20|Feb|2022

So, any thing LIS with Sequence Maximum output these three will

Ok so our two output let's try to understand LIS with Sequence.

let's take one Example

max of { 1, lis(arr, n-1), 1 + lis(arr, n-1) }

arr[] = {10, 9, 3, 5, 4, 1, 1, 7, 8}

n = 8 // sizeof arr / sizeof(arr[0]) ← This will give you value n.

Let say your recursive function

lis(n) means lis method called for n elements of array "arr"

This is Actually like

lis(arr, n) it is same as if I say

i have lis(arr, n-1) and nth element

① nth element is added with lis(arr, n-1) also it follows the Increasing Sequence rule

meaning

for all j 0 < j < n

arr[j] <= arr[n]

and so in this case our output is Lis(arr, n-1) + 1

else

② lis[n] = 1

② I ignore nth element and start finding the longest increasing sequence, so the longest Increasing Sequence is equivalent to lis(arr, n-1)

max of $\{1, lis(arr, n-1), 1 + lis(arr, n-1)\}$

Practically ok, but how do you do collection of your longest increasing subsequence. ?

(A)

Another improvement in this problem is how you will find all the possible longest Increasing Subsequence ~~for~~ from this problem.

Input : $A = \{10, 2, 5, 3, 7, 101, 18\}$

Longest Increasing Sequence

Output { length = 4

$\{2, 5, 7, 101\}$, $\{2, 5, 7, 18\}$ $\{2, 3, 7, 101\}$, $\{2, 3, 7, 18\}$ etc. }

At this point, I can see one or two clarification is required.

1. Do we have duplicate number also ?
   Ans: NO
2. Do we require to store the subsequence also ?
   Ans : yes

Let's go through one more time with different approach

Input arr: {1, 2, 5, 3, 4}

Another clarity we might have that in input Array we don't have duplicated numbers.

NO

- Increasing Subsequence
  - Subsequence means not necessarily contiguous
  - current number is greater than previous number
    - (Increasing)

Let's see with Recursive tree

Today's topic is Dropbox : system Design

Similar service : Google Drive, one Drive etc.

FR

1. Understand the Requirement and notedown some key functional and Non functional requirement after consulting the Interviewer.

Let's list down few top level Requirement

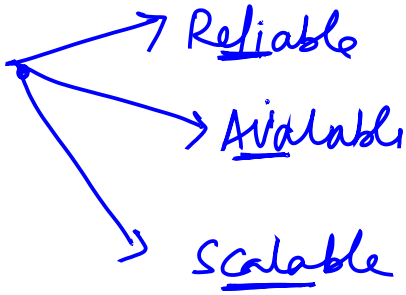
1. User can upload/download the video/photo/files etc. <<cloud storage>>
2. Used Can able to do from anywhere, means through browser or through mobile App etc. <<multiple clients>>
3. User can able to share their files/photo's to their friends/family. <<sharing>>
4. User Can download/upload the files and it will be available across the devices User usually Usage. <<Auto sync>>
5. In this ~~operation~~ service, we must have Acid-ity in features. ACID (Atomicity, Consistency, Isolation and Durability is must feature of this file storage service. We will see how certain database type is helping us in achieving ACID properties. For Ex: ~~the~~ SQL database are always concidered as ACID Complined service.

6. This service should support large files, for example files in GB's.

7. File is a large data and so incase you want to modify files, online editing is not best suitable so, system should allow offline modification of these files.

Non function Requirement NFR

1. system should be highly Reliable and durable
2. system should be highly available for use
3. system should be Scalable to Infinite theoritically



---

(3) Before moving to capacity estimation, let's discuss few more points with your Interviewer. (Additional design Considerations)

(A) What you think about this system, is this system is Read bias or write bias? OR do you think both?

(B) is their any Ration of Read vs write in your mind?

(C) share your thought if any comes in your mind which having big imbact on design

decision. Confirm with your Interviewer. This could be the good strategy sometime.

@ Note: Here Interview may or may not confirm their opinion but your job is to share your view without re-iterating yourself. observe Interviewer reaction on your sharing.

(C) for example in this service where you have very large files may be in GB can be dropped into drive, and so to handle multiple cases ex: failure, performance, replication, etc. (modification) it is recommended to divide the files into fragments of some standard size (4MB) and upload into drive. OR read your fragments from server and merge into files on your local device before play/preview.

(D) In very large file, we might have issue in modification because if you haven't divided file into chunks/fragment than on each small modification also you need to upload the complete file. Which is complete mess in performance. So this approach can't work. The Alternative is to devide large files into chunks/fragments and upload this into Block storage server. You also required to maintain some metadata for managing files vs chunks mapping. it stored in seperate Metadata Server. (Seperate from Block Storage server)

(E) Another key decision I have to make to store metadata on local device also to avoid too many round trips with metadata servers.

So, Above mentioned all points are key decisions of our future design strategy.

Another key strategy in system design is Estimation of your system and their constraints if any.

(4) Capacity Estimation and Constraints.

- Let's assume 500M total user, and 100M daily active user (DAU) <<user>> (1 Million user per second)
- This system is kind of service where usually user's are connected with multiple device and roughly i am taking average 03 different devices
- I am assuming that each user have on an average 200 files/photos in their account so total (500 × 200) Million = 100 Billion files/photos into your system.
- Let's assume each file on an average of size 100 KB (Kilo byte)

(100 Billion × 100) KB (Huge data requirement)

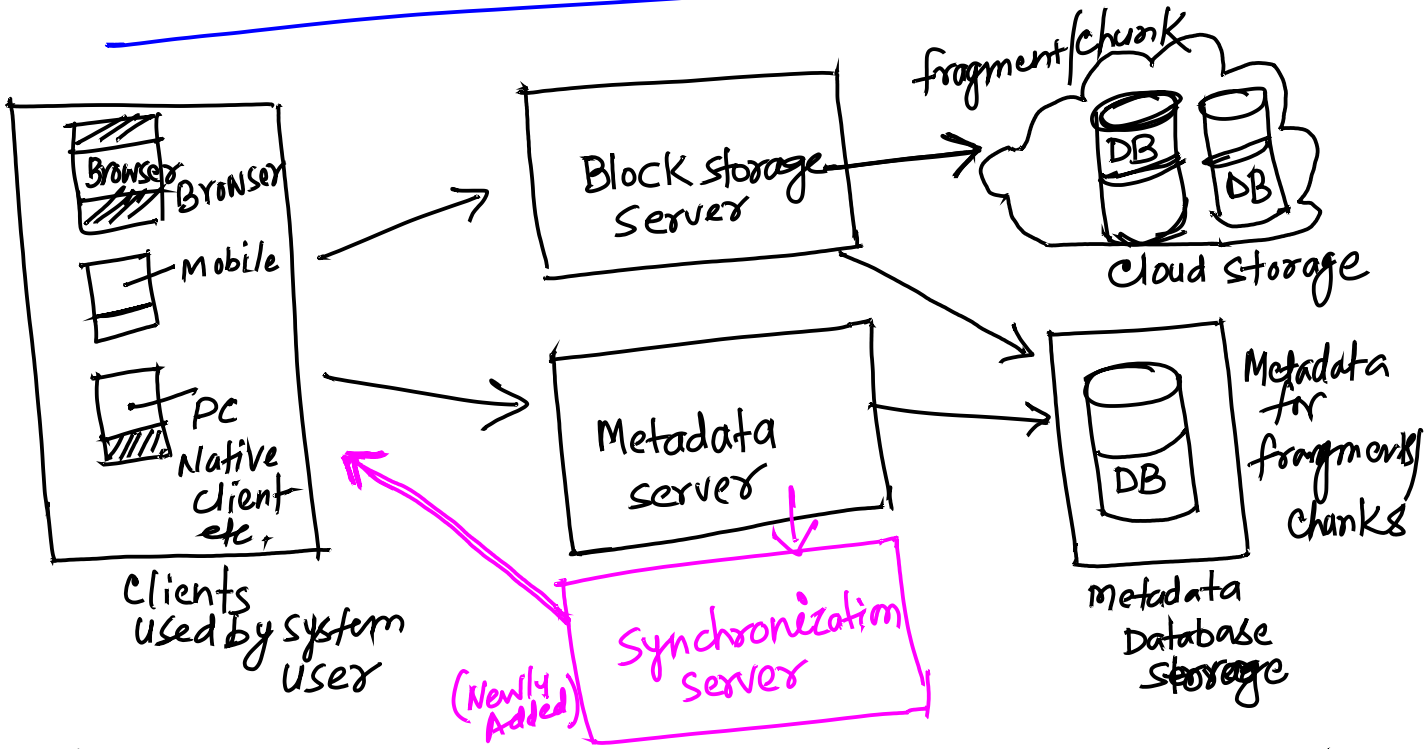
$\leftarrow (100 \times 10^9 \times 100)$ KB = 10 Peta Byte

10^{10} MB
10^7 GB
10^4 TB

- Let's calculate per minute or per second Active user.

  if you remember, I mentioned earlier as my assumption like 100 million daily active user.

So from now, we will move on our design decision consideration because we already covered most of its clarification part. So let's move to our system design high level blocks.



At initial stage this looks fine to me. But at this point may your interviewer interrupt you and asked, how you will handle synchronization among different user device login and their content sync. Why they will bring this aspect, because this is one of its key requirement of your system.