Network Programming
Unit I

By

Khushbu Kumari Sarraf

EMBA,BE

Introduction.

Networks

Two or more computers connected together in a way that allows resource sharing.

- A network contains any combination of computers, computer terminals, printers, display devices, cables, or wireless connections.
- A network is a collection of computers or other hardware devices that are connected together using special hardware and software.

Introduction.

Resources

Resources may be:

- Files.
- Folders.
- Printers.
- Memory.
- Applicati

Introduction.

Computer networking

Advantages	**Disadvantages**
Communication between computer processing units (CPUs)	Access restrictions
Data sharing	Server failures
Hardware sharing	Privacy concerns
Internet access	Security threats
Data management	Redundancy

Types of Networks.

Network types

Computer networks vary in shape (topology) and size depending on their application.

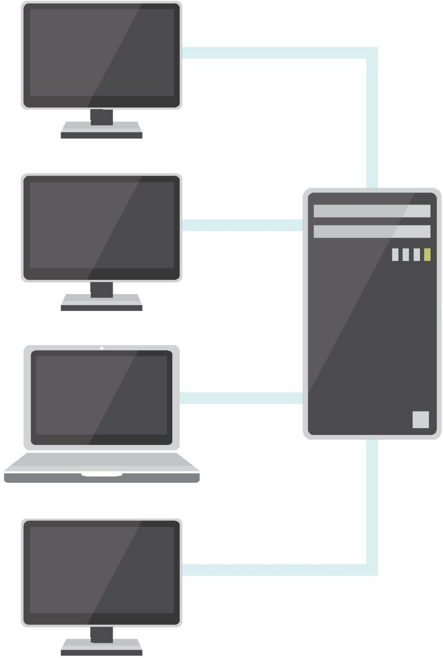
Some of the major types are:

- Local area networks (LANs).
- Wide area networks (WANs).

Types of Networks.

Local area networks (wireless and wired)

- Span a small geographic area.
- Usually confined to a building, a group of buildings, or a vehicle, for example a train or a streetcar.
- Data travels between network



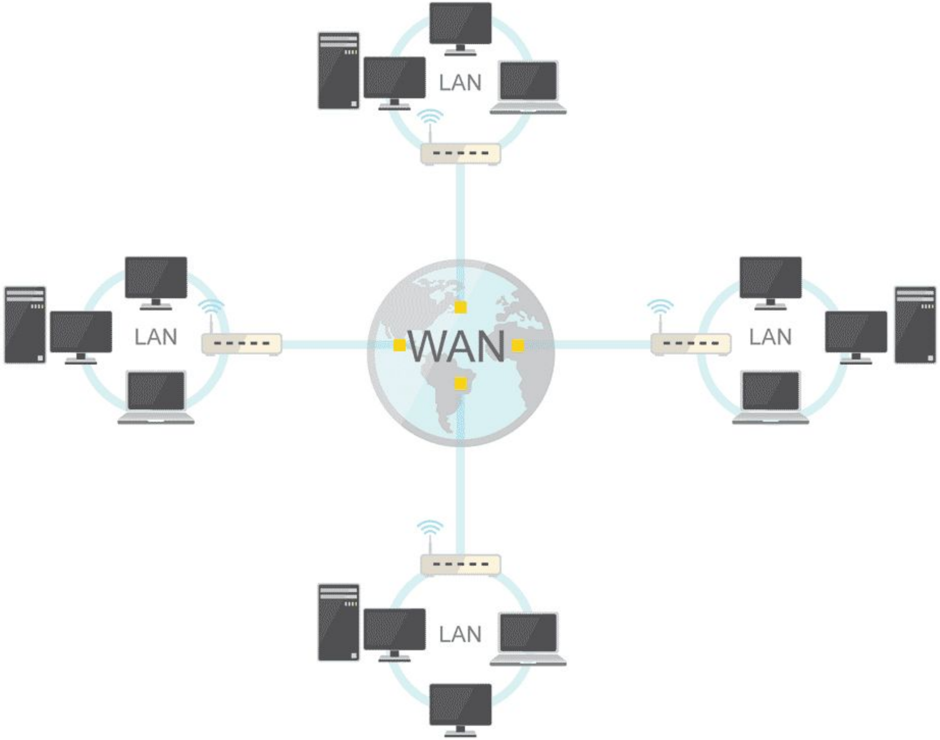
Types of Networks.

Wide area networks (WAN)

- A WAN is a computer network that covers a large area (any network whose communication links across metropolitan, regional, or national boundaries).
- A network that uses routers, modems, and public communication links.
- The world’s largest WAN is the Internet.

Types of Networks.

WAN



Network Topologies.

Network topologies

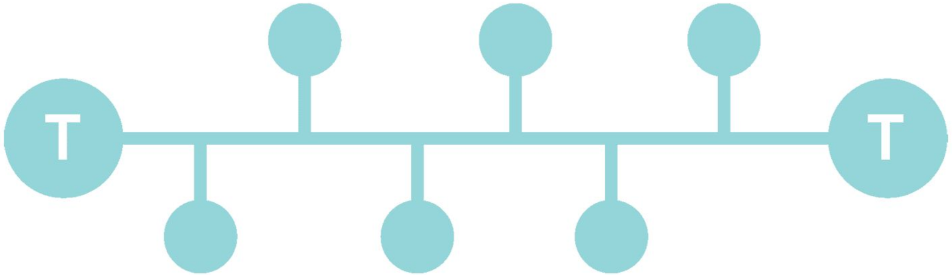
Network topology refers to the shape of a network, or the network’s layout. A network’s topology determines how different devices in a network are connected to each other and how they communicate.

The different network topologies are as follows:

Network Topologies.

Bus topology

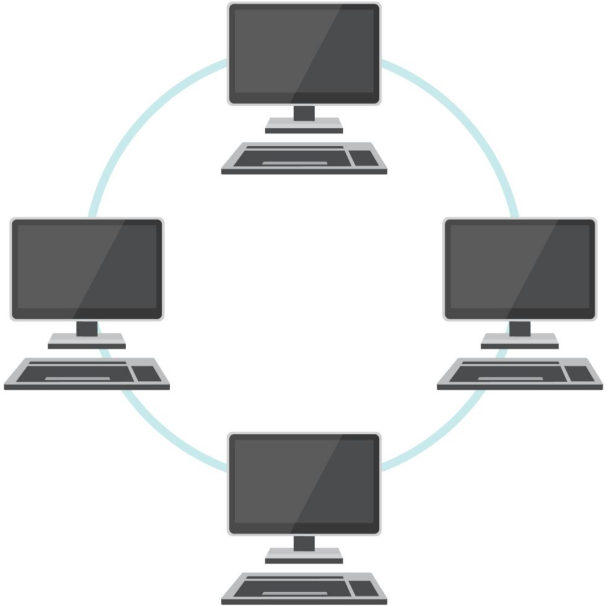
- All devices are connected to a central cable, called a bus or a backbone.
- The simplest physical topology–least amount of cables–but also covers the shortest distance.
- There are terminators at each end of the bus that stop the signals and keep them from travelling backwards.
- All computers share the same data and address path. Messages pass through the central cable and each computer checks to see if the message is addressed to itself. If the address of the message matches the computer’s address, the network adapter copies the message



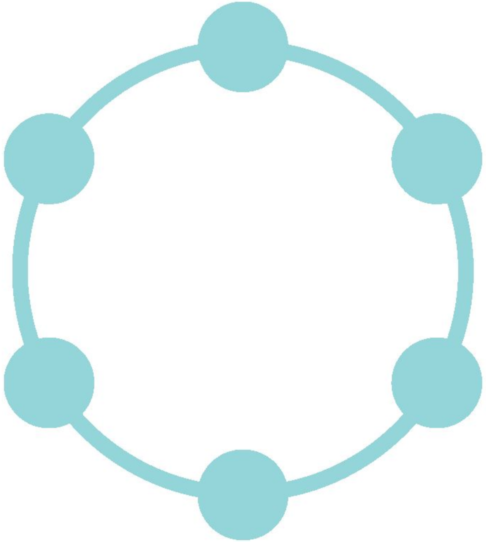
BUS TOPOLOGY

Network Topologies.

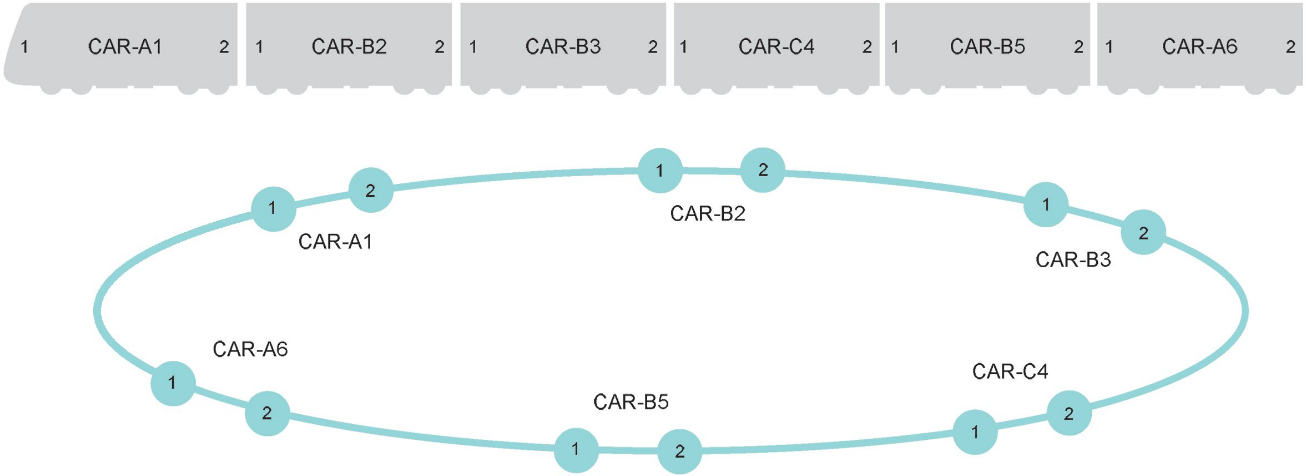
Ring topology



- All devices are connected to one another in the shape of a closed loop.
- Each device is connected directly to two other devices, one on either side of it.
- An equivalent system exists on the trains (TIN) and on streetcars (the



RING TOPOLOGY

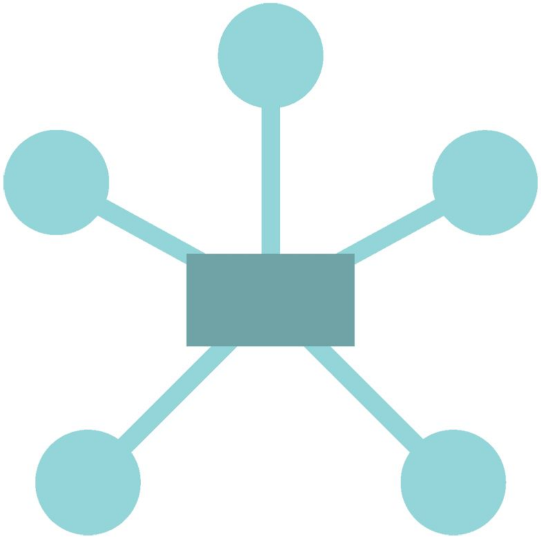


RING TOPOLOGY

Network Topologies.

Star topology

- Devices are not directly connected to each other, rather through a central hub.
- Devices communicate across the network by passing data through the hub or switch.

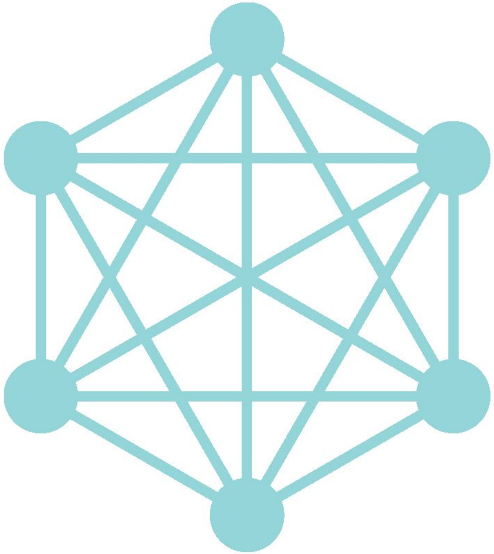


STAR TOPOLOGY

Network Topologies.

Mesh topology

- The simplest logical topology in terms of data flow, and the most complex topology in terms of physical design.
- Each device is connected to every other device.
- This topology is rarely found in LANs, mainly because of the complexity of the cabling.
- Because of its design, the physical mesh topology is very expensive to install and maintain.

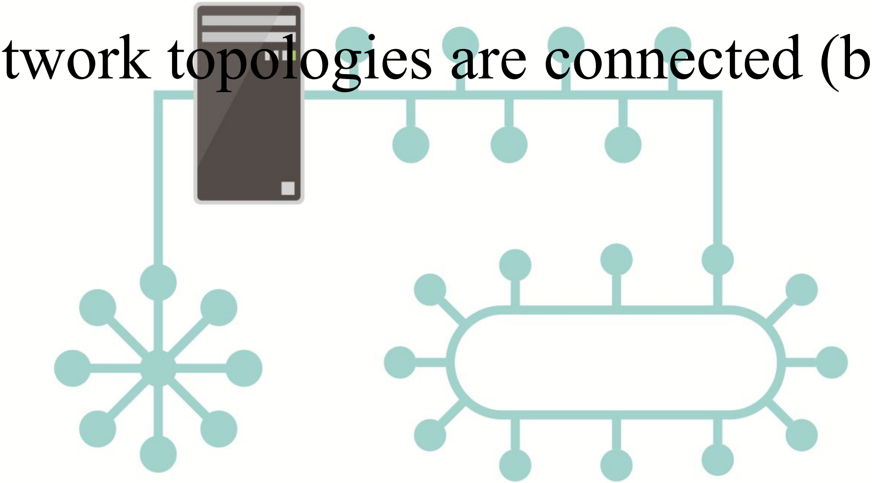


MESH TOPOLOGY

Network Topologies.

Hybrid topology

A hybrid topology is produced when two, or more different basic network topologies are connected (bus, star, ring).



Internet and Intranet.

Internet vs. intranet

Internet

A worldwide system of computer networks. A network of networks in which users at any one computer, with the necessary permissions, can get information from any other computer.

The most commonly used protocol is TCP/IP, it stands for:

Transmission Control Protocol

Internet

Internet protocol

The most common network protocol in public use is the IP.

- The basic protocol that enables home computing devices and LANs across the Internet to communicate with each other.
- Works well for moving individual messages from one network to another.
- TCP allows continuous transmission of data (streaming).

Internet and Intranet.

Internet vs. intranet

Intranet

- A self-contained private network.
- It may consist of many interlinked local area networks and also use leased lines in a wide area network.
- Uses TCP/IP, hypertext transfer protocol (HTTP) and other Internet protocols.
- Companies can send messages through the public network, using encryption/decryption and other security safeguards to connect one part of the

Protocol, Interface, Service

A protocol is an agreement between the communicating parties on how communication is to proceed.

In reality, no data are directly transferred from layer n on one machine to layer n on another machine. (virtual communication is shown by dotted lines and physical communication by solid lines).

Between each pair of adjacent layers is an interface. The interface defines which primitive operations and services the lower layer makes available to the upper one. (minimizing information + Simplify replacing).

Protocol

Network/communication protocols

A protocol is simply an agreed on set of rules and procedures for transmitting data between two or more devices. Hundreds of different protocols have been developed, each designed for specific purposes and environments.

The protocol defines:

- How the sending device indicates it has finished sending the message.
- How the receiving device indicates it has received the message.
- How data is transmitted from source to destination.

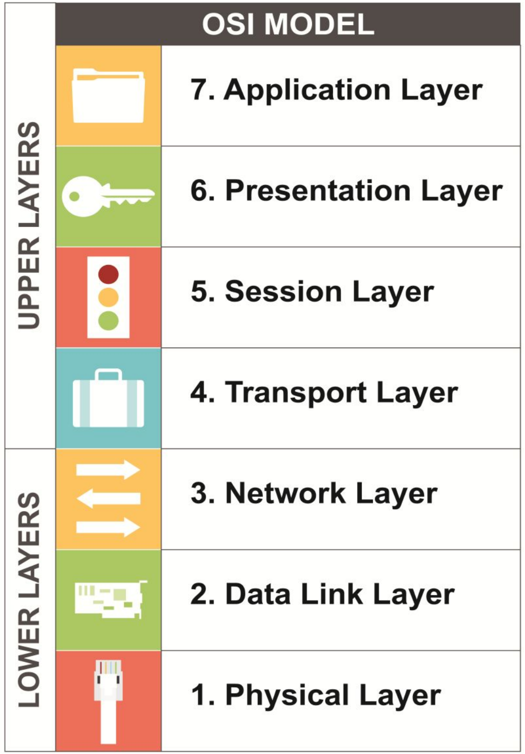
Network/communication protocols

- Network protocols include mechanisms for devices to identify and make connections with each other, as well as formatting rules that specify how data is packaged into messages, sent, and received.
- Hundreds of different computer protocols have been developed, each designed for specific purposes and environments.

Internet and Intranet.

Network protocols

- Network protocols are layered such that each one relies on the protocols that underlie it. Sometimes referred to as the protocol stack.



OSI MODEL

UPPER LAYERS

7. Application Layer

6. Presentation Layer

5. Session Layer

4. Transport Layer

LOWER LAYERS

3. Network Layer

2. Data Link Layer

1. Physical Layer

- Both TCP and IP operate

Internet and Intranet.

The Open System Interconnection (OSI) model

A logical representation of the path data must travel in order to go through the network.

- **Upper layers:**

  Represent software that implements network services like encryption and connection management.

- **Lower layers:**

The Open System Interconnection Model (OSI)

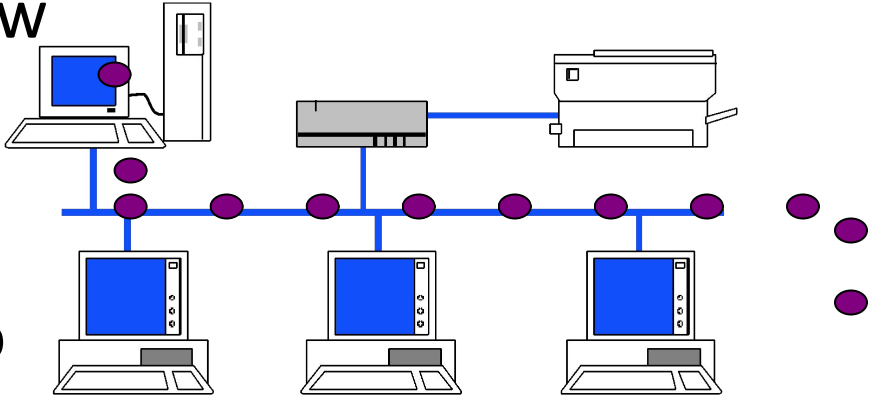
- The International Standards Organization (ISO) has developed a universal architecture for computer communications.
- This standard, Known as the *open Systems Interconnection model, or OSI model.*
- *The purpose of OSI is* to permit communications

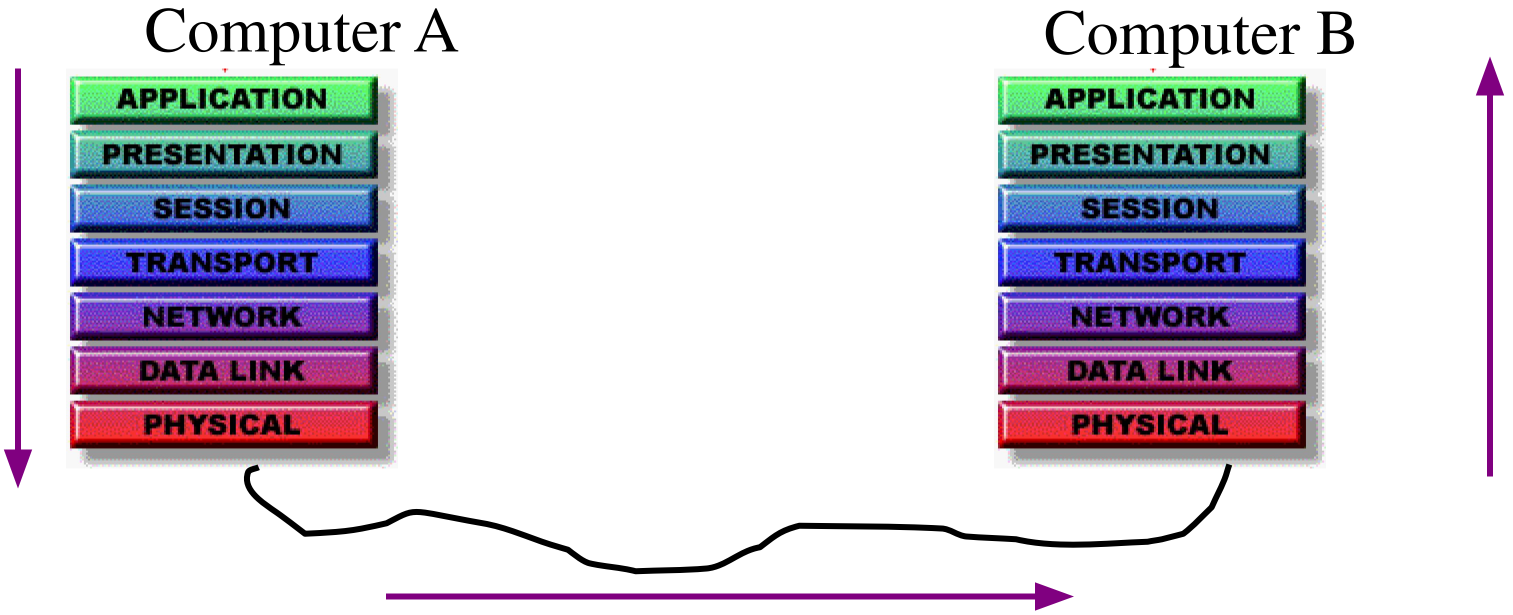
OSI Layers

- OSI has seven layers.
- Each layer represents a particular function.
- It could be, each function is performed by a separate piece of hardware or software.
- Sometimes, a single program may performed the functions of several layers.

OSI Reference Model

- The OSI reference model describes how data makes its way from application programs through a network medium to another application located on another computer on a network

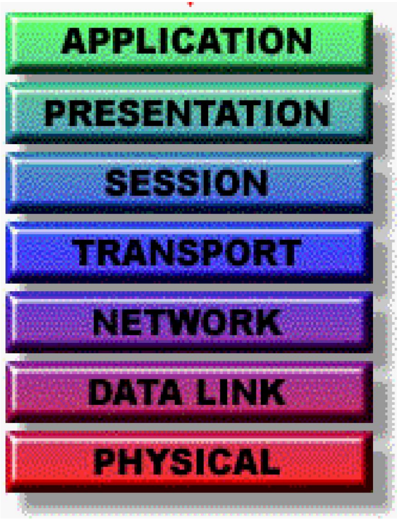


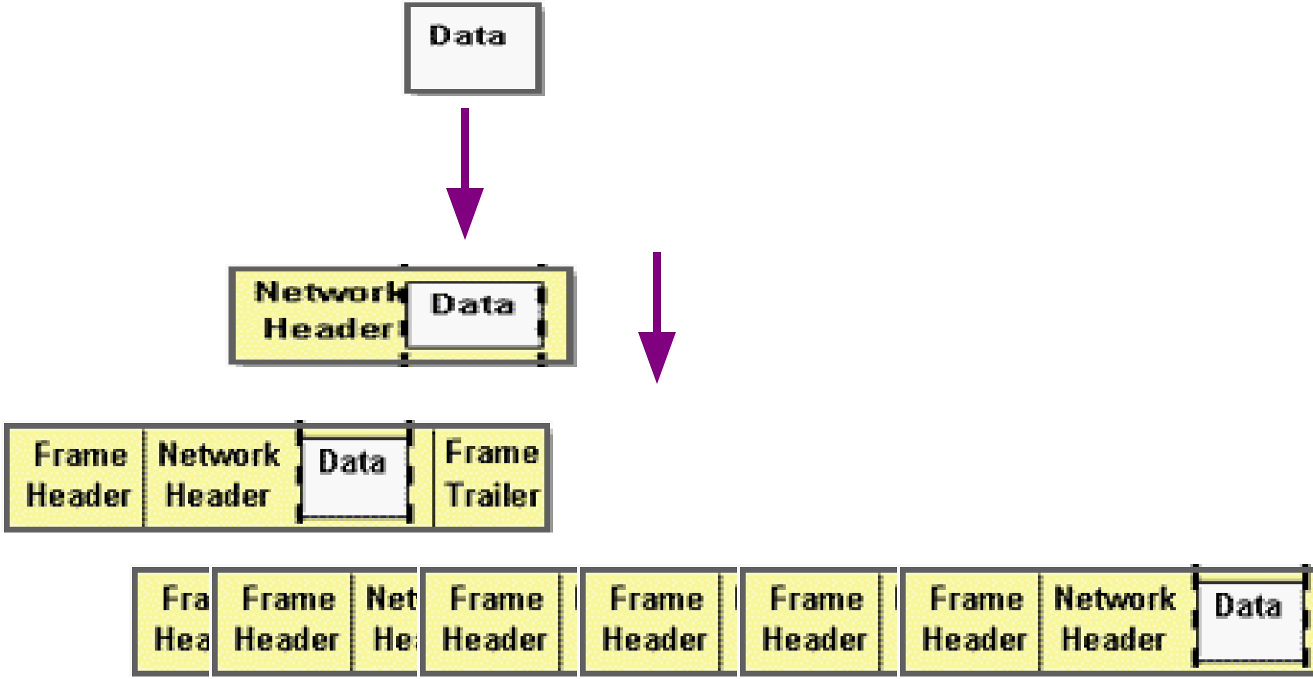


- To accomplish this task data must travel from the application layer to the physical layer on one computer across the network media and from the physical layer to the application layer of another computer

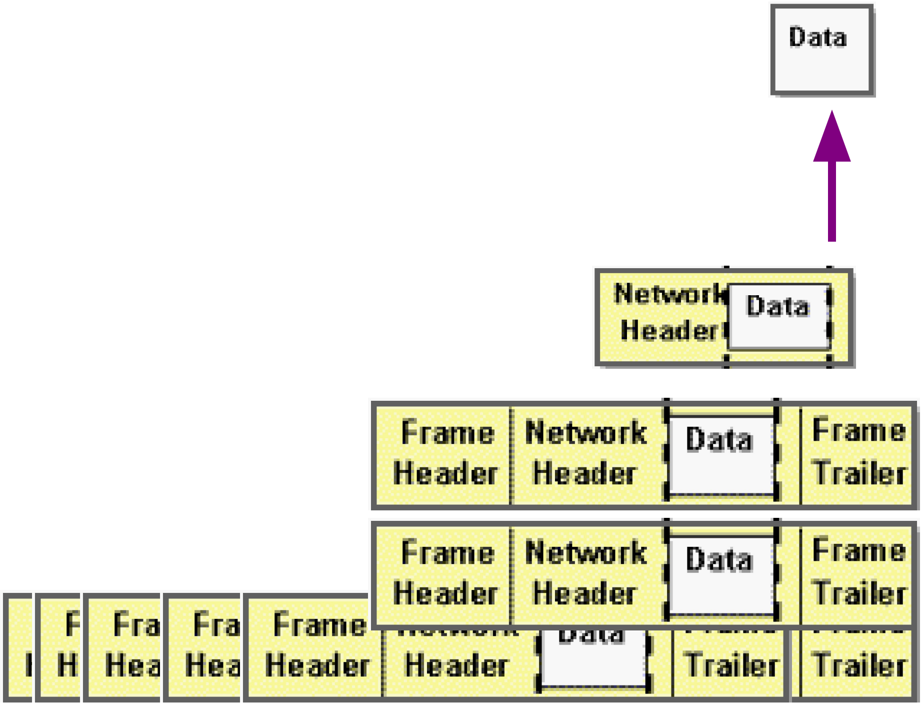
- As data moves down through the layers of the OSI model, headers are added.

Computer A

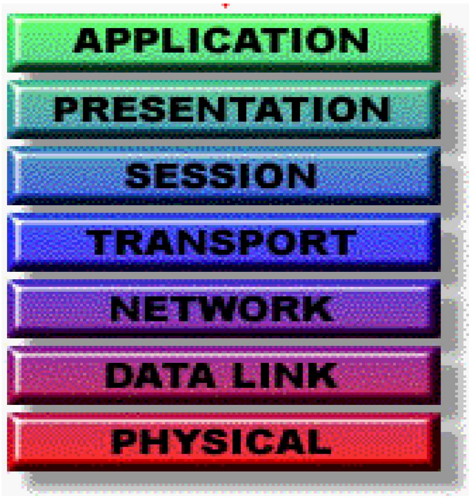




- As data moves up through the layers of the OSI model, headers are removed.



Computer B



OSI Layers

- The Lowest layer, Known as *physical Layer or Layer 1,*
  - is responsible for transmission of bits.
  - Is always implemented by using hardware.
  - Is encompasses the mechanical, electrical, and functional interface.
  - Is the interface to the outside world
  - using electronic signals as specified by interface standards.

OSI Layers

- The *Data Link Layer, Or Layer 2,*
  - is responsible for ensuring error-free,
  - reliable transmission of data.
  - examine the bits received to determine if errors occurred during transmission.
  - Is able to request retransmission or correction of any errors using protocols.

OSI Layers

- The *Network Layer, or Layer 3,*
  - is responsible for setting up the appropriate routing of messages throughout a network
  - is concerned with he types of switching networks used to route the data
- Note:

  Physical, Data Link, and Network layers are usually referred to as the *lower layers*

OSI Layers

- The *Transport Layer, or Layer 4,*
  - is responsible for isolating the function of the lower layers from the higher layers
  - is responsible for monitoring the quality of the communication channel
  - is responsible for selecting the most cost efficient communication service.
  - accepts messages from higher layers, and breaks them down into messages that can be accepted by the lower layers

OSI Layers

- The *Session Layer, or Layer 5,*
  - is responsible for terminating the connection
  - requests a logical connection be established based on the end user's request
  - handles any necessary "log-on" and password procedures.

OSI Layers

- The *Presentation Layer, or Layer 6,*
  - provides format and code conversion services
  - handles any necessary conversion different character codes; example

    ASCII-to- EBCDIC

OSI Layers

- *The Application Layer or Layer 7,*
  - provides access to the network for the end user
  - determines the user's capabilities on the network
  - some Application Layer software, permit remote terminal to only access a host computer; other Application Layer software might also permit file transfers.

The TCP/IP Protocol

- The TCP/IP Suite
  - is a collection of protocols originally designed for use on an network connecting U.S. government agencies with universities performing research
  - specifies protocols at various levels of the OSI model and covers a wide variety of tasks likely to be performed on an open network

Comparison of ISO-OSI Model and the (TCP/IP) Model

Application	Application
Presentation	
Session	
Transport	Host-to-Host/Transport
Network	Internet
Data Link	Network Access
Physical	

Application	Provides access to the OSI environment for users and also provides distributed information service
Presentation	Provides independence to the application process from difference in data representation (syntax)
Session	Provides the control structure for communication between application; establishes, manages ___ terminates connection (session) between cooperating applications.
Transport	Provides reliable, transparent transfer data between end points; provides end-to-end error recovery and flow control.
network	Provides upper layer with independence from the data transmission and switching technologies used to connect systems; responsible for establishing, maintaining, and terminating connections.
Data Link	Provides for the reliable transfer of information across the physical link; sends blocks of data (frames) with the necessary synchronization, error control ,and flow control
Physical	Concerned with transmission of unstructured bit stream over physical medium; deals with the mechanical, electrical, functional and procedural characteristics to access the physical

TCP/IP Layers

- no official model but a working one
  - Application layer
  - Host-to-host, or transport layer
  - Internet layer
  - Network access layer
  - Physical layer

Physical Layer

- concerned with physical interface between computer and network
- concerned with issues like:
  - characteristics of transmission medium
  - signal levels
  - data rates
  - other related matters

Network Access Layer

- exchange of data between an end system and attached network
- concerned with issues like :
  - destination address provision
  - invoking specific services like priority
  - access to & routing data across a network link between two attached systems
- allows layers above to ignore link specifics

Internet Layer (IP)

- routing functions across multiple networks
- for systems attached to different networks
- using IP protocol
- implemented in end systems and routers
- routers connect two networks and relays data between them

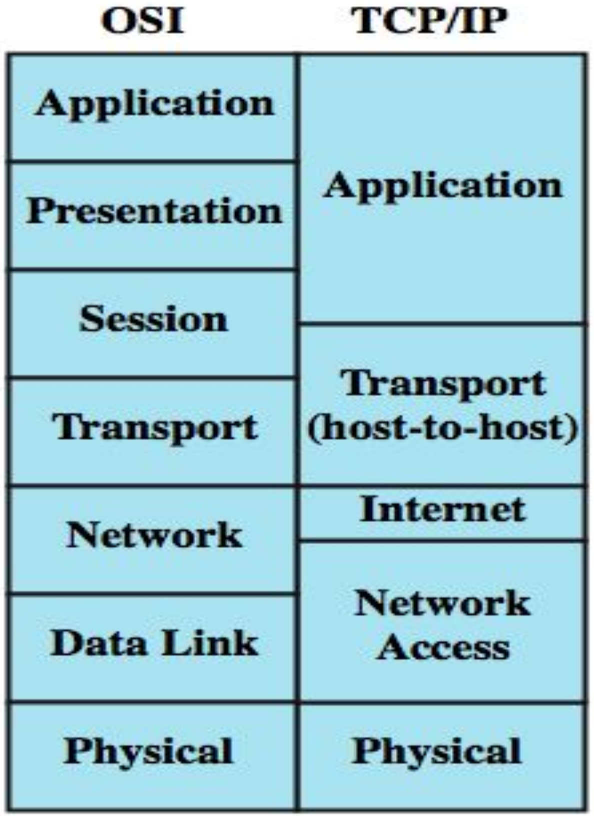
Transport Layer (TCP)

- common layer shared by all applications
- provides reliable delivery of data
- in same order as sent
- commonly uses TCP

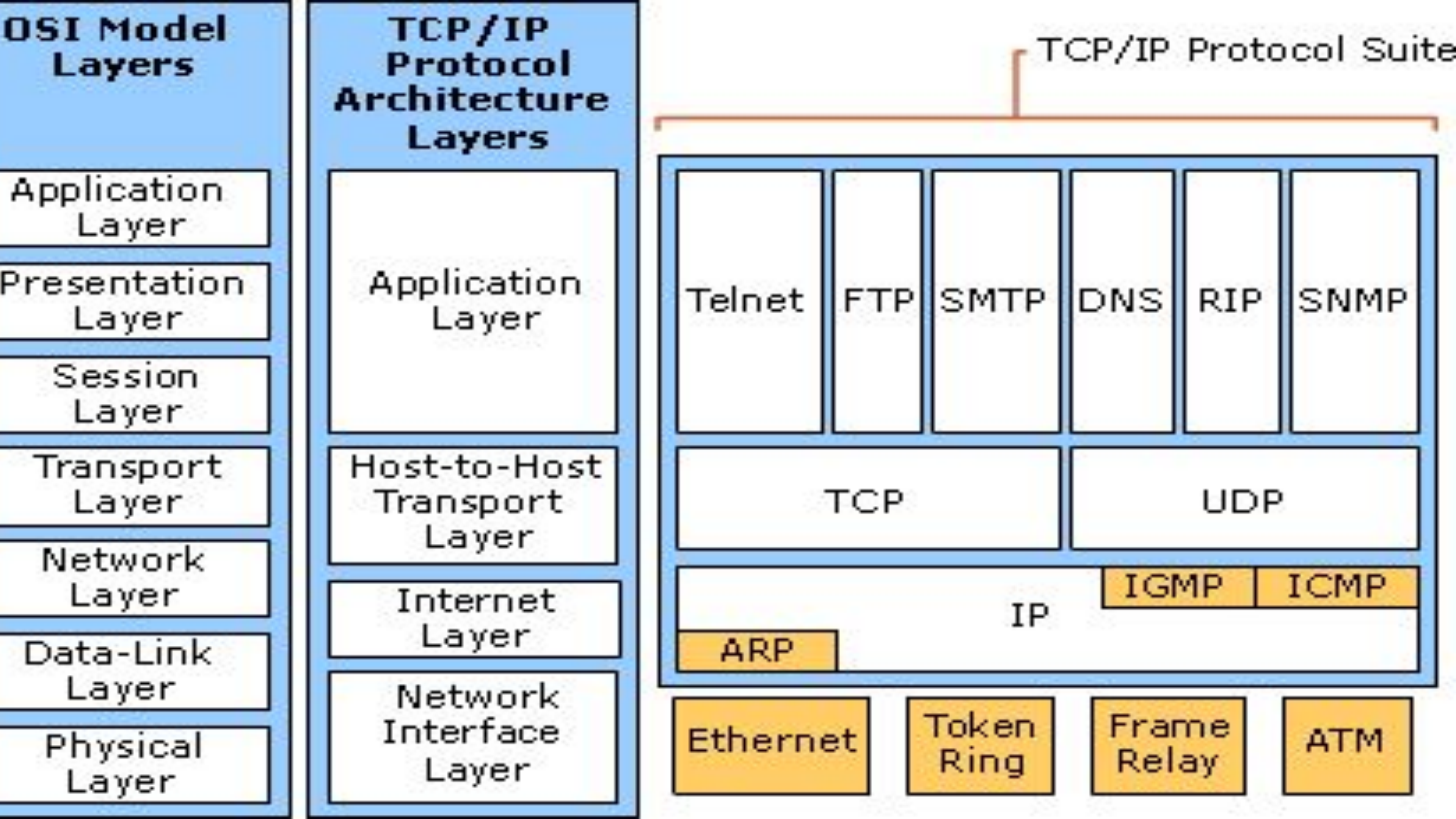
Application Layer

- provide support for user applications
- need a separate module for each type of application

OSI v TCP/IP



OSI	TCP/IP
Application	Application
Presentation	
Session	
Transport	Transport (host-to-host)
Network	Internet
Data Link	Network Access
Physical	Physical



OSI Model Layers

Application Layer

Presentation Layer

Session Layer

Transport Layer

Network Layer

Data-Link Layer

Physical Layer

TCP/IP Protocol Architecture Layers

Application Layer

Host-to-Host Transport Layer

Internet Layer

Network Interface Layer

TCP/IP Protocol Suite

Telnet

FTP

SMTP

DNS

RIP

SNMP

TCP

UDP

IGMP

ICMP

IP

ARP

Ethernet

Token Ring

Frame Relay

ATM

Connection Oriented and Connectionless Services

These are the two services given by the layers to layers.

These services are:

1. Connection Oriented Service
2. Connectionless Services

Connection-Oriented

- A connection-oriented service is a network service that was designed and developed after the telephone system.

- A connection-oriented service is used to create an end to end connection between the sender and the receiver before transmitting the data over the same or different networks.

- In connection-oriented service, packets are transmitted to the receiver in the same order the sender has sent them.

Connectionless Service

- A connection is similar to a postal system, in which each letter takes along different route paths from the source to the destination address.

- Connectionless service is used in the network system to transfer data from one end to another end without creating any connection. So it does not require establishing a connection before sending the data from the sender to the receiver.

- It is not a reliable network service because it does not guarantee the transfer of data packets to the receiver, and data packets can be received in any order to the receiver.

TCP Protocol

TCP is extremely reliable, and is used for everything from surfing the web (HTTP), sending emails (SMTP), and transferring files (FTP).

- TCP is used in situations where it's necessary that all data being sent by one device is received by another completely intact.
- For example, when you visit a website, TCP is used to guarantee that everything from the text, images, and code needed to render the page arrives. Without TCP, images or text could be missing, or arrive in the incorrect order, breaking the page.
- TCP is a connection-oriented protocol, meaning that it establishes a connection between two devices before transferring data, and maintains that connection throughout the transfer process.
- To establish a connection between two devices, TCP uses a method called a three-way

UDP Protocol

- UDP, or User Datagram Protocol, is another one of the major protocols that make up the internet protocol suite. UDP is less reliable than TCP, but is much simpler.
- UDP is used for situations where some data loss is acceptable, like live video/audio, or where speed is a critical factor like online gaming.
- While UDP is similar to TCP in that it's used to send and receive data online, there are a couple of key differences.
- First, UDP is a connectionless protocol, meaning that it does not establish a connection beforehand like TCP does with its three-way handshake.
- Next, UDP doesn't guarantee that all data is successfully transferred. With UDP, data is sent to any device that happens to be listening, but it doesn't care if some of it is lost along the way. This is one of the reasons why UDP is also known as the "fire-and-forget" protocol.
- A good way to think about these differences is that TCP is like a conversation between two people. Person A asks person B to talk. Person B says sure, that's fine. Person A agrees and they both start speaking.
- UDP is more like a protester outside with a megaphone. Everyone who is paying attention to the protester should hear

TCP vs UDP

TCP

UDP

Transmission control protocol (TCP)	User datagram protocol (UDP)
TCP is a connection-oriented protocol. Connection-orientation means that the communicating devices should establish a connection before transmitting data and should close the connection after transmitting the data.	UDP is the Datagram oriented protocol. This is because there is no overhead for opening a connection, maintaining a connection, and terminating a connection. UDP is efficient for broadcast and multicast type of network transmission.
TCP is reliable as it guarantees the delivery of data to the destination router.	The delivery of data to the destination cannot be guaranteed in UDP.
TCP provides extensive error checking mechanisms. It is because it provides flow control and acknowledgement of data.	UDP has only the basic error checking mechanism using checksums.
Acknowledgement segment is present.	No acknowledgement segment.
Sequencing of data is a feature of Transmission Control Protocol (TCP). this means that packets arrive in-order at the receiver.	There is no sequencing of data in UDP. If the order is required, it has to be managed by the application layer.

What are Service Primitives?

service is formally specified by a set of primitives (operations) available to a user process to access the service. These primitives tell the service to perform some action or report on an action taken by a peer entity. If the protocol stack is located in the operating system, as it often is, the primitives are normally system calls. These calls cause a trap to kernel mode, which then turns control of the machine over to the operating system to send the necessary packets. The set of primitives available depends on the nature of the service being provided. The primitives for connection-oriented service are different from those of connection-less service

What are Service Primitives?

There are five types of service primitives :

- **LISTEN :** When a server is ready to accept an incoming connection it executes the LISTEN primitive. It blocks waiting for an incoming connection.
- **CONNECT :** It connects the server by establishing a connection. Response is awaited.
- **RECIEVE:** Then the RECIEVE call blocks the server.
- **SEND** : Then the client executes SEND primitive to transmit its request followed by the execution of RECIEVE to get the reply. Send the message.
- **DISCONNECT** : This primitive is used for terminating the connection. After this primitive one can't send any message. When the client sends DISCONNECT packet then the server

Connection Oriented Service Primitives

LISTEN	Block waiting for an incoming connection
CONNECTION	Establish a connection with a waiting peer
RECEIVE	Block waiting for an incoming message
SEND	Sending a message to the peer
DISCONNECT	Terminate a connection

CONNECTION Establish a connection with a waiting peer

IP Address

An IP address is a unique address that identifies a device on the internet or a local network.

An IP address is a number identifying of a computer or another device on the Internet. It is similar to a mailing address, which identifies where postal mail comes from and where it should be delivered. IP addresses uniquely identify the source and destination of data transmitted with the Internet Protocol.

IPv4 addresses are 32 bits long (four bytes). An example of an IPv4 address is 216.58.216.164.

The range of valid addresses which can be assigned is 0.0.0.0 to

The IP Class System

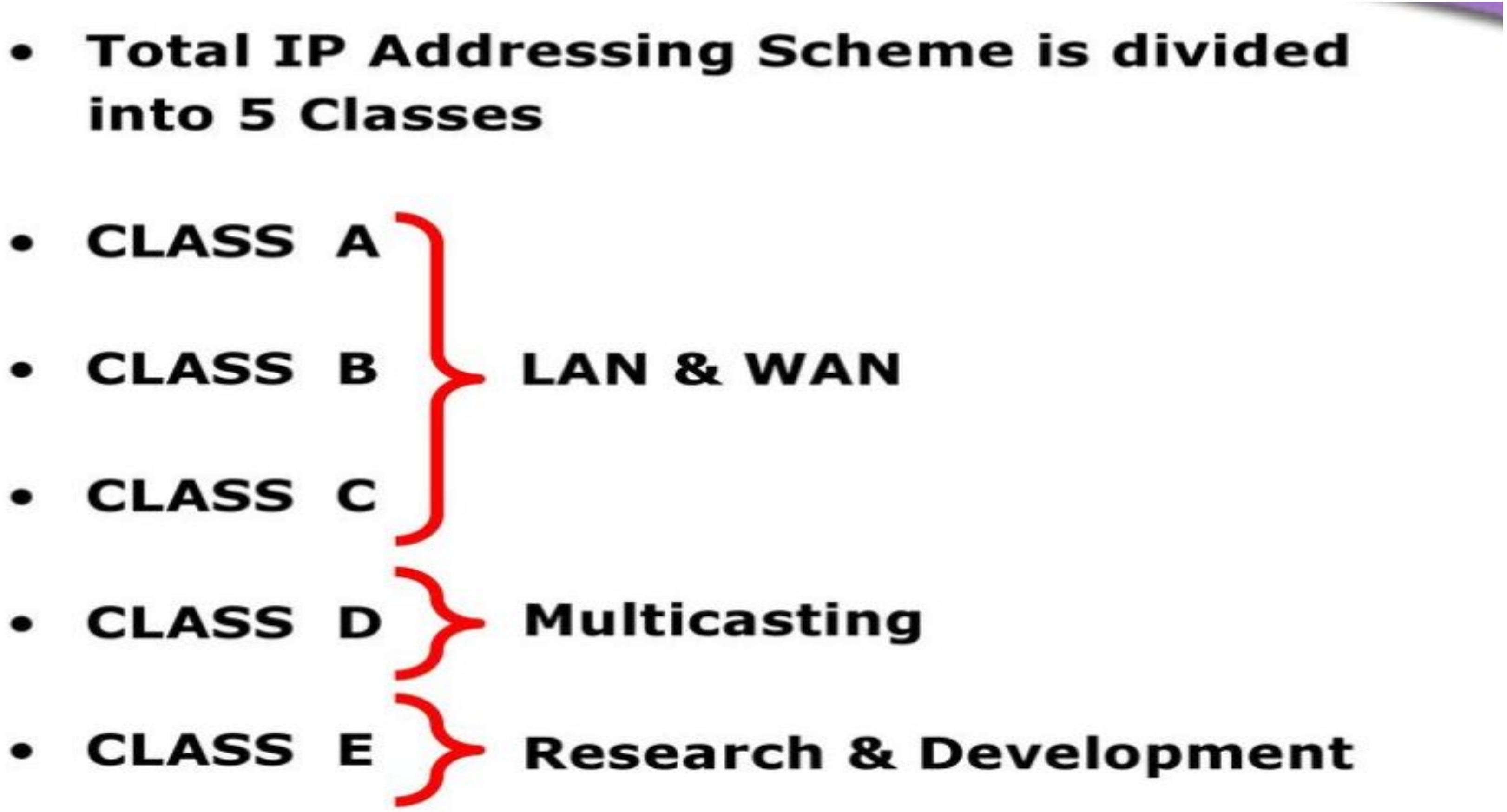
The IPV4 Addresses were broken into 5 classes: A through E.

A, B and C are the primary classes for the addressing, D and E were reserved.

Class D is used for multicasting..

Class E is reserved for future use.

The IP Class System



- Total IP Addressing Scheme is divided into 5 Classes

- CLASS A
- CLASS B
- CLASS C

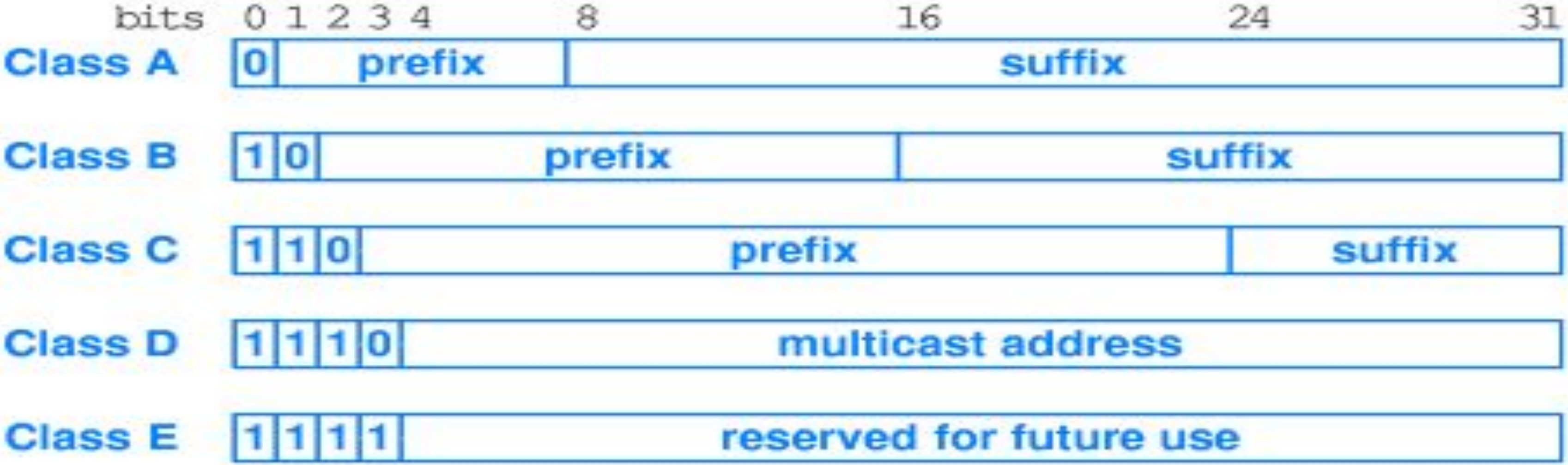
LAN & WAN

- CLASS D

Multicasting

- CLASS E

Research & Development


bits
0 1 2 3 4
8
16
24
31
Class A
0
prefix
suffix
Class B
1 0
prefix
suffix
Class C
1 1 0
prefix
suffix
Class D
1 1 1 0
multicast address
Class E
1 1 1 1
reserved for future use

Class A

Class A is self-identified by the leftmost bit being a 0.

Class A uses the first octet from the left to identify the network and the rest to identity the nodes.

It has 7 bits (first octet minus first bit used to indicate class A) to identify networks, so there can be 128 = 2^7 Class A networks.

It has 24 bits (the last three octets) to identify nodes, so there

Class B

Class B is self-identified by the first two bits being a 10.

Class B uses the first two octets from the left to identify the network and the rest to identity the nodes

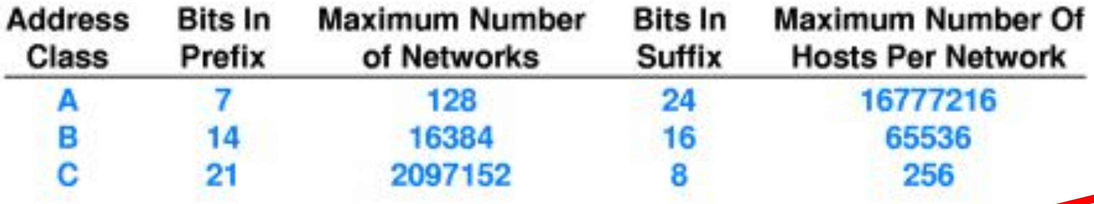
It has 14 bits (first two octet minus first two bits used to

Class C

Class C is self-identified by the first three bits being a 110.

Class C uses the first three octets from the left to identify the network and the remaining one to identity the nodes.

It has 21 bits (first three octet minus first three bits used to



Address Class	Bits In Prefix	Maximum Number of Networks	Bits In Suffix	Maximum Number Of Hosts Per Network
A	7	128	24	16777216
B	14	16384	16	65536
C	21	2097152	8	256

These are all off by 2 because it is neglected by node addresses (suffixes) reserved for the network and broadcasting

Class	Address range	Supports
Class A	1.0.0.1 to 126.255.255.254	Supports 16 million hosts on each of 127 networks.
Class B	128.1.0.1 to 191.255.255.254	Supports 65,000 hosts on each of 16,000 networks.
Class C	192.0.1.1 to 223.255.254.254	Supports 254 hosts on each of 2 million networks.
Class D	224.0.0.0 to 239.255.255.255	Reserved for multicast groups.
Class E	240.0.0.0 to 254.255.255.254	Reserved for future use, or research and development purposes.

Client/Server Model:

Client–server model is a computing model that acts as distributed application which partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters, called clients.

- Often clients and servers communicate over a computer network on separate hardware, but both client and server may reside in the same system.

A server machine is a host that is running one or more server programs which share their resources with clients.

- A client does not share any of its resources, but requests a server's content or

Client/Server Architecture:

- Client server network architecture consists of two kinds of computers: clients and servers.
- Clients are the computers that that do not share any of its resources but requests data and other services from the server computers and server computers provide services to the client computers by responding to client computers requests.
- Normally servers are powerful computers and clients are less powerful personal computers. Web servers are included as part of a larger package of internet and intranet related programs for serving e- mail, downloading requests for FTP files and building and publishing web pages.

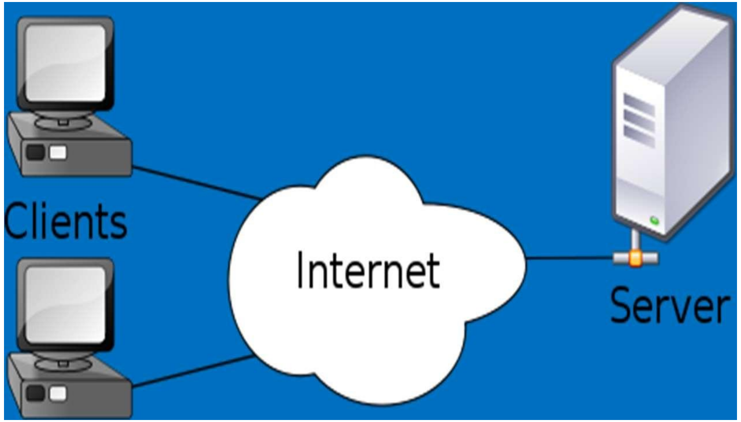
Client Side

- The **client-side** (or simply, client) is the application that runs on the end-user computer; it provides a user-interface (**UI**) that handles what the application feels and looks like and how it interacts with end-user. It may employ and consume resources on the user's machine (computing device) such as temporary and local storage, etc.

Server Side

- The **server-side** (or simply, server) is the application that receives requests from the clients, and contains the logic to send the appropriate data back to the client. Instead of user-interface, the server usually has an application programming interface (**API**). Moreover, the server often includes a **database**, which will persistently store all of the data for the application.

Client/Server Architecture:



Advantages

- The client/ server architecture reduces network traffic by providing a query response to the user rather than transferring total files.
- The client/ server model improves multi-user updating through a graphical user interface (GUI) front end to the shared database.
- Easy to implement security policies, since the data are stored in central location

Disadvantages

- Failure of the server causes whole network to be collapsed.
- Expensive than P2P, Dedicated powerful servers are needed.
- Extra effort are needed for administering and managing the server.

Layerd Architecture

Generally there are three layers:-

- Presentation
- Business
- Data access layer

Layers

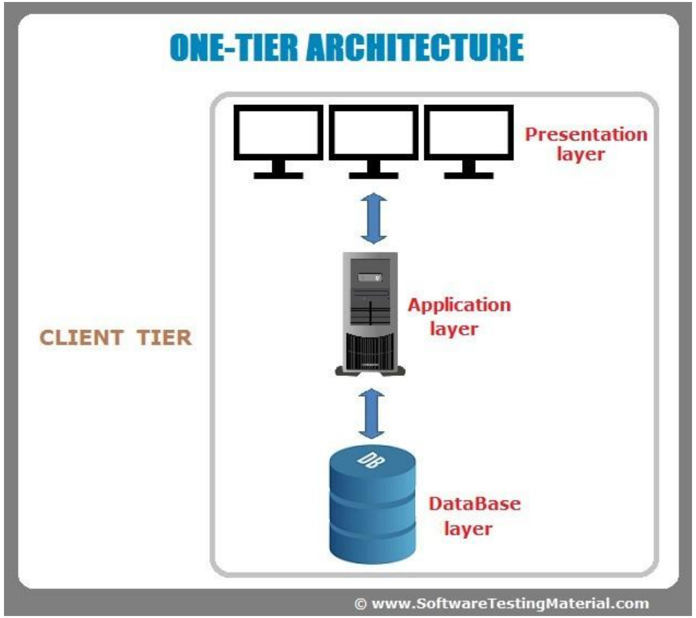
- **Presentation Layer**:- involves with client and application interaction. Provides user friendly interface for the clients.

  Business Layer:- contains the application code or the core functionalities of the application or what the application will perform.

- **Data access Layer:-** involves with the maintaining database and storage of data.

Single Tier or 1-Tier Architecture:

- It is the simplest one as it is equivalent to running the application on the personal computer.
- All of the required components for an application to run are on a single application or server.
- Presentation layer, Business logic layer, and data layer are all located on a single machine.


ONE-TIER ARCHITECTURE
Presentation layer
Application layer
CLIENT TIER
DB
DataBase layer
© www.SoftwareTestingMaterial.com

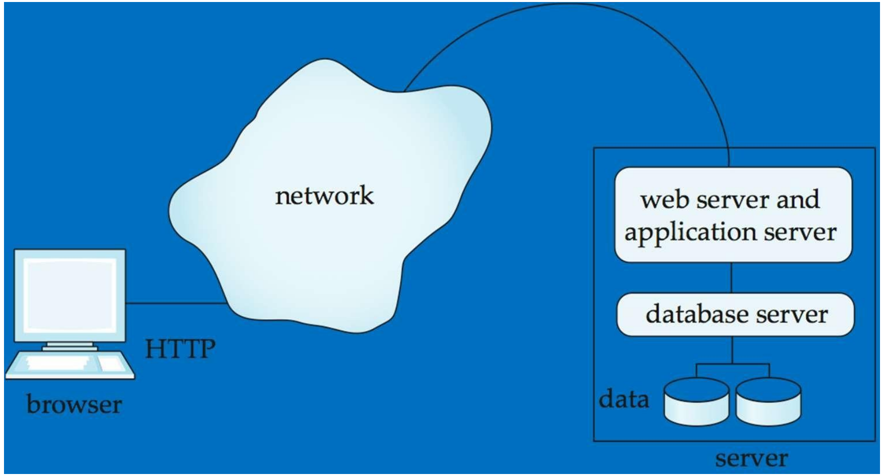
2-Tier Architecture

- In this type of software architecture, the presentation layer or user interface layer runs on the client side while dataset layer gets executed and stored on server side.

- Presentation layer, the business logic layer are separated from the data access layer.
- The advantages of this layer is that the code of the data access layer can be changed any time without affecting the code of the other layer i.e. the whole database and the layer can be changed anytime.

- The database(i.e. the data access layer) can be present anywhere around but the other two layers should be together(tightly connected).

2-Tier Architecture

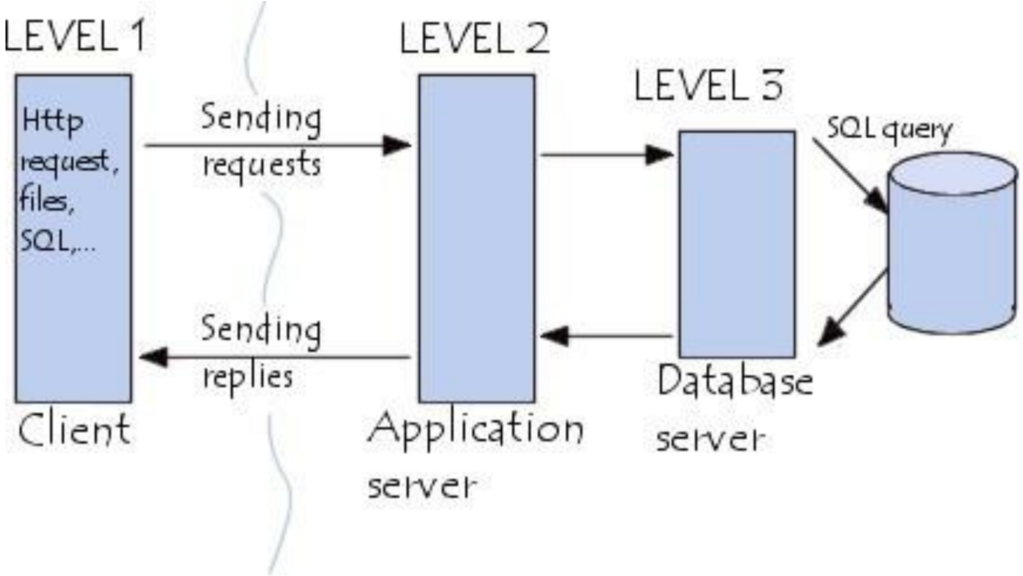


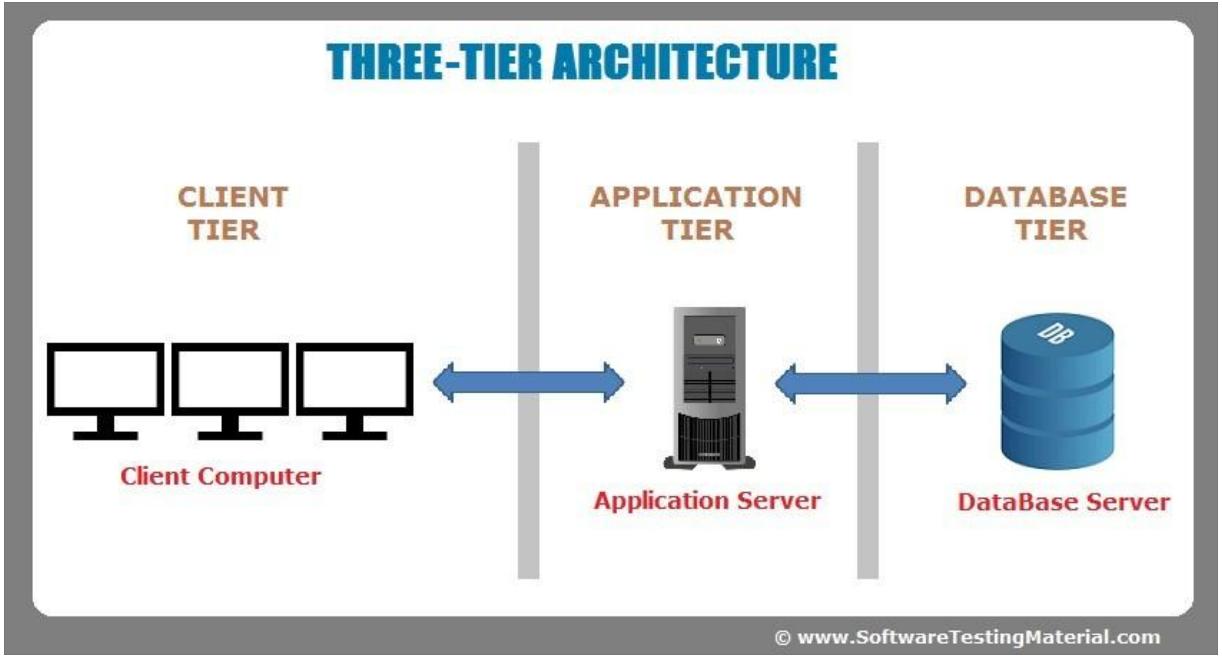
3-Tier Architecture

- In this type of architecture the presentation layer, the business logic layer and the data access layer are separated from each other and are present on three different tiers therefore they are loosely connected.
- The main advantages is that any change in the code in one layer will not affect the other layers and the platform can also be changed independently.
- Now the web designer can concentrate on the design of the user interface i.e. the presentation logic, the application developer concentrate on developing the application i.e. the business logic and the database manager can handle the database independently.

3-Tier Architecture

- In 3-tier architecture, there is an intermediary level, meaning the architecture is generally split up between:
    - A client, i.e. the computer, which requests the resources, equipped with a user interface (usually a web browser) for presentation purposes
    - The application server (also called **middleware**), whose task it is to provide the requested resources, but by calling on another server
    - The data server, which provides the application server with the data it requires
- **Today's application are based on 3-tier architecture which are**


LEVEL 1
LEVEL 2
LEVEL 3
Http request, files, SQL,...
Sending requests
SQL query
Sending replies
Client
Application server
Database server


THREE-TIER ARCHITECTURE
CLIENT TIER
APPLICATION TIER
DATABASE TIER
Client Computer
Application Server
DB
DataBase Server
© www.SoftwareTestingMaterial.com

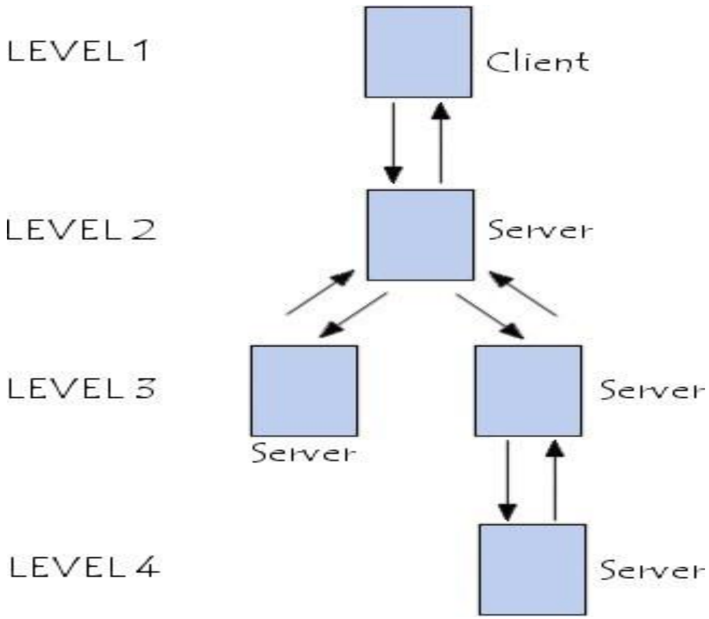
Advantage of 3-Tier Architecture

- Central Database Server accessed by multiple Application Servers
- In turn, each Application Server could independently manage thousands of users
- **Application Servers** can be added to support more users or **DIFFERENT APPLICATIONS**

N-Tier Architecture (Multi-Tier)

N-tier architecture (with N more than 3) is really 3 tier architectures in which the middle tier is split up into new tiers. The application tier is broken down into separate parts. What these parts are differs from system to system. The following picture shows it:

- The primary advantage of N-tier architectures is that they make load balancing possible. Since the application logic is distributed between several servers, processing can then be more evenly distributed among those servers.
- N-tiered architectures are also more easily scalable, since only servers experiencing high demand, such as the application server, need be upgraded. The primary disadvantage of N-tier architectures is that it is also more difficult to program and

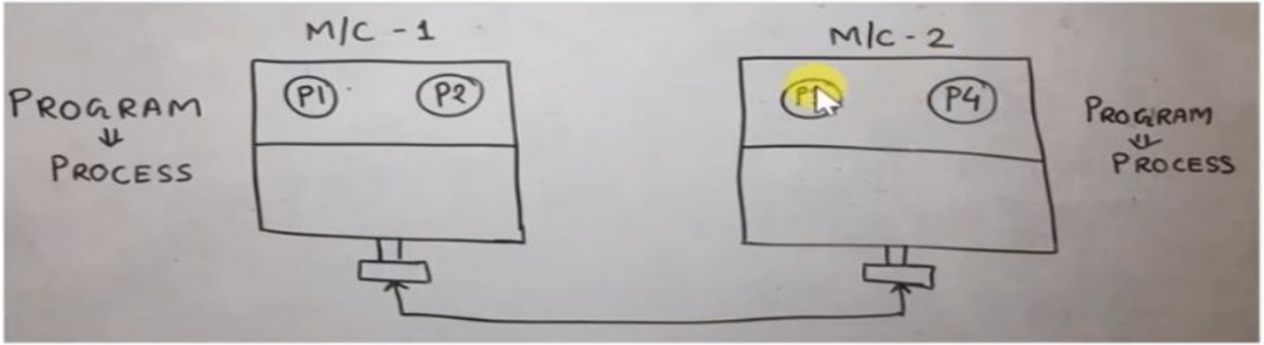

LEVEL 1
Client
LEVEL 2
Server
LEVEL 3
Server
Server
LEVEL 4
Server

Advantages and Disadvantages of Multi-Tier Architectures

Advantages	Disadvantages
· Scalability	· Increase in Effort
· Data Integrity	· Increase in Complexity
· Reusability	
· Reduced Distribution	
· Improved Security	
· Improved Availability	

What is Network Programming

- Developing a program by which one device can communicate with another device.



- A fundamental entity in a computer network is a process.
- A process is a program in execution by the computers operating system.

Computer network programming involves writing computer programs that enable processes to communicate with each other across a computer network.

Network Programming

- The term network programming refers to writing programs that execute across multiple devices in which the device are all connected to each other using network
- Network Programming involves writing programs that communicate with other programs across a computer network.
- As we know that computer Network means a group of computers connect with each other via some medium and transfer data between them as and when require.
- In **network programming** we can make such a program in which the machines connected in network and will send and receive data from other machine in the network by programming.

Network Programming

- The first and simple logic to send or receive any kind of data or message is **we must have the address of receiver or sender.** So when the computer needs to communicate with another computer, it's required the other computer's address.
- Network programming can be done using various other APIs. Most current network programming, however, is done either using sockets directly, or using various other layers on top of sockets (e.g., quite a lot is done over HTTP, which is normally implemented with TCP over sockets). TCP/IP and UDP/IP (as well as a number of other IP-based protocols) are done primarily via the sockets interface.
- **Network socket-** A network socket is a software structure within a network node of a computer network that serves as an endpoint for sending and receiving data across the network.

Examples of Network Programming

- Client Server Application
  - Web Client: Mozilla Firefox, Google Chrome, Safari, Internet Explorer, Opera etc.
  - Web Server: Apache Tomcat, Oracle iPlanet, Tornado etc.
  - Chat Application
  - Email
  - Echo Server

Network Programming Language, Tools & Platforms

Nmap

- Nmap is an open source utility which can quickly scan broad ranges of devices and provide valuable information about the devices on your network.
- Nmap provides utilities to determine what hosts are available on the network, what ports are available on those hosts, the services that are enabled, the operating system and version of the host.

Network Programming Language, Tools & Platforms

Wireshark

- Wireshark is a network packet/protocol analyzer.
- It is tools which enables engineers to quickly get to the packet level of problem.
- It determine if the issue is due to the network, server, service or client.
- Network security engineers use it to examine security problems.
- Wireshark is perhaps one of the best open source packet analyzers available today for UNIX and Windows.

Network Programming Language, Tools & Platforms

iPerf3

- iPerf3 is a tool for active measurements of the maximum achievable bandwidth on IP networks.
- It used to measure packet loss from end to end.
- iperf3 is a commonly used network testing tool that can create TCP and UDP data streams and measure the throughput of a network.
- It is open-source software and runs on various platforms including Linux, Unix and Windows.
- This tools pinpoint whether the network is causing the performance problem or not.

Network Programming Language, Tools & Platforms

Netstat

- It is command line tool that will display the current status of TCP and UDP conversation.
- It is helpful in tracking down server load, mapping connection and monitoring the security of a system that is under attack.

Network Programming Language, Tools & Platforms

Angry IP Scanner

- Angry IP Scanner is a free, lightweight, cross-platform, and open source tool to scan networks.
- It helps you to scan a range of IP addresses to find live hosts, open ports, and other relevant information of each and every IP address.

Thank You!