

Sanchit Sharma.

Internet Technology Final Notes. Typed for Revision.

## **Basic :**

Q : What is a network?

A : A network is a carrier of information between two or more devices. Mediums of communication are Wireless, Wired, Bluetooth....

Q : What is a single link network?

A : A single link network is a network between two devices on one link. Bits of data are sent in frames and/ or packets to one another.

Q : What is a single link multiple access network?

A : A single link multiple access network is a network on one link which has more than 2 devices connected to it. Bits of data are sent in frames/packets to devices. These frames/packets have a header which specifies the destination address. Since there are many receivers, they are differentiated through the use of MAC address.

Q : How to connect multiple links?

A : To connect networks of multiple links we use a router. This router then has the job to figure out how packets/frames will be routed from the source to the destination.

Q : Advantages of Networks ?

A :

- Availability of Resources
- High Reliability
- Load Sharing between machines
- Human to Human communication

Q : What are the types of ISPs?

A : There are 3 types of ISPs, represented by Tiers.

- Tier 3 : Local ISPs
- Tier 2 : Regional ISPs
- Tier 1 : Global ISPs

Tier 3 is a customer of Tier 2 ISP, Tier 2 is a customer of Tier 1 ISP.

Q : What are peering ISPs?

A : Mutual relationship between ISPs about forwarding traffic of each others customers. No money involved in this relationship.

Q : What are transit ISPs?

A : ISPs which provide access to any reachable customer. Money is involved.

Q : What are components of the Internet?

A :

- Link : Communication links for transmission of data.
- Host : Computer running application of end user.
- Router : Computer for routing packets from input line to output line.
- Gateway : A device directly connected to two or more possibly different networks.
- Network : Group of link, hosts and routers capable of sending data to members of the network.

## **DATA TRANSMISSION :**

Q : What are the types of Switching Schemes?

A :

1. Circuit Switching
2. Message Switching
3. Packet Switching

Q : Explain Circuit Switching.

A : Circuit Switching provides service by setting up the total path of connecting lines from the origin to the destination.

The protocol is as follows : (ANALOGY TELEPHONE SERVICE)

1. Origin sets up path by sending control messages to the destination.
2. Return signal from destination informs origin to begin transmission.
3. Data transmission begins.
4. Once a path is selected, it remains allocated for the transmission (Used or Not)
5. When transmission is finished, origin releases the circuit.

Control message sent from origin to router, to router, until destination reached, destination returns signal back to origin. There is a propagation delay at each router for control messages. Data transmission begins, and circuit is closed after finishing.

Q : Explain Message Switching.

A : Message Switching provides service by setting up path from origin to router to destination. The total path is unknown, and the route is decided at each intermediary router.

The protocol is as follows : (ANALOGY POSTAL SERVICE)

1. Each Message contains a header which contains information about the message.
2. Header includes the destination address for the message.
3. When the entire message is received at a router, the next step of its journey is selected. If the selected channel is busy at the time, the message is put on a queue and waits until the channel is free.
4. The messages 'hop' from node to node, allocating one channel at a time, until it reaches the destination.

Each message sent from a node contains a propagation delay, also could contain a queue delay.

Q : Explain Packet Switching.

A : Packet switching provides service similar to message switching. Packets have a destination address from the origin to destination. The total path of the packet is unknown, and the path is selected at intermediary router nodes.

The protocol is as follows :

1. Messages are divided into pieces called packets.
2. Each packet contains a header which contains information about the packet.
3. Each packet is numbered and addressed, and sent to the destination one at a time.
4. Each packets next step in the journey is determined by intermediary router nodes, until reaching the destination.

Packet switching is very similar to message switching. One advantage it creates is the ability of pipelining.

Q : Compare Switching Techniques

A :

1. Header overhead : Packet > Message > Circuit
2. Transmission Delay:
  - Short Bursty Data : Circuit > Message > Packet
  - Long Continuous Data : Packet > Message > Circuit

## **Internet Architecture :**

Q : Why use Protocols?

A : Protocols are the building blocks of network architecture. Each protocol object has two interfaces :

1. Server : Operations on this protocol.
2. Peer-To-Peer : Messages exchanged with peer.

The term “Protocol” refers to both the specification and implementation of a given module.

Q : What are the Internet Layers?

A:

1. Physical Layer
2. Data Link Layer
3. Network Layer
4. Transport Layer
5. Application Layer

Q : Give a functional overview of all Internet layers.

A :

1. Physical Layer :
  - Transmission of a raw bit stream.
  - Forms a physical interface between devices on the network.
2. Data Link Layer :
  - Provides reliable transmission of data between two adjacent nodes.
  - Creates packets/frames from bits and vice-versa.
  - Provides flow control.
  - Provides frame level error control: Provides network layer with error-free links for frames.
3. Network Layer :
  - Responsible for routing decisions.
    - Fixed Routing or Dynamic Routing.
  - Provides congestion control.
4. Transport Layer :
  - Hide the details of the network from the session layer.
  - Provides reliable end-to-end communication.
  - Performs end-to-end flow control.
  - Performs retransmission of packets lost in the network.
5. Application Layer :
  - Application layer protocols are application dependent.
  - Implements communication between two applications of the same type.
  - Provides session and presentation support.

Q : Why is there layering?

A : Because the network architecture is very complex, layering makes it simple and easy to understand. Testing and maintenance on layers is also simplified, if there is a need to update or change a layer then that is also easier to do.

## Measuring Network Performance :

### Definitions:

1. Packet Length : Size of a packet. (Bytes or Bits)
2. Channel Speed or Bandwidth : How fast the channel can transmit. (B/sec or b/sec)
3. Packet Transmission Time : Amount of time taken to transmit one entire packet. (Seconds)
4. Propagation Delay : Delay imposed by properties of the link. (Seconds)
5. Total Transfer Time : (Propagation Delay ) + (Packet Transmission Time)

### Units:

- |                                |                                     |
|--------------------------------|-------------------------------------|
| 1. 1 Kb = $1 \times 10^3$ bits | 1 msec = $1 \times 10^{-3}$ seconds |
| 2. 1 Mb = $1 \times 10^6$ bits | 1 usec = $1 \times 10^{-6}$ seconds |
| 3. 1 Gb = $1 \times 10^9$ bits | 1 nsec = $1 \times 10^{-9}$ seconds |

## Application Layer :

Q : What does the application layer protocol define?

A :

1. Types of Messages exchanged:
  - Either Request or Response.
2. A Message format:
  - Syntax: The types of fields in messages and what those field delineate
  - Semantics: The meaning of information in fields.
3. Rules for when and how processes send and respond to messages.

Q : What is needed for one host to communicate with another host.

A : IP Address and Port # of the host.

- A Non-Network application is an application which runs on a single host.

- Socket is the door between the network and application.

-A network is a 4-tuple:  **$IP^{Source}, Port \#^{Source}, IP^{Destination}, Port \#^{Destination}$**

Q : Explain the Client-Server architecture in Application Layer

A :

- Server:
  1. Always on host.
  2. Permanent IP Address.
  3. Farms for Scaling.
- Client:
  1. Only communicates with the server, not with other clients.
  2. Client may run on Dynamic IP Addresses.
  3. Clients may be intermittently connected to the server.

## **DNS: Domain Name Server**

Q : What is the need for DNS service?

A : IP Addresses are hard to remember for the human brain, thus to compensate we replace them with alphanumeric host names. DNS contains Address Resolution, a service which maps alphanumeric host names to their binary IP Addresses.

Q : Explain Simple DNS and Why it is not used.

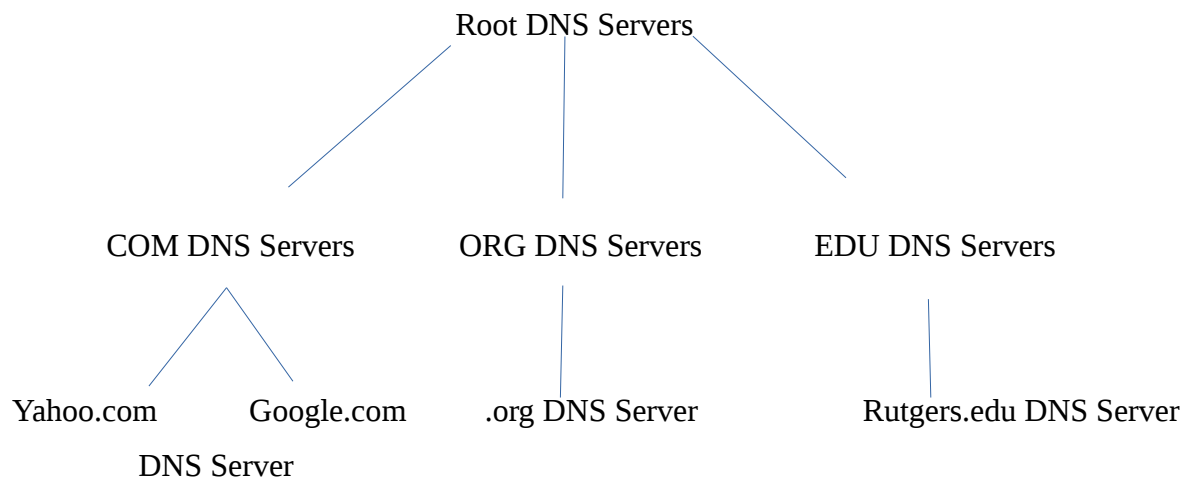
A : Simple DNS is a very simplified technique of implementing DNS. The DNS server contains an Address Resolution table which consists of mapping between alphanumeric hostnames and their IP Addresses. The reasons simple DNS fails is because the Address Resolution table would need to update the table for every host. Thus the DNS server would not SCALE with high volume.

Q : Explain Centralized DNS and Why it is not used.

A : Centralized DNS is when there is a single point of failure. This DNS has a distant database which reduces maintenance capabilities. There is also high traffic volume and high maintenance on the server. Just like simple DNS, centralized DNS does not SCALE.

Q : Explain what kind of DNS is used for the Internet and Why.

A : DNS we use today is Distributed, Hierarchical DNS. This DNS is Scalable with high traffic.



Q : Generally explain the DNS Protocol.

A :

- Communication between the client and server.
- Client connects to port 53.
- IP Address of DNS server should be known before hand
  - Either Automatically or Manually.
- Two types of messages exchanged
  - Either Query or Response
- Two types of Query messages.
  - Both types have a common message format that follows the header.
  - Standard Query (0x0): Request IP Address of a given hostname.
  - Update Query (0x5): Provide a binding of IP Address to a hostname.



Q : Describe the process when a Client wants to know an IP Address of a given hostname.

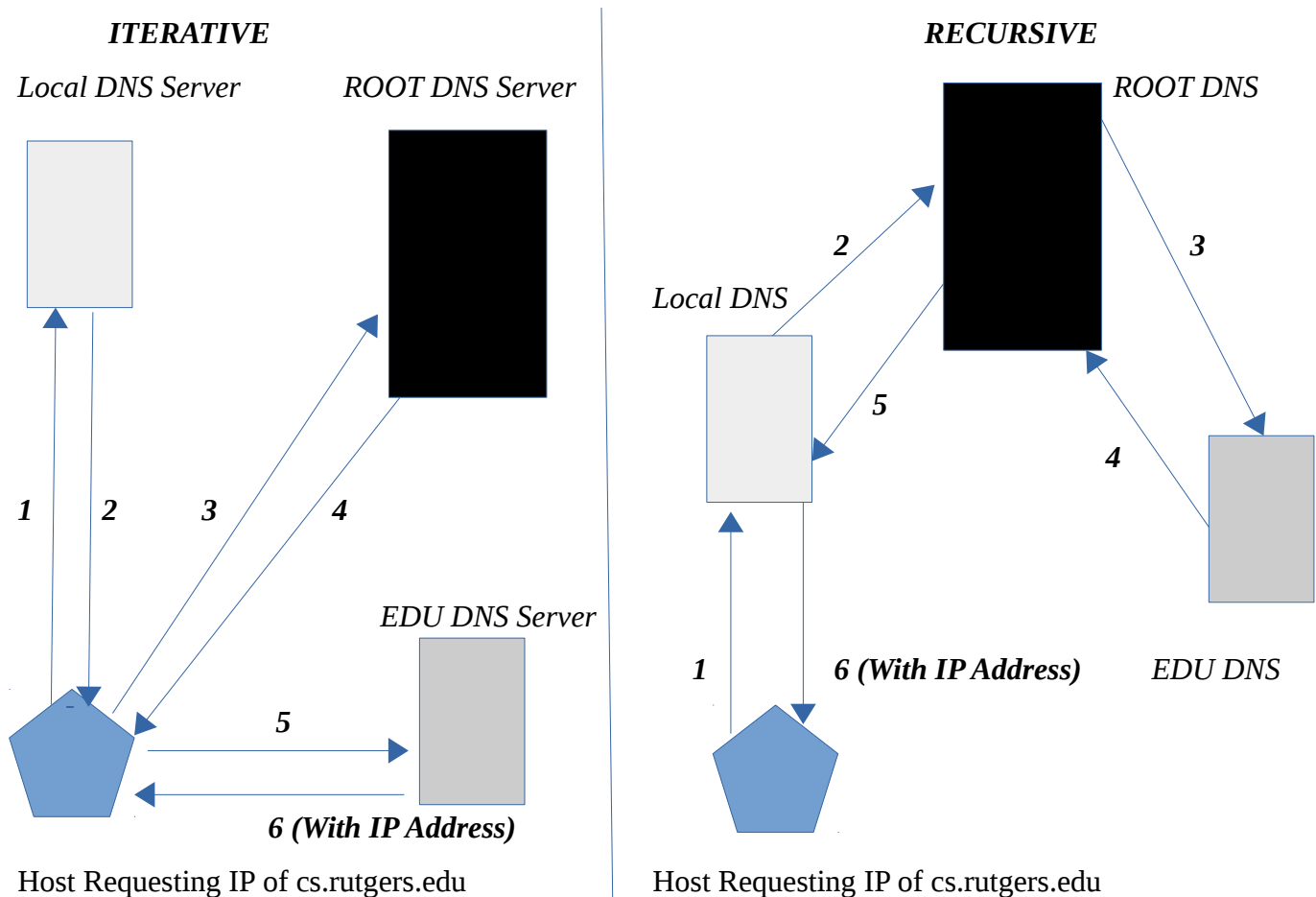
A :

1. The client will first send a standard query to its primary DNS name server, with QR = 0.
2. If the DNS name server contains the mapping, it will return back the IP Address to Client.
3. If it does not, DNS name server will forward the request to the root DNS Server.
4. Root DNS Server will then send the request down the tree to the host, until a DNS name server is found which contains the mapping.

Q : What are the different methods for handling a Query.

A :

- Iterative Query: If a server does not contain the mapping, it will return the name of a DNS server which should be contacted.
- Recursive Query: Recursively send request from contacted DNS server to another until the mapping is found. Each server has a burden of Address Resolution.



Q : What is DNS Caching?

A : Once a DNS server learns a new mapping, it caches the mapping.

- Cache entries timeout after some time period.
- Top Level Domain Server typically cached in lower name servers to reduce the load on Root Servers.

-Distributed databases store Resource Records.

***RR : {Name, Type, Class, TTL, ADDR}***

Q : What are types of DNS records, Specify and Describe.

A : There are 5 types of DNS records.

- A: Name – Hostname  
Value – IPv4 Address
- AAAA: Name - Hostname  
Value – IPv6 Address
- NS: Name – Domain  
Value – Hostname of Authoritative Name Server for the given domain.
- MX: Name – Domain  
Value – Mail Server for it.
- CName: Name – Domain  
Value – Canonical Hostname

Q : Generally describe DNS services.

A :

1. Hostname to IP Address Resolution
2. Host Aliasing (Canonical and Alias Names)
3. Load Sharing (Replicated Web Servers, set of IP Addresses for one Canonical Hostname)
4. Mail Server Aliasing

Q : How does Host contact the Name server with only the Name and no IP Address

A : IP Address of atleast 1 Name server should be configured into computer manually. Can also DHCP.

Q : What is the general THEME of application level protocols?

A :

- Request/Response message exchanges.
- How messages are structured
  - HTTP, SMTP, FTP – Simple ASCII Protocols.
- Caching
- Name Lookup
  - Division of concerns. (zones)
  - Hierarchy Structure.

## HTTP: HyperText Transfer Protocol

**Jargon:**

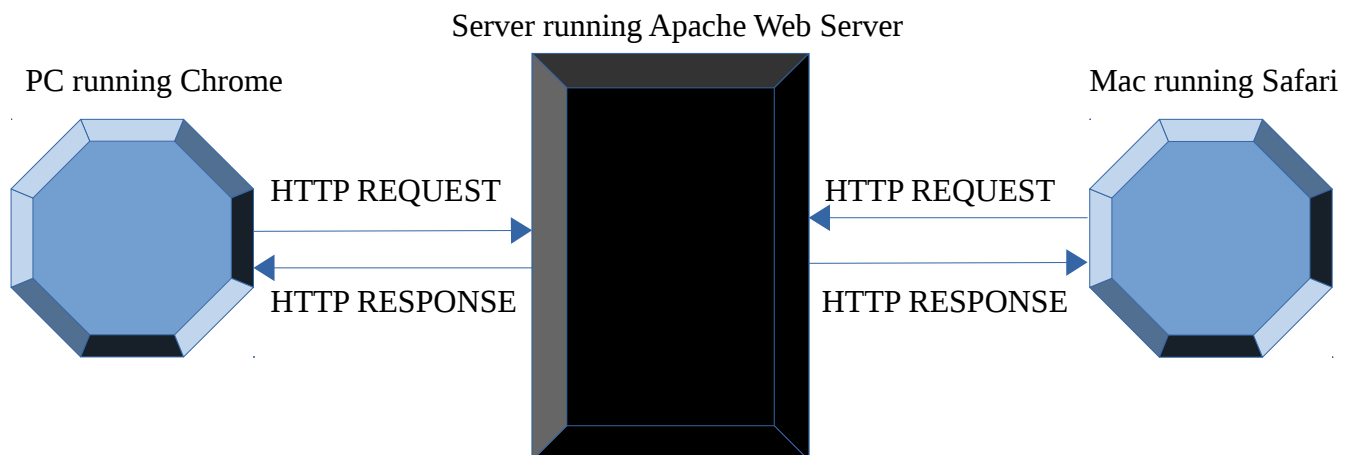
- Webpages consists of objects.
- Objects can be HTML File, JPEG Image, Java Applet, Audio File...
- Webpages consists of base HTML-File which includes several references objects.
- Each object is addressable by a URL.
  - Example : [www.sanchitsharma.com/sanchit.jpg](http://www.sanchitsharma.com/sanchit.jpg)

**HTTP Overview:**

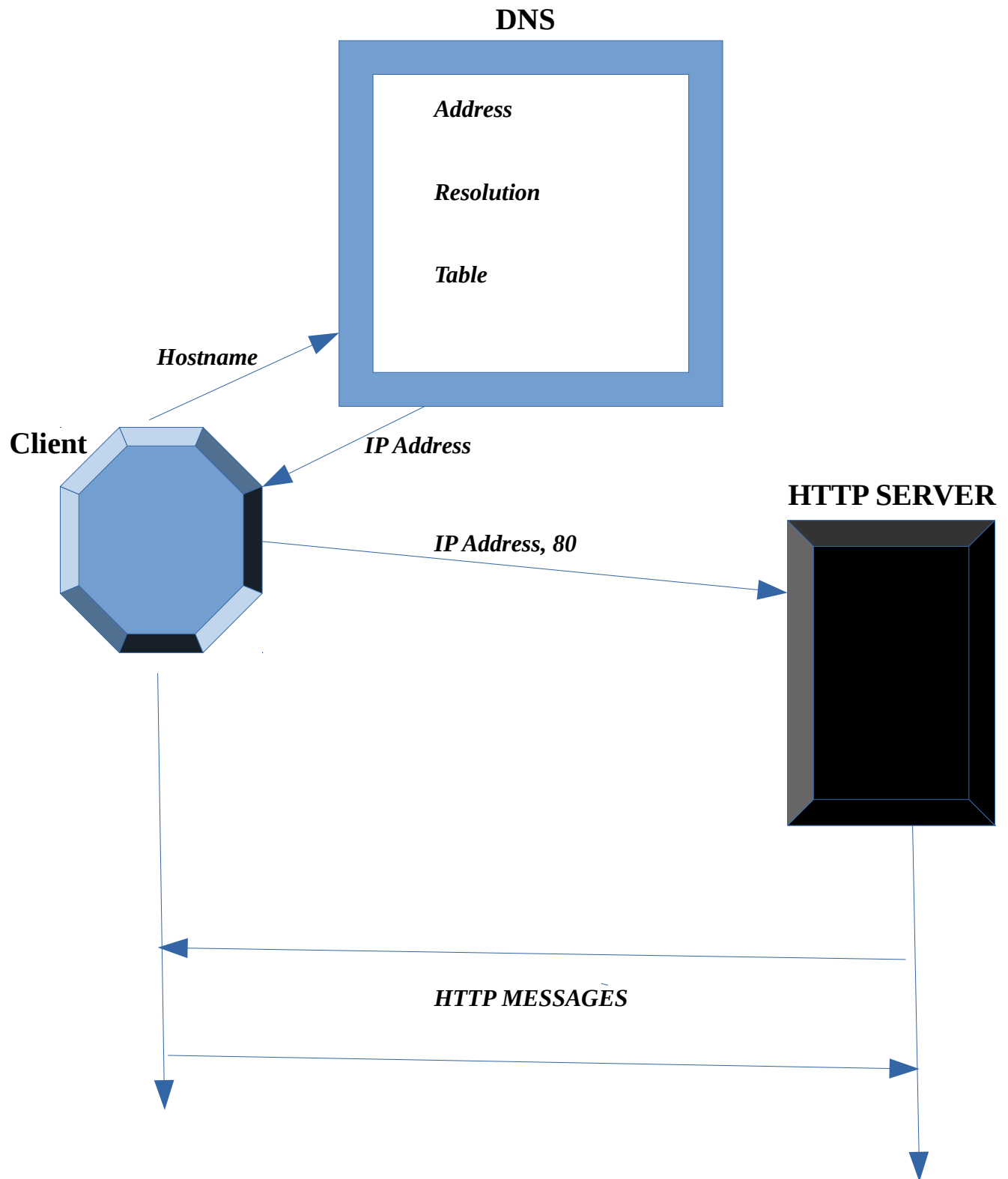
Q : What is the client-server architecture in HTTP?

A :

- Client: Browser that requests, receives, and 'displays' web objects.
- Server: Web server sends objects in response to requests.



## *Client-Server Connection in HTTP:*



## HTTP MESSAGES:

Q : Describe the format of HTTP Request Messages.

A :

- ASCII – Human Readable Format

GET /someurl/feels.jpeg HTTP/1.1

Host : www.sanchitsharma.com

User-Agent : Chrome/11.0

Connection : Close

Accept-Language : En Fr Hi

Data Data Data

*Request Line*

*Header Line*

*Header Line*

*Header Line*

*Header Line*

*Carriage Return Line Feed*

*Main Data*

- General Format (‘\’ indicate Carriage Return Line Feed)

Method            URL            Version \

Header Field Name : Value \

Header Field Name : Value \

\

Entity Body

*Request Line*

*Header Line*

*Header Line*

*Carriage Return Line Feed*

*Main Data*

Q : What are the HTTP Request Methods?

A :

- GET: Get the file specified in the path URL field in entity body.
- PUT: Uploads file in entity body to path specified in URL field.
- POST: Accept the entity enclosed in entity body as a new subordinate of the resource identified by URL field.
- DELETE: Delete file specified in the URL field
- HEAD: Asks server to leave requested objects out of response.

Q : How to Upload Form Input (Eg : Account Sign up Form)

A :

- Post Method: Webpages can include form input, this input is uploaded to server in Entity Body.
- URL Method: Uses Get method, Input is uploaded in URL field of Request line.

Q : Describe the format of HTTP Response Messages.

A :

- ASCII – Human Readable Format

HTTP/1.1	200	OK	<i>Status Line</i>
Connection:	Close		<i>Header Line</i>
Date :	Today		<i>Header Line</i>
Last-Modified:	Yesterday		<i>Header Line</i>
Content-Type:	Food and Images		<i>Header Line</i>
			<i>Carriage Return Line Feed</i>
Data Data Data Data			<i>Main Data</i>

- General Format (‘\’ indicate Carriage Return Line Feed)

Version	Status Code	Status Phrase \	<i>Status Line</i>
Header Field : Value \			<i>Header Line</i>
Header Field : Value \			<i>Header Line</i>
Header Field : Value \			<i>Header Line</i>
\			<i>Carriage Return Line Feed</i>
Entity Body			<i>Main Data</i>

Q : What are some HTTP Response Status Codes?

A :

- 200 OK : Request Succeeded, requested object in Entity Body
- 301 Moved Permanently : Requested object moved, new location in Entity Body
- 400 Bad Request : Request Message not understood by Server
- 404 Not found : Requested document was not found on Server.
- 505 HTTP Version not supported.

## **HTTP Connections:**

Q : What are the two types of HTTP Connections?

A : Two types of HTTP connections are

- Non-Persistent: At most one object is sent over a TCP connection. Used by HTTP/1.0
- Persistent: Multiple objects can be sent over a single TCP connection. HTTP/1.1 Default

Q : Describe process of non-persistent HTTP connection, when user enters a URL with 10 images.

A :

1. HTTP Client initiates a TCP connection to HTTP Server.
2. HTTP Server of host “accepts” connection, notifying client.
3. HTTP Client sends HTTP request message.
4. HTTP Server receives request message, replies with HTTP response message containing requested object.
5. HTTP Server closes TCP connection.
6. HTTP Client receives response message containing HTML File. Client parses this file and finds 10 referenced JPEG Objects.
7. STEPS 1-5 Are repeated for each JPEG Object.

## ***HTTP Response Times:***

Q : What is the definition of RTT?

A : Full-form of RTT is Round Trip Time. It is the time taken to send a small packet from Client to Server, and back to Client.

Q : What is the Response Time for Non-Persistent HTTP Connection?

A :

- One RTT to initiate TCP connection between client and server.
- One RTT for HTTP Request message from client to server and HTTP Response message from server to client.
- File Transmission Time.
- **Total Response Time = (2 \* RTT) + (File Transmission Time)**

Q : What are issues of using Non-Persistent HTTP Connection? Why should you use one connection over another?

A :

- Using Non-Persistent connection is inefficient as it requires 2 RTTs per object. A solution to this is for the browser to open parallel TCP connections to fetch referenced objects.
- Persistent connections have the ability to GET multiple objects with one TCP connection. Server leaves the TCP connection open after sending response. Subsequent message exchanges between client and server are sent over this open TCP connection.

### HTTP User-Server State:

Q : What does a 'Stateless' Server mean?

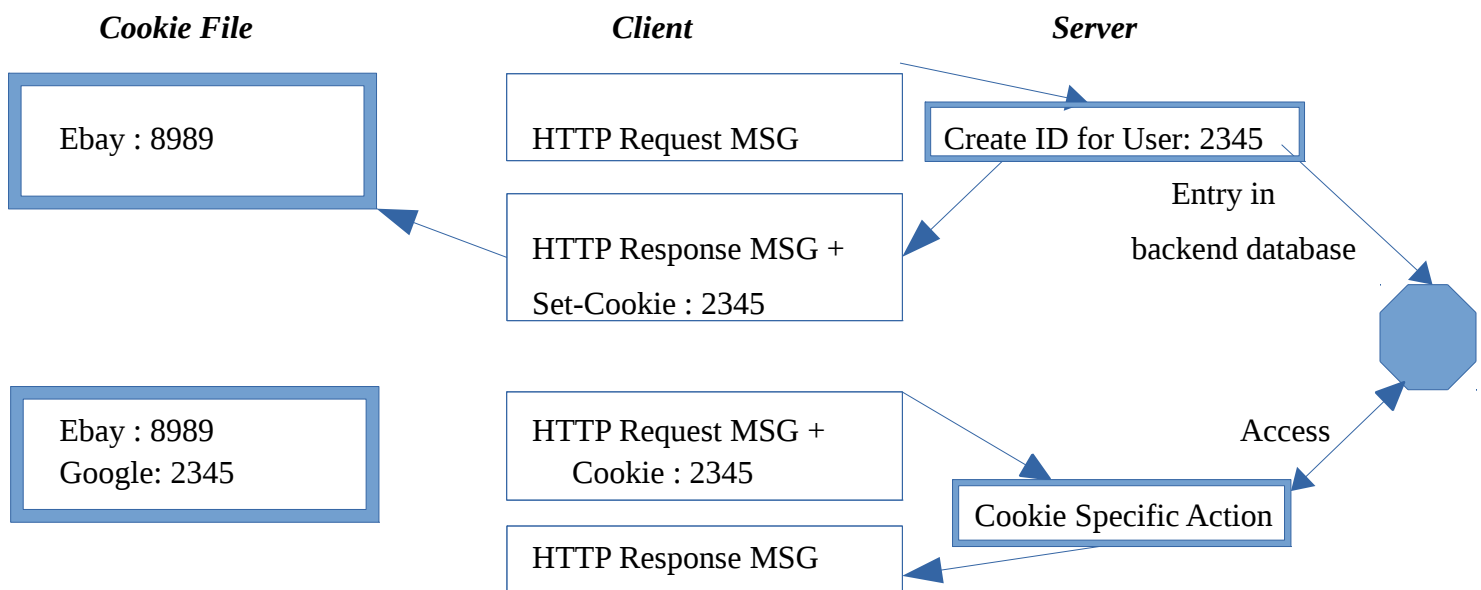
A : This means that the server does not maintain information about past client request.

Q : What are the advantages of implementing 'state'?

A :

1. Authorization
2. Shopping Carts
3. Recommendations
4. User Session state

- Cookies can be used to implement State in HTTP.





Q : What are the components of Cookies?

A :

1. Cookie header line in HTTP Request Message.
2. Cookie header line of HTTP Response Message.
3. Cookie File kept on user's host, managed by user's browser.
4. Back-end database at Website.

Q : Why use Web caches (Proxy Servers)?

A :

- Reduce response time for clients request.
- Reduce traffic on an institutions access link.

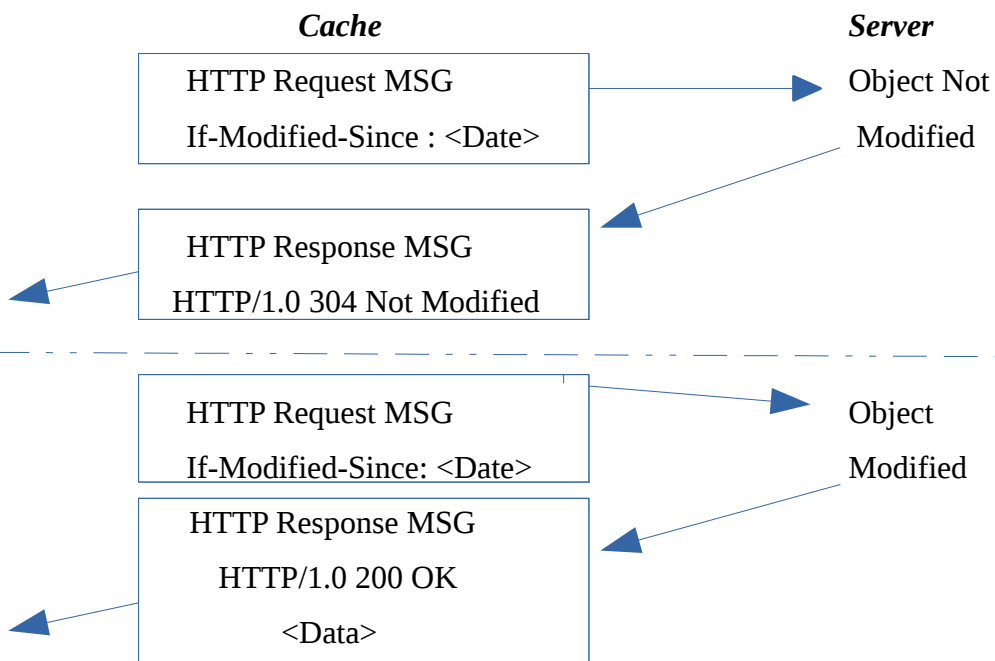
Q : What is the browsers role in Web caching?

A : Browser sends all HTTP request to cache.

- Hit: Cache returns object.
- Miss: Cache requests object from origin server, then returns object to client.

Q : What is 'conditional get' in Web caching?

A : Conditional get guarantees cache content is up-to-date and also saves traffic and response time whenever possible.



## **CDN: Content Distribution Networks**

### **Definitions:**

- Origin Server: Server that holds the authoritative copy of the content.
- CDN Server: Replica server owned by CDN provider.
- CDN Name Server: DNS like name server used for redirection.

Q : What is the need for CDNs?

A :

1. Reduce Bandwidth requirements of content provider.
2. Reduce cost of maintaining servers.
3. Cache for server content.
4. Reduce traffic on the link of content provider.
5. Improve response time to user.

## **FTP: File Transfer Protocol**