|  |
| --- |
| **ISO/OSI Model**    **Lt Col. T V Perumal(Retd)**  **23MX128** |

**ISO/OSI Model**

The **Open Systems Interconnection model** (**OSI model**) is a conceptual model from the **International Organization for Standardization (ISO)** that provides a common basis for the coordination of standards development for the purpose of computing systems interconnection.

In this model, the communications between a computing system are split into seven different abstraction layers: Physical, Data Link, Network, Transport, Session, Presentation, and Application.

The model partitions the flow of data in a communication system into seven abstraction layers to describe networked communication from the physical implementation of transmitting bits across a communications medium to the highest-level representation of data of a distributed application. Each intermediate layer serves a class of functionality to the layer above it and is served by the layer below it.

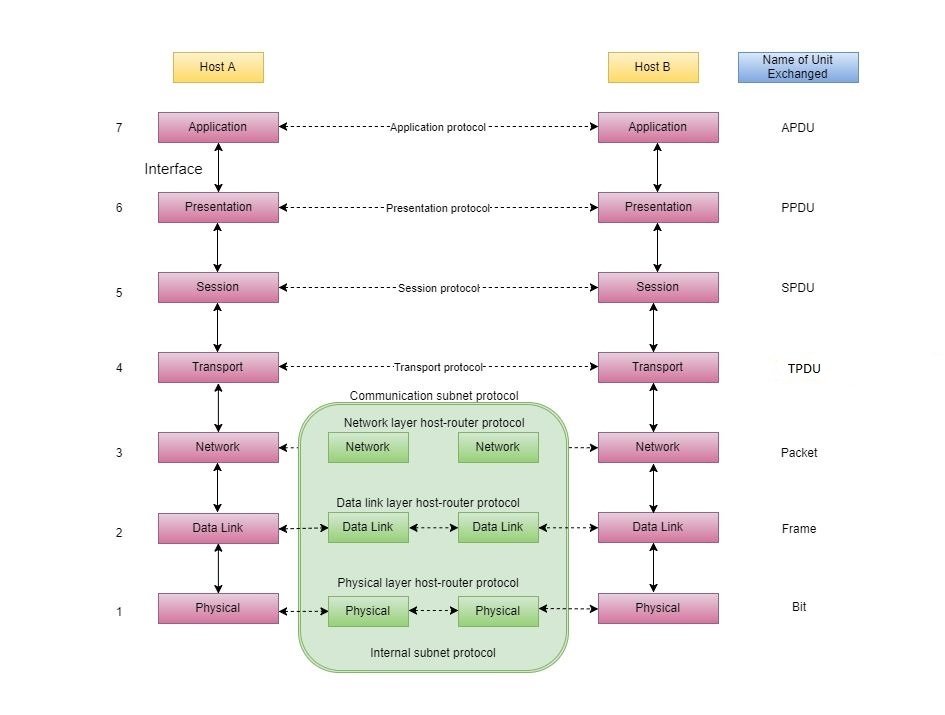
Each layer in the OSI model has well-defined functions, and the methods of each layer communicate and interact with those of the layers immediately above and below as appropriate.

The benefits of this layered architecture are: -

* Reduces complexity – by dividing the task into manageable parts - breaks Network communication into smaller, simpler components thus aiding component development, design, and trouble shooting.
* Standardizes interface – lays down standards for the network components there by allowing vendor development and support.
* Facilitates modular engineering – allows different types of Network hardware and software to communicate with each other.
* Interoperability between vendors – allows multiple-vendors to contribute to development as various components of the Network are standardized. Defines the process for connecting two layers together and eases trouble shooting and implementation.
* Ensures interoperability technology - change in one layer does not affect the other layers, allowing for quicker developments and advancement.
* Accelerates evolution – provides for effective updates and improvements to individual components without affecting other components or having to re-write the entire protocol.
* The layered architecture provides abstraction from other layers.
* Due to the abstraction, any layer can be changed independently. Each layer can be changed, tested, and analyzed independently, thus facilitating easier incorporation of newer technologies.

## The 7 Layers of the OSI Model

The OSI model is a layered framework for designing network systems that allows communication between all types of computer systems.



The top three Upper layers are known as the **Software layers.**

The upper layers are responsible for all the application-related issues. The layers include: -

1. Application Layer
2. Presentation Layer
3. Session Layer

The bottom three layers are known as the **Hardware layers.**

The lower layers are responsible for all the data transfer issues. The layers include: -

1. Network Layer
2. Data Link Layer
3. Physical Layer.

The **Transport layer** is the heart of the OSI model.

**Physical Layer**

The Physical layer is the lowest layer of the OSI model. This **layer** coordinates the functions required for movement of individual bits from one node to the next Node, over the physical medium. The main work of the physical layer is to activate, maintain, and deactivate the physical connection. The physical layer is also responsible for the transmission and reception of the unstructured raw data over the network.

The data in the physical layer consists of a stream of bits. The bits of data must be encoded into the form of signals for transmission. For the transmission, the physical layer sets the voltages, light speed (in the case of fiber optics cable), and data rates (numbers of bits to be transmitted per second).

**The functions provided by the Physical layer are as follows:**

* The physical layer encodes the signals at the sender's end and decodes the signals at the receiver's end. It also defines the type of encoding scheme to be used (how 0's and 1's are to be changed into signals).
* It deals with the synchronization of the sender and the receiver so that the receiver and the sender are at the same bit level. (Their clocks must be synchronized).
* The physical layer deals with the line configuration i.e., how the devices are connected through a dedicated link.
* It deals with the type of topology to be used - for example ring, mesh, bus, star, hybrid, etc. **Network topology** is the physical and logical arrangement of nodes and connections in a computer network.
* The physical layer also deals with the direction and type of transmission between two or more devices. The mode of transmission can be simplex, half-duplex, and full duplex.
* **Simplex:** communication channel that sends information in only one direction.
* **Half-duplex:** communication channel that sends information in both directions, but one at a time.
* **Full duplex:** communication channel that sends information in both directions on a network at the same time.

Image: Primary transmission of data and working of the Physical layer.

A diagram of a computer data processing process

Description automatically generated

**The various protocols used in the Physical layer are:**

* Digital Subscriber Line.
* Integrated Services Digital Network.
* Ethernet etc.

**The various devices used in the Physical layer are:**

* Network adapters
* Hubs
* Cables
* Repeaters
* Modem etc.

**Data Link Layer**

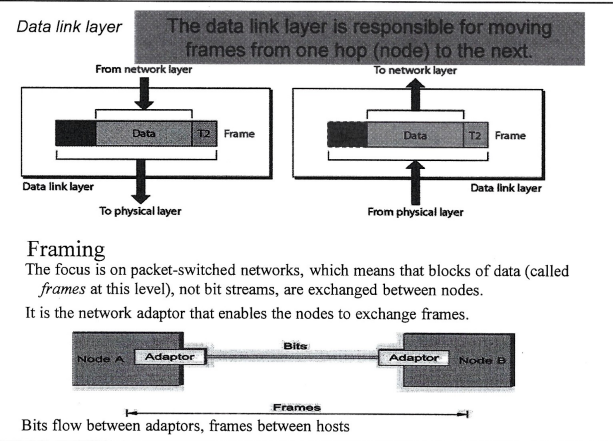
The Data Link Layer is the second layer of the OSI model, which is used to transmit the **error-free** frames from one node to the other. Provides reliable communication over the un-reliable Physical layer link. If the two computer nodes are on the same networks, then the Data link layer provides a connection between the two nodes. The main work of the Data link layer is to convert the data into the form of frames.

**The functions provided by the Data Link layer are as follows:**

* Provides reliable and efficient communication between two or more devices by using ARQ mechanism implemented as **piggy backing** –a process of attaching acknowledgement with the data packet to be sent. This improves the efficiency of the bi-directional protocols.
* Framing - the technique in which the data is divided into streams of bits (called **frames**) received from the Network layer. Along with the conversion of data into frames, the Data Link layer adds a header and trailer to the frames. The **header** (present at the starting of the frame) contains the hardware's physical address (MAC address) of source and destination. The **physical address** is also known as the MAC (Media Access Control) address. The MAC address is a unique address of each computer present on the NIC (Network Interface card). The **trailer** (present at the end of the frame) contains error detection and correction bits.
* Flow control - The Data Link layer also maintains the flow control of data during transmission. Decides on how many frames of to send without overwhelming the receiver. If the rate of data transmission and data absorption varies then there is likely loss of data, so the Data Link layer maintains the flow control to prevent loss of frames.
* Error control – Checks errors by implementing Acknowledgement and retransmission schemes. The Data Link layer adds the error detection and correction bits at the end of the frames in the form of trailers. These bits are used to detect the errors and then retransmit the damaged or lost data to prevent any kind of duplication.
* Line discipline – Manages the start of communication between the two ends. Data Link layer maintains the access control. In situations where two or more devices are connected to the same communication medium then the Data Link layer protocols determine the device that can transmit the data.

Images below shows the primary transmission of data and working of the Data Link layer.A diagram of a data flow

Description automatically generated



A diagram of a machine

Description automatically generated

**The various protocols used in the Data Link layer are:**

* PPP (Point-to-Point Protocol)
* Frame Relay
* ATM (The asynchronous transfer mode protocol) etc.

**The various devices used in Data Link layer are:**

* Bridges
* Switches
* NIC cards (Network Interface Cards), etc. The **physical address** is also known as the MAC (media access control) address. The MAC address is a unique address of each computer present on the NIC card.

The Data Link layer follows the Hop-to-Hop delivery system.

A diagram of a link

Description automatically generated

**Network Layer**

The Network layer is the third layer of the OSI model which provides Host-to-Host communication. The Network layer divides the data received from the Transport layer in the form of **Packets.**

**The Network layer at the source computer provides four services, namely – packetizing, finding the logical address of the next Hop, finding the physical (MAC) address of the next Hop, and fragmenting the datagram received from Transport layer if necessary.**

**The Network layer receives several pieces of information from the upper layers – Data, length of data, logical destination address, the protocol ID that is being used (mainly TCP/IP or UDP) and the service type. These pieces of information are processed to create fragmented datagrams(packets) along with the next Hop MAC address which is delivered to the Data link layer for further processing.**

The Network layer provides connectionless service.

* **Connectionless Services: Connectionless services are often associated with the Network Layer (Layer 3) of the OSI model. This layer is responsible for routing packets from the source to the destination across a network. Protocols like IP (Internet Protocol) operate at this layer and provide connectionless services. Connectionless services do not establish a dedicated connection before sending data; instead, they rely on the best-effort delivery of packets without guaranteeing delivery or ordering.**

**The functions provided by the network layer:**

* **Packetizing: It divides the outgoing data into packets.**
* **Logical Addressing:** The network layer adds the logical address i.e., IP address (Internet Protocol address) if the packet crosses the network boundary. This helps in the proper identification of devices on the network. The Network layer adds the source and destination address that forms the Network layer header which gets attached to the segments received from the Transport layer. This encapsulation forms the Packet.
* **Routing:** The Network layer chooses the best or optimum route and channel paths from source host Network to destination host Network for transmission of data (use routing algorithms). It manages the subnet traffic.
* **Address Resolution:** Determines and finds the physical address of the next hop.
* **Switching:** Uses packet switching to route packet from input router to appropriate forward router
* **Error control:** Header part of the IP datagram (Packet) is validated at each router using checksum.
* **Flow control:** There is no flow control service in the Network layer. Since flow control is provided for most of the upper layers that use the services of the Network layer, another level of flow control will make the Network layer more complicated and the whole system less efficient.
* **Congestion control**: In a connection less network, congestion control is implemented using a Choke packet – a special packet sent from a router to sender router when it experiences congestion. The Network layer uses the ICMP (Internet Control Message protocol) to send the Choke packet to indicate the sender to slow down transmission.
* **Security**: There is no security provided for in a connection less connection. To provide security, another virtual layer is created that changes the connection less service to connection-oriented service. This virtual layer is called IPSec, which is implemented as optional in the latest TCP/IP protocol.

Image shows the working of the Network layer.

A diagram of a data layer

Description automatically generated with medium confidence

**The various protocols used in Network layer are:**

* IPv4 (Internet Protocol version 4)
* IPv6 (Internet Protocol version 6)
* ICMP (Internet Control Message Protocol)
* IPSec (IP Security)
* ARP (Address Resolution Protocol)
* RARP (Router Address Resolution Protocol)
* MPLS (Multiprotocol Label Switching) etc.

**The various devices used in Network layer are:**

* Routers
* Layer 3 Switch
* Gateway
* Firewall
* Load balancer.
* Router Interface
* Network Address Translation (NAT) device

**Transport Layer**

The Transport layer is the fourth layer of the OSI model which is responsible for the process-to-process delivery of data. It provides the logical connection between the two hosts. The main aim of the Transport layer is to maintain order so that the data must be received in the same sequence as it was sent by the sender. The Transport layer provides connection-oriented services.

**Connection-Oriented Services:** Connection-oriented services are typically provided at the Transport Layer (Layer 4) of the OSI model. This layer is responsible for establishing, maintaining, and terminating connections, as well as ensuring the reliable and ordered delivery of data. Protocols such as TCP (Transmission Control Protocol) operate at this layer and provide connection-oriented services.

**The functions provided by the Transport layer are:**

* The Transport layer maintains the order of data.
* It receives the data from the upper layer and converts it into smaller parts known as **segments**.
* One of the major tasks of the Transport layer is to add the **port addressing** (addition of a port number to the header of the data). The port number is added so that the data can be sent to the respective process only.
* The Transport layer on the receiver's end re-assembles the segments to form the actual data.
* The Transport layer provides reliable service and prevents data loss by ensuring synchronized flow control using the Sliding window protocol.
* **Segmentation and Addressing:** Attaches the port address (16 bits) to each of the segments formed. Every local and remote process and application running on hosts are identified by their port numbers. Most of these port numbers are pre-determined and allocated by the ICANN.
* **Data integrity and error detection:** achieved using check sum and ACK NACK packets.
* **Congestion control:** Uses Open loop congestion control to prevent congestion. Uses Closed loop congestion control to clear congestion in the Network once it has occurred. TCP provides AIMD (Additive Increase Multiplicative Decrease) and Leaky bucket technique for congestion control.
* **Connection multiplexing and de-multiplexing** are essential functions performed at the Transport Layer. These processes are integral to the proper functioning of Transport layer protocols like TCP (Transmission Control Protocol) and UDP (User Datagram Protocol). These processes are crucial for efficient and organized communication at the Transport Layer of the OSI model.
  + **Connection Multiplexing:** Connection multiplexing is the process of combining multiple data streams or communication sessions from higher-layer applications into a single network connection or Transport layer connection.
    - It's done to optimize the utilization of network resources and reduce overhead by minimizing the number of connections established at the transport layer.
    - Example: In the case of TCP, multiple applications running on a computer can create different sockets, but these sockets can all be multiplexed onto a single TCP connection when they are destined for the same remote endpoint. This allows efficient use of the underlying network infrastructure.
  + **Connection De-Multiplexing:** Connection de-multiplexing is the process of correctly directing incoming data packets from the network to the appropriate higher-layer application or socket.
    - It's done to ensure that data from different sources or applications is delivered to the correct destination within the receiving host.
    - Example: When a host receives data on a network connection, it needs to determine which socket or application should receive that data. TCP and UDP use port numbers in their headers to de-multiplex incoming packets. The destination port number in the Transport layer header helps identify the appropriate application or service running on the receiving host.
  + **Summary:**
    - Connection multiplexing - bundles multiple application-level data streams into a single Transport layer connection.
    - Connection de-multiplexing - directs incoming packets to the appropriate higher-layer application or service based on port numbers.
* **Encapsulation and de-capsulation** are fundamental processes that occur at the Transport Layer (Layer 4) of the OSI model. These processes are essential for packaging, transmitting, and unpacking data when it is sent across a network. Two primary Transport layer protocols, TCP (Transmission Control Protocol) and UDP (User Datagram Protocol), illustrate these concepts:
  + **Encapsulation in the Transport Layer:**
    - Data Segmentation: When an application sends data to be transmitted over a network, it hands the data to the Transport Layer.
    - Segmentation: The Transport Layer may break the data into smaller segments or packets for transmission. These segments typically have headers added to them, containing information such as source and destination port numbers (in the case of TCP and UDP) and sequence numbers (in the case of TCP). This process is known as encapsulation.
    - For TCP:
      * TCP encapsulates data into segments.
      * It adds a TCP header that contains control information like sequence numbers, acknowledgment numbers, and other control flags.
      * The resulting TCP segment is then passed down to the Network Layer for further encapsulation (e.g., adding IP headers).
    - For UDP:
      * UDP encapsulates data into datagrams.
      * It adds a UDP header with source and destination port numbers.
      * The resulting UDP datagram is passed down to the Network Layer.
  + **De-Capsulation in the Transport Layer:**
    - Receiving Data: When data arrives at a destination host it is initially received at the Network Layer. The Network Layer forwards the received packets to the appropriate Transport Layer protocol (TCP or UDP) based on the destination port number.
    - De-capsulation: At the Transport Layer, the received segments (for TCP) or datagrams (for UDP) are de-capsulated.
    - For TCP, the TCP header is removed to extract the payload, and any necessary reordering and reassembly of segments may occur.
    - For UDP, the UDP header is removed to extract the payload.
    - Delivery to Application: Finally, the Transport Layer delivers the extracted data payload to the appropriate application or process running on the receiving host. The destination port number is used to determine which application should receive the data.
  + **Summary:** Encapsulation involves adding headers to data for transmission, and De-capsulation is the process of removing those headers to retrieve the original data at the receiving end. The Transport Layer plays a crucial role in these processes, and protocols like TCP and UDP are responsible for performing these functions as data traverses a network.
* **Socket addressing:** The Transport layer in TCP/IP suite needs both the IP address and the Port address (16 bits) of each end to make a connection and facilitating applications to communicate with each other. Sockets are a fundamental concept in network programming. It allows multiple services to run on a single device and enables communication between devices across a network. By combining an IP address and a port number, networked applications can ensure that data is sent to the correct service or endpoint on the target device.
  + Addressing refers to the method used to identify and locate a specific endpoint or service in a network using a combination of an IP address and a port number.
  + **IP Address**: An IP (Internet Protocol) address is a numerical label assigned to each device connected to a network. It serves as a unique identifier for a device within that network. IP addresses are used for routing data packets to their intended destinations across the Internet or a local network.
  + **Port Number**: A port number is a 16-bit unsigned integer that identifies a specific service or application running on a device with a particular IP address. Ports are used to distinguish between different services on the same device. Port numbers can range from 0 to 65535, with well-known ports (e.g., port 80 for HTTP) reserved for specific services and higher-numbered ports available for custom or less common applications.
  + **Combining IP Address and Port Number**: In socket addressing, a combination of an IP address and a port number uniquely identifies an endpoint where a service is running. This combination is often represented as IP:Port. For example, "192.168.1.1:80" specifies a specific device (identified by its IP address, 192.168.1.1) running a service on port 80.
  + **Socket**: A socket is an endpoint for sending or receiving data in a network. It is created in software and is associated with a specific IP address and port number. Sockets enable communication between applications running on different devices. When one application wants to communicate with another, it specifies the destination socket's IP address and port number.

Image: Primary transmission of data and working of the transport layer.

A diagram of a layer of data

Description automatically generated

**The various protocols used in Transport layer:**

* TCP (Transmission Control Protocol)
* UDP (User Datagram Protocol) etc.

Transmission Control Protocol

* + Reliable stream transport –
    - Reliable services never lose/corrupt data.
    - Reliable services cost more.
    - Example of application needing reliable service – File transfer.
    - Example of application Not needing reliable service – Voice traffic.
    - Not all Applications need connections or reliability.
  + Connection oriented (full duplex virtual circuit) – concept is like the telephone system.
    - Place call, the 2 ends (Hosts) communicate and agree on mode and details.
    - After agreement, the applications are notified of connection.
    - During transfer of data, ends (Session layer) communicate continuously to verify data received correctly.
    - When done, ends terminate the connection.
  + If UDP is like regular post mail, TCP is like a phone call.
  + Provides buffering and flow control.
  + Takes care of lost packets, out of order, duplicates, long delays – by use of Positive Ack re-transmission protocol.
  + Helps isolate Application programs from Network hardware details.

**The various devices used in Transport layer are:**

* Segments,
* Load Balancers/Firewalls, etc.

**Session Layer**

The Session layer is the fifth layer of the OSI model whose main aim is to establish, manage, and terminate the connection between Applications.

Facilitates the users from different platforms to set up an active communication session between themselves.

Provides the synchronization between distinctive Applications. Synchronization is necessary for efficient delivery of data, and without any loss that is to be received at the destination.

Session layer places different check points while sending a large amount of data.

**There are mainly two functions provided by the Session layer:**

* The Session layer is responsible for creating a **dialog box** which allows two systems to enter a dialog and transmit the data in either half-duplex or full-duplex mode.
* The Session layer is also responsible for adding synchronization bits or checkpoints into the stream of data. These checkpoints help to detect any kind of error that may have occurred during the data transmission. So, if an error has occurred in between the transmission then the re-transmission will take place from the last checkpoint only.
* **Simplex**: Only in one direction transmission possible, from sender to receiver.
* **Half-duplex**: Transmission of signals in both directions but only one way at a time (not simultaneously).
* **Full Duplex:** Transmission and reception of signals both directions simultaneously.

**The various protocols used in Session layer are:**

* PAP (Password Authentication Protocol)
* PPTP (Point-to-Point Tunneling Protocol)
* RPC (Remote Procedure Call Protocol)
* RTCP (Real-time Transport Control Protocol) etc.

**The various devices used in Session layer are:**

* Gateway etc.

**Presentation Layer**

The Presentation layer is the sixth layer of the OSI model which mainly concentrates on the syntax and semantics of the information exchanged between the systems. The main aim of the **Presentation layer** is to convert the data from one presentation format to the other format as different applications may use different applications.

**There are mainly three functions provided by the Presentation layer. They are as follows:**

* Present the data to its end user in a form in which it can easily be understood.
* Since different computer systems use different encoding systems so the Presentation layer must translate the data into a computer-dependent format. Plays the role of a translator between two systems so that they come on the same platform for communication and will easily understand each other.
* This layer converts the data, which is in the form of characters and numbers, to bits before transmission. It translates the data in the format that is required by different devices – like for network devices, or for Phones, PC, audio devices etc.
* The Presentation layer deals with the encryption and decryption of the data so that the data can be transmitted securely.
* The Presentation layer also deals with the compression and de-compression for efficient use of bandwidth for large file transmissions.

Image: basic transmission of data and the working of the presentation layer.

A diagram of a data processing process

Description automatically generated

**The various protocols used in this layer are:**

* AFP (Apple Filing Protocol)
* ICA (Independent Computing Architecture)
* Citrix system core protocol
* LPP (Lightweight Presentation Protocol)
* NCP (NetWare Core Protocol)
* NDR (Network Data Representation)
* Tox protocol etc.
* NetBIOS
* RPC (Remote Procedure Call)
* PPTP (Point to Point Tunneling Protocol)
* SMB (Sever Message Block)
* NFS (Network File System)
* Session Control protocols

**Application Layer**

The Application layer is the seventh and the last layer of the OSI model, which mainly concentrates on providing services to the users.

This layer communicates with the Operating system of the Computer, with the End user and user Applications. Checks whether the sender’s computer has the necessary communication interfaces – such as Ethernet or Wi-Fi interfaces.

This layer enables a direct interface and access for users to access the network. Services provided by this layer include - users to share files, access emails, FTP, GUI based software, remote login, telnet network devices, access webpages (via the world wide web) etc.

Enables authentication between devices for an extra layer of network security.

On the receiver end the data from the Application layer is presented to the user application.

Image: working of the Application layer.

A diagram of a process

Description automatically generated

**The various protocols used in this layer are:**

* DNS (Domain Name System)
* DHCP (Dynamic Host Control Protocol)
* SMTP (Simple Mail Transfer Protocol)
* SNMP (Simple Network Management Protocol)
* FTP (File Transfer Protocol)
* Telnet
* POP (Post Office Protocol)
* HTTP (Hypertext Transfer Protocol)
* HTTPS
* NTP (Network Time Protocol)
* SSH (Secure Shell)
* TFTP (Trivial File Transfer Protocol)

**The various devices used in this layer are:**

* PC's (Personal Computer)
* Phones
* Servers
* Firewalls