常见系统设计面试题汇总

如何设计Twitter

短网址系统设计

如何设计Uber

分布式系统

1) Inverted index

2) Anagram

3) Word Count

4) Distributed File System Design 设计

实时位置系统

1) Design Yelp

2) Design Uber

3) Design Whatsapp

搜索系统设计

1) crawler

2) typeahead

3) inverted index

failure rate, DNS, web server, file server, timeout, content delivery network, cookie, HTTP, divide and conquer, Internet service provider, hosts, hijack, retention rate, cache, lazy load, rate limiter, QPS, counter, expire, request list, token bucket

网站系统设计

1) What happend if you visit www.google.com?

2) How to design rate limiter?

3) How to design data log?

面向对象设计

数据库系统设计

database, primary/foreign key, table, row, attribute, index, transaction, dlog, lock, lifecycle graph, binary search tree, B+Tree, atomicity, consistency, isolation, durability, session

1. 设计文件系统

2. 数据结构for spreadsheet

3. 一个app 需要用cache，怎么实现thread safe

4. social network, billions id, every id has about 100 friends roughly, what is max connections between any two ppls. Write algorithm to return min connections between two ids: int min_connection(id1, id2)

you can call following functions

expand(id) return friends list of id

expandall(list) return friends union of all the ids in the list

intersection(list1, list2) return intersection

removeintersection(list1, list2)

5. Open google.com, you type some words in the edit box to search something, it will return lots of search results. Among these returning results (that is, website link), you may only CLICK some results that are interesting to you. The system will record the "CLICK" action. Finally, you will have the search results (i.e. url) and "CLICK" informatin at hand.

Question: how do you find the similarity of these searching?

6. 如何找出最热门的话题(根据tweets)。如果一个话题一直热门，我们不想考虑怎么办

7. Discuss design challenges of a distributed web crawler running on commercial PCs. How to utilize network bandwidth of those PCs efficiently?

8. Design a site similar to tinyurl.com

9. large log file,含有 customer id, product id, time stamp 想得到在某一天中某个custom看网页的次数1. 足够memory 2. limited memory

10. 设计一个actor 和movie 的数据库的schema, 支持从movie 得到它的actors 和从actor 得到ta 出现过的moive (Google, phone, 2006)

11. 某建筑有五十层高,打算装俩电梯,设计该电梯系统

12. how to design facebook's newsfeed?

13. 一个文件里n 行m 列 ,每行是一个record ,每列一个feature ,你时不时要按不同 feature 排序和查找。不能用数据库，文件大小内存能装下，数据结构和算法不限，语言不限，给出你最好的办法。

14. Design online game

15. static 变量用来在整个class 中共享数据.基于此，各种synchronization 技术， 以及 busy waiting 的优缺点 ,啥时候要用基于busy waiting 的 spinlock 主要是基于性能的探讨。 如果有一个应用程序运行时没有达到timing constraint，你如何去分析问题出在哪儿， 可以用什么工具或者技术。

16. 设计题，有一个多台机器构成的cluster。现在有大量公司的数据文件（并有多个备份）。如果设计一个算法，使得每台机器尽量均衡的使用，并且 每个公司文件的不同 copy 不能存在于同一台机器上。主要的Idea 就是用round-`robin 的方式assign 每个公司的原数据文件到一台机器，再结合使用hashtable。

Interviewer 提到我的解法正是他现在在使用的解法。

17. Design a class providing lock function which provides lock only if it sees there are no possible deadlocks.

18. 设计一个分布式文件系统，给定path name，可以读写文件。具体的system design 这里就不提了。其中一个细节是，给定path name，怎么知道哪个node 拥有这个文件。我提出需要实现一个lookup function，它可以是一个hash function，也可以是一个lookup table。如果是lookup table，为了让所有client sysnc，可以考虑额外做一个lookup cluster。然后 Interviewer 很纠结，既然可以用hash function，为什么还搞得那么复杂。我就告诉他hash function 的缺点。假定一开始有N 个node， hash function 把M 个文件 uniformly

distribute 到N 个node 上。某天发现capacity 不够，加了一个node。首先，要通知所有的client machine，configuration 改变了。如果不想重启client machine 的process，这不是一个trivial job。其次，文件到node 的mapping 也变了。比如，本来按照hash function, 一个文件是放在node 1。加了一个node 后 ,它可能就map 到node 2 了。平均来说 ,N/(N+1)的文件需要move 到新的node。这个data migration 还是很大的。然后我就提出一些hash function 的design ,可以减少data migration。最后他提了一个问题 ,说要实现一个function，要统计distributed file system 所有目录的大小。前提是，一个目录下的文件可能放在不同的node 上。我说这个不就是在每个node 上统计，然后发到一个merge 吗。

他说对 ,但是又问用什么data structure 来表示。我说这就是hash table ,key 就是directory name，value 就是大小。因为directory 本身是树结构，这个hash table 的key 可以用tree 来组织。最后让我实现一个function，把我说得这个data structure serialize 成byte array。因为这个byte array 就是网络传输的data。我用了depth first traverse。不过等我程序写完，才发现，用 breathfirst traverse 会更方便，code 也会很简洁

19. 超大图的存储问题

20. 给个Lock w/ two atomic method lock() and unlock()，请用lock 实现一个文件读写的系统，要求：

1: reader blocks writer;;

2: writer blocks reader;;

3: writer blocks writer;;

21.设计一个web cache server，假设存储网页数量是10 个billion，打算怎么设计

22.你可以得到网站访问记录，每条记录有user IP，写一个程序，要随时能算出过去5 分钟

内访问次数最多的1000 个IP. 这个好像跟着这个rolling window 的precision 有关，所以我们暂且定为5 秒钟update 一次window

23. Design free and malloc.

24. How to design data structures for a facebook network and how to design an algorithm to find connection? How to optimize it if data is distributed into multiple computers?

25. Design a deck class and member function to randomly select a card from those cards which haven’ t been selected before. (You can assume the number of this function call will never be larger than the number of cards) For example, we have a deck of four cards: 1,2,3,4. First it may select 3, then next time it should randomly select one from 1,2,4… And design a member function to reset.

26. Google search design problem. How to distribute data and how to a design backup system.

27. 设计一个online chat system

28. design bit.ly url shortening web service。算法设计，后端存储，中间层cache，前端load balance，最后是web analytics。

29. Design and implement an algorithm that would correct typos: for example, if an extra letter is added, what would you do?

30. Suppose there are 2 persons A and B on FB. A should be able to view the pictures of B only if either A is friend of B or A and B have at least one common friend. The

interviewer discussed it for nearly 30 minutes. The discussion mainly included following points：

[1] How are you going to store the list of friends for a given user?

[2] File system vs DB

[3] Given list of friends of 2 users, how are you going to find common friends?

[4] If you are going to store the friends in DB then how will the table look like?

[5] How many servers do you need?

[6] How are you going to allocate work to servers?

[7] How many copies of data will you need?

[8] What problems will you face if you are maintaining multiple copies of data.

31. Design structure for auto completion

32. 如何实现search suggestions。

33. 设计 fb 的系统支持like 那个button

34. design 股票#，time，price；

-ˇ设计一个client side 显示股票信息，给出尽可能多的user case

-ˇ在给出的user case 里面，怎么设计客户端，使得客户段性能提高

-ˇ怎么设计server 端

-ˇ数据如何传输

-ˇserver 端如何保存数据

-ˇ怎么设计database table 保存数据

-ˇ不用index 怎么提高数据查询速度

-ˇdatabase 是怎么实现数据查询的（要求从database implementation 角度解释）_