**Note** : In design pattern problems , interview may not be interested in data structre,

So leave details on method call.

**Call centre problem**

Imagine you have a call center with three levels of employees: respondent,

manager, and director. An incoming telephone call must be first allocated to a

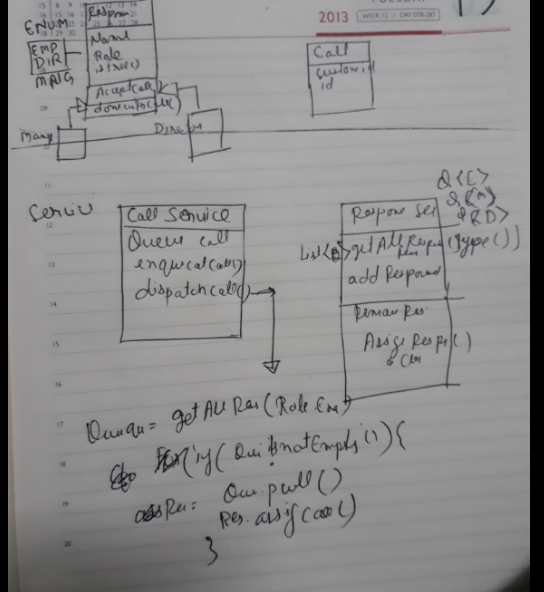
respondent who is free. If the respondent can't handle the call, he or she must

escalate the call to a manager. If the manager is not free or not able to handle it,

then the call should be escalated to a director. Design the classes and data structures

for this problem. Implement a method dispatchCall() which assigns a

call to the first available employee.



Note, here details of how the call will be handled can be abstracted ,

Parking Lot problem

Design a railway Railway Reservation system

<https://www.ibm.com/developerworks/rational/tutorials/ar-designpat1/index.html>

Situational problem: Design problems

Design Step-by step

1. Scope the problem
2. Make reasonable assumption
3. Draw major components
4. Identify key issues
5. May redesign for key issues

Algorithm that scale

Step by –step

Ask questions – understand the problems

Focus of MVP minimum viable product – agile development

Focus on number priority

Make believe

Pretend that the data can all fit on one machine and there are no memory limitations.

How would you solve the problem?

*Get Real*

Now, go back to the original problem. How much data can you fit on one machine,

and what problems will occur when you split the data up

*Solve Problems*

Finally, think about how to solve the issues you identified in Step 2. Remember that the

solution for each issue might be to actually remove the issue entirely,

Use Data De-Normalization for speed

Joins slows the read operation, add the reduduent information in the table to make them read faster

Database portioning – Sharding

1. Vertical – some table in one location, and rest in others based on the feature, social network example
2. Key based
3. Directory Based

Caching

Networking matrix

Bandwith

Latency

ThroughPut

Low Latency and high throughput

Map reduce

**Considerations**

Failures – Any part of the system can fail , need to plan for that

Availbility vs Realiability

Read Heavy –Write Heavey – If write heavy then queue up the requests

Secutity

Questios

Imagine you are building some sort of service that will be called by up to 1000 client applications to get simple end-of-day stock price information (open,close, high, low). You may assume that you already have the data, and you can store it in any format you wish. How would you design the client-facing service which provides the information to client applications? You are responsible for the development, rollout, and ongoing monitoring and maintenance of the feed. Describe the different methods you considered and why you would recommend your approach. Your service can use any technologies you wish, and can distribute the information to the client applications in any mechanism you choose.

Solution

Discusses approach to solve the problem

1. FTP file
2. download the txt file from web
3. XML or FIX format transfer
4. Rest web call

Discuss prons and cons of each

Q2 How would you design the data structures for a very large social network like Facebook or Linkedln? Describe how you would design an algorithm to show the connection, or path, between two people (e.g., Me -> Bob -> Susan -> Jason -> You).

Ans. Do it through BFS for a one machine, then expand to multiple machines

Web Crawler

How would you design a web crawler

**Duplicate URLs**

You have 10 billian urls , how to detect duplicates .

The problem of performing operation on a large set of data lets say 400 GB of records

**Convert the records into GB, this can help to devide the problem , if not able to convert use a dummy number 400 GB**

Solution : 1 url has 100 words means, 1 word = bytes , 1 url = 400 bytes

10 billian url = 1 E12\*400 bytes , or 4000 GB not possible to store this data in memory

**How to solve the problem for small scale**

Create a hashMap and set value as true when found duplicate.

**How to scale the problem to 400GB records**

Devide it into 400 files, on the basis of hashMap ,

Place the urls with same hash in a single file.

Process each file individually   
or expand it to 400 machine in a cluster, each machine will process for a hash records/

**Caching best problem**

Search Engine Caching

*Imagine a web server for a simplified search engine. This system has WO machines to*

*respond to search queries,which may then call out using processSearch (string*

*query) to another cluster of machines to actually get the result. The machine*

*which responds to a given query is chosen at random, so you can not guarantee that*

*the same machine will always respond to the same request. The method process -*

*Search is very expensive. Design a caching mechanism to cache the results of the*

*most recent queries. Be sure to explain how you would update the cache when data*

*changes.*

*Caching and updating the data, invalidting , how to store multiple copies of data*

*Lets do it for a single machine*

1. Design a cache a simple cache , a map , which query as id and result as a list : which stores most recent access values

Change the order in the list as search result , basis of how

Move to the front and delete last