Chapter Application Layer

- It works for the user, it needs support protocols, DNS (domain name system) which handles naming within the internet, electronic mail & www comes under this layer.

DNS
→ IP addresses are hard to remember.
→ ASCII names were introduced to decouple machine names from machine addresses.
→ Networks itself understand only numeral addresses therefore ASCII to numerical conversion is required.
→ At the time of ARPANET, there was a file host.txt that listed all the host & IP addresses.
→ When thousands of mini computers & personal computers connected to net, this approach did not work.
→ To solve this problem domain name system was invented. In this system hierarchical domain based naming scheme & it distributed database system for implementing this naming scheme it was invented.
→ To map a name to an IP address, an application program calls a library procedure called a resolver passing name as a parameter.
→ Resolver sends a packet to local DNS server then DNS server looks up the name and return the

IP address to resolver. Resolver sends it to the caller. ~~Asset~~ And then the TCP server is ~~connected~~ established.

→ Internet is divided into over 200 top level domains where each domain covers many hosts. Each domain is partitioned into sub domains & so on.

→ Top level domains are of 2 types generic and countries.

→ Generic - eg: com, edu, gov, int (international), mil, net, org (not non profit organisation like NGO's), biz, info, name, pro, professionals), coop

→ country domains include one entry for every country.

→ for getting a domain, it requires going to a registrar for corresponding top level domain to check if the desired name is available and nobody else has a trade mark for it.

→ Domain names can be _absolute_ or _relative_. _Absolute_ domain name always ends with a dot.

→ Domain names are case sensitive. Full path names must not exceed 255 characters.

WWW

→ It is an architectural framework for accessing linked documents spread out over millions of machines all over the internet.

→ In 1994, an agreement was signed b/w ~~&~~ cern and ~~and~~ MIT sign an agreement setting up the www consortium which is the organisation devoted to further development of web standardizing protocols & ~~int~~ encourage inter-operability of size.

→ The web consist of a vast collection of documents called web pages. Each page contain a link to other pages this link is called hypertext. The pages

are viewed with a program called browser. The browser fetches the page requested, interprets the text and formating commands on it, then displays the page on the screen.
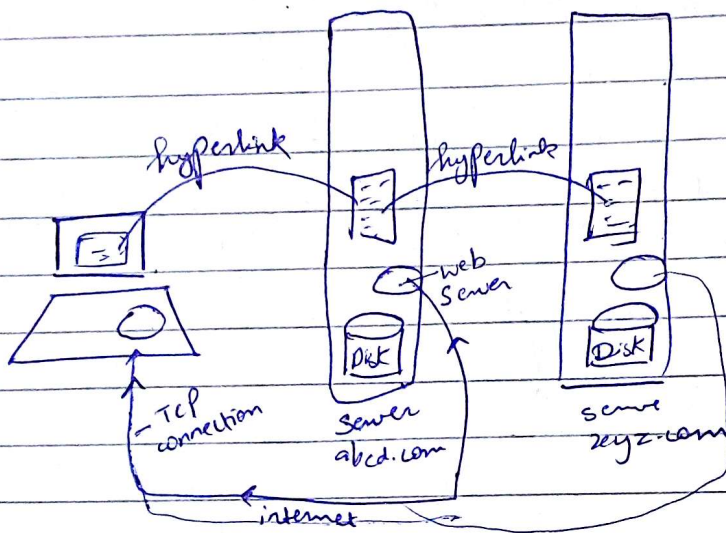
There are some steps followed at client & server side.

Client side

1) The browser determines the URL
2) The browser asked DNS for IP address.
3) DNS replies with IP address
4) The browser makes a TCP connection to port 80 on that IP address
5) It then sends over a request asking for a particular file.
6) The server sends the requested file.
7) TCP connection is released.
8) Browser displays all the text and then fetches & displays all images.
Not all pages contain the html code. a page may contain formatted document in pdf or icon in

gif to format or photograph in jpeg, song in mp3, video in mpeg etc.

→ The browser has a problem when it encounters a a page that it cannot interpret.

→ The solution to the above problem is when a server returns a page, it also return some additional info about the page, this includes the mime type of the page (multiple interfaces multimedia extension).

→ The 2 possibilities are pluggin & helper application.

→ A pluggin is a code module that & browser fetches from a special directory it from a disk & & install it & as an extension.

→ The other way is to use helper application which is a complete program running as a separate process it offers no interface to the browser & do not make use of browser services. It just accepts the name of the file where the content has been stored, opens the file & displays the contents.

Server side

1) ~~&~~ Accept a TCP connection from a client.
2) Get the name of the file requested
3) Get the file from the disk.
4) Return the file to the client. ~~release the TCP~~
5) Release the TCP connection.

The steps performed by modern web servers

1) Resolve the name of the web page requested
2) Authenticate the client (verifying client side identity)
3) Perform access control on the client.
4) Perform access control on the web page.

5) Check the cache.

6) Fetch the requested page from disk

7) Determine the mime type from file extension.

8) Take care of any odds & even

9) Return back reply to client

10) May can entry into server log.

→ 11) If many request come, CPU cannot handle. The solution for this is the use of <u>CPU server farm model</u>.

→ In this we add more nodes with the replicated disks the front end accepts all incoming requests and splits it into multiple CPU's to reduce the load on each computer.

→ Problem with the server farm is that there is no longer a shared cache because each processing node has its own memory.

→ The solution is frontend to keeps track of each request and sends the subsequent request for the same page to same node.

URL

→ Web pages point to other web pages. How these pointers are implemented is a big issue. Web pages require few mechanisms one for naming & second for locating.

→ If every page has a unique name then there is no ambiguity.

→ The solution to this problem is each page should be assigned a URL which has 3 parts 1) protocol 2) DNS, 3) local name.