CHAPTER - 4

REFERENCE MODEL

PROTOCOL LAYER:

The communication between the nodes in a packet data network must be precisely defined to ensure correct interpretation of the packets by the receiving intermediate and the end systems. The packets exchanged between nodes are defined by a protocol or communications language.

There are many functions which may be need to be performed by a protocol. These range from the specification of connectors, addresses of the communications nodes, identification of interfaces, options, flow control, reliability, error reporting, synchronization, etc. In practice there are so many different functions, that a set (also known as suite or stack) of protocols are usually defined. Each protocol in the suite handles one specific aspect of the communication.

The protocols are usually structured together to form a layered design (also known as a "protocol stack"). All major telecommunication network architectures currently used or being developed use layered protocol architectures. The precise functions in each layer vary. In each case, however, there is a distinction between the functions of the lower (network) layers, which are primarily designed to provide a connection or path between users to hide details of underlying communications facilities, and the upper (or higher) layers, which ensure data exchanged are in correct and understandable form. The upper layers are sometimes known as "middleware" because they provide software in the computer which convert data between what the applications programs expect, and what the network can transport. The transport layer provides the connection between the upper (applications-oriented) layers and the lower (or network-oriented) layers.

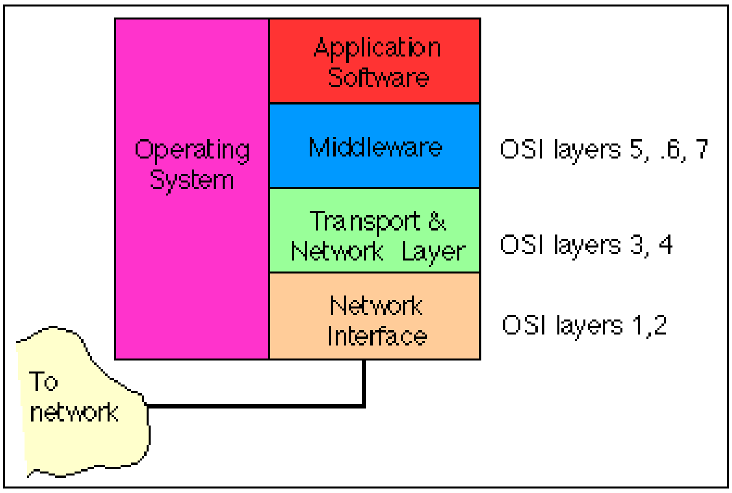


Fig: Overview of layering of protocol functions

The basic idea of a layered architecture is to divide the design into small pieces. Each layer adds to the services provided by the lower layers in such a manner that the highest layer is provided a full set of services to manage communications and run distributed applications. A basic principle is to ensure independence of layers by defining services provided by each layer to the next higher layer without defining how the services are to be performed. This permits changes in a layer without affecting other layers. Prior to the use of layered protocol architectures, simple changes such as adding one terminal type to the list of those supported by an architecture often required changes to essentially all communications software at a site.

Protocol Stacks:

The protocol stacks were once defined using proprietary documentation each manufacturer wrote a comprehensive document describing the protocol. This approach was appropriate when the cost of computers was very high and communications software was "cheap" in comparison. Once computers became readily available at economic prices, users saw the need to interconnect the computers from different manufacturers using computer networks. It was costly to connect computers with different proprietary protocols, since for each pair of protocols a separate "gateway" product had to be developed. This process was made more complicated in some cases, since variants of the protocol existed and not all variants were defined by published documents.

The Open Systems Interconnection (OSI) architecture has been developed by the International Organization for Standardization (ISO) to describe the operation and design of layered protocol architectures. This forms a valuable reference model and defines much of the language used in data communications.

An example protocol stack and the corresponding layers:

Protocol	Layer
HTTP	Application
TCP	Transport
IP	Network/Internet
Ethernet	Data Link/ Link
IEEE 802.3u	Physical

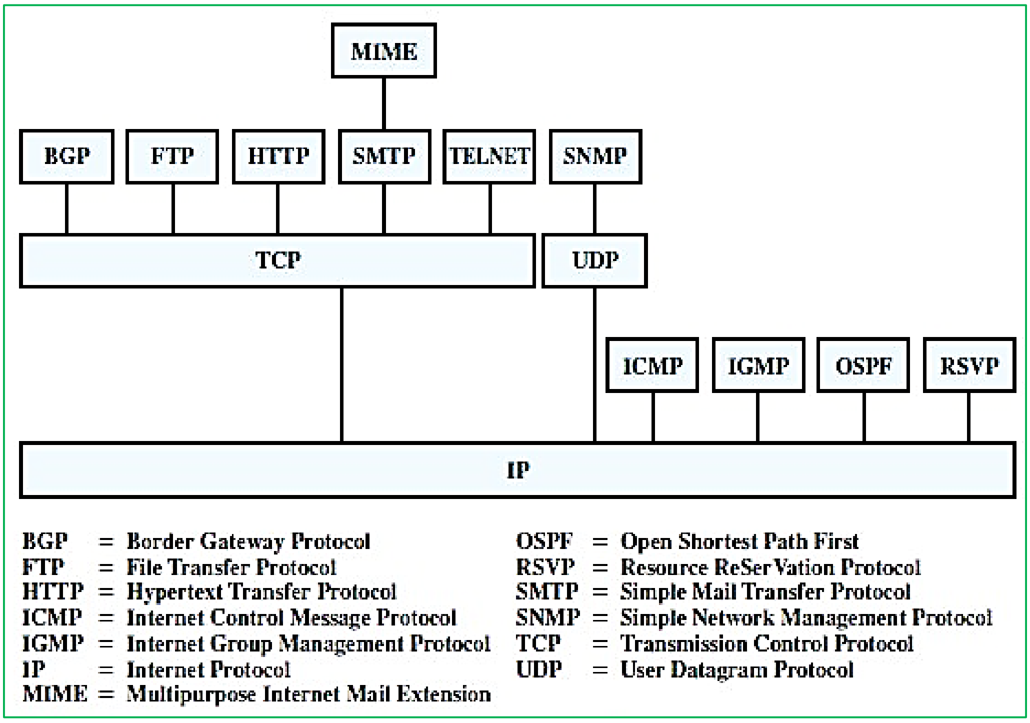

MIME
BGP
FTP
HTTP
SMTP
TELNET
SNMP
TCP
UDP
ICMP
IGMP
OSPF
RSVP
IP
BGP = Border Gateway Protocol
FTP = File Transfer Protocol
HTTP = Hypertext Transfer Protocol
ICMP = Internet Control Message Protocol
IGMP = Internet Group Management Protocol
IP = Internet Protocol
MIME = Multipurpose Internet Mail Extension
OSPF = Open Shortest Path First
RSVP = Resource ReSerVation Protocol
SMTP = Simple Mail Transfer Protocol
SNMP = Simple Network Management Protocol
TCP = Transmission Control Protocol
UDP = User Datagram Protocol

Fig: Some Protocols in the TCP/IP Protocol Suite

Network Entities and Layers:

Layer 7: Application Layer

- Protocols: HTTP, FTP, SMTP, POP, DNS, DHCP, Telnet, etc.
- Devices: Hosts, PC, Servers, Mobile Phones, etc.

Layer 6: Presentation Layer

- Protocols: Apple Filing Protocol (AFP), Independent Computing Architecture (ICA), the Citrix system core protocol, Lightweight Presentation Protocol (LPP), NetWare Core Protocol (NCP), Network Data Representation (NDR), Telnet (a remote terminal access protocol)
- Graphics: TIFF, JPEG, GIF, etc
- Text: ASCII, EBCDIC, Unicode, Encrypted
- Audio: Midi, MPEG, WAV, MP3
- Video: Quicktime, AVI

Layer 5: Session Layer

- Protocols: NFS - Network File System, RPC - Remote Procedure Call, ASP - Appletalk Session Protocol, PPTP
- Devices: Gateways, Firewalls, PC's

Layer 4: Transport Layer

- Protocols: Connection Oriented (TCP), Connectionless (UDP)
- Devices: Gateways, Firewalls.

Layer 3: Network Layer

- Protocols: VLAN, IP (IPv4, IPv6), ICMP, IPSec, OSPF, etc.
- Devices: Routers

Layer 2: Data Link Layer

- Protocols: MAC, LLC, ARP, SLIP, Ethernet-Frame relay, PPP
- Devices : Bridges, Switches, NIC (Layers 1 & 2)

Layer 1: Physical Layer

- Protocols: EIA RS-232, DSL, 802.11a/b/g/n PHY, Ethernet, SONET/SDH
- Devices: Hubs, Repeater, NIC (Layers 1 & 2), IrDA
- Media: Coax, Fiber, Twisted Pair, Wireless

OSI Reference Model:

OSI model is based on the proposal developed by the International Standards Organization (ISO). This model is called ISO OSI (Open Systems Interconnection) Reference model because it deals with connecting open systems (systems that are open for communication with other systems). We call it as OSI Model. **Open Systems Interconnection (OSI)** model is developed by ISO (International organization for standardization) in **1984.**

OSI reference model is a logical framework for standards for the network communication. OSI reference model is now considered as a primary standard for internetworking and inter computing. Today many network communication protocols are based on the standards of OSI model. In the OSI model the network/data communication is defined into seven layers.

A layer should be created where different level of abstraction is needed. Each layer should perform a well-defined function. The function of each layer should be chosen according to the internationally standardized protocols. The number of layers should be large enough that distinct functions should not be put in the same layer and small enough that the architecture does not become very complex.

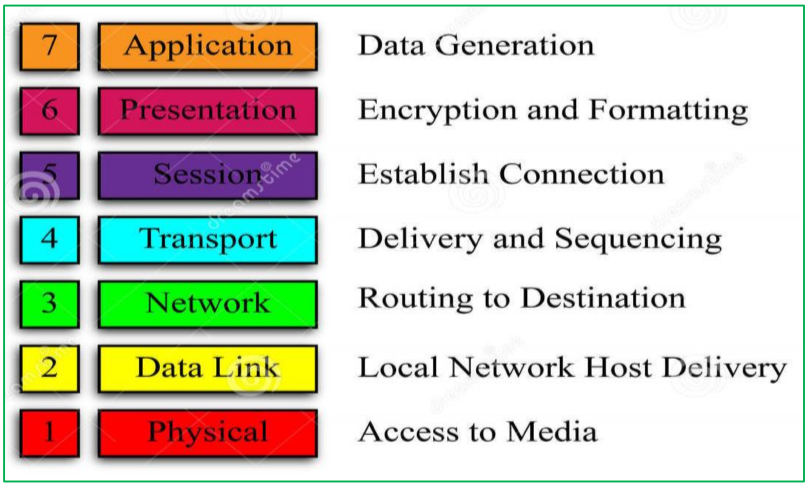

7 Application Data Generation
6 Presentation Encryption and Formatting
5 Session Establish Connection
4 Transport Delivery and Sequencing
3 Network Routing to Destination
2 Data Link Local Network Host Delivery
1 Physical Access to Media

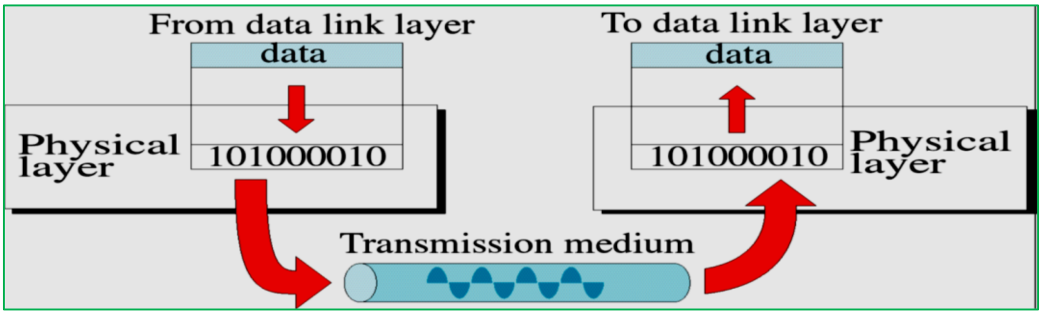
Fig: OSI Reference Model

Physical Layer:

It is the bottom layer of OSI Model. It is responsible for the actual physical connection between the devices. Such physical connection may be made by using twisted pair cable. It is concerned with transmitting bits over a communication channel.

Functions:

- Transforming bits into signals
- Provides synchronization of bits by a clock.
- Physical layer manages the way a device connects to network media.
- It defines the transmission rate.
- It defines the way in which the devices are connected to the medium.
- It provides physical topologies
- It can use different techniques of multiplexing.



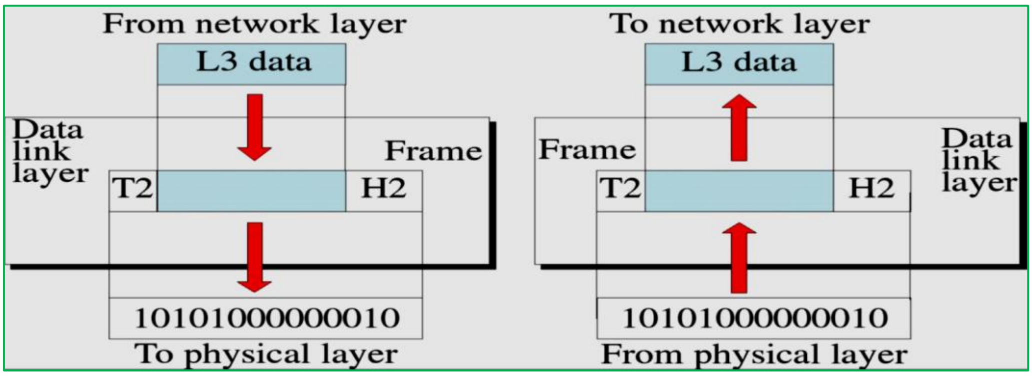
DATA LINK LAYER:

It is responsible for node-to-node delivery of data. It receives the data from network layer and creates FRAMES, add physical address to these frames & pas them to physical layer. It consist of 2 layers:

- **Logical Link Layer (LLC):** Defines the methods and provides addressing information for communication between network devices.
- **Medium Access Control (MAC):** establishes and maintains links between communicating devices.

Functions:

- **Framing: DLL** divides the bits received from N/W layer into frames. (Frame contains all the addressing information necessary to travel from S to D).
- **Physical addressing:** After creating frames, DLL adds physical address of sender/receiver (MAC address) in the header of each frame.
- **Flow Control:** DLL prevents the fast sender from drowning the slow receiver.
- **Error Control:** It provides the mechanism of error control in which it detects & retransmits damaged or lost frames.
- **Access Control:** When single comm. Channel is shared by multiple devices, MAC layer of DLL provides help to determine which device has control over the channel.



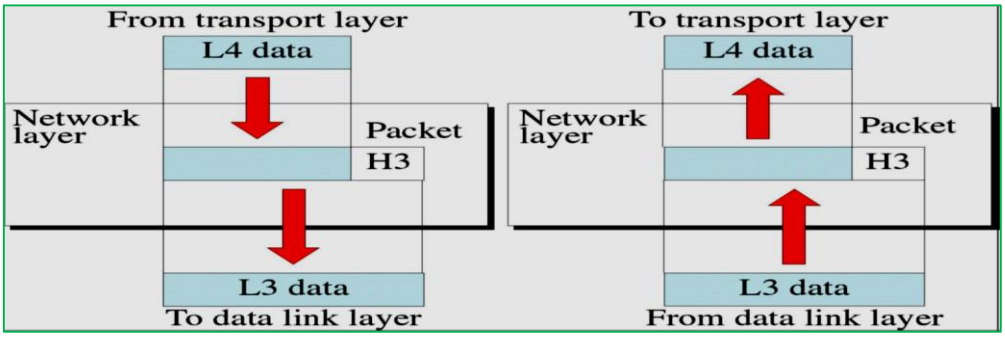
NETWORK LAYER:

It is responsible for the source to destination delivery of a packet across multiple networks. If two systems are attached to different networks with devices like routers, then N/W layer is used. Thus DLL overseas the delivery of the packet between the two systems on same network and the network layer ensures that the packet gets its point of origin to its final destination.

Functions:

- **Internetworking:** It provides Internetworking.
- **Logical Addressing:** When packet is sent outside the network, N/W layer adds Logical (network) address of the sender & receiver to each packet. Network addresses are assigned to local devices by n/w administrator and assigned dynamically by special server called DHCP (Dynamic Host Configuration Protocol)

- **Routing:** When independent n/w are connected to create internetwork several routes are available to send the data from S to D. These n/w are interconnected by routers & gateways that route the packet to final destination.



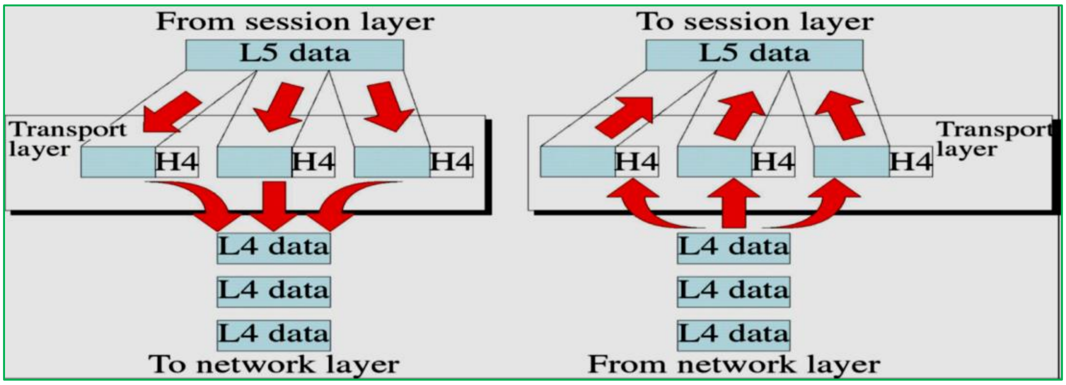
Transport Layer:

It is responsible for process-to-process delivery of the entire message. Transport layer looks after the delivery of entire message considering all its packets & make sure that all packets are in order. On the other hand network layer treated each packet independently. At the receiver side, Transport layer provides services to application layer & takes services form network layer. At the source side, Transport layer receives message from upper layer into packets and reassembles these packets again into message at the destination. Transport Layer provides two types of services:

- **Connection Oriented Transmission:** In this type of transmission the receiving devices sends an acknowledge back to the source after a packet or group of packet is received. It is slower transmission method.
- **Connectionless Transmission:** In this type of transmission the receiving devices does not sends an acknowledge back to the source. It is faster transmission method.

Functions:

- **Segmentation of message into packet & reassembly of packets into message.**
- **Port addressing:** Computers run several processes. TL header include a port address with each process.
- **Flow Control:** Flow control facility prevents the source form sending data packets faster than the destination can handle.
- **Error control**: TL ensures that the entire message arrives at the receiving TL without error.

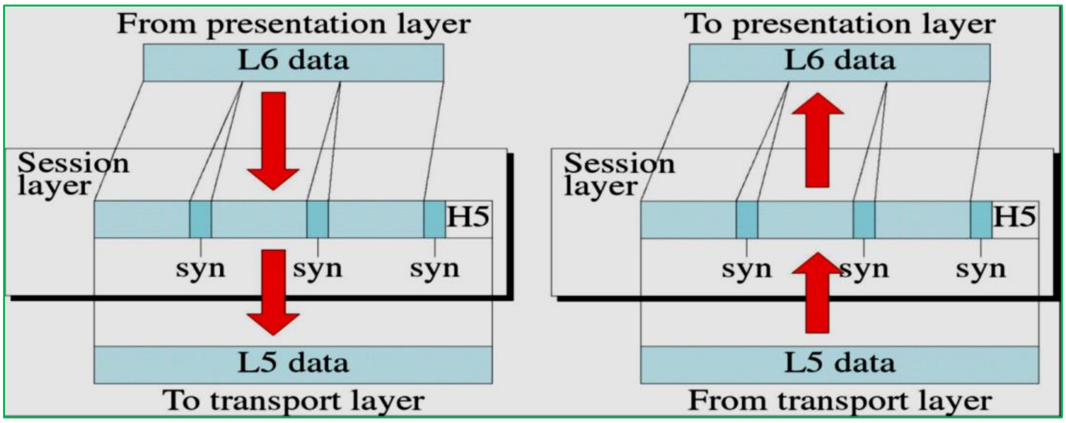


SESSION LAYER:

Session layer is the fifth layer of OSI Model. It has the responsibility of beginning, maintaining and ending the communication between two devices, called session. It also provides for orderly communication between devices by regulating the flow of data.

Functions:

- **Establishing, Maintaining and ending a session:** When sending device first contact with receiving device, it sends ***syn*** (synchronization) packet to establish a connection & determines the order in which information will be sent. Receiver sends ***ack*** (acknowledgement). So the session can be set & end.
- **Dialog Control:** This function determines that which device will communicate first and the amount of data that will be sent.
- **Dialog separation:** Process of adding checkpoints & markers to the stream of data is called dialog separation.

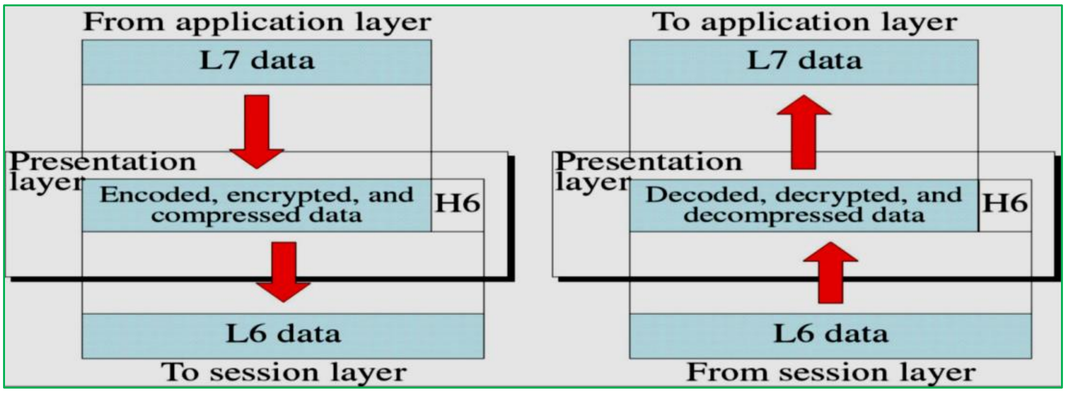


PRESENTATION LAYER:

Presentation layer is the sixth layer of OSI Model. It is concerned with the syntax & semantics of the information exchanged between the two devices. It was designed for data encryption, decryption and compression.

Functions:

- **Data Presentation or Translation:** Because different computers use different encoding systems. It ensures that the data being sent is in the format that the recipient can process.
- **Data Encryption:** Presentation Layer provides this facility by which hides the information from everyone except the person who originally sent the information & the intended recipient. When encrypted data arrives at destination, Presentation Layer decrypts the message.
- **Data Compression:** Presentation Layer shrinks large amount of data into smaller pieces i.e. it reduces the size of data.

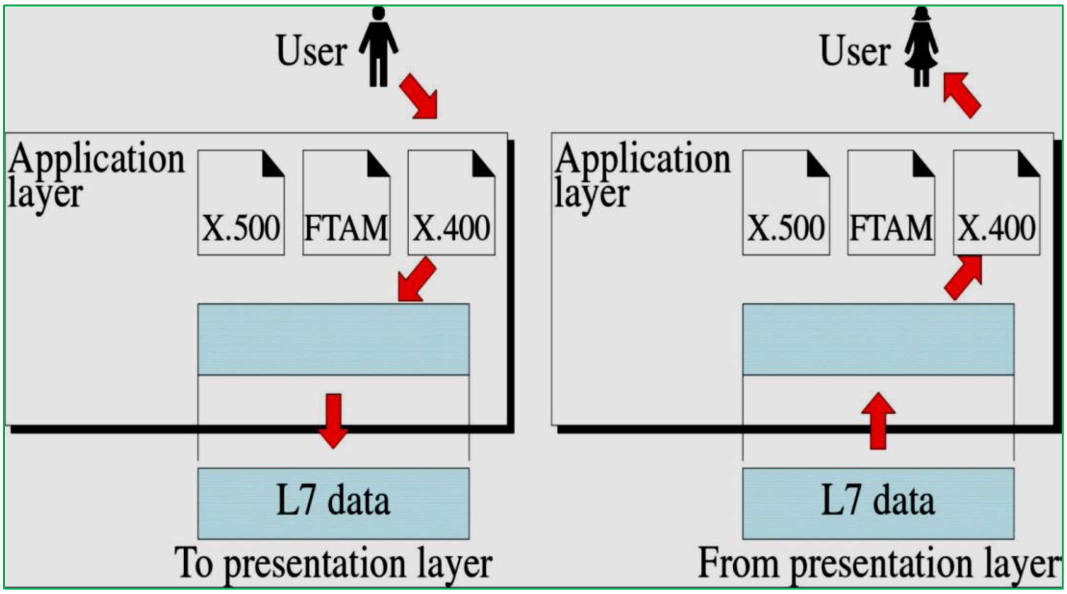


Application Layer:

It is the topmost i.e. seventh layer of OSI Model. It enables the user to access the network. It provides user interface & supports for services such as e-mail, file transfer, access to the World Wide Web. So it provides services to different user applications.

Functions:

- **Mail Services:** This application provides various e-mail services.
- **File transfer & Access:** It allows users to access files in a remote host, to retrieve files from remote computer for use etc.
- **Remote log-in:** A user can log into a remote computer and access the resources of that computer.
- **Accessing the World Wide Web:** Most common application today is the access of the World Wide Web.


User
Application layer
X.500
FTAM
X.400
L7 data
To presentation layer
User
Application layer
X.500
FTAM
X.400
L7 data
From presentation layer