INTERNET AND WEB TECHNOLOGY

6TH SEM COMPUTER SCIENCE AND ENGINEERING

1. INTERNET FUNDAMENTALS

INTERNET

The internet is the wider network that allows computer networks around the world run by companies, governments, universities and other organisations to talk to one another. The result is a mass of cables, computers, data centres, routers, servers, repeaters, satellites and wi-fi towers that allows digital information to travel around the world.

MOTIVATION FOR INTERNET WORKING

* LAN technologies provide high speed communication across short distances
* WAN technologies serves large areas
* No single networking technology is best for all needs
* Ex: Ethernet might be the best solution for connecting computers in an office
* Ex: Frame relay might be the best solution for interconnecting computers in one city to another

INTERNET ARCHITECRE BOARD

The Internet Architecture Board (IAB) is a board of researchers and professionals that manages the engineering and technical development related to the Internet. IAB offers assistance and insight on a wide range of Internet-related concerns. Professional entities, standards agencies and other organizations frequently use IAB as a reference for network expertise.

IAB manages several task forces, including the Internet Research Task Force (IRTF) and the Internet Engineering Task Force (IETF). IAB was originally established as the Internet Configuration Control Board (ICCB) in 1979. Adopting several names afterwards, it finally became the IAB in 1992. Initially, the U.S. government and Federal Research Internet Configuration Committee Techopedia explains Internet Architecture Board (IAB)

During the 1980s, Internet developments were implemented for the promotion of the Internet and Internet standards. IAB was established to manage oversight of the following responsibilities:

- Manage and publish Request for Comments (RFC)
- Oversee the Internet standard process
- Oversee the IETF

Eventually, IAB responsibilities developed as follows:

- Architectural Supervision: Responsible for overseeing different network and Internet Protocol (IP) architectural standards.
- Appeals and Standards Process Supervision: The appeal board was established to review standard issues and related appeals.
- Advice for Internet Societies: Provides guidance to ISOC officials.

Although IAB arranges groups for developing technical principles and ideas, it generally does not build comprehensive implementation plans. IAB's main objective is to help the IETF improve Internet standardization. The IAB is rarely associated with policy decisions and usually does not address the Internet's operational or commercial elements.

INTERNET PROTOCOL AND STANDARDIZATION

Many of the protocols that make up the TCP/IP protocol suite have been standardized or are in the process of standardization. By universal agreement, an organization known as the *Internet Society* is responsible for the development and publication of these standards. The Internet Society is a professional membership organization that oversees a number of boards and task forces involved in Internet development and standardization. Three organizations under the Internet Society are responsible for the actual work of standards development and publication:

- **Internet Architecture Board (IAB):** Responsible for defining the overall architecture of the Internet, providing guidance and broad direction to the IETF.
- **Internet Engineering Task Force (IETF):** The protocol engineering and development arm of the Internet.
- **Internet Engineering Steering Group (IESG):** Responsible for technical management of IETF activities and the Internet standards process.

The actual development of new standards and protocols for the Internet is carried out by working groups chartered by the IETF. Membership in a working group is voluntary; any

interested party can participate. During the development of a specification, a working group will make a draft version of the document available as an *Internet Draft*, which is placed in the IETF's "Internet-Drafts" online directory. The document may remain as an Internet Draft for up to six months, and interested parties can review and comment on the draft. During that time, the IESG may approve publication of the draft as an *RFC* (Request for Comment). If the draft has not progressed to the status of an RFC during the six-month period, it is withdrawn from the directory. The working group may subsequently publish a revised version of the draft.

The IETF is responsible for publishing the RFCs, with approval of the IESG. The RFCs are the working notes of the Internet research and development community. A document in this series may cover essentially any topic related to computer communications and may be anything from a meeting report to the specification of a standard.

The work of the IETF is divided into eight areas, each with an area director and composed of numerous working groups:

- **General:** IETF processes and procedures. An example is the process for development of Internet standards.
- **Applications:** Internet applications. Examples include Web-related protocols, EDI-Internet integration, LDAP.
- **Internet:** Internet infrastructure. Examples include IPv6, PPP extensions.
- **Operations and management:** Standards and definitions for network operations. Examples include SNMPv3, remote network monitoring.
- **Routing:** Protocols and management for routing information. Examples include multicast routing, OSPF.
- **Security:** Security protocols and technologies. Examples include Kerberos, IPSec, X.509, S/MIME, TLS.
- **Transport:** Transport layer protocols. Examples include differentiated services, IP telephony, NFS, RSVP.
- **User services:** Methods to improve the quality of information available to users of the Internet,. Examples include responsible use of the Internet, user services, FYI documents.

The decision of which RFCs become Internet standards is made by the IESG, on the recommendation of the IETF. To become a standard, a specification must meet the following criteria:

- Be stable and well understood
- Be technically competent
- Have multiple, independent, and interoperable implementations with substantial operational experience
- Enjoy significant public support
- Be recognizably useful in some or all parts of the Internet

ROLES OF ISP

ISP is an acronym that stands for *Internet Service Provider*. An Internet Service Provider is a company that provides Internet access to organizations and home users.

An ISP provides you with Internet access, usually for a fee. Without an ISP, you wouldn't be able to shop online, access Facebook, or read this page. Connecting to the Internet requires specific telecommunications, networking, and routing equipment. ISPs allow users access to networks that contain the required equipment, enabling users to establish Internet connectivity.

ISPs are responsible for making sure you can access the Internet, routing Internet traffic, resolving domain names, and maintaining the network infrastructure that makes Internet access possible.

While the core function of an ISP is to provide Internet access, many ISPs do much more. ISPs also offer services like web hosting, domain name registration, and email services.

FACTORS FOR CHOOSING ISP

Availability

Unfortunately, this is the biggest deciding factor in rural areas. A high-speed cable or fiber connection doesn't do your business any good if the provider doesn't service your area. A surprising number of businesses and homeowners have just a few options, usually one of which is a satellite internet option and either some type of broadband (AT&T U-Verse,

Comcast Xfinity, etc) or 4G-LTE network (which can be surprisingly good with the right equipment and plan).

Speed

As a business, you must ensure that you have sufficient speed to not disrupt daily use, even when demand is at its highest. To some customers, speed is the most important factor when determining an ISP. They simply want the fastest internet they can get in their area. This is completely based on location and what's offered to you as a business or consumer. The number you are looking at when comparing plans is called the "Bandwidth". Bandwidth is simply the volume of information per unit of time that the transmission medium can sustain. Some customers get lucky and have access to fiber connections with of over 1000 Megabits (Mbps) per second while some rural businesses are stuck on 3 to 6Mbps DSL connections. Also, just because the speed is advertised, doesn't mean that's the speed you'll be receiving. It's definitely worth checking with neighbouring businesses to see what kind of speed you can realistically expect.

Cost

In order for an ISP to make sense for you, it needs to have a good balance between speed and price. For example, if you're running a small business out of your home, $1000 a month for a dedicated fiber connection probably won't make sense for you. To some companies, the price doesn't matter as much as the speed & reliability. The same $1000-a-month dedicated fiber connection mentioned above might be a no-brainer for a growing small business with 25 employees. As with most aspects of businesses, you'll need to weigh the pros and cons.

Type of Connection

The type of connection has a big influence on how fast the internet "feels". Satellite internet is notorious for seeming "slow", despite having respectable download speeds (Up to 25Mbps on HughesNet). The reasoning behind this is sheer physics. The signal is sent from your satellite and travels around 22,000 miles out to space. From there, the satellite in orbit contacts a network center to find the requested site. That information is then sent back to the satellite in orbit and then back to you. Even at the speed of light, this process takes almost 500 milliseconds plus any additional processing time for the request, which takes place on

both the server and client side. I know it doesn't sound like much, but adding an extra 1/2 second to every action makes it seem so slow if you're used to a traditional connection. By contrast, even the latency on 4G-LTE signals is around 100 milliseconds versus 400+ milliseconds for satellite connections. Other connections, such as Fiber, offer much lower latency, often under 20 milliseconds.

Reliability

Reliability is likely the biggest factor, especially for business customers. Having unreliable internet is stressful and counterproductive. If you are in an industry that can't risk internet service interruptions, it's wise to consider an ISP that offers a Service Level Agreement (SLA). SLA's are service contracts that specifically state how reliable the connection should be. Customer Services goes hand-in-hand with reliability. Regardless of how good the connection is, something will one day go wrong. Whether it's dying hardware or a physically damaged line, it's extremely likely there will be issues one day. Good customer service is a measure of how fast they can help get you back up and running. Most businesses can't wait several days for new hardware to be shipped to them. They require a higher level of service and a good service provider understands that.

INTERNET SERVICE PROVIDER IN INDIA

- BSNL.
- MTNL. ...
- Bharti Airtel. ...
- Hathway Cable. ...
- Tata Communications. ...
- You Telecom. ...
- Reliance Communications. ...
- Sify **Broadband**.

TYPES OF CONNECTIVITY

There are many ways a personal electronic device can connect to the internet. They all use different hardware and each has a range of connection speeds. As technology changes, faster internet connections are needed to handle those changes.

Dial-Up

Dial-up access is cheap but slow. A modem (internal or external) connects to the Internet after the computer dials a phone number. This analog signal is converted to digital via the modem and sent over a land-line serviced by a public telephone network. Telephone lines are variable in quality and the connection can be poor at times. The lines regularly experience interference and this affects the speed, anywhere from 28K to 56K. Since a computer or other device shares the same line as the telephone, they can't be active at the same time.

Leased

Leased connection is also known as direct Internet access or Level Three connection. It is the secure, dedicated and most expensive, level of Internet connection. With leased connection, your computer is dedicatedly and directly connected to the Internet using high speed transmission lines. It is on-line twenty-four hours a day, seven days a week.

VSAT

Short for very small aperture terminal, an earthbound station used in satellite communications of data, voice and video signals, excluding broadcast television. A VSAT consists of two parts, a transceiver that is placed outdoors in direct line of sight to the satellite and a device that is placed indoors to interface the transceiver with the end user's communications device, such as a PC. The transceiver receives or sends a signal to a satellite transponder in the sky. The satellite sends and receives signals from a ground station computer that acts as a hub for the system. Each end user is interconnected with the hub station via the satellite, forming a star topology. The hub controls the entire operation of the network. For one end user to communicate with another, each transmission has to first go to the hub station that then retransmits it via the satellite to the other end user's VSAT.

Advantages

Satellite communication systems have some advantages that can be exploited for the provision of connectivity. These are:

- Costs Insensitive to Distance
- Single Platform service delivery (one-stop-shop)

• Flexibility

• Upgradeable

• Low incremental costs per unit

Disadvantages

However like all systems there are disadvantages also. Some of these are

• High start-up costs (hubs and basic elements must be in place before the services can be provided)

• Higher than normal risk profiles

• Severe regulatory restrictions imposed by countries that prevent VSAT networks and solutions from reaching critical mass and therefore profitability

• Some service quality limitations such the high signal delays (latency)

• Natural availability limits that cannot be mitigated against

• Lack of skills required in the developing world to design, install and maintain satellite communication systems adequately .

Gateway Access

Gateway Access is also known as Level-One connection. It is the access to the Internet from a network, which is not on the Internet. The gateway allows the two different types of networks to "talk" to each other. But the users of the Gateway Internet have limited access to the Internet. They might not be able to use all the tools available on Internet. The local Internet Service Provider (ISP) normally defines this limitation. Good example of network with Level One connectivity within India is that of VSNL (Videsh Sanchar Nigam Limited). All access to Internet from India are through VSNL gateway.

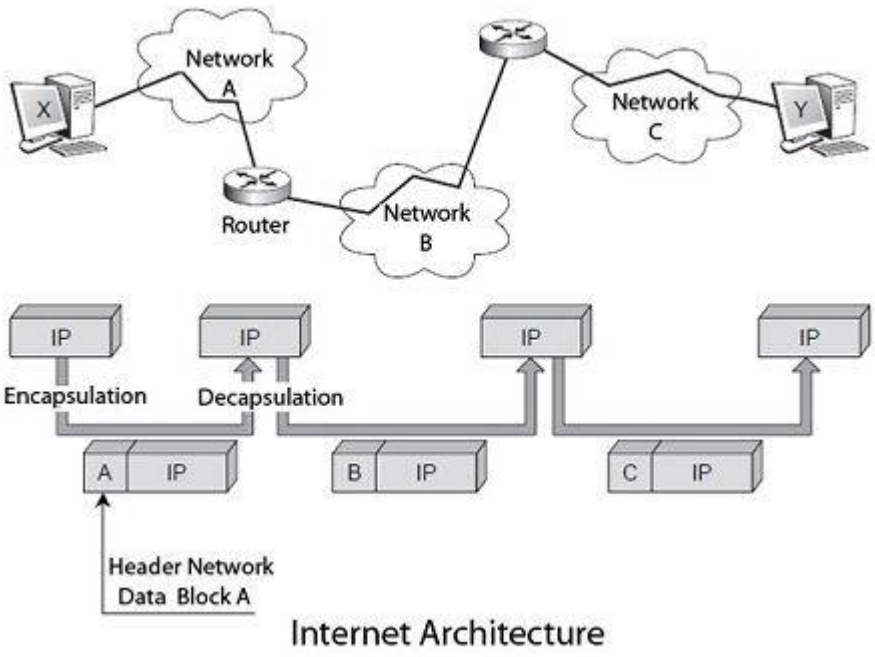
PROPERTIES OF INTERNET

1. its global nature;
2. interactivity;

3. its potential to shift the balance of power in the offline world;
4. accessibility;
5. anonymity;
6. its facilitation of republication;
7. the prominence of intermediaries;
8. its reliance on hyperlinks/hypertext;
9. its long-term impact — the use of permanent archives;
10. its multimedia character; and
11. it's temporal indeterminacy.

INTERNET ARCHITECTURE

The **Internet architecture** is based on a simple idea: ask all networks want to be part of carrying a single packet type, a specific format the IP protocol. In addition, this IP packet must carry an address defined with sufficient generality in order to identify each computer and terminals scattered throughout the world. This architecture is illustrated in Figure.



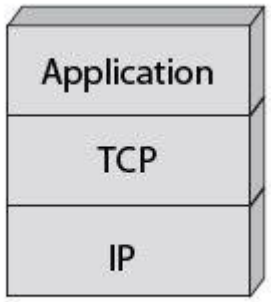
Internet Architecture

The user who wishes to make on this internetwork must store its data in IP packets that are delivered to the first network to cross. This first network encapsulates the IP packet in its own packet structure, the package A, which circulates in this form until an exit door, where it is decapsulated so as to retrieve the IP packet. The IP address is examined to locate, thanks to a

routing algorithm, the next network to cross, and so on until arriving at the destination terminal.

To complete the IP, the US Defence added the TCP protocol; specify the nature of the interface with the user. This protocol further determines how to transform a stream of bytes in an IP packet, while ensuring quality of transport this IP packet. Both protocols, assembled under the TCP / IP abbreviation, are in the form of a layered architecture. They correspond to the packet level and message-level reference model.

The Internet model is completed with a third layer, called the application level, which includes different protocols on which to build Internet services. Email (SMTP), the file transfer (FTP), the transfer of hypermedia pages, transfer of distributed databases (World Wide Web), etc., are some of these services. Figure shows the three layers of the Internet architecture.



The Three Layers of the Internet

IP packets are independent of each other and are individually routed in the network by interconnecting devices subnets, routers. The quality of service offered by IP is very small and offers no detection of lost or possibility of error recovery packages.

TCP combines the functionality of message-level reference model. This is a fairly complex protocol, which has many options for solving all packet loss problems in the lower levels. In particular, a lost fragment can be recovered by retransmission on the stream of bytes. TCP uses a connection-oriented mode.

The flexibility of the Internet architecture can sometimes be a default, to the extent that global optimization of the network is carried out by sub-network subnet, by a succession of local optimizations. This does not allow a homogeneous function in different subnets traversed.

Another important feature of this architecture is to place the entire control system, that is to say, intelligence and control of the network, in the terminal machine leaving virtually nothing in the network, at least in the current version, IPv4, the IP protocol. The control intelligence is in the TCP software on the PC connected to the network.

It is the TCP protocol which takes care of sending more or fewer packets according to network load. Precise control window the maximum number of unacknowledged fragments that may be issued. The TCP window control increases or decreases the traffic following the time required to complete a round trip. Over this time increases, Considering the more congested network, and the transmission rate must decrease to counter saturation. In return, the infrastructure cost is extremely low; no intelligence is not in the network. The service provided by the network of networks corresponds to a quality called best effort, which means that the network does its best to carry the traffic. In other words, the service quality is not assured.

The new generation of IP, IPv6, introduces new features that make the nodes of the network smarter. The new generation of routers comes with QoS management algorithms, which allow them to provide transportation can meet time constraints or packet loss. We expect the arrival of IPv6 for ten years, but it's still IPv4 IP that governs the world. The reason for this is that every new need achievable with IPv6, IPv4 has been able to find the algorithms needed to do as well.

In IPv4, each new customer is treated the same way as those already connected with resources being distributed equitably among all users. The resource allocation policies of telecom operator’s networks are totally different, since, on these networks, a customer who already has a certain quality of service does not suffer any penalty because of the arrival of a new customer. As discussed, the now advocated solution in the Internet environment is to encourage customers with real-time requirements, using appropriate protocols, using priority levels.

The IP protocol for thirty years, but remained almost confidential for twenty years before taking off, unless its properties as a result of the failure of the protocols directly related to the reference model, too many and often incompatible. The IP world growth comes from the simplicity of its protocol, with very few options, and it’s free.

INTERCONNECTION THROUGH IP ROUTERS

Incoming datagrams will be checked to see if the local host is the IP destination host:

yes

The datagram is passed to the higher-level protocols.

no

The datagram is for a different host. The action depends on the value of the *ipforwarding* flag.

true

The datagram is treated as an outgoing datagram and is routed to the *next hop* according to the algorithm described below.

false

The datagram is discarded.

In the internet protocol, outgoing IP datagrams pass through the *IP routing* algorithm which determines where to send the datagram according to the destination IP address.

- If the host has an entry in its *IP routing table* which matches the destination IP address, the datagram is sent to the address in the entry.
- If the network number of the destination IP address is the same as the network number for one of the host's network adapters (that is, the destination and the host are on the same network) the datagram is sent to the physical address of the host matching the destination IP address.
- Otherwise, the datagram is sent to a *default router*.

This base algorithm, needed on all IP implementations, is sufficient to perform the base routing function.

As noted above, a TCP/IP host has basic router functionality included in the IP protocol. Such a router is adequate for simple routing, but not for complex net.

The IP routing mechanism combined with the ``layered'' view of the TCP/IP protocol stack, is represented in Internet Figure Router. This shows an IP datagram, going from one IP address (network number X, host number A) to another (network number Y, host number B), through two physical networks. Note that at the intermediate router, only the lower part of the TCP/IP

protocol stack (the internetwork and the network interface layers) are involved.

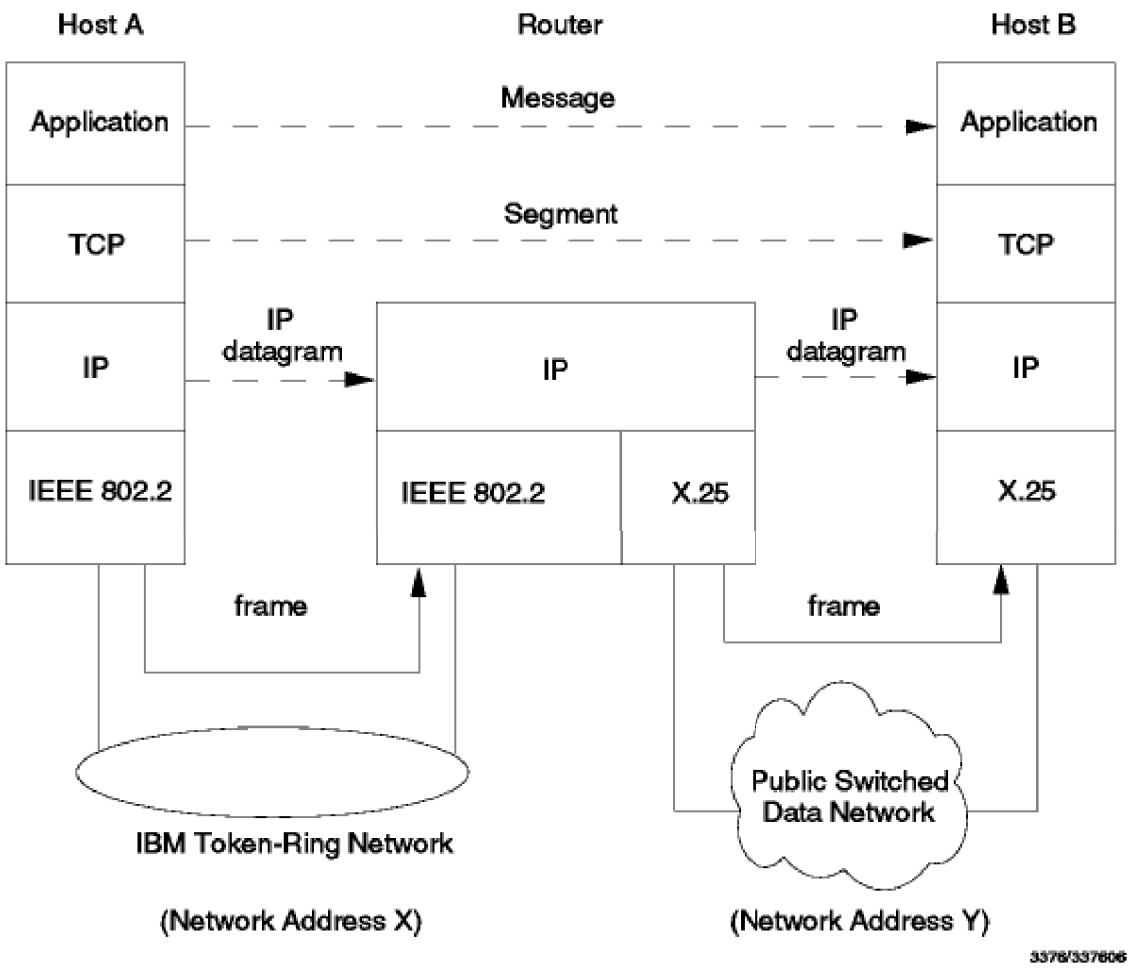


Figure: Internet Router - The router function is performed by the IP protocol.

INTERNET ADDRESS

An **Internet Protocol address** (**IP address**) is a numerical label assigned to each device connected to a computer network that uses the Internet Protocol for communication. An IP address serves two main functions: host or network interface identification and location addressing.

Internet Protocol version 4 (IPv4) defines an IP address as a 32-bit number. However, because of the growth of the Internet and the depletion of available IPv4 addresses, a new version of IP (IPv6), using 128 bits for the IP address, was standardized in 1998. IPv6 deployment has been ongoing since the mid-2000s.

IP addresses are written and displayed in human-readable notations, such as *172.16.254.1* in IPv4, and *2001:db8:0:1234:0:567:8:1* in IPv6. The size of the routing prefix of the address is designated in CIDR notation by suffixing the address with the number of significant bits, e.g., *192.168.1.15/24*, which is equivalent to the historically used subnet mask *255.255.255.0*.

The IP address space is managed globally by the Internet Assigned Numbers Authority (IANA), and by five regional Internet registries (RIRs) responsible in their designated territories for assignment to local Internet registries, such as Internet service providers, and other end users. IPv4 addresses were distributed by IANA to the RIRs in blocks of approximately 16.8 million addresses each, but have been exhausted at the IANA level since 2011. Only one of the RIRs still has a supply for local assignments in Africa. Some IPv4 addresses are reserved for private networks and are not globally unique.

ORIGINAL CLASSFULL ADDRESSING SCHEME

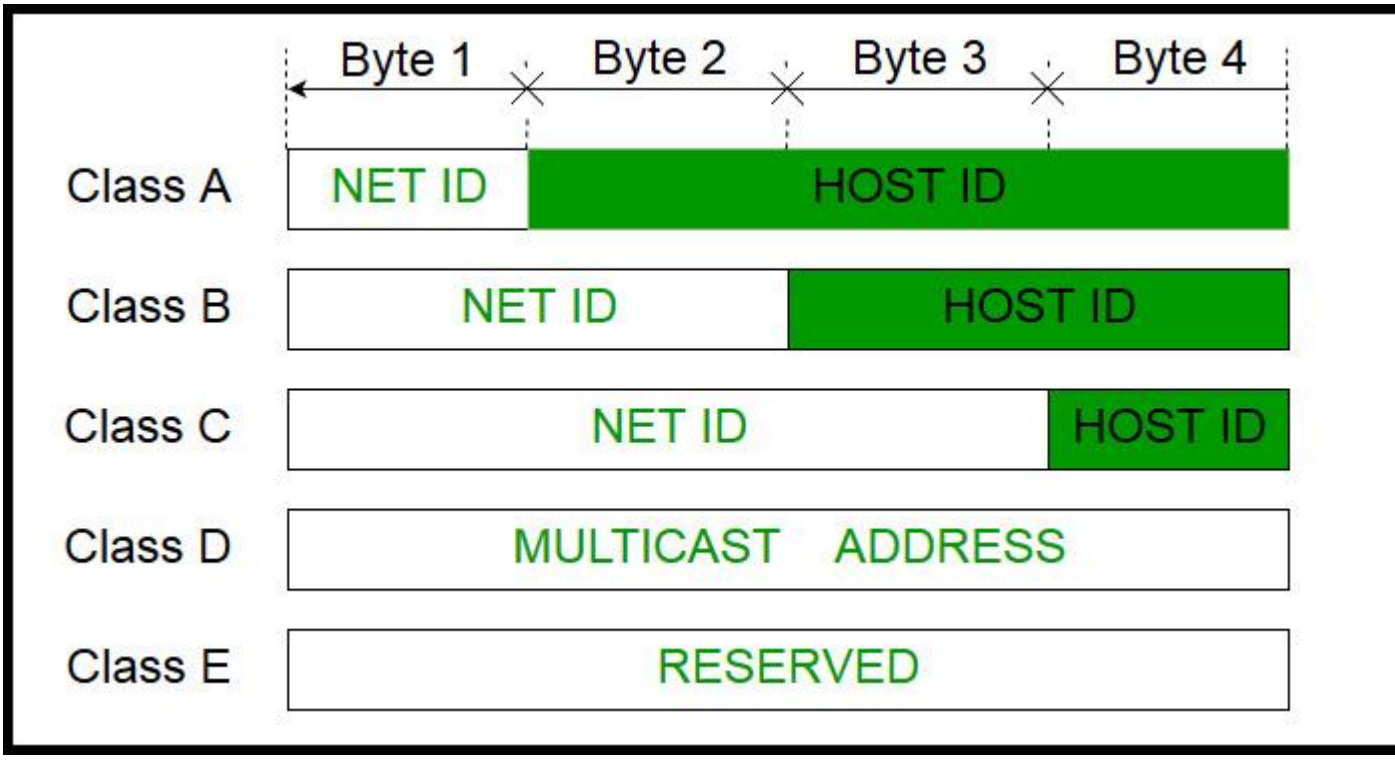
The 32 bit IP address is divided into five sub-classes. These are:

- Class A
- Class B
- Class C
- Class D
- Class E

Each of these classes has a valid range of IP addresses. Classes D and E are reserved for multicast and experimental purposes respectively. The order of bits in the first octet determine the classes of IP address.
IPv4 address is divided into two parts:

- **Network ID**
- **Host ID**

The class of IP address is used to determine the bits used for network ID and host ID and the number of total networks and hosts possible in that particular class. Each ISP or network administrator assigns IP address to each device that is connected to its network.



Note: IP addresses are globally managed by Internet Assigned Numbers Authority(IANA) and regional Internet registries(RIR).

Note: While finding the total number of host IP addresses, 2 IP addresses are not counted and are therefore, decreased from the total count because the first IP address of any network is the network number and whereas the last IP address is reserved for broadcast IP.

Class A:

IP address belonging to class A are assigned to the networks that contain a large number of hosts.

- The network ID is 8 bits long.
- The host ID is 24 bits long.

The higher order bit of the first octet in class A is always set to 0. The remaining 7 bits in first octet are used to determine network ID. The 24 bits of host ID are used to determine the host in any network. The default subnet mask for class A is 255.x.x.x. Therefore, class A has a total of:

- 2^7-2= 126 network ID(Here 2 address is subracted because 0.0.0.0 and 127.x.y.z are special address.)
- 2^24 – 2 = 16,777,214 host ID

IP addresses belonging to class A ranges from 1.x.x.x – 126.x.x.x

Class B:

IP address belonging to class B are assigned to the networks that ranges from medium-sized to large-sized networks.

- The network ID is 16 bits long.
- The host ID is 16 bits long.

The higher order bits of the first octet of IP addresses of class B are always set to 10. The remaining 14 bits are used to determine network ID. The 16 bits of host ID is used to determine the host in any network. The default sub-net mask for class B is 255.255.x.x. Class B has a total of:

- 2^14 = 16384 network address
- 2^16 – 2 = 65534 host address

IP addresses belonging to class B ranges from 128.0.x.x – 191.255.x.x.

Class C:

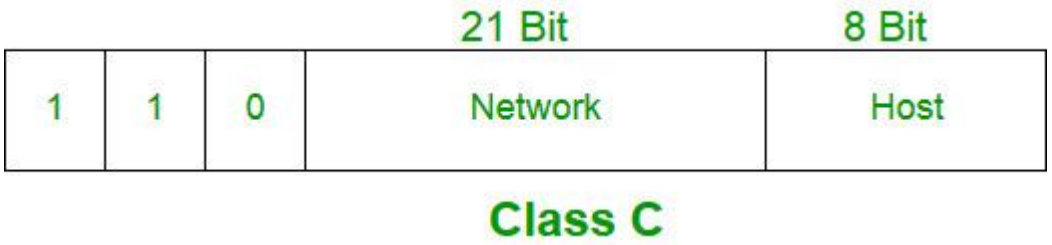
IP address belonging to class C are assigned to small-sized networks.

- The network ID is 24 bits long.
- The host ID is 8 bits long.

The higher order bits of the first octet of IP addresses of class C are always set to 110. The remaining 21 bits are used to determine network ID. The 8 bits of host ID is used to determine the host in any network. The default sub-net mask for class C is 255.255.255.x. Class C has a total of:

- 2^21 = 2097152 network address
- 2^8 – 2 = 254 host address

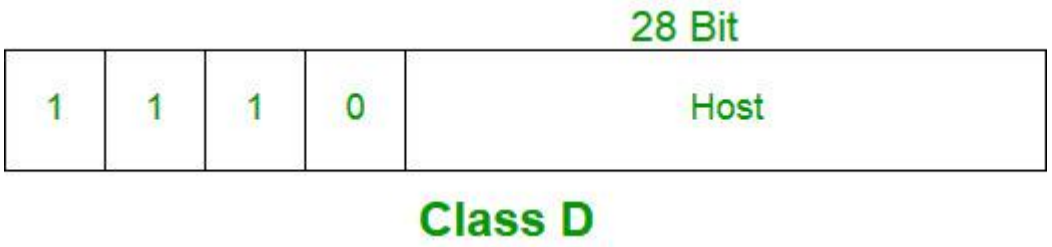
IP addresses belonging to class C ranges from 192.0.0.x – 223.255.255.x.



Class D:

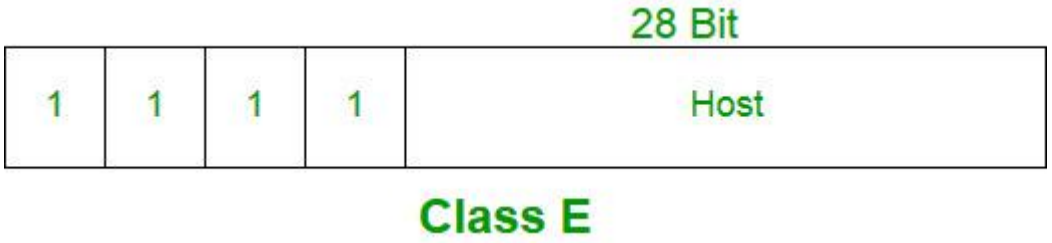
IP address belonging to class D are reserved for multi-casting. The higher order bits of the first octet of IP addresses belonging to class D are always set to 1110. The remaining bits are for the address that interested hosts recognize.

Class D does not posses any sub-net mask. IP addresses belonging to class D ranges from 224.0.0.0 – 239.255.255.255.



Class E:

IP addresses belonging to class E are reserved for experimental and research purposes. IP addresses of class E ranges from 240.0.0.0 – 255.255.255.254. This class doesn't have any sub-net mask. The higher order bits of first octet of class E are always set to 1111.



Range of special IP addresses:

169.254.0.0 – 169.254.0.16 : Link local addresses
127.0.0.0 – 127.0.0.8 : Loop-back addresses
0.0.0.0 – 0.0.0.8 : used to communicate within the current network.

Rules for assigning Host ID:

Host ID's are used to identify a host within a network. The host ID are assigned based on the following rules:

- Within any network, the host ID must be unique to that network.

- Host ID in which all bits are set to 0 cannot be assigned because this host ID is used to represent the network ID of the IP address.
- Host ID in which all bits are set to 1 cannot be assigned because this host ID is reserved as a broadcast address to send packets to all the hosts present on that particular network.

Rules for assigning Network ID:

Hosts that are located on the same physical network are identified by the network ID, as all host on the same physical network is assigned the same network ID. The network ID is assigned based on the following rules:

- The network ID cannot start with 127 because 127 belongs to class A address and is reserved for internal loop-back functions.
- All bits of network ID set to 1 are reserved for use as an IP broadcast address and therefore, cannot be used.
- All bits of network ID set to 0 are used to denote a specific host on the local network and are not routed and therefore, aren't used.

DOTTED DECIMAL NOTATION

Dot-decimal notation is a presentation format for numerical data expressed as a string of decimal numbers each separated by a full stop. For example, the hexadecimal number *0xFF000000* may be expressed in dot-decimal notation as *255.0.0.0*.

In computer networking, the notation is associated with a specific use to represent IPv4 addresses and used as a synonym for *dotted quad notation*, or *quad-dotted notation*.

Object identifiers use a style of dot-decimal notation to represent an arbitrarily deep hierarchy of objects identified by decimal numbers.

INTERNET ASSIGNED NUMBERS AUTHORITY

The Internet Assigned Numbers Authority (IANA) is responsible for maintaining a collection of registries that are critical in ensuring global coordination of the DNS root zone, IP addressing, and other Internet protocol resources. Since 1997, this role has been performed by

ICANN, under a contract awarded by the National Telecommunications and Information Administration (NTIA), an agency in the U.S. Department of Commerce.

The IANA registries fall into three categories, each of which relate to a specific function in the Internet infrastructure:

IP Addresses

The IANA includes the global registry for IPv4 and IPv6 addresses and Autonomous System Numbers (ASNs). These lists contain entries for all the IP address ranges and ASN blocks that are allocated for use on the Internet, as well as the Regional Internet Registry (RIR) to whom responsibility for these resources has been delegated. For instance, the entry for 185.0.0.0/8 points to the RIPE NCC as the responsible registry.

The IANA will make changes to the global IP address registries (such as allocating a block of IP address space to an RIR) according to policies developed and agreed on by the global community.

DNS Root Zone

The Domain Name System (DNS) is a hierarchical distributed database that links domain names such as www.ripe.net to an IP address, which is then used to send data between computers. This can be compared to a phone book.

IANA maintains the top level of this hierarchy, the DNS root zone, which contains pointers to where information about second level domains, such as .com, .net and .nl can be found.

Protocol Parameters

In order to make sure computers understand each other when communicating, certain numbers used in networking protocols need to have a globally unique meaning. These protocol parameters are defined as part of the technical protocol standards produced by the IETF. The IANA maintains and publishes these registries, which can then be used by software makers to ensure stable and predictable communications.

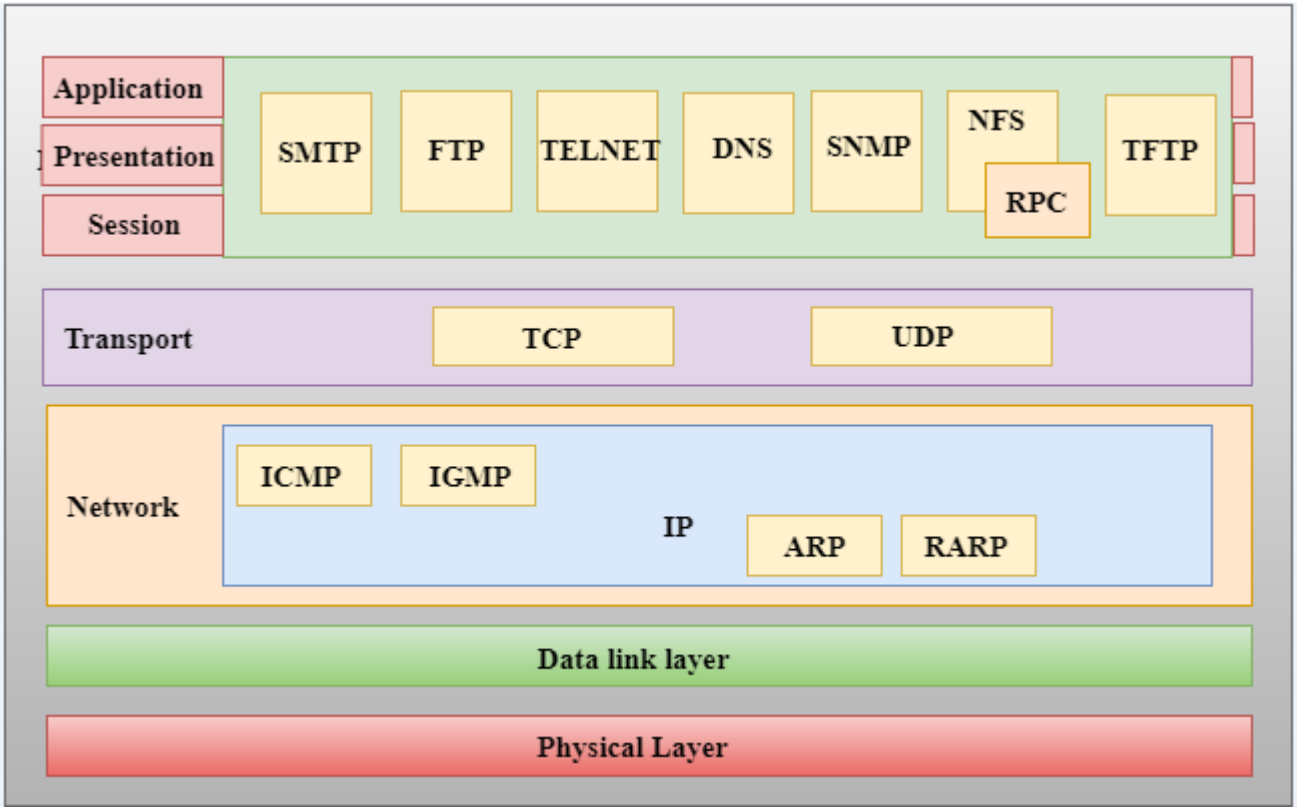
2. TCP/IP

TCP/IP INTERNET LAYERING MODEL

- The TCP/IP model was developed prior to the OSI model.
- The TCP/IP model is not exactly similar to the OSI model.
- The TCP/IP model consists of five layers: the application layer, transport layer, network layer, data link layer and physical layer.
- The first four layers provide physical standards, network interface, internetworking, and transport functions that correspond to the first four layers of the OSI model and these four layers are represented in TCP/IP model by a single layer called the application layer.
- TCP/IP is a hierarchical protocol made up of interactive modules, and each of them provides specific functionality.

Here, hierarchical means that each upper-layer protocol is supported by two or more lower-level protocols.

Functions of TCP/IP layers:



Network Access Layer

- A network layer is the lowest layer of the TCP/IP model.
- A network layer is the combination of the Physical layer and Data Link layer defined in the OSI reference model.
- It defines how the data should be sent physically through the network.
- This layer is mainly responsible for the transmission of the data between two devices on the same network.
- The functions carried out by this layer are encapsulating the IP datagram into frames transmitted by the network and mapping of IP addresses into physical addresses.
- The protocols used by this layer are ethernet, token ring, FDDI, X.25, frame relay.

Internet Layer

- An internet layer is the second layer of the TCP/IP model.
- An internet layer is also known as the network layer.
- The main responsibility of the internet layer is to send the packets from any network, and they arrive at the destination irrespective of the route they take.

Following are the protocols used in this layer are:

IP Protocol: IP protocol is used in this layer, and it is the most significant part of the entire TCP/IP suite.

Following are the responsibilities of this protocol:

- **IP Addressing:** This protocol implements logical host addresses known as IP addresses. The IP addresses are used by the internet and higher layers to identify the device and to provide internetwork routing.
- **Host-to-host communication:** It determines the path through which the data is to be transmitted.
- **Data Encapsulation and Formatting:** An IP protocol accepts the data from the transport layer protocol. An IP protocol ensures that the data is sent and received securely, it encapsulates the data into message known as IP datagram.
- **Fragmentation and Reassembly:** The limit imposed on the size of the IP datagram by data link layer protocol is known as Maximum Transmission unit (MTU). If the

size of IP datagram is greater than the MTU unit, then the IP protocol splits the datagram into smaller units so that they can travel over the local network. Fragmentation can be done by the sender or intermediate router. At the receiver side, all the fragments are reassembled to form an original message.

- **Routing:** When IP datagram is sent over the same local network such as LAN, MAN, WAN, it is known as direct delivery. When source and destination are on the distant network, then the IP datagram is sent indirectly. This can be accomplished by routing the IP datagram through various devices such as routers.

ARP Protocol

- ARP stands for **Address Resolution Protocol**.
- ARP is a network layer protocol which is used to find the physical address from the IP address.
- **The two terms are mainly associated with the ARP Protocol:**
    - **ARP request:** When a sender wants to know the physical address of the device, it broadcasts the ARP request to the network.
    - **ARP reply:** Every device attached to the network will accept the ARP request and process the request, but only recipient recognize the IP address and sends back its physical address in the form of ARP reply. The recipient adds the physical address both to its cache memory and to the datagram header

ICMP Protocol

- **ICMP** stands for Internet Control Message Protocol.
- It is a mechanism used by the hosts or routers to send notifications regarding datagram problems back to the sender.
- A datagram travels from router-to-router until it reaches its destination. If a router is unable to route the data because of some unusual conditions such as disabled links, a device is on fire or network congestion, then the ICMP protocol is used to inform the sender that the datagram is undeliverable.
- An ICMP protocol mainly uses two terms:
    - **ICMP Test:** ICMP Test is used to test whether the destination is reachable or not.

    - **ICMP Reply:** ICMP Reply is used to check whether the destination device is responding or not.
- The core responsibility of the ICMP protocol is to report the problems, not correct them. The responsibility of the correction lies with the sender.
- ICMP can send the messages only to the source, but not to the intermediate routers because the IP datagram carries the addresses of the source and destination but not of the router that it is passed to.

Transport Layer

The transport layer is responsible for the reliability, flow control, and correction of data which is being sent over the network.

The two protocols used in the transport layer are **User Datagram protocol and Transmission control protocol**.

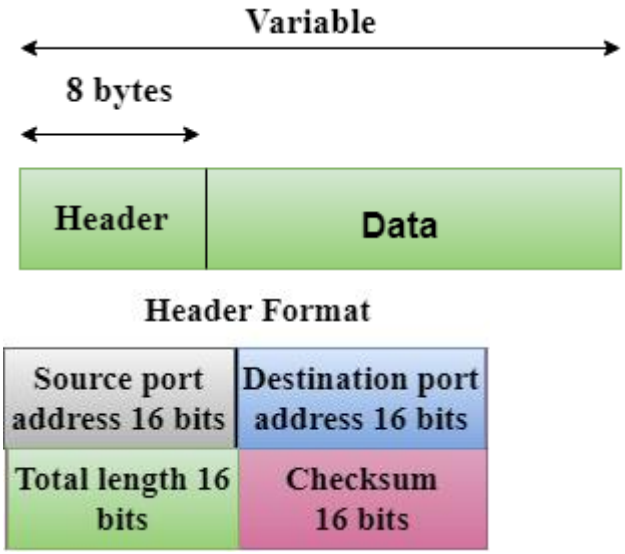
- **User Datagram Protocol (UDP)**
    - It provides connectionless service and end-to-end delivery of transmission.
    - It is an unreliable protocol as it discovers the errors but not specify the error.
    - User Datagram Protocol discovers the error, and ICMP protocol reports the error to the sender that user datagram has been damaged.
    - **UDP consists of the following fields:**

      Source port address: The source port address is the address of the application program that has created the message.

      Destination port address: The destination port address is the address of the application program that receives the message.

      Total length: It defines the total number of bytes of the user datagram in bytes.

      Checksum: The checksum is a 16-bit field used in error detection.
    - UDP does not specify which packet is lost. UDP contains only checksum; it does not contain any ID of a data segment.



- **Transmission Control Protocol (TCP)**
  - It provides a full transport layer services to applications.
  - It creates a virtual circuit between the sender and receiver, and it is active for the duration of the transmission.
  - TCP is a reliable protocol as it detects the error and retransmits the damaged frames. Therefore, it ensures all the segments must be received and acknowledged before the transmission is considered to be completed and a virtual circuit is discarded.
  - At the sending end, TCP divides the whole message into smaller units known as segment, and each segment contains a sequence number which is required for reordering the frames to form an original message.
  - At the receiving end, TCP collects all the segments and reorders them based on sequence numbers.

Application Layer

- An application layer is the topmost layer in the TCP/IP model.
- It is responsible for handling high-level protocols, issues of representation.
- This layer allows the user to interact with the application.
- When one application layer protocol wants to communicate with another application layer, it forwards its data to the transport layer.
- There is an ambiguity occurs in the application layer. Every application cannot be placed inside the application layer except those who interact with the communication

system. For example: text editor cannot be considered in application layer while web browser using **HTTP** protocol to interact with the network where **HTTP** protocol is an application layer protocol.

Following are the main protocols used in the application layer:

- **HTTP:** HTTP stands for Hypertext transfer protocol. This protocol allows us to access the data over the world wide web. It transfers the data in the form of plain text, audio, video. It is known as a Hypertext transfer protocol as it has the efficiency to use in a hypertext environment where there are rapid jumps from one document to another.
- **SNMP:** SNMP stands for Simple Network Management Protocol. It is a framework used for managing the devices on the internet by using the TCP/IP protocol suite.
- **SMTP:** SMTP stands for Simple mail transfer protocol. The TCP/IP protocol that supports the e-mail is known as a Simple mail transfer protocol. This protocol is used to send the data to another e-mail address.
- **DNS:** DNS stands for Domain Name System. An IP address is used to identify the connection of a host to the internet uniquely. But, people prefer to use the names instead of addresses. Therefore, the system that maps the name to the address is known as Domain Name System.
- **TELNET:** It is an abbreviation for Terminal Network. It establishes the connection between the local computer and remote computer in such a way that the local terminal appears to be a terminal at the remote system.
- **FTP:** FTP stands for File Transfer Protocol. FTP is a standard internet protocol used for transmitting the files from one computer to another computer.

RELIABLE STREAM TRANSPORT SERVICE(TCP)

Transmission Control Protocol adds substantial complexity and functionality to ARP, Proxy ARP, and UDP. TCP is not a piece of software it is a communication protocol.

TCP is a connection oriented protocol that requires both endpoints to agree.
TCP provides a full duplex connection between two machines, allowing them to exchange large volumes of data efficiently.
TCP is flexible enough to operate over a large variety of delivery systems.
TCPs basic unit of transfer is a segment.

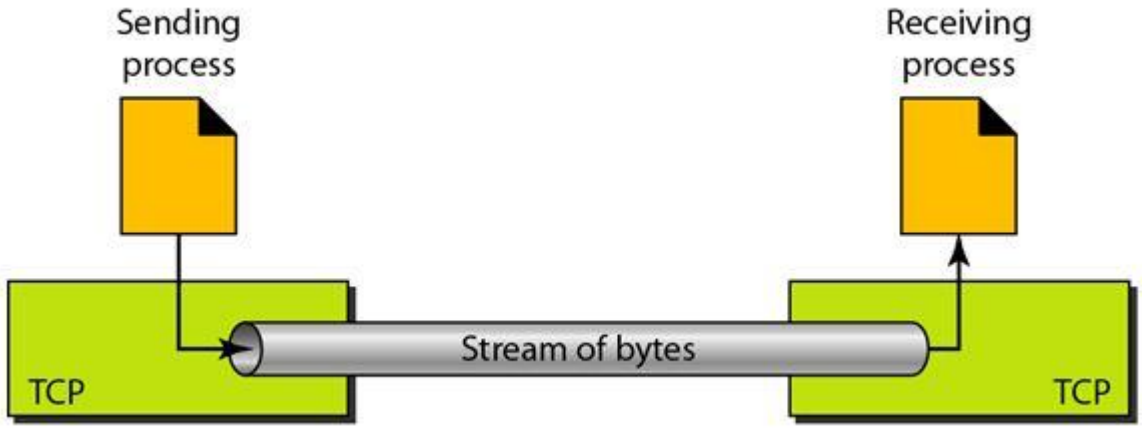
TCP implements flow control and supports out-of-band messages.
TCPs standard specifies exponential back off for retransmission timers and congestion avoidance algorithms.
TCP uses heuristics to avoid transferring small packets.

NEEDS OF STREAM DELIVERY

TCP, unlike UDP, is a stream-oriented protocol. TCP, on the other hand, allows the sending process to deliver data as a stream of bytes and allows the receiving process to obtain data as a stream of bytes. TCP creates an environment in which the two processes seem to be connected by an imaginary "tube" that carries their data across the Internet. This imaginary environment is depicted in the following figure. The sending process produces (writes to) the stream of bytes, and the receiving process consumes (reads from) them.

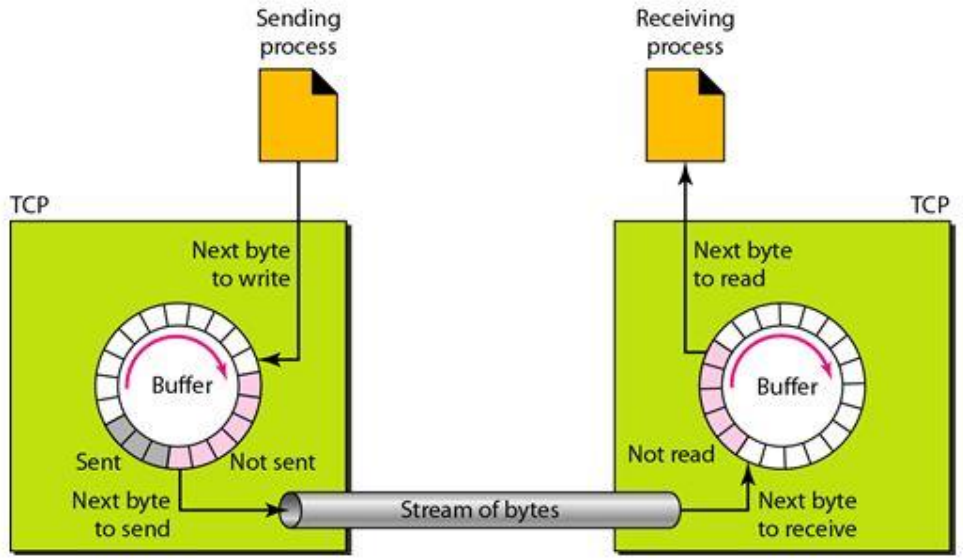
Process-to-Process Delivery Concepts
User Datagram Protocol (UDP)
TCP Segment

How To Create a TCP Connection?



Sending and Receiving Buffers:

Because the sending and the receiving processes may not write or read data at the same speed, TCP needs buffers for storage. There are two buffers, the sending buffer and the receiving buffer, one for each direction and these buffers are also necessary for flow and error control mechanisms used by TCP.) One way to implement a buffer is to use a circular array of 1-byte locations as shown in the following figure.



The above figure shows the movement of the data in one direction.

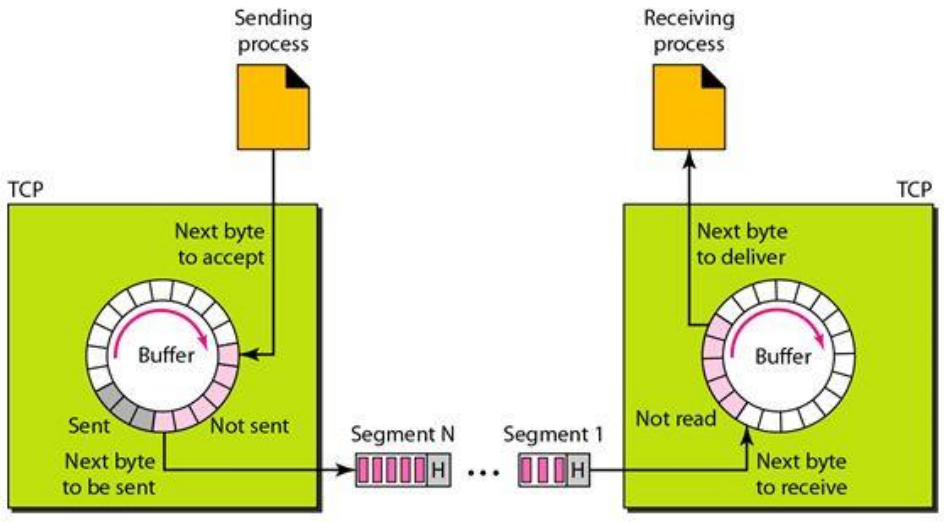
At the sending site, the buffer has three types of chambers. The white section contains empty chambers that can be filled by the sending process (producer). The gray area holds bytes that have been sent but not yet acknowledged. TCP keeps these bytes in the buffer until it receives an acknowledgment. The colored area contains bytes to be sent by the sending TCP. However, TCP may be able to send only part of this colored section. This could be due to the slowness of the receiving process or perhaps to congestion in the network. Also note that after the bytes in the gray chambers are acknowledged, the chambers are recycled and available for use by the sending process. This is why we show a circular buffer.

At the receiving site, the operation of the buffer at the receiver site is simpler. The circular buffer is divided into two areas (shown as white and colored). The white area contains empty chambers to be filled by bytes received from the network. The colored sections contain received bytes that can be read by the receiving process. When a byte is read by the receiving process, the chamber is recycled and added to the pool of empty chambers.

Segments:

The buffering handles the disparity between the speed of the producing and consuming processes, we need one more step before we can send data. The IP layer, as a service provider for TCP, needs to send data in packets, not as a stream of bytes. At the transport layer, TCP groups a number of bytes together into a packet called a segment. TCP adds a header to each segment (for control purposes) and delivers the segment to the IP layer for transmission.

The segments are encapsulated in IP datagrams and transmitted. This entire operation is transparent to the receiving process. The segments may be received out of order, lost, or corrupted and resent. All these are handled by TCP with the receiving process unaware of any activities. The following figure shows how segments are created from the bytes in the buffers.



Full-Duplex Communication:

TCP offers full-duplex service, in which data can flow in both directions at the same time. Each TCP then has a sending and receiving buffer, and segments move in both directions.
Connection-Oriented Service:
TCP, unlike UDP, is a connection-oriented protocol. When a process at site A wants to send and receive data from another process at site B, the following occurs:

1. The two TCPs establish a connection between them.

2. Data are exchanged in both directions.

3. The connection is terminated.

The TCP segment is encapsulated in an IP datagram and can be sent out of order, or lost, or corrupted, and then resent. Each may use a different path to reach the destination. There is no physical connection. TCP creates a stream-oriented environment in which it accepts the responsibility of delivering the bytes in order to the other site.

Reliable Service

TCP is a reliable transport protocol. It uses an acknowledgment mechanism to check the safe and sound arrival of data. We will discuss this feature further in the section on error control.

PROPERTIES OF RELIABLE DELIVERY SERVICES

Stream Orientation: Stream delivery service on destination passes to the receiver exact same sequence of bytes that the sender passes it to the source.

2. Virtual Circuit Connection: Protocol software on both the ends communicate by verifying that the transfer is authorized and both sides are ready. Once all details have been settled, the protocol modules inform the application programs that the connection has been established and that transfer can begin.

3. Buffered transfer : When transferring data, each application uses whatever size pieces it finds convenient, which can be as small as a single octet.

4. Unstructured stream : Application programs using the stream service must understand stream content and agree on stream format before they initiate a connection.

5. Full duplex connection : A full duplex connection consists of two independent streams flowing in opposite directions, with no apparent interaction. The advantage of a full duplex connection is that the underlying protocol software can send control information for one stream back to the source in datagrams carrying data in the opposite direction. Such piggybacking reduces network traffic.

IDEA BEHIND SLIDING WINDOW

Sliding window is a technique for controlling transmitted data packets between two network computers where reliable and sequential delivery of data packets is required, such as when using the Data Link Layer (OSI model) or Transmission Control Protocol (TCP).

In the sliding window technique, each data packet (for most data link layers) and byte (in TCP) includes a unique consecutive sequence number, which is used by the receiving computer to place data in the correct order. The objective of the sliding window technique is to use the sequence numbers to avoid duplicate data and to request missing data.

Sliding window is also known as windowing.

The sliding window technique places varying limits on the number of data packets that are sent before waiting for an acknowledgment signal back from the receiving computer. The number of data packets is called the window size. The limits on window size vary depending on the rate at which the receiving computer can process the data packets, and on the capacity of its buffer.

If the application in the receiving computer processes the data packets at a slower rate than the sending computer is sending them, the acknowledgment signal from the receiving computer will tell the sending computer to decrease the number of packets in the window size in the next transmission, or to temporarily stop transmission to free the buffer. If, on the other hand, the receiving application can process the data packets faster than the sending computer

is sending them, the acknowledgment signal will tell the sending computer to increase the number of packets in the next transmission.

For efficient data packet transmission, the transmitter must not be forced to stop sending for an unnecessarily long time. This will happen if the receiving computer sends an acknowledgment signal to stop and does not send another signal to begin transmitting when its buffer has available space or is empty. Other considerations for efficient data packet transmission include:

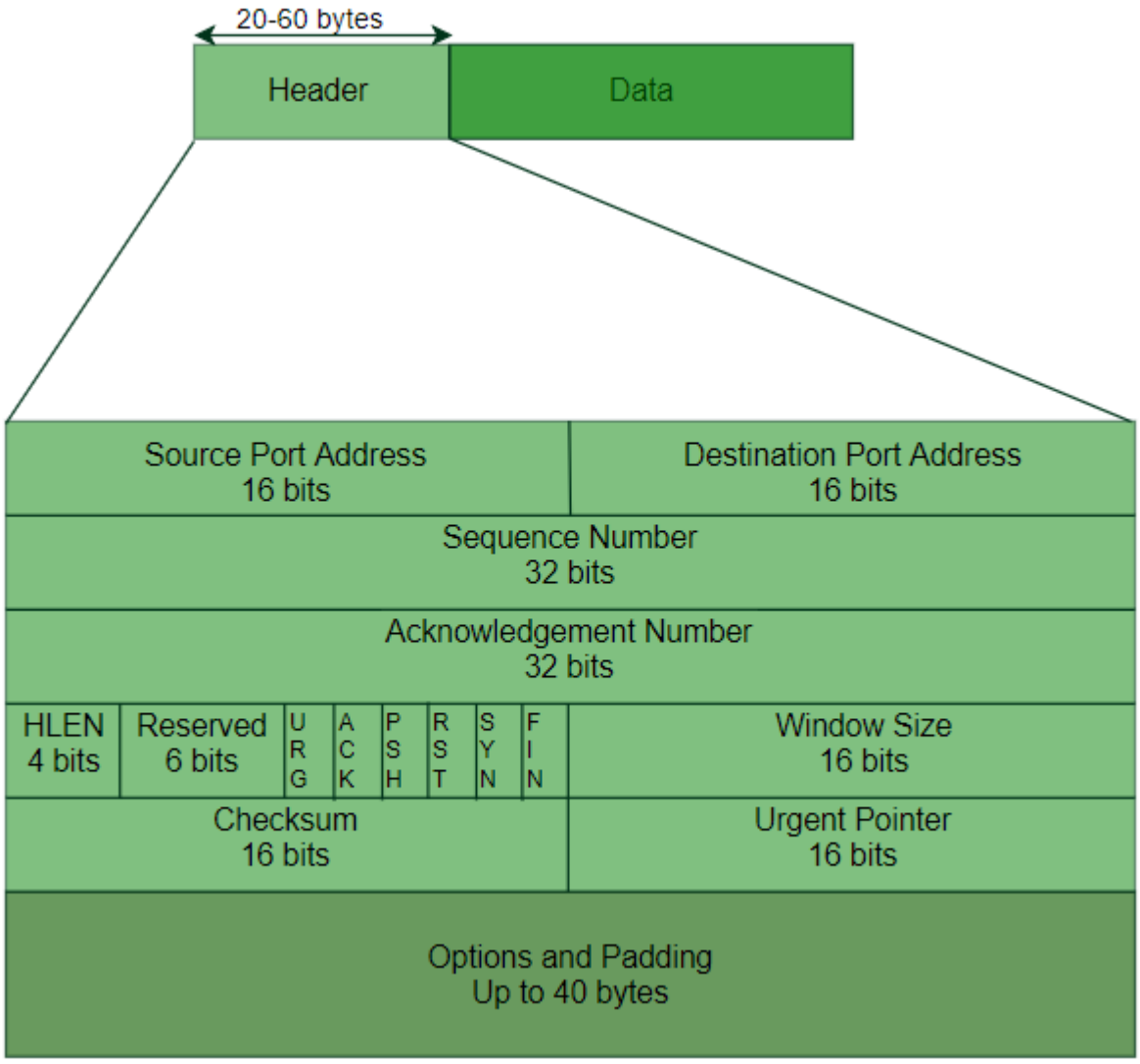
- Round-trip delay time
- End-to-end delay
- Bandwidth delay

PORT CONNECTIONS AND END POINTS

In TCP/IP and UDP networks, a port is an endpoint to a logical connection and the way a client program specifies a specific server program on a computer in a network. The port number identifies what type of port it is. For example, port 80 is used for HTTP traffic. Some ports have numbers that are assigned to them by the IANA, and these are called the ***"well-known ports"*** which are specified in RFC 1700.

TCP SEGMENT FORMAT

TCP segment consists of data bytes to be sent and a header that is added to the data by TCP as shown:



The header of a TCP segment can range from 20-60 bytes. 40 bytes are for options. If there are no options, header is of 20 bytes else it can be of upmost 60 bytes.
Header fields:

- **Source Port Address –**
  16 bit field that holds the port address of the application that is sending the data segment.
- **Destination Port Address –**
  16 bit field that holds the port address of the application in the host that is receiving the data segment.
- **Sequence Number –**
  32 bit field that holds the sequence number, i.e, the byte number of the first byte that

is sent in that particular segment. It is used to reassemble the message at the receiving end if the segments are received out of order.

- **Acknowledgement Number –**
  32 bit field that holds the acknowledgement number, i.e, the byte number that the receiver expects to receive next. It is an acknowledgment for the previous bytes being received successfully.
- **Header Length (HLEN) –**
  This is a 4 bit field that indicates the length of the TCP header by number of 4-byte words in the header, i.e. if the header is of 20 bytes (min length of TCP header), then this field will hold 5 (because 5 x 4 = 20) and the maximum length: 60 bytes, then it'll hold the value 15(because 15 x 4 = 60). Hence, the value of this field is always between 5 and 15.
- **Control flags –**
  These are 6 1-bit control bits that control connection establishment, connection termination, connection abortion, flow control, mode of transfer etc. Their function is:
  - URG: Urgent pointer is valid
  - ACK: Acknowledgement number is valid(used in case of cumulative acknowledgement)
  - PSH: Request for push
  - RST: Reset the connection
  - SYN: Synchronize sequence numbers
  - FIN: Terminate the connection
- **Window size –**
  This field tells the window size of the sending TCP in bytes.
- **Checksum –**
  This field holds the checksum for error control. It is mandatory in TCP as opposed to UDP.
- **Urgent pointer –**
  This field (valid only if the URG control flag is set) is used to point to data that is urgently required that needs to reach the receiving process at the earliest. The value of this field is added to the sequence number to get the byte number of the last urgent byte.

TCP CHECKSUM

TCP includes checksum field in the TCP header to detect the risk of *errors* being introduced into a TCP segment during its travel across the inter-network or in simple words TCP wants to check if the segment got corrupted(intentionally or unintentionally) while segment was on travelling in order to reach the destination.

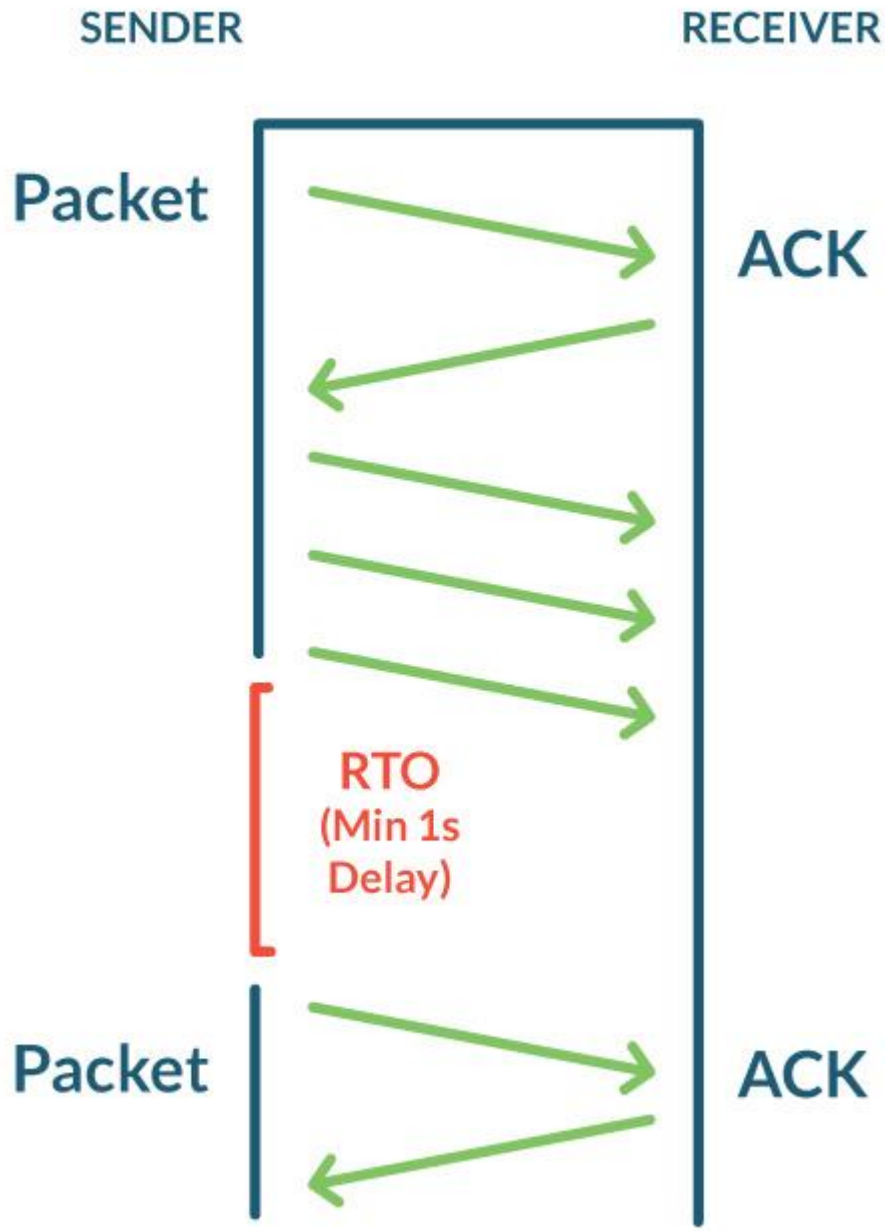
TCP creates a 96-bit header Pseudo header to create the checksum. This Pseudo header is not sent with the packet, it is only used for checksum calculation.

TCP RETRANSMISSION

TCP (the Transmission Control Protocol) connects network devices to the internet. When an outbound segment is handed down to an IP and there's no acknowledgment for the data before TCP's automatic timer expires, the segment is retransmitted. This actually happens all the time, and typically doesn't cause much of a problem: as the retransmission timer counts down, the packets are resent, and the network continues to hum along.

A retransmission timeout (RTO), on the other hand, is quite a different beast. An RTO occurs when the sender is missing too many acknowledgments and decides to take a time out and stop sending altogether. After some amount of time, usually at least one second, the sender cautiously starts sending again, testing the waters with just one packet at first, then two packets, and so on.

As a result, an RTO causes, at minimum, a one-second delay on your network. We've seen sites that show millions of RTOs in a 24-hour window, with one million RTOs translating to 277 hours of application delay. These retransmission timeouts add up to significant problems for network and application performance and certainly require some tuning and optimization.

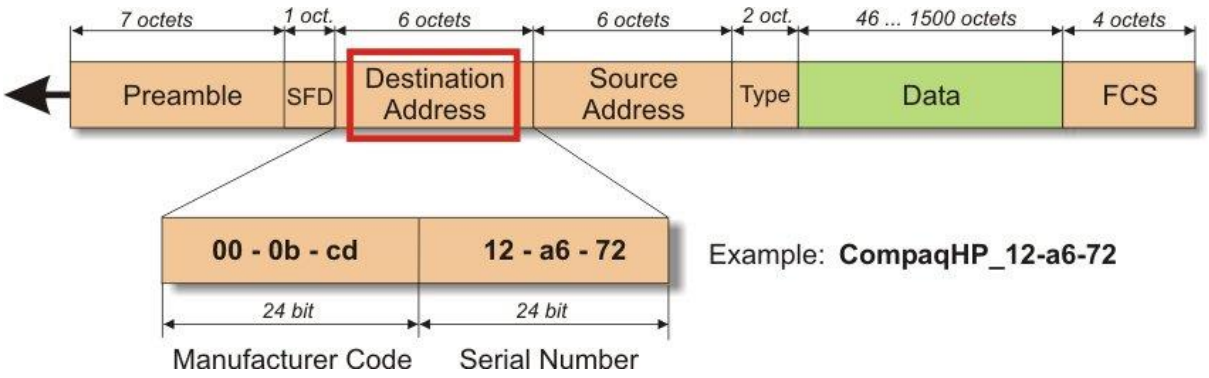


COMMON CAUSES OF RTO'S

- Duplex mismatch on the switch
- A bad cable
- Bad checksums
- Driver issues

DESTINATION ADDRESS AND SOURCE ADDRESS

The address to which a frame or packet of data is sent over a network. The destination address is used by hosts on the network to determine whether the packet or frame is intended for them or for other hosts. The destination address is also used by routers to determine how to forward the packet or frame through an internetwork.



The destination address can be one of the following:

- The physical address, such as the MAC address of an Ethernet frame
- The logical address, such as the IP address of an IP packet

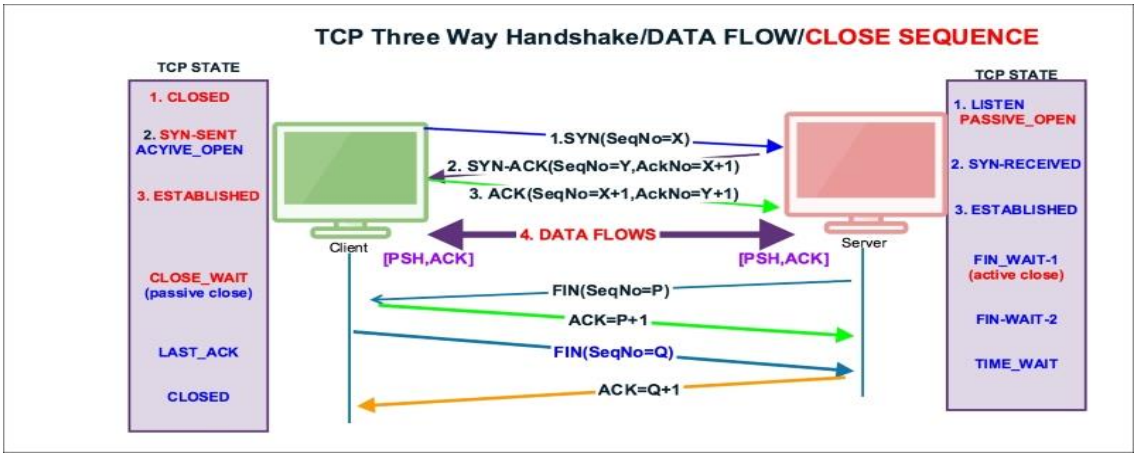
Destination addresses can be either specific or general. Specific addresses point to a specific host on the network. A general address points the packet or frame to all hosts on the network or multicasts it to a specific multicast group of hosts on the network.

The other kind of address in a packet or frame is the source address. This is the address of the host from which the packet originates (unless the source address is being spoofed).

CLOSING CONNECTION OF TCP

There are three ways a TCP connection is closed:

1. The **client** initiates **closing** the **connection** by sending a FIN packet to the server.
2. The server initiates **closing** the **connection** by sending a FIN packet to the **client**.
3. Both **client** and server initiate **closing** the **connection**.



TCP RESEST

In a stream of packets of a TCP connection, each packet contains a TCP header. Each of these headers contains a bit known as the "reset" (RST) flag. In most packets this bit is set to 0 and has no effect; however, if this bit is set to 1, it indicates to the receiving computer that the computer should immediately stop using the TCP connection; it should not send any more packets using the connection's identifying numbers, called ports, and discard any further packets it receives with headers indicating they belong to that connection. A TCP reset basically kills a TCP connection instantly.

When used as designed, this can be a useful tool. One common application is the scenario where a computer (computer A) crashes while a TCP connection is in progress. The computer on the other end (computer B) will continue to send TCP packets since it does not know that computer A has crashed. When computer A reboots, it will then receive packets from the old pre-crash connection. Computer A has no context for these packets and no way of knowing what to do with them, so it might send a TCP reset to computer B. This reset lets computer B know that the connection is no longer working. The user on computer B can now try another connection or take other action.

3.INTERNET PROTOCOL

CONNECTIONLESS DATAGRAM DELIVERY

Connectionless protocols behave in a manner similar to sending a letter in the marl. Let's say I write you a letter, put it in an envelope, address it, add postage, and drop it in a mailbox. What happens? On a best-effort basis the postal office routes the letter through their system and delivers it to you. However, notice that there is no absolute guarantee of delivery; there is no notification if the letter is lost or mangled in transit. Further, there is no assurance that letters will be delivered in the order in which they were sent. The nice thing about this mode of exchange is that you do not need any pre-established relationships in order to communicate.

Connectionless protocols operate in this manner. One casts a datagram onto the network with the understanding that it will be delivered on a best-effort basis to whomever it is addressed to. In addition, we accept that there is no notification of a failure, nor can we make assumptions about the sequence of delivery. UDP is a great example of this sort of communication.

CONCEPT OF UNRELIABLE DELIVERY

Unreliable protocols make no effort to set up a connection, they don't check to see if the data was received and usually don't make any provisions for recovering from errors or lost data. Unreliable protocols work best over physical medium with low loss and low error rates. User Datagram Protocol (UDP) is an example of an unreliable protocol. UDP makes no provisions for verifying whether data arrived or is intact. However, UDP adds a minimum of overhead when compared to TCP and is thus much faster for data transfers over high quality physical links that are high speed and exhibit little or no errors in communication.

CONNECTIONLESS DELIVERY SYSTEM

A Connectionless delivery is a data communication between two nodes where the sender sends data without ensuring whether the receiver is available to receive the data. Here, each data packet has the destination address and is routed independently irrespective of the other packets. Thus the data packets may follow different paths to reach the destination. There's no need to setup connection before sending a message and relinquish it after the message has been sent. The data packets in a connectionless service are usually called datagrams.

Protocols for connectionless services are −

- Internet Protocol (IP)
- User Datagram Protocol (UDP)
- Internet Control Message Protocol (ICMP)

Connectionless services may be of the following types:

- A datagram with Acknowledgement: e.g. text messages with delivery report
- Request-Reply: e.g. queries from remote databases

Advantages of Connectionless Services

- It has low overhead.
- It enables to broadcast and multicast messages, where the sender sends messages to multiple recipients.
- It is simpler and has low overhead.
- It does not require any time for circuit setup.
- In case of router failures or network congestions, the data packets are routed through alternate paths. Hence, communication is not disrupted.

Disadvantages of Connectionless Services

- It is not a reliable connection. It does not guarantee that there will not be a loss of packets, wrong delivery, out – of – sequence delivery or duplication of packets.
- Each data packet requires longer data fields since it should hold all the destination address and the routing information.
- They are prone to network congestions.

PURPOSES OF IP PROTOCOL

The **Internet Protocol** (**IP**) is the principal communications protocol in the Internet protocol suite for relaying datagrams across network boundaries. Its routing function enables internetworking, and essentially establishes the Internet.

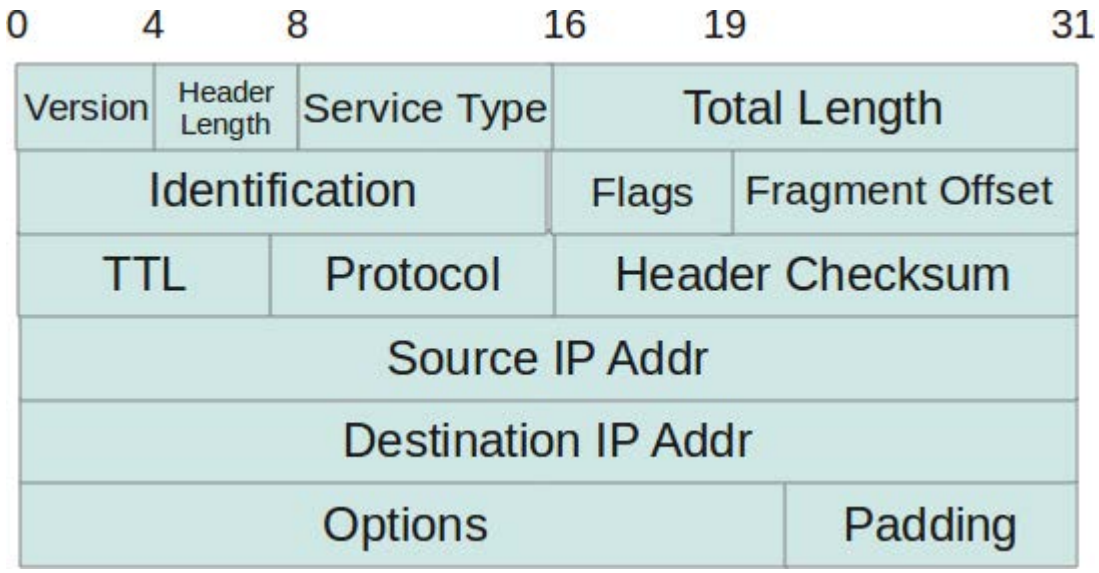
IP has the task of delivering packets from the source host to the destination host solely based on the IP addresses in the packet headers. For this purpose, IP defines packet structures that

encapsulate the data to be delivered. It also defines addressing methods that are used to label the datagram with source and destination information.

Historically, IP was the connectionless datagram service in the original Transmission Control Program introduced by Vint Cerf and Bob Kahn in 1974, which was complemented by a connection-oriented service that became the basis for the Transmission Control Protocol (TCP). The Internet protocol suite is therefore often referred to as *TCP/IP*.

The first major version of IP, Internet Protocol Version 4 (IPv4), is the dominant protocol of the Internet. Its successor is Internet Protocol Version 6 (IPv6), which has been in increasing deployment on the public Internet since c. 2006.

IP HEADER



- **Protocol Version(4 bits)** : This is the first field in the protocol header. This field occupies 4 bits. This signifies the current IP protocol version being used. Most common version of IP protocol being used is version 4 while version 6 is out in market and fast gaining popularity.
- **Header Length(4 bits)** : This field provides the length of the IP header. The length of the header is represented in 32 bit words. This length also includes IP options (if any). Since this field is of 4 bits so the maximum header length allowed is 60 bytes. Usually when no options are present then the value of this field is 5. Here 5 means five 32 bit words ie 5 *4 = 20 bytes.
- **Type of service(8 bits)** : The first three bits of this field are known as precedence bits and are ignored as of today. The next 4 bits represent type of service and the last bit is left unused.

The 4 bits that represent TOS are : minimize delay, maximize throughput, maximize reliability and minimize monetary cost.

 Total length(16 bits): This represents the total IP datagram length in bytes. Since the header length (described above) gives the length of header and this field gives total length so the length of data and its starting point can easily be calculated using these two fields. Since this is a 16 bit field and it represents length of IP datagram so the maximum size of IP datagram can be 65535 bytes. When IP fragmentation takes place over the network then value of this field also changes. There are cases when IP datagrams are very small in length but some data links like ethernet pad these small frames to be of a minimum length ie 46 bytes. So to know the exact length of IP header in case of ethernet padding this field comes in handy.

 Identification(16 bits): This field is used for uniquely identifying the IP datagrams. This value is incremented every-time an IP datagram is sent from source to the destination. This field comes in handy while reassembly of fragmented IP data grams.

 Flags(3 bits): This field comprises of three bits. While the first bit is kept reserved as of now, the next two bits have their own importance. The second bit represents the 'Don't Fragment' bit. When this bit is set then IP datagram is never fragmented, rather its thrown away if a requirement for fragment arises. The third bit represents the 'More Fragment' bit. If this bit is set then it represents a fragmented IP datagram that has more fragments after it. In case of last fragment of an IP datagram this bit is not set signifying that this is the last fragment of a particular IP datagram.

 Fragment offset(13 bits): In case of fragmented IP data grams, this field contains the offset(in terms of 8 bytes units) from the start of IP datagram. So again, this field is used in reassembly of fragmented IP datagrams.

 Time to live(8 bits) : This value represents number of hops that the IP datagram will go through before being discarded. The value of this field in the beginning is set to be around 32 or 64 (lets say) but at every hop over the network this field is decremented by one. When this field becomes zero, the data gram is discarded. So, we see that this field literally means the effective lifetime for a datagram on network.

 Protocol(8 bits) : This field represents the transport layer protocol that handed over data to IP layer. This field comes in handy when the data is demultiplexed at the destination as in that case IP would need to know which protocol to hand over the data to.

 Header Checksum(16 bits) : This field represents a value that is calculated using an algorithm covering all the fields in header (assuming this very field to be zero). This value is

calculated and stored in header when IP data gram is sent from source to destination and at the destination side this checksum is again calculated and verified against the checksum present in header. If the value is same then the datagram was not corrupted else its assumed that data gram was received corrupted. So this field is used to check the integrity of an IP datagram.

- **Source and destination IP(32 bits each)** : These fields store the source and destination address respectively. Since size of these fields is 32 bits each so an IP address os maximum length of 32 bits can be used. So we see that this limits the number of IP addresses that can be used. To counter this problem, IP V6 has been introduced which increases this capacity.
- **Options(Variable length)** : This field represents a list of options that are active for a particular IP datagram. This is an optional field that could be or could not be present

PROTOCOL NUMBER

The numeric identification of the upper layer protocol that an IP packet should be sent to. The number is stored in the header that is prefixed to an IP packet. Note that the IP protocol number is not the same as the port number (see TCP/IP port), which refers to a higher level, such as the application layer.

ROUTING IN AN INTERNET

Internet routing is the process of transmitting and routing IP packets over the Internet between two or more nodes.

It is the same as standard routing procedures but incorporates packet routing techniques and processes on external networks or those that are hosted or Internet enabled. It utilizes IP-based networks, but mainly those which are publicly accessible such as that of ISPs.

Internet routers route packets from internal networks to external Internet-based routers. Internet routing enables a user to access web pages and other data stored on a remote website. Internet routing involves broadcasting or sending a message from an internal network to an external network using Internet-based networks. Such routing generally involves sending a message that travels between several Internet service providers (ISP) or autonomous systems (AS) before reaching the destination.

DIRECT AND INDIRECT DELIVERY

- **Direct Deliveries:** When datagrams are sent between two devices on the same physical network, it is possible for datagrams to be delivered directly from the source to the destination. Imagine that you want to deliver a letter to a neighbor on your street. You probably wouldn't bother mailing it through the post office; you'd just put the neighbor's name on the envelope and stick it right into his or her mailbox.

- **Indirect Deliveries:** When two devices are not on the same physical network, the delivery of datagrams from one to the other is *indirect*. Since the source device can't see the destination on its local network, it must send the datagram through one or more intermediate devices to deliver it. Indirect delivery is analogous to mailing a letter to a friend in a different city. You don't deliver it yourself—you put it into the postal system. The letter journeys through postal system, possibly taking several intermediate steps, and ends up in your friend's neighborhood, where a postal carrier puts it into his or her mailbox.

TABLE DRIVEN IP ROUTING

A routing table is a set of rules, often viewed in table format, that is used to determine where data packets traveling over an Internet Protocol (IP) network will be directed. All IP-enabled devices, including routers and switches, use routing tables.

A routing table contains the information necessary to forward a packet along the best path toward its destination. Each packet contains information about its origin and destination. When a packet is received, a network device examines the packet and matches it to the routing table entry providing the best match for its destination. The table then provides the device with instructions for sending the packet to the next hop on its route across the network.

A basic routing table includes the following information:

- Destination: The IP address of the packet's final destination
- Next hop: The IP address to which the packet is forwarded
- Interface: The outgoing network interface the device should use when forwarding the packet to the next hop or final destination

- Metric: Assigns a cost to each available route so that the most cost-effective path can be chosen
- Routes: Includes directly-attached subnets, indirect subnets that are not attached to the device but can be accessed through one or more hops, and default routes to use for certain types of traffic or when information is lacking.

  Routing tables can be maintained manually or dynamically. Tables for static network devices do not change unless a network administrator manually changes them. In dynamic routing, devices build and maintain their routing tables automatically by using routing protocols to exchange information about the surrounding network topology. Dynamic routing tables allow devices to "listen" to the network and respond to occurrences like device failures and network congestion.

DEFAULT ROUT

The default route is a setting on a computer that defines the packet forwarding rule to use when no specific route can be determined for a given Internet Protocol (IP) destination address. All packets for destinations not established in the routing table are sent via the default route.

The default route generally points to another router, which treats the packet the same way: if a route matches, the packet is forwarded accordingly, otherwise the packet is forwarded to the default route of that router. The route evaluation process in each router uses the longest prefix match method to obtain the most specific route. The network with the longest subnet mask that matches the destination IP address is the next-hop network gateway. The process repeats until a packet is delivered to the destination. Each router traversal counts as one hop in the distance calculation for the transmission path.

The device to which the default route points is often called the default gateway, and it often carries out other functions such as packet filtering, firewalling, or proxy server operations.

The default route in Internet Protocol Version 4 (IPv4) is designated as the zero-address 0.0.0.0/0 in CIDR notation,[1] often called the quad-zero route.[*citation needed*] The subnet mask is given as /0, which effectively specifies all networks, and is the shortest match possible. A route lookup that does not match any other route, falls back to this route. Similarly, in IPv6, the default route is specified by ::/0.

HOST SPECIFIC ROUTING

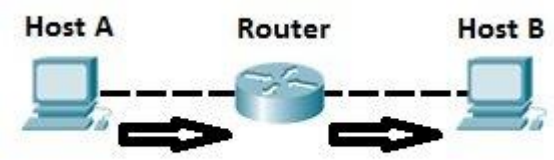
Host specific routing determines the packet forward route based on the exact matching of a packet's IP address with the routing table entry that records the route towards the host. Most of the existing routers can support a small number of host specific routes in their routing tables. However, due to the scalability issues and memory limitation, host specific routing capability has never been widely used in Internet. By limiting the use of host specific routing to smaller network domains such as enterprise, metropolitan, and various access networks, scalability will be less an issue.

Host specific routing has the advantages of supporting host mobility, fast packet forwarding, and flattening the network hierarchy. There is a trade-off in selecting host specific routing or prefix routing for the different network domains. Use prefix based routing in Internet backbone and host specific routing in metropolitan area networks, access networks, enterprise networks could be the ideal solution for optimizing network performance while increasing the network flexibility.

ROUTING WITH IP ADDRESS

IP routing is the process of sending packets from a host on one network to another host on a different remote network. This process is usually done by routers. Routers examine the destination IP address of a packet , determine the next-hop address, and forward the packet. Routers use routing tables to determine the next hop address to which the packet should be forwarded.

Consider the following example of IP routing:



Host A wants to communicate with host B, but host B is on another network. Host A is configured to send all packets destined for remote networks to router R1. Router R1 receives the packets, examines the destination IP address and forwards the packet to the outgoing interface associated with the destination network.

Routing table

Each router maintains a routing table and stores it in RAM. A routing table is used by routers to determine the path to the destination network. Each routing table consists of the following entries:

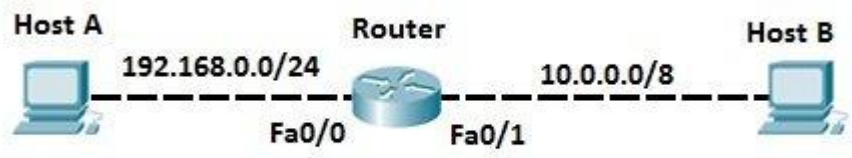
- **network destination and subnet mask** – specifies a range of IP addresses.
- **remote router** – IP address of the router used to reach that network.
- **outgoing interface** – outgoing interface the packet should go out to reach the destination network.

There are three different methods for populating a routing table:

- directly connected subnets
- using static routing
- using dynamic routing

Each of this method will be described in the following chapters.

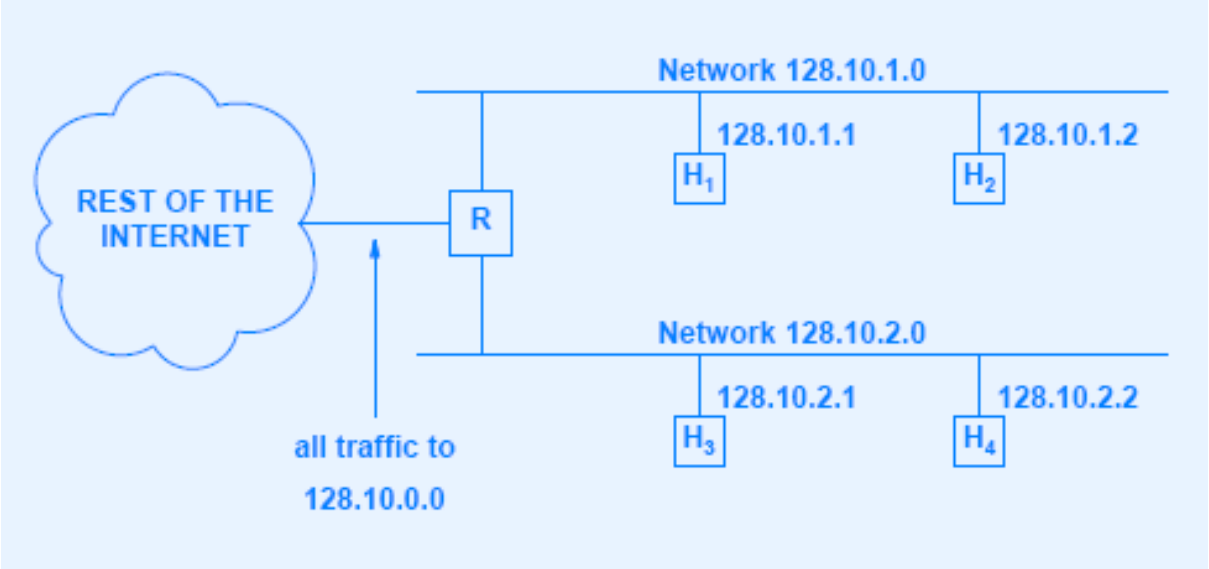
Consider the following example. Host A wants to communicate with host B, but host B is on another network. Host A is configured to send all packets destined for remote networks to the router. The router receives the packets, checks the routing table to see if it has an entry for the destination address. If it does, the router forwards the packet out the appropriate interface port. If the router doesn't find the entry, it discards the packet.

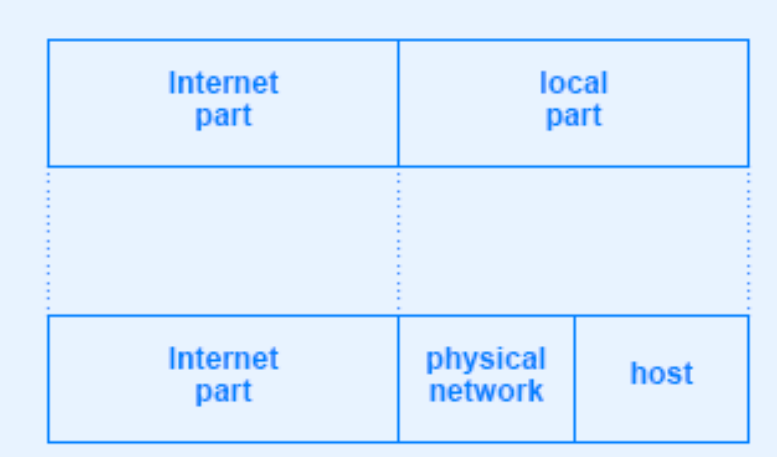


4. SUBNET ADDRESS EXTENSION

INTRODUCTION

- **Subdivides the host suffix into a pair of fields for physical network and host**
  - Allows an organization to use a single network prefix for multiple physical networks
  - Interpreted only by routers and hosts at the site; treated like normal address elsewhere





- Both physical networks share prefix 128.10
- Router R uses third octet of address to choose physical net

Address Mask: Each physical network is assigned 32-bit address mask (also called subnet mask)

TRANSPARENT ROUTERS

There are two basic models for interconnecting local-area networks and wide-area (or long-haul) networks in the Internet. In the first, the local-area network is assigned a network prefix and all routers in the Internet must know how to route to that network. In the second, the local-area network shares (a small part of) the address space of the wide-area network. Routers that support this second model are called address sharing routers or transparent routers. The focus of this memo is on routers that support the first model, but this is not intended to exclude the use of transparent routers.

The basic idea of a transparent router is that the hosts on the local-area network behind such a router share the address space of the wide-area network in front of the router. In certain situations this is a very useful approach and the limitations do not present significant drawbacks.

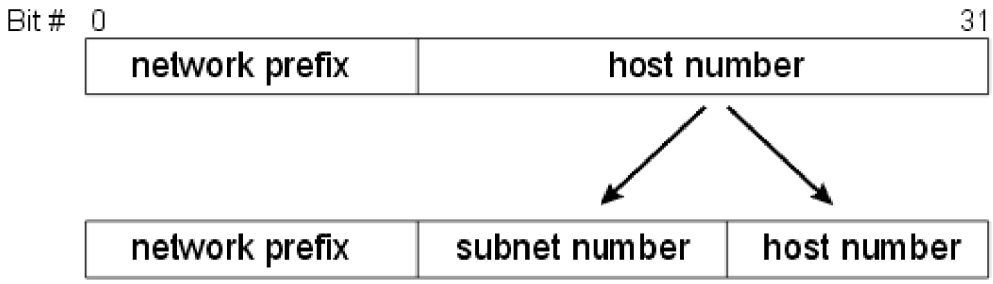
The words in front and behind indicate one of the limitations of this approach: this model of interconnection is suitable only for a geographically (and topologically) limited stub environment. It requires that there be some form of logical addressing in the network level addressing of the wide-area network. IP addresses in the local environment map to a few (usually one) physical address in the wide- area network. This mapping occurs in a way consistent with the {IP address <-> network address } mapping used throughout the wide-area network.

SUBNET ADDRESSING

In 1985, RFC 950 defined a standard procedure to support the subnetting, or division, of a single Class A, B, or C network number into smaller pieces. Subnetting was introduced to overcome some of the problems that parts of the Internet were beginning to experience with the classful two-level addressing hierarchy:

- Internet routing tables were beginning to grow.
- Local administrators had to request another network number from the Internet before a new network could be installed at their site.

Both of these problems were attacked by adding another level of hierarchy to the IP addressing structure. Instead of the classful two-level hierarchy, subnetting supports a three-level hierarchy. The basic idea of subnetting is to divide the standard classful host-number field into two parts - the subnet-number and the hostnumber on that subnet.



3-level Internet Address Structure

Subnetting attacked the ***expanding routing table problem*** by ensuring that the subnet structure of a network is never visible outside of the organization's private network. The route from the Internet to any subnet of a given IP address is the same, no matter which subnet the destination host is on. This is because all subnets of a given network number use the same network-prefix but different subnet numbers. The routers within the private organization need to differentiate between the individual subnets, but as far as the Internet routers are concerned, all of the subnets in the organization are collected into a single routing table entry. This allows the local administrator to introduce arbitrary complexity into the private network without affecting the size of the Internet's routing tables.

Subnetting overcame the ***registered number issue*** by assigning each organization one (or at most a few) network number(s) from the IPv4 address space. The organization was then free to assign a distinct subnetwork number for each of its internal networks. This allows the organization to deploy additional subnets without needing to obtain a new network number from the Internet.

FLEXIBILITY IN SUBNET ADDRESS ASSIGNMENT

IPv6 addresses have a flexible structure for address assignments. This enables registries, internet service providers, network designers and others to assign address ranges to

organizations and networks based on different criteria, like size of networks, estimated growth rate, etc. Often, the initial assignment doesn't scale well because a small network becomes larger than expected, needing more addresses. But then, the assignment authority cannot allocate contiguous addresses because they were already assigned to another network.

IMPLEMENTATION OF SUBNET WITH MASKING

A subnet mask is a number that defines a range of IP addresses that can be used in a network. (It is not something you wear on your head to keep subnets out.) Subnet masks are used to designate subnetworks, or subnets, which are typically local networks LANs that are connected to the Internet. Systems within the same subnet can communicate directly with each other, while systems on different subnets must communicate through a router. Therefore, subnetworks can be used to partition multiple networks and limit the traffic between them.

A subnet mask hides, or "masks," the network part of a system's IP address and leaves only the host part as the machine identifier. A common subnet mask for a Class C IP address is 255.255.255.0. Each section of the subnet mask can contain a number from 0 to 255, just like an IP address. Therefore, in the example above, the first three sections are full, meaning the IP addresses of computers within the subnet mask must be identical in the first three sections. The last section of each computer's IP address can be anything from 0 to 255. For example, the IP addresses 10.0.1.201 and 10.0.1.202 would be in the same subnet, while 10.0.2.201 would not. Therefore, a subnet mask of 255.255.255.0 allows for close to 256 unique hosts within the network (since not all 256 IP addresses can be used).

ROUTING IN THE PRESENCE OF SUBNET

A **subnetwork** or **subnet** is a logical subdivision of an IP network.[1]:1,16 The practice of dividing a network into two or more networks is called **subnetting**.

Computers that belong to a subnet are addressed with an identical most-significant bit-group in their IP addresses. This results in the logical division of an IP address into two fields, the *network number* or *routing prefix* and the *rest field* or *host identifier*. The *rest field* is an identifier for a specific host or network interface.

The *routing prefix* may be expressed in Classless Inter-Domain Routing (CIDR) notation written as the first address of a network, followed by a slash character (/), and ending with the bit-length of the prefix. For example, *198.51.100.0/24* is the prefix of the Internet Protocol version 4 network starting at the given address, having 24 bits allocated for the network prefix, and the remaining 8 bits reserved for host addressing. Addresses in the range *198.51.100.0* to *198.51.100.255* belong to this network. The IPv6 address specification *2001:db8::/32* is a large address block with 2^{96} addresses, having a 32-bit routing prefix.

For IPv4, a network may also be characterized by its **subnet mask** or **netmask**, which is the bitmask that when applied by a bitwise AND operation to any IP address in the network, yields the routing prefix. Subnet masks are also expressed in dot-decimal notation like an address. For example, *255.255.255.0* is the subnet mask for the prefix *198.51.100.0/24*.

Traffic is exchanged between subnetworks through routers when the routing prefixes of the source address and the destination address differ. A router serves as a logical or physical boundary between the subnets.

5. UDP

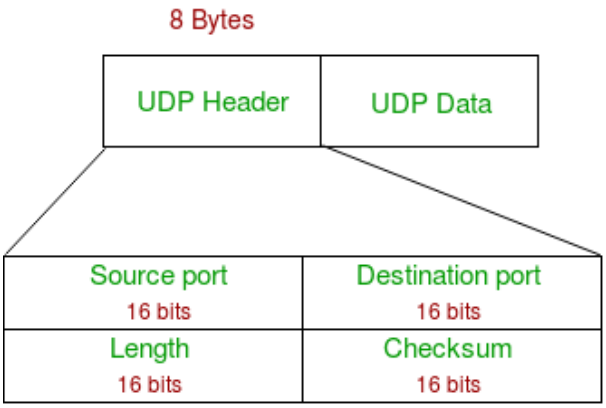
INTRODUCTION TO UDP

User Datagram Protocol (UDP) is a Transport Layer protocol. UDP is a part of Internet Protocol suite, referred as UDP/IP suite. Unlike TCP, it is **unreliable and connectionless protocol.** So, there is no need to establish connection prior to data transfer.

Though Transmission Control Protocol (TCP) is the dominant transport layer protocol used with most of Internet services; provides assured delivery, reliability and much more but all these services cost us with additional overhead and latency. Here, UDP comes into picture. For the realtime services like computer gaming, voice or video communication, live conferences; we need UDP. Since high performance is needed, UDP permits packets to be dropped instead of processing delayed packets. There is no error checking in UDP, so it also save bandwidth.
User Datagram Protocol (UDP) is more efficient in terms of both latency and bandwidth.

FORMAT OF UDP MESSAGE

UDP header is **8-bytes** fixed and simple header, while for TCP it may vary from 20 bytes to 60 bytes. First 8 Bytes contains all necessary header information and remaining part consist of data. UDP port number fields are each 16 bits long, therefore range for port numbers defined from 0 to 65535; port number 0 is reserved. Port numbers help to distinguish different user requests or process.



1. **Source Port:** Source Port is 2 Byte long field used to identify port number of source.

2. **Destination Port:** It is 2 Byte long field, used to identify the port of destined packet.
3. **Length:** Length is the length of UDP including header and the data. It is 16-bits field.
4. **Checksum:** Checksum is 2 Bytes long field. It is the 16-bit one's complement of the one's complement sum of the UDP header, pseudo header of information from the IP header and the data, padded with zero octets at the end (if necessary) to make a multiple of two octets

Applications of UDP:

- Used for simple request response communication when size of data is less and hence there is lesser concern about flow and error control.
- It is suitable protocol for multicasting as UDP supports packet switching.
- UDP is used for some routing update protocols like RIP (Routing Information Protocol).
- Normally used for real time applications which cannot tolerate uneven delays between sections of a received message.
- Following implementations uses UDP as a transport layer protocol:
    - NTP (Network Time Protocol)
    - DNS (Domain Name Service)
    - BOOTP, DHCP.
    - NNP (Network News Protocol)
    - Quote of the day protocol
    - TFTP, RTSP, RIP, OSPF.
- Application layer can do some of the tasks through UDP-
    - Trace Route
    - Record Route
    - Time stamp
- UDP takes datagram from Network Layer, attach its header and send it to the user. So, it works fast.
- Actually UDP is null protocol if you remove checksum field.

6.DOMAIN NAME SYSTEM

HIERARCHIAL NAMES

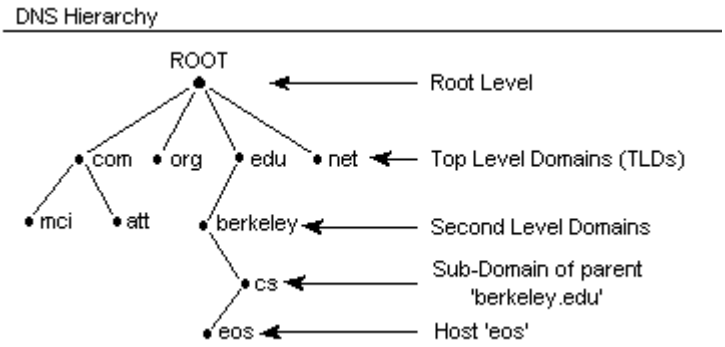
Domain Names are hierarchical and each part of a domain name is referred to as either the root, top level, second level or as a sub-domain. To allow computers to properly recognize a fully qualified domain name, dots are placed between each part of the name. All resolvers treat dots as separators between the parts of the domain name. The fully qualified domain name is split into pieces at the dots and the tree is searched starting from the root of the hierarchial tree structure. All resolvers start their lookups at the root, therefore the root is represented by a dot and is often assumed to be there, even when not shown. The resolver navigates its way down the tree until it gets to the last, left-most part of the domain name and then looks within that location for the information it needs. Information about a host such as its name, its IP address and occasionally even its function are stored in one or more zone files which together compose a larger zone often referred to as a *domain*.

- Top Level Domains (TLD's)
- Second Level Domains
- Sub-Domains
- Host Name (a resource record)

Within the hierarchy, you will start resolution at the top level domain, work your way down to the second-level domain, then through zero, one or more sub-domains until you get to the actual host name you want to resolve into an IP address.

It is traditional to use different DNS servers for each level of the DNS hierarchy. The root of all DNS entries is handled by the DNS servers at the InterNIC [*well, sort of, but we'll get to that later --InetD*]. The InterNIC points the Top Level Domains (TLDs) to the top level domain name servers maintained by all registrars such as Network Solutions, Register.Com, OpenSRS and many others. Next come each domain's server will delegate to the DNS server at the next lower level in the hierarchy.

For example, in the figure below, .edu is the top level domain, berkeley is the second level domain, and .cs is the sub-domain of berkeley. Eos is the host name. A DNS server would store the IP address of the host where its name resides in the tree.



SUBNET AUTHORITY

INTERNET DOMAIN SYSTEM

A **domain name** is an identification string that defines a realm of administrative autonomy, authority or control within the Internet. Domain names are used in various networking contexts and for application-specific naming and addressing purposes. In general, a domain name identifies a network domain, or it represents an Internet Protocol (IP) resource, such as a personal computer used to access the Internet, a server computer hosting a web site, or the web site itself or any other service communicated via the Internet. In 2017, 330.6 million domain names had been registered.

Domain names are formed by the rules and procedures of the Domain Name System (DNS). Any name registered in the DNS is a domain name. Domain names are organized in subordinate levels (subdomains) of the DNS root domain, which is nameless. The first-level set of domain names are the top-level domains (TLDs), including the generic top-level domains (gTLDs), such as the prominent domains com, info, net, edu, and org, and the country code top-level domains (ccTLDs). Below these top-level domains in the DNS hierarchy are the second-level and third-level domain names that are typically open for reservation by end-users who wish to connect local area networks to the Internet, create other publicly accessible Internet resources or run web sites.

The registration of these domain names is usually administered by domain name registrars who sell their services to the public.

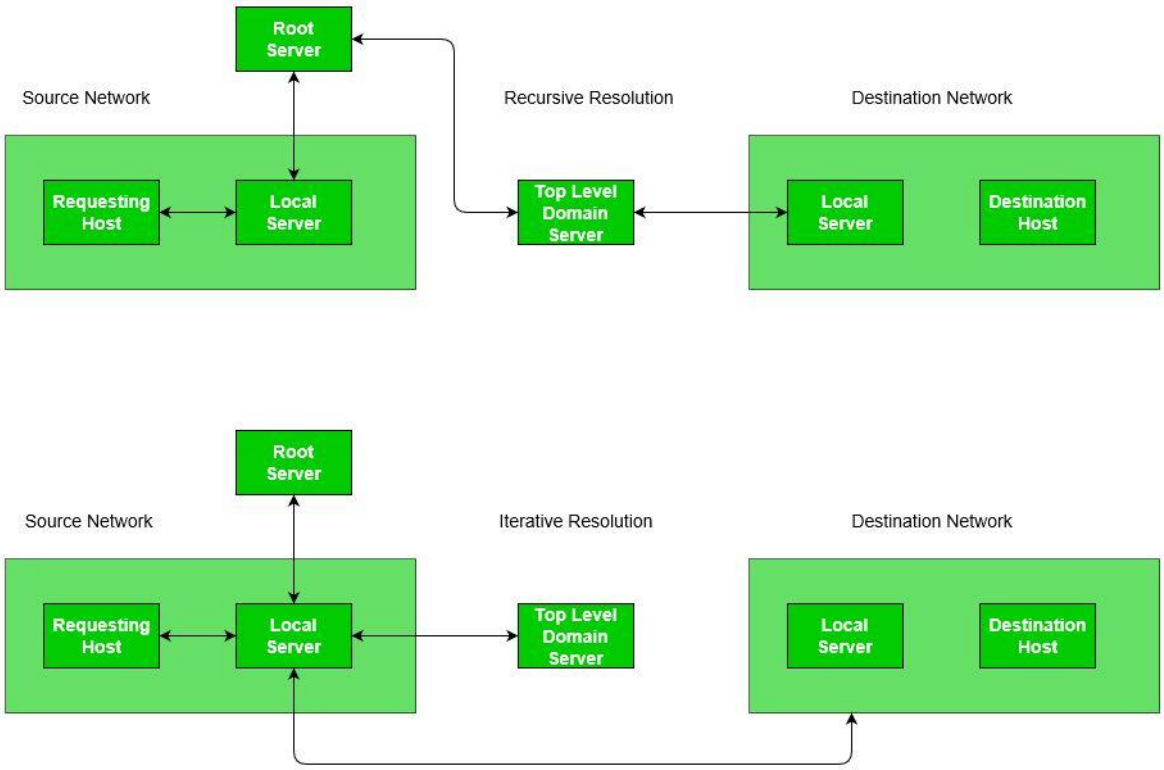
A fully qualified domain name (FQDN) is a domain name that is completely specified with all labels in the hierarchy of the DNS, having no parts omitted.

MAPPING OF DOMAIN NAME TO ADDRESS

Mapping a domain name to an IP Address is known as **Name-Address Resolution**. The Domain Name Server (DNS) Resolver performs this operation by consulting name servers.

In order to find a particular DNS the requesting host place it's query to the Local DNS Server with a mapping request. If it has the information, the resolver is satisfied else the resolver is referred to other servers or other servers are asked to provide the information. After the resolver, gets the response, it checks whether the response is correct or not. If the response is correct, the response is passed to the process that requested it, else the name query fails.

A resolution can be of two types – iterative and recursive.



1.Recursive Resolution –

Here, client requires the Local Server to give either the requested mapping or an error message. A DNS Query is generated by the application program to the resolver to fetch the destination IP Address. The Query is then forward to the local DNS Server. If it knows the IP Address, it sends a response to the resolver. Assuming, it does not know the IP Address, it sends the query to the root name server.

The root name server contains information of about at least one server of Top Level Domain. The query is then sent to the respective Top-Level Domain server. If it contains the mapping, the response is sent back to the root server and then to host's local server. If it doesn't contain the mapping, it should contain the IP Address of destination's local DNS Server. The local DNS server knows the destination host's IP Address. The information is then sent back to the top-level domain server, then to the root server and then to the host's Local DNS Server and finally to the host.

2.Iterative Resolution –

The main difference between iterative and recursive resolution is that, here each server that does not know the mapping sends the IP Address of the next server to the one requested it. Here, client allows the server to return the best answer it can give as a match or as a referral. A DNS Query is generated by the application program to the resolver to fetch the destination IP Address. The Query is then forward to the local DNS Server. Assuming, it does not know the IP Address, it sends the query to the root name server.

The root name server returns the IP Address of the Top-Level Domain Server to the Local Server. The Top-Level Domain server is contacted by Local Server and it returns either the IP of the destination host or its local DNS Server. If it returns the server's address, then by contacting the destination's Local DNS Server, we get the IP Address of the destination host. The response/mapping is then passed from host's local DNS server to the resolver and then finally to the host.

DOMAIN NAME RESOLUTION

Domain Name Resolution is the task of converting domain names to their corresponding IP address. This is all done behind the scenes and is rarely noticed by the user. When you enter a domain name in an application that uses the Internet, the application will issue a command to have the operating system convert the domain name into its IP address, and then connect to that IP address to perform whatever operation it is trying to do.

The way the operating system resolves the domain name is based upon its configuration. For almost all operating systems the default order for Domain Name resolution is as follows:

1. **Hosts File** - There is a file called the HOSTS file that you can use to convert domain names to IP addresses. Entries in the HOSTS file override any mappings that would be resolved via a DNS server.
2. **Domain Name System -** This is the system used on the Internet for converting domain names to their corresponding IP addresses. Your operating system will connect to the DNS server configured on your computer and have that server return to you the IP address for the domain name you queried it with.
3. **Netbios** - This only applies to Windows machines and will only be used to map names to IP addresses if all previous methods failed. This method will attempt to map the netbios name you are trying to connect to with an IP address.

7.INTERNET APPLICATIONS AND SERVICES

E-MAIL NETWORK

Electronic mail (**email** or **e-mail**) is a method of exchanging messages ("mail") between people using electronic devices. Email entered limited use in the 1960s, but users could only send to users of the same computer, and some early email systems required the author and the recipient to both be online simultaneously, similar to instant messaging. Ray Tomlinson is credited as the inventor of email; in 1971, he developed the first system able to send mail between users on different hosts across the ARPANET, using the @ sign to link the user name with a destination server. By the mid-1970s, this was the form recognized as email.

Email operates across computer networks, primarily the Internet. Today's email systems are based on a store-and-forward model. Email servers accept, forward, deliver, and store messages. Neither the users nor their computers are required to be online simultaneously; they need to connect, typically to a mail server or a webmail interface to send or receive messages or download it.

Originally an ASCII text-only communications medium, Internet email was extended by Multipurpose Internet Mail Extensions (MIME) to carry text in other character sets and multimedia content attachments. International email, with internationalized email addresses using UTF-8, is standardized but not widely adopted.

E-MAIL PROTOCOL

The commonly used protocols are- IMAP, POP3, SMTP, and Exchange. These are few protocol types one would come across while accessing an email client. The protocol details can be accessed via the server settings based on the email client being used.

POP3

POP3 stands for Post Office 3 protocol. POP simply reaches out to the mail server and brings back the mail contents. This is a simple yet standardized way which allows users to access mailboxes and quickly download messages to their device.

With POP3, users can configure the server settings. This can be used to allow mail copies to be left on the server or move all emails without leaving any copy on the server. This is usually configurable in most cases. The biggest advantage of POP3 is the low dependency over the Internet. Users can download all emails and read them at leisure even if they are accessing this offline.

The way these emails are stored in local depends on the email client. For instance, Outlook utilizes .pst, while Thunderbird uses .mbox. This is a good option in case you choose to read emails offline. Apart from this, this helps you reduce the server space by storing messages locally.

The default ports for POP3 are:

- Port 110 – This is the default non-encrypted port.
- Port 995 – This is the default port for secure connections.

IMAP

This stands for Internet Message Access Protocol. This again is a standard protocol for accessing emails and is a client/server protocol. Here the emails are received and held by the Internet server. Unlike POP, this does not move the emails. The biggest difference between POP3 and IMAP is the mail sync up. POP3 assumes that a user will be connected to a single device. However, IMAP is suitable for different devices simultaneously.

IMAP requires users to be constantly connected to the Internet. When a user accesses the mailbox, the user is actually connected to an external server. This is more beneficial when there are multiple users. IMAP can work over a relatively low internet connection since it only downloads email messages from the server when a user has requested to read a specific email.

The default ports for IMAP are:

- Port 143 – This is the default non-encrypted port.
- Port 993 – This is default port for secure connections.

SMTP

This stands for Simple Mail Transfer Protocol. This is a standard protocol for sending emails over the Internet. This is a protocol which is used by a Mail Transfer Agent to deliver emails to a recipient's email server. This is a protocol which defines mail sending and cannot be used for mail receiving.

SMTP is the most commonly used protocol for mail transfer between two servers. This requires no authentication to function, unlike POP3 and IMAP. Certain Internet Service Providers block the default port 25 of SMTP. In such cases, the mail server also provides an alternate secondary port.

The default port for SMTP are:

- Port 25 – This is the default non-encrypted port.
- Port 465/ 587 – This is default port for secure connections.

HTTP

This is a commonly known protocol and stands for Hyper Text Transfer Protocol. This is not an email specific protocol. However, HTTP is used for email access using web-based emails. Hotmail or Gmail are examples of using HTTP as an email protocol. This is used to compose and retrieve emails from a web-based account.

The default port for HTTP are:

- Port 80 – This is default non-encrypted port.
- Port 443 – This is default port for secure connections.

Exchange Account (EAS – Exchange ActiveSync)

This is used by Exchange servers like Microsoft Exchange. This not only syncs mail but also syncs contacts, calendars, notes and everything in the outlook. The advantage is that users can have a synced copy of the calendar, contacts over multiple devices.

FORMAT OF E-MAIL MESSAGE

- The first portion of all e-mail addresses, the part before the @ symbol, contains the alias, user, group, or department of a company. In our above example, **support** is the Technical Support department at Computer Hope.
- Next, the @ (at sign) is a divider in the e-mail address; it's required for all SMTP e-mail addresses since the first message was sent by Ray Tomlinson.
- Finally, **computerhope.com** is the domain name to which the user belongs. The .com is the TLD (top-level domain) for our domain.

E-MAIL ROUTING

Typically, incoming emails go through a spam filtering service, and all clean emails are sent to the mail server. Email routing goes a step beyond spam filtering and lets you copy or redirect emails based on customized rules. The routing rules can be based on the sender, recipient, or many other parameters. With email routing, an email can be sent to the original recipient and carbon copied to additional recipient(s), or it can be redirected to a completely different destination.

Why Would I Need Email Routing?

Email routing will help you keep track of important emails and make sure the right people are kept in the loop. There's a lot of ways you can use email routing:

- Send sales inquiries to the right person or department
- Instantly sort large amounts of email
- Automatically copy emails to managers for oversight and accountability
- Selectively archive important emails

PUBLIC DOMAIN SOFTWARE

Public domain software is any software that has no legal, copyright or editing restrictions associated with it. It is free and open-source software that can be publicly modified, distributed or sold without any restrictions. SQLite, I2P and CERN https are popular examples of public domain software.

Public domain software has no ownership and is available for use, modification and commercialization by anyone. Typically, public domain software is intentionally or voluntarily uncopyrighted, unpatented and is unrestricted by its developer/author. It is different from free software and freeware that does has copyrights and patents associated with it.

Although there are no licensing requirements with public domain software, The Unlicense, Creative Commons License and WTFPL are based on a similar approach.

TYPES OF FTP SOFTWARE

A File Transfer Protocol **client** (**FTP client**) is a **software** utility that establishes a connection between a host computer and a remote server, typically an **FTP** server. An **FTP client** provides the dual-direction transfer of data and files between two computers over a TCP network or an Internet connection.

Some ftp software types are

- WinSCP. ...
- Core **FTP** LE. ...
- CuteFTP. ...
- Cyberduck. ...
- FileZilla. ...
- CrossFTP. ...
- gFTP. ...
- Nautilus.

FTP CLIENTS

FTP (File Transfer Protocol) allows you to upload files from your computer to your WordPress site. In order to use FTP, you will need an FTP client which is a desktop app that connects your computer to your WordPress hosting account.

It provides an easy to use graphics user interface, so that you can perform all FTP functions such as copy, upload, delete, rename, and edit files / folders on your WordPress site.

How to Use an FTP Client?

You will need a FTP username and password in order to connect to your WordPress site.

This information can be found in the email you got when you first started your blog and signed up for a web hosting account.

You can also get this information from your web hosting cPanel dashboard or ask the support, and they will email it to you.

Once you have this information, you can connect to your website.

First, you will need to launch your FTP client and enter your FTP username, password, host (usually your website address e.g. wpbeginner.com), and then click on the connect button.

TELNET PROTOCOL

Telnet, developed in 1969, is a protocol that provides a command line interface for communication with a remote device or server, sometimes employed for remote management but also for initial device setup like network hardware. Telnet stands for Teletype Network, but it can also be used as a verb; 'to telnet' is to establish a connection using the Telnet protocol.

Telnet provides users with a bidirectional interactive text-oriented communication system utilizing a virtual terminal connection over 8 byte. User data is interspersed in-band with telnet control information over the transmission control protocol (TCP). Often, Telnet was used on a terminal to execute functions remotely.

The user connects to the server by using the Telnet protocol, which means entering Telnet into a command prompt by following this syntax: telnet hostname port. The user then executes commands on the server by using specific Telnet commands into the Telnet prompt. To end a session and log off, the user ends a Telnet command with Telnet.

Telnet can be used to test or troubleshoot remote web or mail servers, as well as for remote access to MUDs (multi-user dungeon games) and trusted internal networks.

IRC NETWORKS AND SERVICES

Internet Relay Chat (IRC) is an application layer protocol that provides a way of communicating with people from all over the world in the real time.

IRC Network

It consists of different separate networks and their servers which allow users to connect to the IRC and the different clients. The user runs a program (i.e.client) to connect to a server on one of the IRC network. The server copies information and forwards them to the respective destinations as a result relaying of messages happen.

The IRC Protocol is based on the **client-server model**, and is well suited to running on many machines in a distributed fashion. A typical setup involves a single server forming a central point for clients or other servers to connect to, and perform the required message delivery/multiplexing and other functions.

The overall IRC has mainly the following ingredients:

1. Networks

2. Servers

3. Clients

4. Channels

WORLD WIDE WEB

World Wide Web (WWW), byname **the Web**, the leading information retrieval service of the Internet (the worldwide computer network). The Web gives users access to a vast array of documents that are connected to each other by means of hypertext or hypermedia links—i.e., hyperlinks, electronic connections that link related pieces of information in order to allow a user easy access to them. Hypertext allows the user to select a word or phrase from text and thereby access other documents that contain additional information pertaining to that word or phrase. Hypermedia documents feature links to images, sounds, animations, and movies. The Web operates within the Internet's basic client-server format; servers are computer programs that store and transmit documents to other computers on the network when asked to, while clients are programs that request documents from a server as the user asks for them. Browser software allows users to view the retrieved documents.

BROWSERS

A browser is an application program that provides a way to look at and interact with all the information on the World Wide Web. This includes Web pages, videos and images. The word "browser" originated prior to the Web as a generic term for user interfaces that let you browse (navigate through and read) text files online. Many people will use web browsers today for access to the internet and is seen almost as a necessity in how many navigate their daily life.

A Web browser is a client program that uses HTTP (Hypertext Transfer Protocol) to make requests of Web servers throughout the Internet on behalf of the browser user. Most browsers support e-mail and the File Transfer Protocol (FTP), but a Web browser is not required for those Internet protocols and more specialized client programs are more popular. Most Web browsers share standard features such as:

- A home button- which, when selected, will bring a user to a pre-defined homepage.
- A Web address bar, which allows users to input a Web address and visit a website.
- Back and forward buttons- which will take the user to the previous or the next page they were on.
- Refresh- a button which can be used to reload a Web page.
- Stop- a button which makes a Web cease communication with a Web server, stopping a page from loading.

8.HTML AND INTERACTIVE TOOLS

HTML

- HTML stands for Hyper Text Markup Language
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements are represented by tags
- HTML tags label pieces of content such as "heading", "paragraph", "table", and so on
- Browsers do not display the HTML tags, but use them to render the content of the page.

Example

<!DOCTYPE html>

<html>

<head>

<title>Page Title</title>

</head>

<body>

<h1>My First Heading</h1>

My first paragraph.

</body>

</html>

Example Explained

- The <!DOCTYPE html> declaration defines this document to be HTML5
- The <html> element is the root element of an HTML page
- The <head> element contains meta information about the document
- The <title> element specifies a title for the document
- The <body> element contains the visible page content
- The <h1> element defines a large heading
- The element defines a paragraph

HEADER ELEMENTS

HTML headings are defined with the <h1> to <h6> tags.

<h1> defines the most important heading. <h6> defines the least important heading.

Example

<h1>Heading 1</h1>
<h2>Heading 2</h2>
<h3>Heading 3</h3>
<h4>Heading 4</h4>
<h5>Heading 5</h5>
<h6>Heading 6</h6>

<h1> headings should be used for main headings, followed by <h2> headings, then the less important <h3>, and so on.

SECTION HEADINGS

- **HTML Body Element** (<body>) defines all the content of a document. It contains all the content and HTML tags.
- **HTML Header Element** (<header>) defines a page which typically contains the logo, title, and navigation. The header can also be used in other semantic elements such as <article> or <section> — or section header, containing perhaps the section's heading, author name, etc. <article>, <section>, <aside>, and <nav> can have their own <header>. Despite its name, it is not necessarily positioned at the beginning of the page or section.
- **HTML Footer Element** (<footer>) defines a page footer which typically contains the copyright, legal notices and sometimes some links — or section footer, containing perhaps the section's publication date, license information, etc. <article>, <section>, <aside>, and <nav> can have their own <footer>. Despite its name, it is not necessarily positioned at the end of the page or section.

BLOCK ORIENRED ELEMENTS

A block-level element always starts on a new line and takes up the full width available (stretches out to the left and right as far as it can).

The element is a block-level element.

Example

Hello World

Block level elements in HTML:

<address> <article> <aside> <blockquote> <canvas> <dd> <dl> <dt>

<fieldset> <figcaption> <figure> <footer> <form> <h1>-<h6> <header> <hr>

 <main> <nav> <noscript> <p> <pre> <section> <table> <tfoot>

 <video>

LIST

HTML offers web authors three ways for specifying lists of information. All lists must contain one or more list elements. Lists may contain −

- **<ul>** − An unordered list. This will list items using plain bullets.
- **<ol>** − An ordered list. This will use different schemes of numbers to list your items.
- **<dl>** − A definition list. This arranges your items in the same way as they are arranged in a dictionary.

HTML Unordered Lists

An unordered list is a collection of related items that have no special order or sequence. This list is created by using HTML **<ul>** tag. Each item in the list is marked with a bullet.

Example

```
<!DOCTYPE html>
<html>

  <head>
    <title>HTML Unordered List</title>
  </head>

  <body>
    <ul>
```

```
      <li>Beetroot</li>
      <li>Ginger</li>
      <li>Potato</li>
      <li>Radish</li>
    </ul>
  </body>

</html>
```

This will produce the following result –

- Beetroot
- Ginger
- Potato
- Radish

The type Attribute

You can use **type** attribute for <ul> tag to specify the type of bullet you like. By default, it is a disc. Following are the possible options −

```
<ul type = "square">
<ul type = "disc">
<ul type = "circle">
```

Example

Following is an example where we used <ul type = "square">

```
<!DOCTYPE html>
<html>

  <head>
    <title>HTML Unordered List</title>
  </head>
```

```
  <body>
    <ul type = "square">
      <li>Beetroot</li>
      <li>Ginger</li>
      <li>Potato</li>
      <li>Radish</li>
    </ul>
  </body>

</html>
```

This will produce the following result −

- Beetroot
- Ginger
- Potato
- Radish

HTML Ordered Lists

If you are required to put your items in a numbered list instead of bulleted, then HTML ordered list will be used. This list is created by using **<ol>** tag. The numbering starts at one and is incremented by one for each successive ordered list element tagged with <li>.

Example

Live Demo

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Ordered List</title>
   </head>

   <body>
      <ol>
         <li>Beetroot</li>
         <li>Ginger</li>
         <li>Potato</li>
         <li>Radish</li>
      </ol>
   </body>

</html>
```

This will produce the following result −

The type Attribute

You can use **type** attribute for <ol> tag to specify the type of numbering you like. By default, it is a number. Following are the possible options −

<ol type = "1"> - Default-Case Numerals.
<ol type = "I"> - Upper-Case Numerals.
<ol type = "i"> - Lower-Case Numerals.
<ol type = "A"> - Upper-Case Letters.
<ol type = "a"> - Lower-Case Letters.

Example

Following is an example where we used <ol type = "1">

```
<!DOCTYPE html>
<html>

  <head>
    <title>HTML Ordered List</title>
  </head>

  <body>
    <ol type = "1">
      <li>Beetroot</li>
      <li>Ginger</li>
      <li>Potato</li>
      <li>Radish</li>
    </ol>
  </body>

</html>
```

This will produce the following result −

1. Beetroot
2. Ginger
3. Potato
4. Radish

The start Attribute

You can use **start** attribute for <ol> tag to specify the starting point of numbering you need. Following are the possible options −

<ol type = "1" start = "4"> - Numerals starts with 4.
<ol type = "I" start = "4"> - Numerals starts with IV.
<ol type = "i" start = "4"> - Numerals starts with iv.
<ol type = "a" start = "4"> - Letters starts with d.
<ol type = "A" start = "4"> - Letters starts with D.

Example

Following is an example where we used <ol type = "i" start = "4" >

```
<!DOCTYPE html>
<html>

  <head>
    <title>HTML Ordered List</title>
  </head>

  <body>
    <ol type = "i" start = "4">
      <li>Beetroot</li>
      <li>Ginger</li>
      <li>Potato</li>
      <li>Radish</li>
    </ol>
  </body>
```

</html>

This will produce the following result −

iv. Beetroot
v. Ginger
vi. Potato
vii. Radish

HTML Definition Lists

HTML and XHTML supports a list style which is called **definition lists** where entries are listed like in a dictionary or encyclopedia. The definition list is the ideal way to present a glossary, list of terms, or other name/value list.

Definition List makes use of following three tags.

- <dl> − Defines the start of the list
- <dt> − A term
- <dd> − Term definition
- </dl> − Defines the end of the list

Example

```
<!DOCTYPE html>
<html>

  <head>
    <title>HTML Definition List</title>
  </head>

  <body>
    <dl>
      <dt><b>HTML</b></dt>
      <dd>This stands for Hyper Text Markup Language</dd>
      <dt><b>HTTP</b></dt>
      <dd>This stands for Hyper Text Transfer Protocol</dd>
```

```
    </dl>
  </body>

</html>
```

This will produce the following result −

HTML

This stands for Hyper Text Markup Language

HTTP

This stands for Hyper Text Transfer Protocol

INLINE ELEMENTS

An inline element does not start on a new line and only takes up as much width as necessary.

This is an inline element inside a paragraph.

Example

```
<span>Hello World</span>
```

Inline elements in HTML:

<a> <abbr> <acronym> <b> <bdo> <big>
 <button> <cite> <code> <dfn>

 <i> <img> <input> <kbd> <label> <map> <object> <output> <q> <samp>

<script> <select> <small>   <sub> <sup> <textarea> <time>

<tt> <var>

The Element

The element is often used as a container for other HTML elements.

The element has no required attributes, but style, class and id are common.

When used together with CSS, the element can be used to style blocks of content:

Example

```
<div style="background-color:black;color:white;padding:20px;">
  <h2>London</h2>
  <p>London is the capital city of England. It is the most populous city in the United Kingdom, with a metropolitan area of over 13 million inhabitants.</p>
</div>
```

The Element

The element is often used as a container for some text.

The element has no required attributes, but style, class and id are common.

When used together with CSS, the element can be used to style parts of the text:

Example

```
<h1>My <span style="color: red">Important</span> Heading</h1>
```

HYPERTEXT LINKS

HTML links are hyperlinks.

You can click on a link and jump to another document.

When you move the mouse over a link, the mouse arrow will turn into a little hand.

Hyperlinks are defined with the HTML <a> tag:

link text

Example

```
<a href="https://www.w3schools.com/html/">Visit our HTML tutorial</a>
```

UNIFORM RESOURCE LOCATOR(URL)

A **Uniform Resource Locator** (**URL**), colloquially termed a **web address**,[1] is a reference to a web resource that specifies its location on a computer network and a mechanism for retrieving it. A URL is a specific type of Uniform Resource Identifier (URI),[2][3] although many people use the two terms interchangeably.[4][a] URLs occur most commonly to reference web pages (http), but are also used for file transfer (ftp), email (mailto), database access (JDBC), and many other applications.

Most web browsers display the URL of a web page above the page in an address bar. A typical URL could have the form http://www.example.com/index.html, which indicates a protocol (http), a hostname (www.example.com), and a file name (index.html).

IMAGES

HTML, images are defined with the <img> tag.

The <img> tag is empty, it contains attributes only, and does not have a closing tag.

The src attribute specifies the URL (web address) of the image:

<img src="*url*">

The alt Attribute

The alt attribute provides an alternate text for an image, if the user for some reason cannot view it (because of slow connection, an error in the src attribute, or if the user uses a screen reader).

The value of the alt attribute should describe the image:

Example

<img src="img_chania.jpg" alt="Flowers in Chania">

If a browser

cannot find an image, it will display the value of the alt attribute:

Example

```
<img src="wrongname.gif" alt="Flowers in Chania">
```

Image Size - Width and Height

You can use the style attribute to specify the width and height of an image.

Example

```
<img src="img_girl.jpg" alt="Girl in a jacket" style="width:500px;height:600px;">
```

Alternatively, you can use the width and height attributes:

TABLE

An HTML table is defined with the <table> tag.

Each table row is defined with the <tr> tag. A table header is defined with the <th> tag. By default, table headings are bold and centered. A table data/cell is defined with the <td> tag.

Example

```
<table style="width:100%">
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
```

</table>

HTML Table - Adding a Border

If you do not specify a border for the table, it will be displayed without borders.

A border is set using the CSS border property:

Example

```
table, th, td {
  border: 1px solid black;
}
```

SPECIAL CHARACTERS

HTML, special characters are typically those that can't be easily typed into a keyboard or may cause display issues if typed or pasted into a web page.

If you plan to use any of the special characters on this page, you should use either the HTML entity name or the HTML entity number. This will ensure that it displays correctly in most/all browsers.

COMMON GATEWAY INTERFACE(CGI)

The **Common Gateway Interface (CGI)** provides the middleware between WWW servers and external databases and information sources. The World Wide Web Consortium (W3C) defined the Common Gateway Interface (CGI) and also defined how a program interacts with a Hyper Text Transfer Protocol (HTTP) server. The Web server typically passes the form information to a small application program that processes the data and may send back a confirmation message. This process or convention for passing data back and forth between the server and the application is called the common gateway interface (CGI).

Features of CGI:

- It is a very well defined and supported standard.
- CGI scripts are generally written in either Perl, C, or maybe just a simple shell script.
- CGI is a technology that interfaces with HTML.

- CGI is the best method to create a counter because it is currently the quickest
- CGI standard is generally the most compatible with today's browsers

Advantages of CGI:

- The advanced tasks are currently a lot easier to perform in CGI than in Java.
- It is always easier to use the code already written than to write your own.
- CGI specifies that the programs can be written in any language, and on any platform, as long as they conform to the specification.
- CGI-based counters and CGI code to perform simple tasks are available in plenty.

Disadvantages of CGI:

There are some disadvantages of CGI which are given below:

- In Common Gateway Interface each page load incurs overhead by having to load the programs into memory.
- Generally, data cannot be easily cached in memory between page loads.
- There is a huge existing code base, much of it in Perl.
- CGI uses up a lot of processing time.

VB SCRIPT

VBScript (**V**isual **B**asic **Script**ing Edition) is an Active Scripting language developed by Microsoft is modelled on Visual Basic. It is designed as a "lightweight" language with a fast *interpreter* for use in a wide variety of Microsoft environments. VBScript uses the *Component Object Model* to access elements of the environment within which it is running; for example, the FileSystemObject (FSO) is used to create, read, update and delete files.

VBScript has been installed by default in every desktop release of Microsoft Windows since Windows 98 in Windows Server since Windows NT 4.0 Option Pack;[] and optionally with Windows CE (depending on the device it is installed on).

A VBScript script must be executed within a host environment, of which there are several provided with Microsoft Windows, including: Windows Script Host (WSH), Internet Explorer (IE), and Internet Information Services (IIS). Additionally, the VBScript hosting

environment is embeddable in other programs, through technologies such as the Microsoft Script Control (msscript.ocx).

EXAMPLE

```
<html>
<body>
<script type="text/vbscript">
document.write("Hello World!")
</script>
</body>
</html>
```

To insert a VBScript into an HTML page, we use the <script> tag. Inside the <script> tag we use the type attribute to define the scripting language.

So, the <script type="text/vbscript"> and </script> tells where the VBScript starts and ends:

```
<html>
<body>
<script type="text/vbscript">
...
</script>
</body>
</html>
```

JAVASCRIPT

JavaScript makes HTML pages more dynamic and interactive.

Example

```
<!DOCTYPE html>

<html>
```

<body>

<h1>My First JavaScript</h1>

<button type="button"

onclick="document.getElementById('demo').innerHTML = Date()">

Click me to display Date and Time.</button>

</body>

</html>

The HTML <script> Tag

The <script> tag is used to define a client-side script (JavaScript).

The <script> element either contains script statements, or it points to an external script file through the src attribute.

Common uses for JavaScript are image manipulation, form validation, and dynamic changes of content.

To select an HTML element, JavaScript most often uses the document.getElementById() method.

This JavaScript example writes "Hello JavaScript!" into an HTML element with id="demo":

Example

```
<script>
document.getElementById("demo").innerHTML = "Hello JavaScript!";
</script>
```

The HTML <noscript> Tag

The <noscript> tag is used to provide an alternate content for users that have disabled scripts in their browser or have a browser that doesn't support client-side scripts:

Example

```
<script>
document.getElementById("demo").innerHTML = "Hello JavaScript!";
</script>
<noscript>Sorry, your browser does not support JavaScript!</noscript>
```

XML

XML is a software- and hardware-independent tool for storing and transporting data.

- XML stands for eXtensible Markup Language
- XML is a markup language much like HTML
- XML was designed to store and transport data
- XML was designed to be self-descriptive
- XML is a W3C Recommendation

EXAMPLE

```
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

XML APPLICATION

- **Web publishing**: XML allows you to create interactive pages, allows the customer to customize those pages, and makes creating e-commerce applications more intuitive. With XML, you store the data once and then render that content for different viewers or devices based on style sheet processing using an Extensible Style Language (XSL)/XSL Transformation (XSLT) processor.
- **Web searching and automating Web tasks**: XML defines the type of information contained in a document, making it easier to return useful results when searching the Web:

  For example, using HTML to search for books authored by Tom Brown is likely to return instances of the term 'brown' outside of the context of author. Using XML

restricts the search to the correct context (for example, the information contained in the <author> tag) and returns only the information that you want. By using XML, Web agents and robots (programs that automate Web searches or other tasks) are more efficient and produce more useful results.

- **General applications**: XML provides a standard method to access information, making it easier for applications and devices of all kinds to use, store, transmit, and display data.
- **e-business applications**: XML implementations make electronic data interchange (EDI) more accessible for information interchange, business-to-business transactions, and business-to-consumer transactions.
- **Metadata applications**: XML makes it easier to express metadata in a portable, reusable format.
- **Pervasive computing**: XML provides portable and structured information types for display on pervasive (wireless) computing devices such as personal digital assistants (PDAs), cellular phones, and others. For example, WML (Wireless Markup Language) and VoiceXML are currently evolving standards for describing visual and speech-driven wireless device interfaces.

XML RULES

- All **XML** must have a root element.
- All tags must be closed.
- All tags must be properly nested.
- Tag names have strict limits.
- Tag names are case sensitive.
- Tag names cannot contain spaces.
- Attribute values must appear within quotes ("").
- White space is preserved.

DISPLAY XML DOCUMENTS

An XML file can be displayed using two ways. These are as follows :-

1. Cascading Style Sheet
2. Extensible Stylesheet Language Transformation

Displaying XML file using CSS :

CSS can be used to display the contents of the XML document in a clear and precise manner. It gives the design and style to whole XML document.

- **Basic steps in defining a CSS style sheet for XML :**

  For defining the style rules for the XML document, the following things shoulde be done :-

  1. Define the style rules for the text elements such as font-size, color, font-weight, etc.
  2. Define each element either as a block, inline or list element, using the display property of CSS.
  3. Identify the titles and bold them.

- **Linking XML with CSS :**

  In order to display the XML file using CSS, link XML file with CSS. Below is the syntax for linking the XML file with CSS:

  <?xml-stylesheet type="text/css" href="name_of_css_file.css"?>

- **Example 1.**

  In this example, the XML file is created that contains the information about five books and displaying the XML file using CSS.

  - **XML file :**

    Creating Books.xml as :-

    filter_none

    brightness_4

    ```
    <?xml version="1.0" encoding="UTF-8"?>
    <?xml-stylesheet type="text/css" href="Rule.css"?>
    <books>
       <heading>Welcome To GeeksforGeeks </heading>
       <book>
    ```

```
    <title>Title -: Web Programming</title>
    <author>Author -: Chrisbates</author>
    <publisher>Publisher -: Wiley</publisher>
    <edition>Edition -: 3</edition>
    <price> Price -: 300</price>
</book>
<book>
    <title>Title -: Internet world-wide-web</title>
    <author>Author -: Ditel</author>
    <publisher>Publisher -: Pearson</publisher>
    <edition>Edition -: 3</edition>
    <price>Price -: 400</price>
</book>
<book>
    <title>Title -: Computer Networks</title>
    <author>Author -: Foruouzan</author>
    <publisher>Publisher -: Mc Graw Hill</publisher>
    <edition>Edition -: 5</edition>
    <price>Price -: 700</price>
</book>
<book>
    <title>Title -: DBMS Concepts</title>
    <author>Author -: Navath</author>
    <publisher>Publisher -: Oxford</publisher>
    <edition>Edition -: 5</edition>
    <price>Price -: 600</price>
</book>
<book>
    <title>Title -: Linux Programming</title>
    <author>Author -: Subhitab Das</author>
    <publisher>Publisher -: Oxford</publisher>
    <edition>Edition -: 8</edition>
    <price>Price -: 300</price>
```

```
    </book>
</books>
```

In the above example, Books.xml is linked with Rule.css which contains the corresponding style sheet rules.

- **CSS FILE :**
  Creating Rule.css as:-

  filter_none

  brightness_4

```
books {
     color: white;
     background-color : gray;
     width: 100%;
}
 heading {
     color: green;
     font-size : 40px;
     background-color : powderblue;
}
 heading, title, author, publisher, edition, price {
     display : block;
}
 title {
     font-size : 25px;
     font-weight : bold;
}
```

PARTS OF XML DOCUMENT

An XML document consists of three parts, in the order given:

1. An XML declaration (which is technically optional, but recommended in most normal cases)
2. A document type declaration that refers to a DTD (which is optional, but required if you want validation)
3. A body or document instance (which is required)

Collectively, the XML declaration and the document type declaration are called the XML prolog.

XML Declaration

The XML declaration is a piece of markup (which may span multiple lines of a file) that identifies this as an XML document. The declaration also indicates whether the document can be validated by referring to an external Document Type Definition (DTD). DTDs are the subject of chapter 4; for now, just think of a DTD as a set of rules that describes the structure of an XML document.

The minimal XML declaration is:

```
<?xml version="1.0" ?>
```

XML is case-sensitive (more about this in the next subsection), so it's important that you use lowercase for xml and version. The quotes around the value of the version attribute are required, as are the ? characters. At the time of this writing, "1.0" is the only acceptable value for the version attribute, but this is certain to change when a subsequent version of the XML specification appears.

Document Type Declaration

The document type declaration follows the XML declaration. The purpose of this declaration is to announce the root element (sometimes called the *document element*) and to provide the location of the DTD.[4] The general syntax is:

```
<!DOCTYPE RootElement (SYSTEM | PUBLIC)
    ExternalDeclarations? [InternalDeclarations]? >
```

where <!DOCTYPE is a literal string, RootElement is whatever you name the outermost element of your hierarchy, followed by either the literal keyword SYSTEM or PUBLIC. The optional ExternalDeclarations portion is typically the relative path or URL to the DTD that describes your document type. (It is really only optional if the entire DTD appears as an InternalDeclaration, which is neither likely nor desirable.) If there are InternalDeclarations, they must be enclosed in square brackets. In general, you'll encounter far more cases with ExternalDeclarations than InternalDeclarations, so let's ignore the latter for now.

Document Body

The document body, or instance, is the bulk of the information content of the document. Whereas across multiple instances of a document of a given type (as identified by the DOCTYPE) the XML prolog will remain constant, the document body changes with each document instance (in general). This is because the prolog defines (either directly or indirectly) the overall structure while the body contains the real instance-specific data. Comparing this to data structures in computer languages, the DTD referenced in the prolog is analogous to a struct in the C language or a class definition in Java, and the document body is analogous to a runtime instance of the struct or class.

Because the document type declaration specifies the root element, this *must* be the first element the parser encounters. If any other element but the one identified by the DOCTYPE line appears first, the document is immediately invalid.

CPNCEPT OF DTD

The XML Document Type Declaration, commonly known as DTD, is a way to describe XML language precisely. DTDs check vocabulary and validity of the structure of XML documents against grammatical rules of appropriate XML language.

An XML DTD can be either specified inside the document, or it can be kept in a separate document and then liked separately.

Syntax

Basic syntax of a DTD is as follows −

```
<!DOCTYPE element DTD identifier
[
  declaration1
  declaration2
  ........
]>
```

In the above syntax,

- The **DTD** starts with <!DOCTYPE delimiter.
- An **element** tells the parser to parse the document from the specified root element.
- **DTD identifier** is an identifier for the document type definition, which may be the path to a file on the system or URL to a file on the internet. If the DTD is pointing to external path, it is called **External Subset.**
- **The square brackets [ ]** enclose an optional list of entity declarations called *Internal Subset*.

Internal DTD

A DTD is referred to as an internal DTD if elements are declared within the XML files. To refer it as internal DTD, *standalone* attribute in XML declaration must be set to **yes**. This means, the declaration works independent of an external source.

Syntax

Following is the syntax of internal DTD −

<!DOCTYPE root-element [element-declarations]>

where *root-element* is the name of root element and *element-declarations* is where you declare the elements.

Example

Following is a simple example of internal DTD −

```
<?xml version = "1.0" encoding = "UTF-8" standalone = "yes" ?>
<!DOCTYPE address [
  <!ELEMENT address (name,company,phone)>
  <!ELEMENT name (#PCDATA)>
  <!ELEMENT company (#PCDATA)>
  <!ELEMENT phone (#PCDATA)>
]>

<address>
```

```
   <name>Tanmay Patil</name>
   <company>TutorialsPoint</company>
   <phone>(011) 123-4567</phone>
</address>
```

Let us go through the above code −

Start Declaration − Begin the XML declaration with the following statement.

```
<?xml version = "1.0" encoding = "UTF-8" standalone = "yes" ?>
```

DTD − Immediately after the XML header, the *document type declaration* follows, commonly referred to as the DOCTYPE −

```
<!DOCTYPE address [
```

The DOCTYPE declaration has an exclamation mark (!) at the start of the element name. The DOCTYPE informs the parser that a DTD is associated with this XML document.

DTD Body − The DOCTYPE declaration is followed by body of the DTD, where you declare elements, attributes, entities, and notations.

```
<!ELEMENT address (name,company,phone)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT company (#PCDATA)>
<!ELEMENT phone_no (#PCDATA)>
```

Several elements are declared here that make up the vocabulary of the <name> document. <!ELEMENT name (#PCDATA)> defines the element *name* to be of type "#PCDATA". Here #PCDATA means parse-able text data.

End Declaration − Finally, the declaration section of the DTD is closed using a closing bracket and a closing angle bracket (**]>**). This effectively ends the definition, and thereafter, the XML document follows immediately.

Rules

- The document type declaration must appear at the start of the document (preceded only by the XML header) − it is not permitted anywhere else within the document.
- Similar to the DOCTYPE declaration, the element declarations must start with an exclamation mark.
- The Name in the document type declaration must match the element type of the root element.

External DTD

In external DTD elements are declared outside the XML file. They are accessed by specifying the system attributes which may be either the legal *.dtd* file or a valid URL. To refer it as external DTD, *standalone* attribute in the XML declaration must be set as **no**. This means, declaration includes information from the external source.

Syntax

Following is the syntax for external DTD −

```
<!DOCTYPE root-element SYSTEM "file-name">
```

where *file-name* is the file with *.dtd* extension.

Example

The following example shows external DTD usage −

```
<?xml version = "1.0" encoding = "UTF-8" standalone = "no" ?>
<!DOCTYPE address SYSTEM "address.dtd">
<address>
   <name>Tanmay Patil</name>
   <company>TutorialsPoint</company>
   <phone>(011) 123-4567</phone>
</address>
```

The content of the DTD file **address.dtd** is as shown −

<!ELEMENT address (name,company,phone)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT company (#PCDATA)>
<!ELEMENT phone (#PCDATA)>

Types

You can refer to an external DTD by using either **system identifiers** or **public identifiers**.

System Identifiers

A system identifier enables you to specify the location of an external file containing DTD declarations. Syntax is as follows −

<!DOCTYPE name SYSTEM "address.dtd" [...]>

As you can see, it contains keyword SYSTEM and a URI reference pointing to the location of the document.

Public Identifiers

Public identifiers provide a mechanism to locate DTD resources and is written as follows −

<!DOCTYPE name PUBLIC "-//Beginning XML//DTD Address Example//EN">

As you can see, it begins with keyword PUBLIC, followed by a specialized identifier. Public identifiers are used to identify an entry in a catalog. Public identifiers can follow any format, however, a commonly used format is called **Formal Public Identifiers, or FPIs**.

ENTITY

Some characters are reserved in HTML.

If you use the less than (<) or greater than (>) signs in your text, the browser might mix them with tags.

Character entities are used to display reserved characters in HTML.

A character entity looks like this:

&entity_name;

OR

&#entity_number;

Non-breaking Space

A common character entity used in HTML is the non-breaking space: ** **

A non-breaking space is a space that will not break into a new line.

Two words separated by a non-breaking space will stick together (not break into a new line). This is handy when breaking the words might be disruptive.

Examples:

- § 10
- 10 km/h
- 10 PM

Another common use of the non-breaking space is to prevent browsers from truncating spaces in HTML pages.

If you write 10 spaces in your text, the browser will remove 9 of them. To add real spaces to your text, you can use the ** ** character entity.