Communication between processes

Client: Software that accesses the services made available by a server

Server: Running instance of an application which can accept requests and give responses accordingly

web browser is a client process. It retrieves, searches and displays information

web server is a server process. It processes requests and gives responses. It uses HTTP (Hypertext Transfer Protocol) to process requests and to distribute information

The peer that is *downloading* content is the **Client**

The peer that is *uploading* content is the **Server**

Communication between application layer protocols and transport layer protocols such as TCP and UDP can be managed using Ports and Sockets

TCP and UDP are preformed in IP (Internet Protocol)

Port: TCP or UDP connection point

Socket: endpoint or connection obtained by pairing the IP address and a Port

For host-to-host communication, you need:  
1. IP address of destination host  
2. Port number of destination host

UDP (User Data Protocol): used to perform the transaction from a remote client to a server

Reasons to use UDP:  
1. Completes transaction in one round trip time  
2. Requires less time to compete the transaction between client and server  
3. Usually reliable enough  
4. Request is sent by client to a UDP socket, server generates response and sends it back to client’s UDP socket

Reasons not to use TCP:  
1. TCP is connection oriented and requires connection between two hosts in a network  
2. Two round trip times, one for connection setup and other is for client to send request and server to give response

Other protocols require application data to be received, TCP is more reliable because it is connection-oriented network where packet transmission is guaranteed

UDP sends only datagram and does not manage retransmission, data sequencing, or connection

Application data will be received without any gaps and in the correct order in TCP, maybe not in UDP

TCP data transmission is accurate

Loss of data in HTTP, SMTP, FTP and POP3 is unacceptable using UDP, therefore they use TCP

1. When client visits e-commerce website, it will return a cookie number  
2. Cookie number is stored on user’s host managed by browser. Username and password is remembered  
3. Cookie number is present in cookie header and generated by server  
4. Client receives response along with header with server name and user ID number  
5. Client sends request that includes cookie header with header line that specifies identification number for server  
6. website finds user by sending user’s cookie number

Web caching: temporary storage of documents such as images, video files, HTML pages to reduce bandwidth usage

brings desired content close to user on same LAN as host

Web cache: Keeps copies of requested objects of client in storage

web caching reduces the delay in receiving requested objects

BitTorrent identifies top four providers that are sending data at highest rate. Each peer gives priority to its top four peers or providers and doesn’t send data to remaining peers. If peer finds better partner, then it modifies the set of top four peers

Client-server: Server is always-on host to manage network resources and client receives service request from hosts to run applications

P2P: Peers act as both client and server.

Instant messaging with centralized index is a hybrid

In HTTP, client must send one request per object. Client sends 4 requests and will receive 4 responses. non-persistent HTTP 4 requests will be sent over 4 TCP connection. Persistent HTTP 4 requests sent over one TCP connection. Must request files on same server for persistent HTTP.

header line in HTTP request message indicates time and date when object in response was created and sent by the server. Header doesn’t contain any details about object itself

Error HTTP messages (404) send with an empty body

Web document can be retrieved by HTTP client at given location or URL. Initially, HTTP server doesn’t know the IP address. UDP, DNS, TCP, HTTP needed for transport layer (all needed except HTTP for retrieval of web document). DNS and HTTP needed for application layer

If cache of DNS is accessible, the most popular web server can be identified by counting the number of requests for a web server among the snapshots of DNS cache taken periodically.

Getting complete file as free-rider can be achieved by not uploading any data to other peers. Also 0