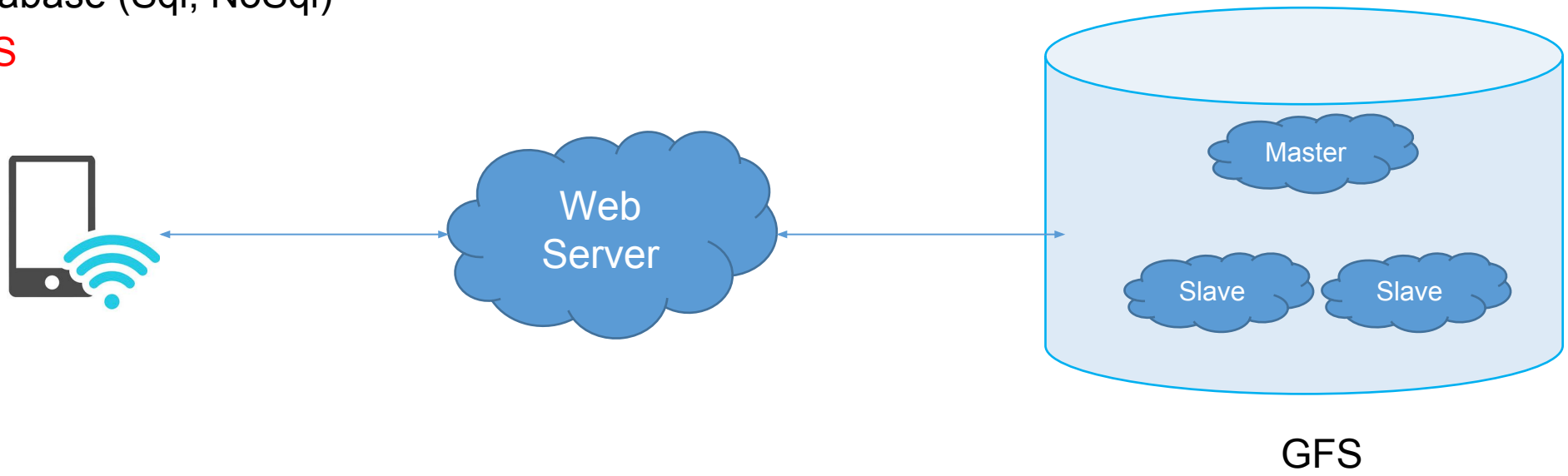


Storage

- **S**torage:
  - Database (Sql, NoSql)
    - Good but not perfect.
    - 1. This is read only system, bigtable is optimized for write, read is not that fast.
    - 2. If the QPS > 10k

Storage

- **S**torage:
  - Hash Map
  - Database (Sql, NoSql)
  - GFS

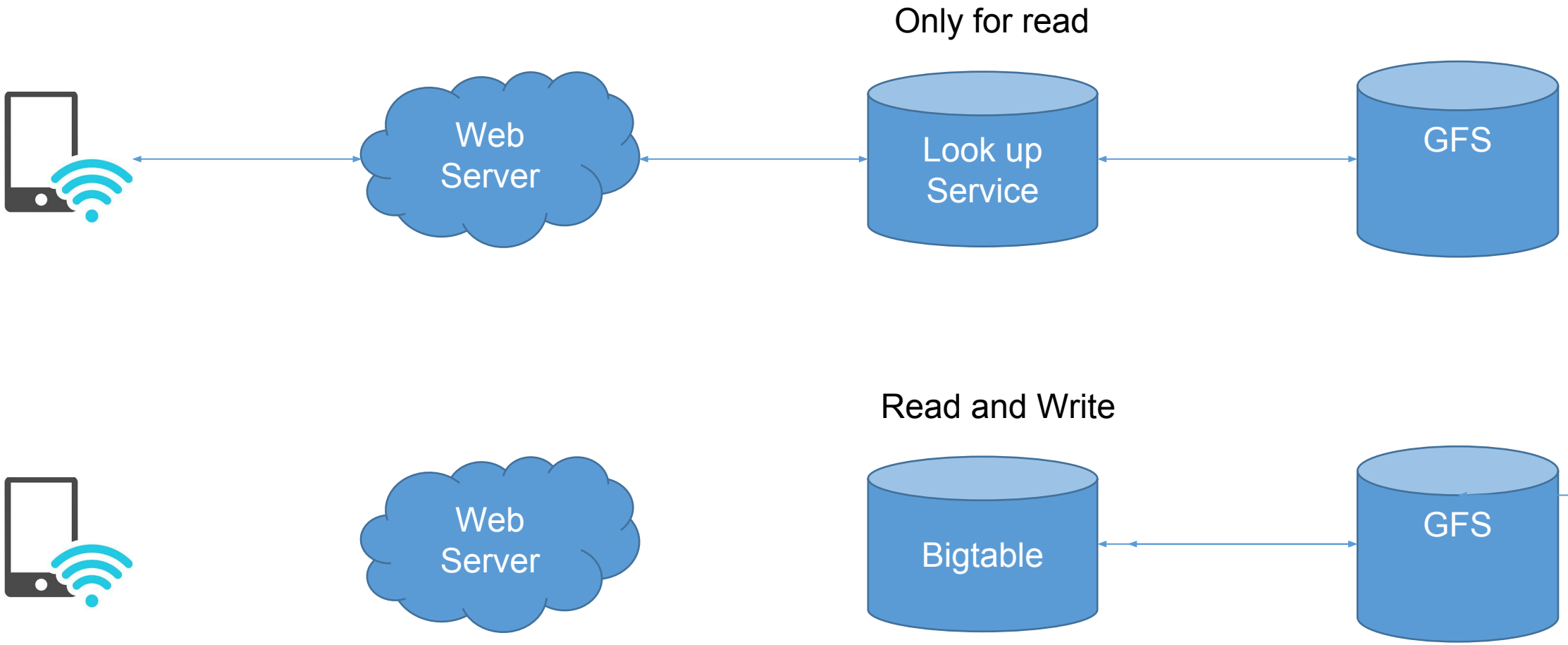


可以直接通过GFS实现key value look up 么？

GFS的操作都是对于文件

所以还是需要在文件系统基础上搭一个look up的系统

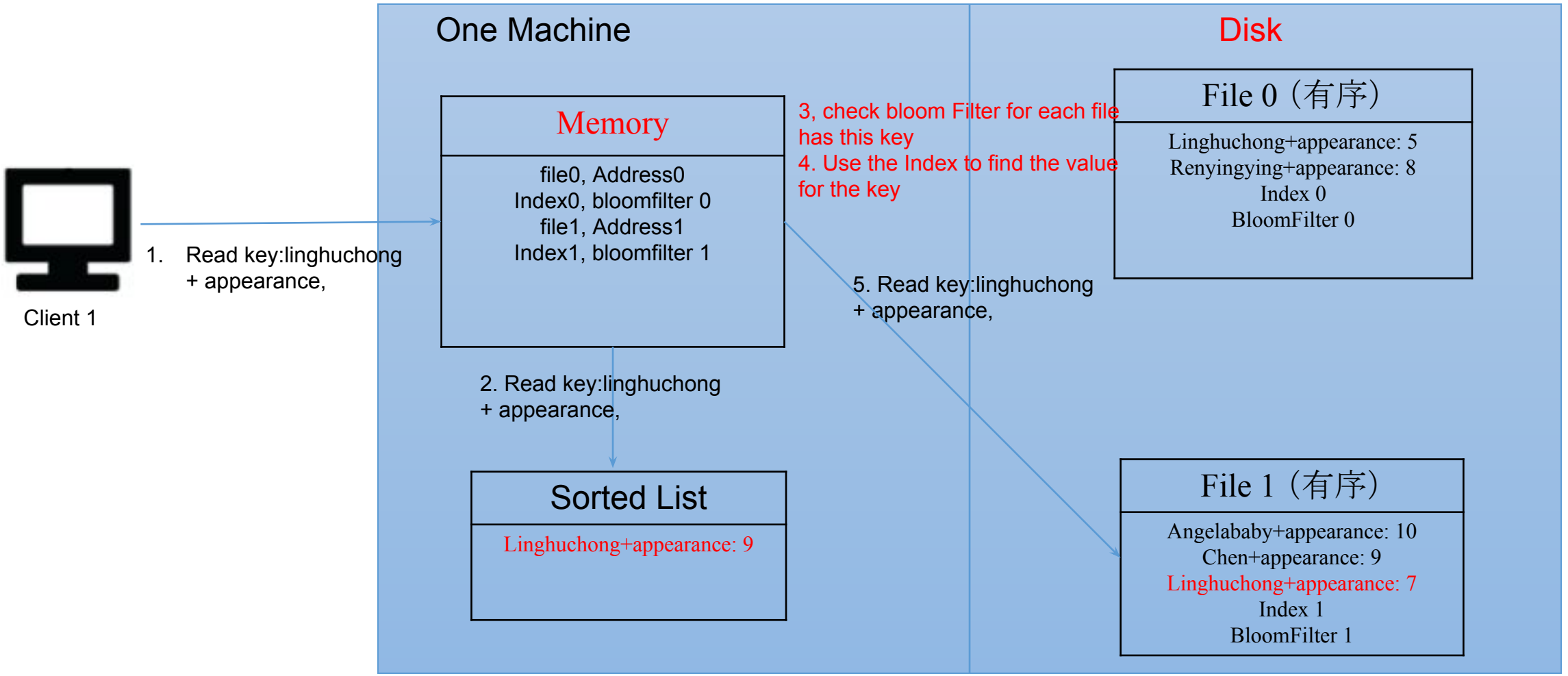
Overview of Design a Lookup Service



回顾一下big table

How to read on One Machine Bigtable

读出过程

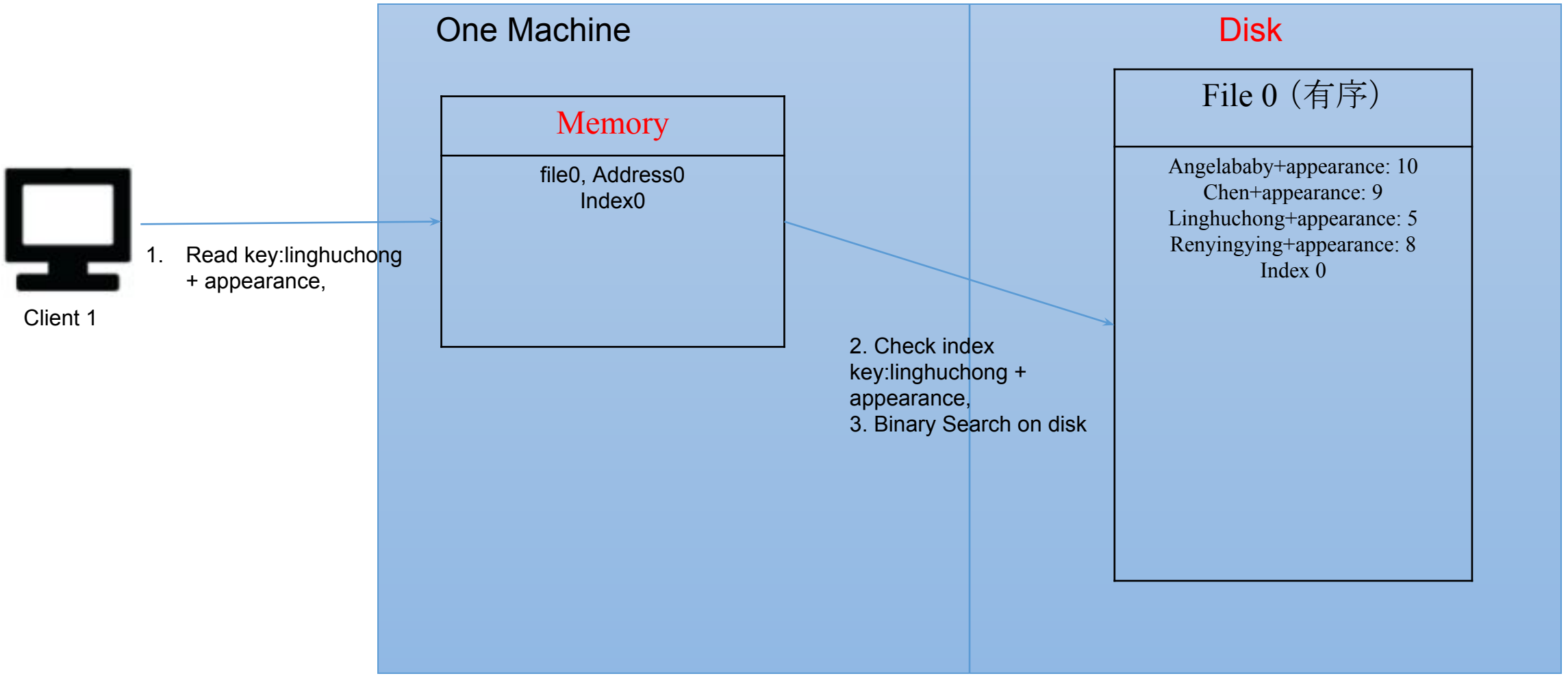


Why so complicated?

- Why sorted list?
  - 因为写的时候想append 操作。
- Why separate into file 0 and file1?
  - 不是分出来的，是每一次sorted list 满了之后写到文件里面的。
- Why build index and bloom filter ?
  - 因为我们要检查所有的文件所以需要建index 还有用bloom filter。
- 这一切的罪魁祸首就是写的时候要append

现在没有写操作

读出过程

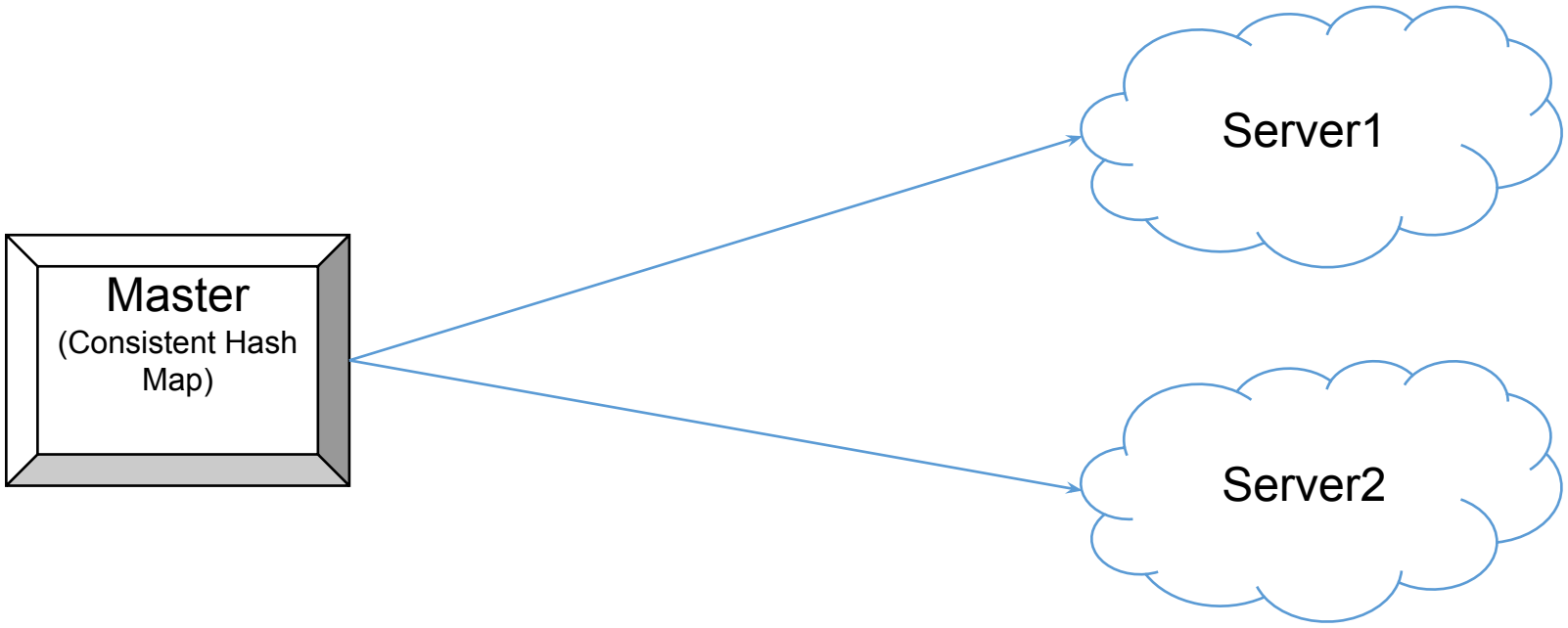


Is One Server Disk big enough for total value size = 1PB

Master + Slave

Interviewer: How to manager server?

- Key
  - Master + Slave
  - Master has HashMap[key, server address]



Sharding

Key is (Latitude37.4088799, longitude-122.0894253)

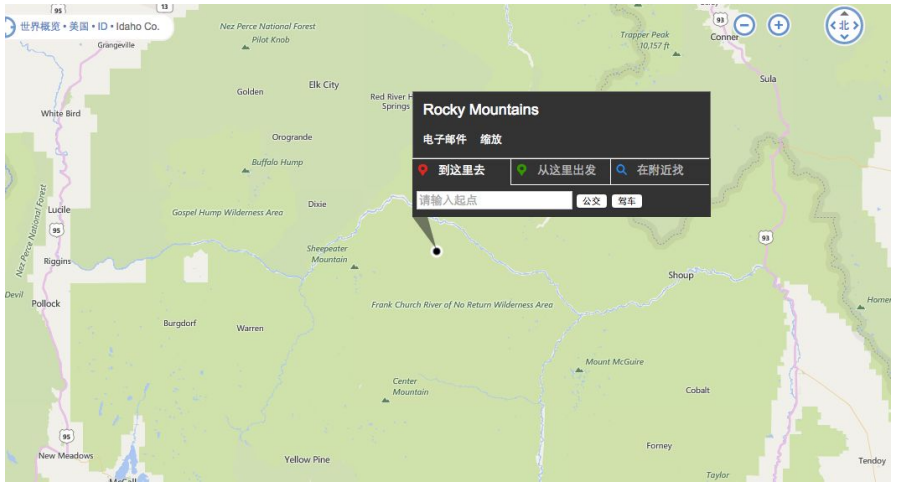
Horizontal Sharding

Vertical Sharding

City sharding

City sharding

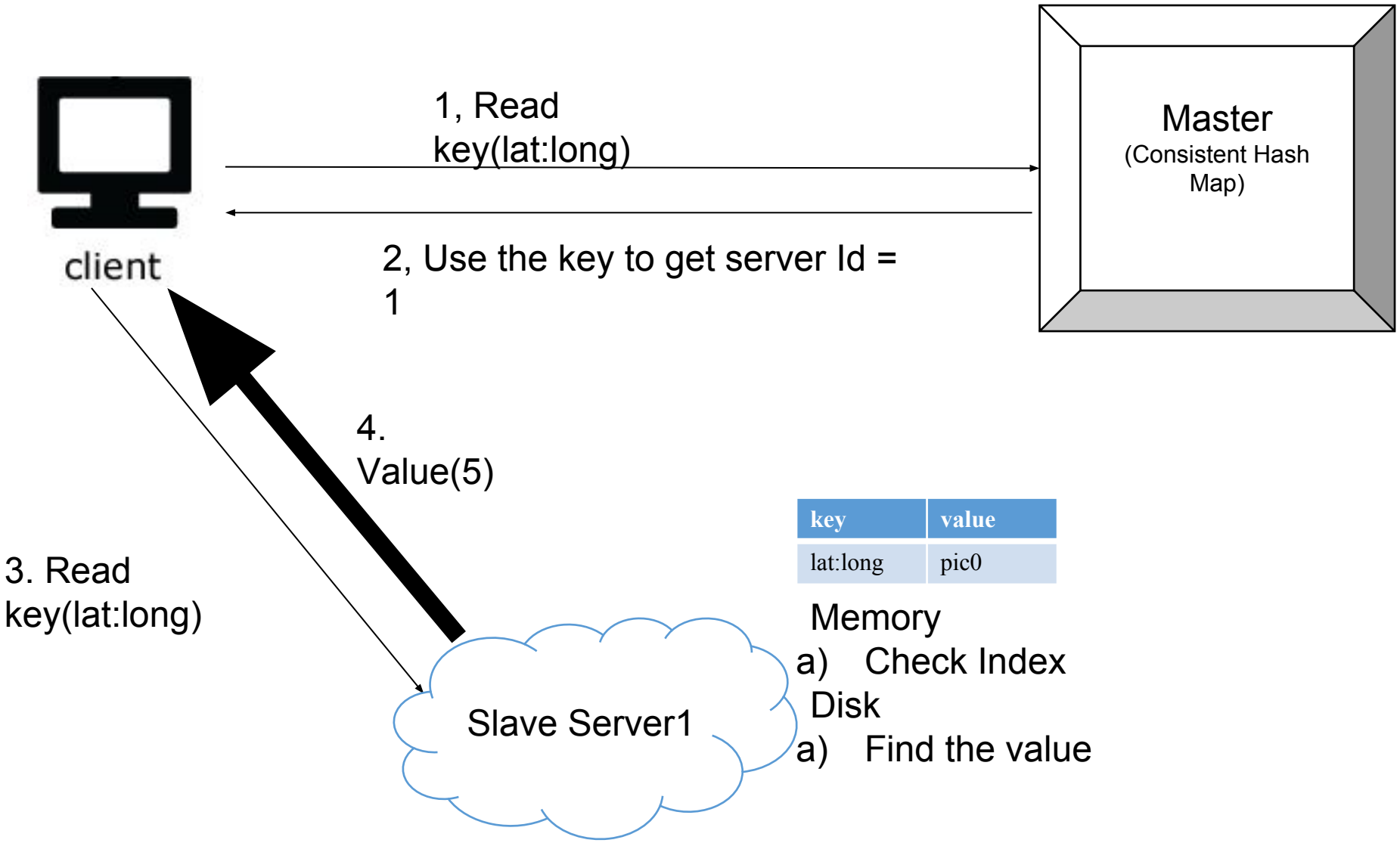
Uber 可以不去山上，
难道用Map 就不去爬山了？



Horizontal Sharding

Key is (Latitude37.4088799, longitude-122.0894253)

Interview: How to read a key?



Interviewer: How to solve big disk size problem? ?

把所有数据存到GFS里面

Advantage:

1. Disk Size
2. Replica
3. Failure and Recovery

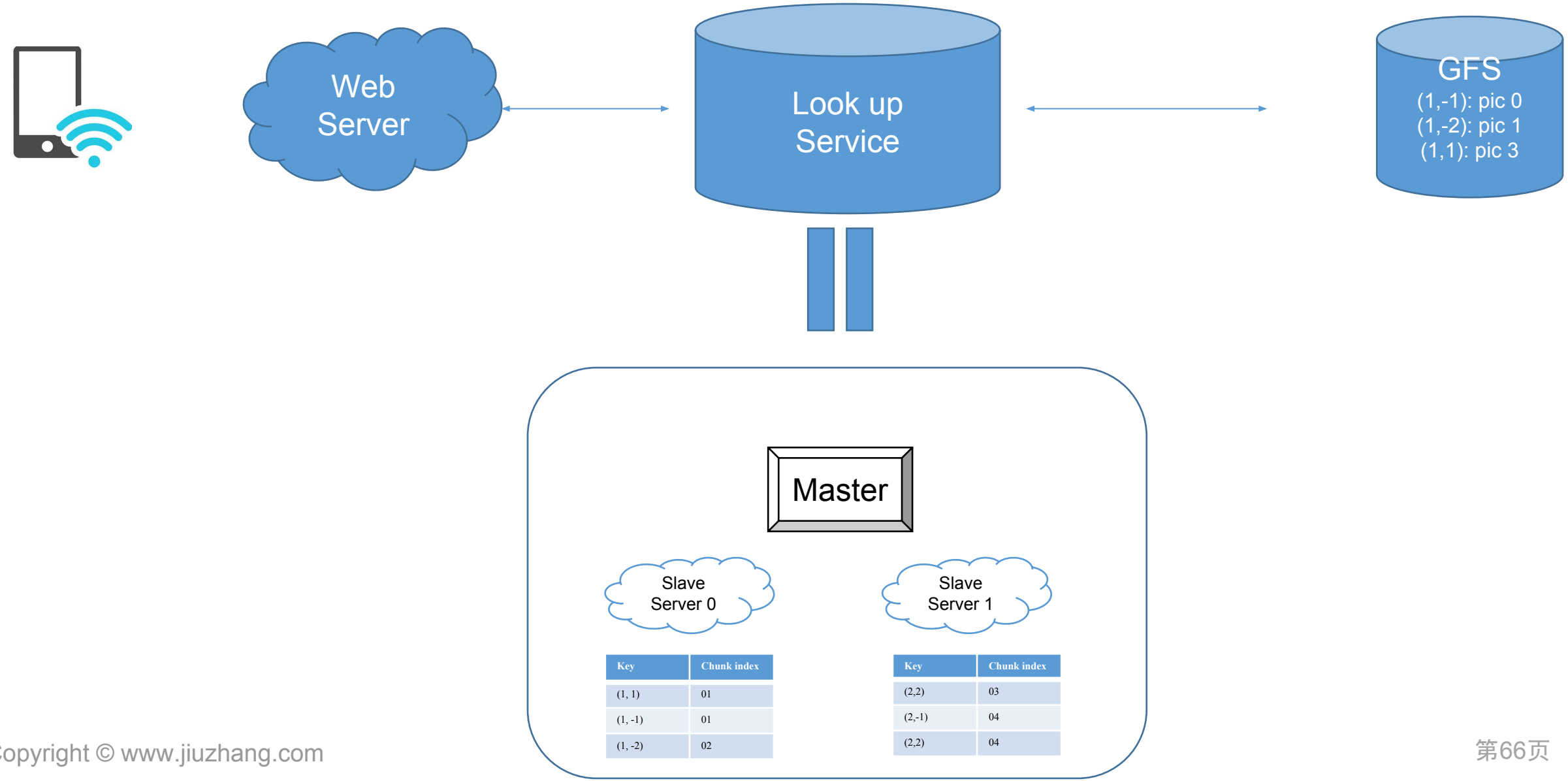
Question: For given key, how do we know which chunk we should read?

Answer: store key: chunk Index

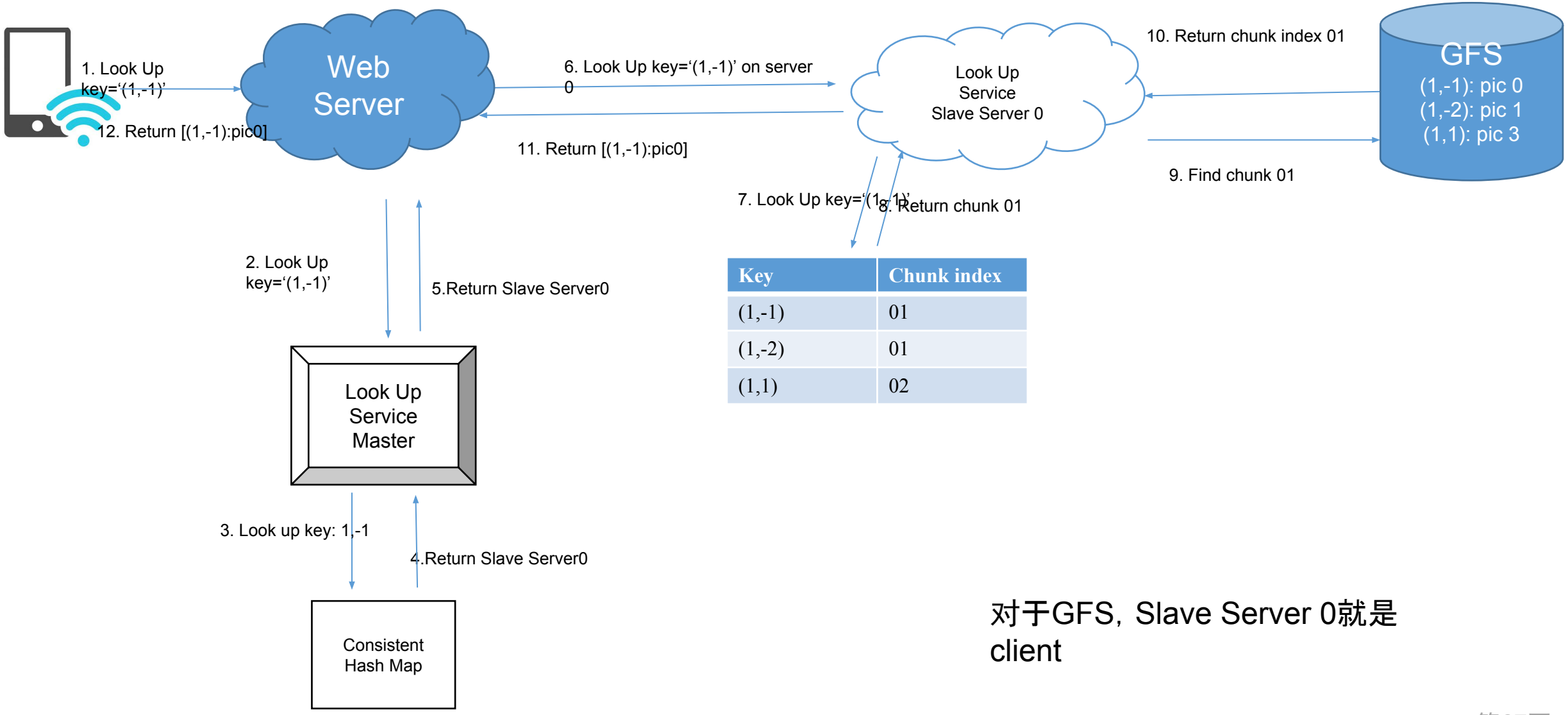
Key	Chunk index
(1,-1)	01
(1,-2)	01
(1,1)	02

这张表会不会很大？

Lookup Service Storage



What is the lookup process?



Key	Chunk index
(1,-1)	01
(1,-2)	01
(1,1)	02

对于GFS，Slave Server 0就是client

When and how to initialize this map?

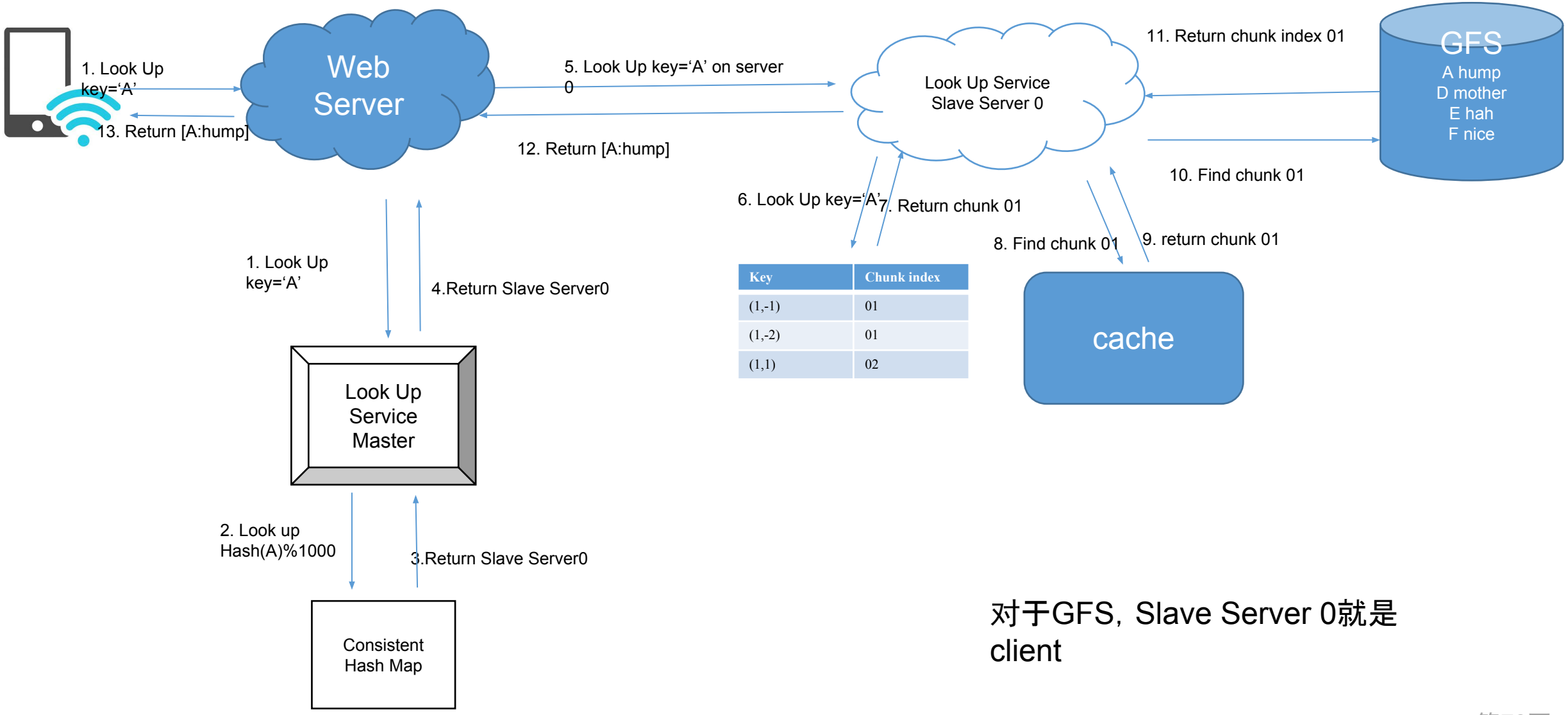
At the beginning, master will distribute the key and initialize this map for every slave server

Key	Chunk index
(1,-1)	01
(1,-2)	01
(1,1)	02

Question: If two keys near each other, do we need to request GFS twice?

Answer: We can use the server as Cache for a 64M chunk

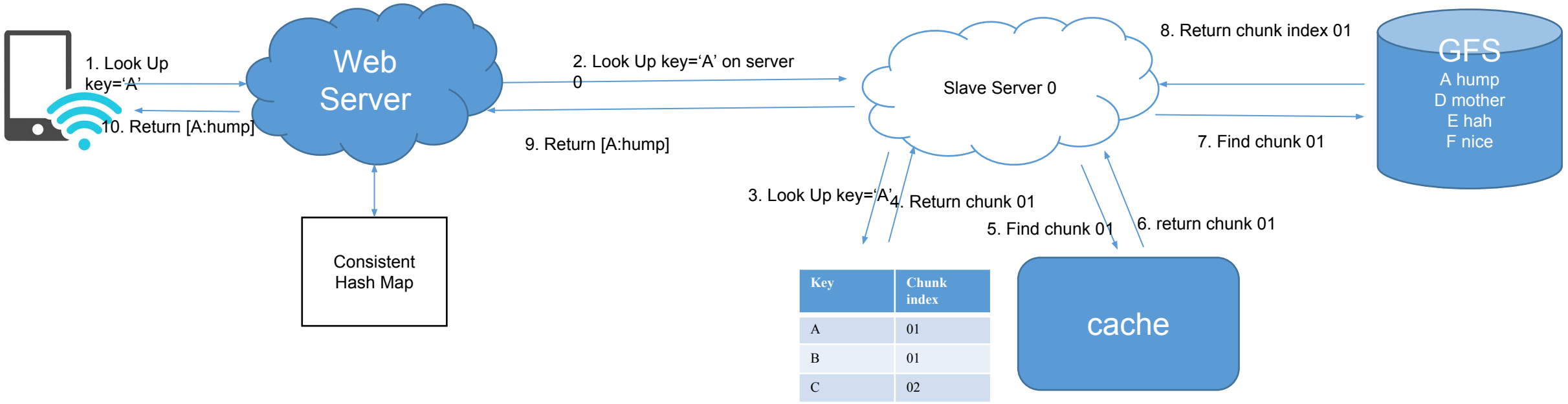
What is the lookup process?



对于GFS，Slave Server 0就是client

Interviewer: Master is bottle neck?

What is the lookup process?



Key	Chunk index
A	01
B	01
C	02

Summary of Lookup Service

- Design
  - Client + Master + Server
- Client
  - Look up
  - Consistent Hash Map
- Server
  - Maintain the Data (Key value pairs)
  - Connect to GFS
- Master
  - Shard the file
  - Maintain the MetaData (Similar to GFS master)
  - Manage the servers health

Summary of Today

- Map Reduce Step
  - Step1 Input
  - Step2 Split
  - Step3 Map
  - Step4 传输
  - Step5 Reduce
  - Step6 Output
- Lookup Service
  - 怎么把已学的东西运用
    - Master
    - Client
    - Server
    - How to connect to GFS