COMMUNICATIONS

Data communications are the exchange of data between two devices via some form of transmission medium such as a wire or cable. For data communications to occur, the communicating devices must be part of a communication system made up of a combination of hardware (physical equipment) and software (programs).

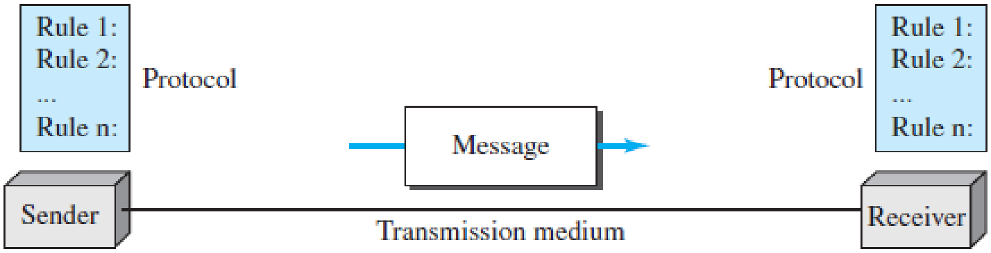
The effectiveness of a data communications system depends on four fundamental characteristics:

1. Delivery
2. Accuracy
3. Timeliness and
4. Jitter.

1. **Delivery**: The system must deliver data to the correct destination.
2. **Accuracy**: The system must deliver the data accurately.
3. **Timeliness**: The system must deliver data in a timely manner. This kind of delivery is called real-time transmission.
4. **Jitter**: Jitter refers to the variation in the packet arrival time.

Components

A data communications system has five components



Five Components of Data Communication System

1. **Message:** The message is the information (data) to be communicated.
   Popular forms of information include text, numbers, pictures, audio, and video.

2. **Sender:** The sender is the device that sends the data message.
   It can be a computer, workstation, telephone handset, video camera, and so on.

3. **Receiver**: The receiver is the device that receives the message.
   It can be a computer, workstation, telephone handset, television, and so on.

4. **Transmission medium**: The transmission medium is the physical path by which a message travels from sender to receiver.
   Examples of transmission media: Twisted-pair wire, coaxial cable, fiber-optic cable, and radio waves etc.

5. **Protocol**: A protocol is a set of rules that govern data communications. It represents an agreement between the communicating devices.
   Example: Without a protocol, two devices may be connected but not communicating, just as a person speaking French cannot be understood by a person who speaks only Japanese.

Data Representation

Information today comes in different forms such as **TEXT, NUMBERS, IMAGES, AUDIO, and VIDEO.**

1. **Text**
   In data communications, Text is represented as a bit pattern, a sequence of bits (0s or 1s). Different sets of bit patterns have been designed to represent text symbols. Each set is called a code, and the process of representing symbols is called coding.

2. **Numbers**
   Numbers are also represented by bit patterns. However, a code such as ASCII is not used to represent numbers; the number is directly converted to a binary number to simplify mathematical operations.

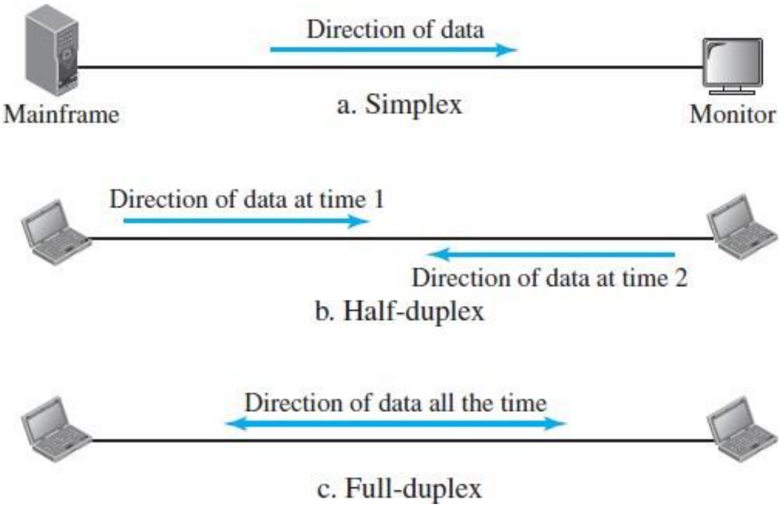
3. **Images**
   Images are also represented by bit patterns. In its simplest form, an image is composed of a matrix of pixels (picture elements), where each pixel is a small dot. The size of the pixel depends on the resolution.

4. **Audio**
   Audio refers to the recording or broadcasting of sound or music. Audio is by nature different from text, numbers, or images. It is continuous, not discrete.

5. **Video**
   Video refers to the recording or broadcasting of a picture or movie. Video can either be produced as a continuous entity (e.g., by a TV camera), or it can be a combination of images, each a discrete entity, arranged to convey the idea of motion.

Data Flow

Communication between two devices can be simplex, half-duplex, or full-duplex


Direction of data
Mainframe
Monitor
Direction of data at time 1
Direction of data at time 2
Direction of data all the time

a. Simplex

b. Half-duplex

c. Full-duplex

Simplex

In **simplex mode,** the communication is unidirectional (one-way). Only one of the two devices on a link can transmit; the other can only receive. The simplex mode can use the entire capacity of the channel to send data in one direction

Example: The keyboard can only introduce input; the monitor can only accept output.

Half-Duplex

In **half-duplex mode,** each station can both transmit and receive, but not at the same time. When one device is sending, the other can only receive, and vice versa. The entire capacity of the channel can be utilized for each transmission direction.

Example: Walkie-talkies and CB (citizens band) radios.

Full-Duplex

In **full-duplex mode**, both stations can transmit and receive simultaneously. The capacity of the channel, however, must be divided between the two directions

Example: Telephone network.

NETWORKS

A network is the interconnection of a set of devices capable of communication. These devices in a network are connected using wired or wireless transmission media such as cable or air.

Example: Device can be a host (Computer, desktop, laptop, work station, cellular phone, or security system) or (Connecting device such as a router, a switch, a modem and so on).

Network Criteria

A network must be able to meet a certain number of criteria. The most important of these are **Performance, Reliability and Security.**

1. **Performance**
   The performance of a network depends on a number of factors, including the number of users, the type of transmission medium, the capabilities of the connected hardware, and the efficiency of the software. Performance is often evaluated by two networking metrics: throughput and delay.

2. **Reliability**
   Network reliability is measured by the frequency of failure, the time it takes a link to recover from a failure, and the network's robustness in a catastrophe.

3. **Security**
   Network security issues include protecting data from unauthorized access, protecting data from damage and development, and implementing policies and procedures for recovery from breaches and data losses.

Transit time and response time:

Transit time is the amount of time required for a message to travel from one device to another. Response time is the elapsed time between an inquiry and a response.

Physical Structures Type of Connection

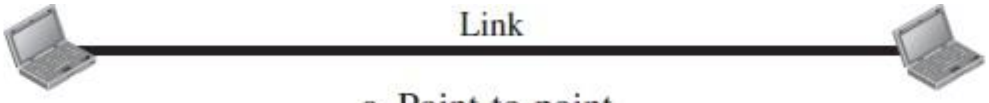
A network is two or more devices connected through links. A link is a communications pathway that transfers data from one device to another. There are two possible types of connections: point- to-point and multipoint.

Point-to-Point

A **point-to-point connection** provides a dedicated link between two devices. The entire capacity of the link is reserved for transmission between those two devices.

Most point-to-point connections use an actual length of wire or cable to connect the two ends, but other options, such as microwave or satellite links, are also possible.

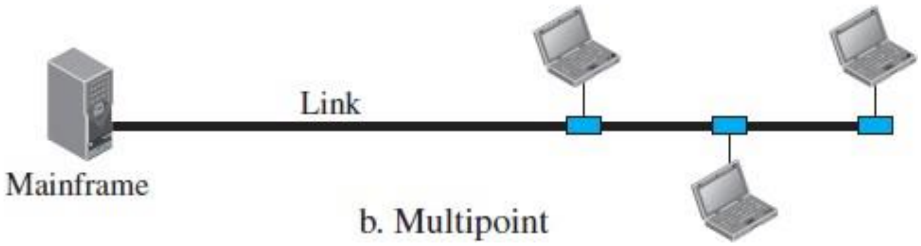
Example: When we change television channels by infrared remote control, we are establishing a point-to-point connection between the remote control and the television’s control system.



a. Point-to-point

Multipoint

A **multipoint** (also called **multidrop**) **connection** is one in which more than two specific devices share a single link. In a multipoint environment, the capacity of the channel is shared, it is a timeshared connection.



b. Multipoint

Physical Topology

The topology of a network is the geometric representation of the relationship of all the links and linking devices is connected physically (usually called **nodes**). There are four basic topologies possible: **MESH, STAR, BUS, and RING.**

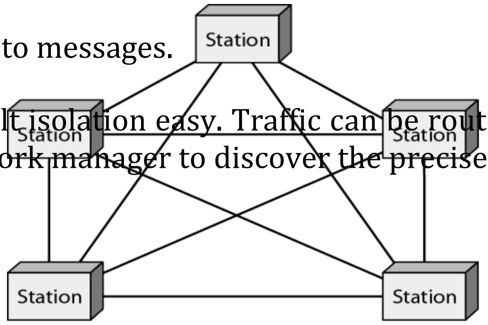
Mesh Topology

In a **mesh topology,** every device has a dedicated point-to-point link to every other device.

A mesh offers several advantages over other network topologies.

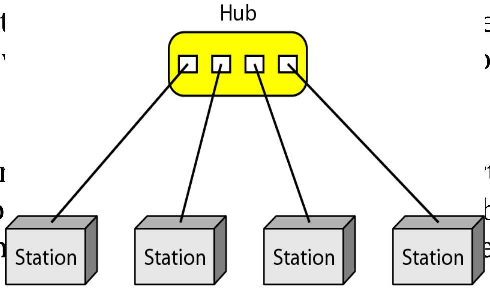
1. First, the use of dedicated links guarantees that each connection can carry its own data load, thus eliminating the traffic problems that can occur when links must be shared by multiple devices.
2. Second, a mesh topology is robust. If one link becomes unusable, it does not incapacitate the entire system.
3. Third, there is the advantage of privacy or security. When every message travels along a dedicated line, only the intended recipient sees it.
   Physical boundaries prevent other users from gaining access to messages.

4. Finally, point-to-point links make fault identification and fault isolation easy. Traffic can be routed to avoid links with suspected problems. This facility enables the network manager to discover the precise location of the fault and aids in finding its cause and solution.



Star Topology

In a star topology, each device has a dedicated point-to-point link only to a central controller, usually called a hub. The devices are not directly linked to one another.



Star topology does not allow direct traffic between devices. The cont evice wants to send data to another, it sends the data to the controller, other connected device.

A star topology is less expensive than a mesh topology; each device t to connect it to any number of others. This factor also makes it easy to bling needs to be housed, and additions, moves, and deletions involve on evice and the hub.

Other advantages include robustness. If one link fails, only that link is affected. All other links remain active. This factor also lends itself to easy fault identification and fault isolation. As long as the hub is working, it can be used to monitor link problems and bypass defective links.

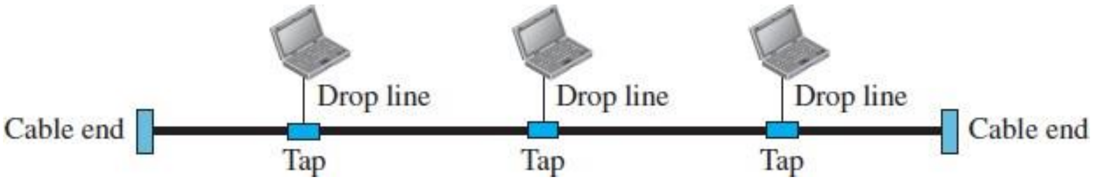
One big disadvantage of a star topology is the dependency of the whole topology on one single, the hub. If the hub goes down, the whole system is dead.

The star topology is used in local-area networks (LANs), High-speed LANs often use a star topology with a central hub.

Bus Topology

A **bus topology** is multipoint. One long cable acts as a **backbone** to link all the devices in a network. Nodes are connected to the bus cable by drop lines and taps.

A drop line is a connection running between the device and the main cable. A tap is a connector to create a contact with the metallic core.



Advantages of a bus topology include ease of installation. Backbone cable can be laid along the most efficient path, and then connected to the nodes by drop lines of various lengths. In this way, a bus uses less cabling than mesh or star topologies.

Disadvantages include difficult reconnection and fault isolation. There is a limit on the number of taps a bus can support and on the distance between those taps.

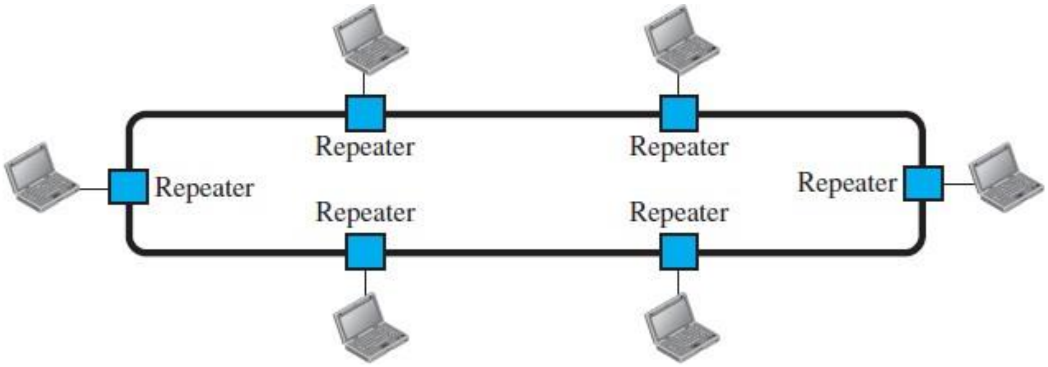
Ring Topology

In a **ring topology,** each device has a dedicated point-to-point connection with only the two devices on either side of it. A signal is passed along the ring in one direction, from device to device, until it reaches its destination. Each device in the ring incorporates a repeater. When a device receives a signal intended for another device, its repeater regenerates the bits and passes them along.

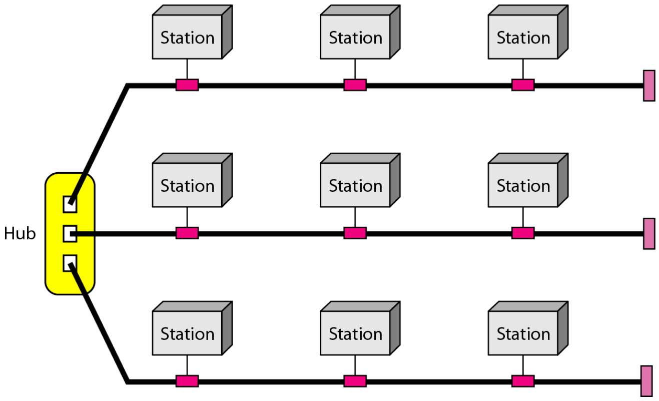
Advantages
A ring is relatively easy to install and reconfigure. Fault isolation is simplified.

Disadvantage
In a simple ring, a break in the ring (such as a disabled station) can disable the entire network.

This weakness can be solved by using a dual ring or a switch capable of closing off the break. Ring topology was prevalent when IBM introduced its local-area network, Token Ring. Today, the need for higher-speed LANs has made this topology less popular.



Hybrid: The combination of the above topologies



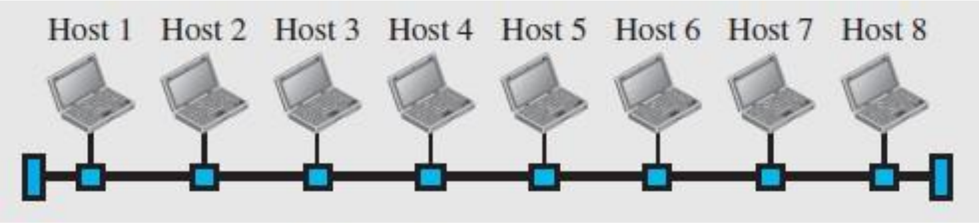
NETWORK TYPES

Two types of networks, LANs and WANs, we define switching, which is used to connect networks to form an internetwork (a network of networks).

Local Area Network

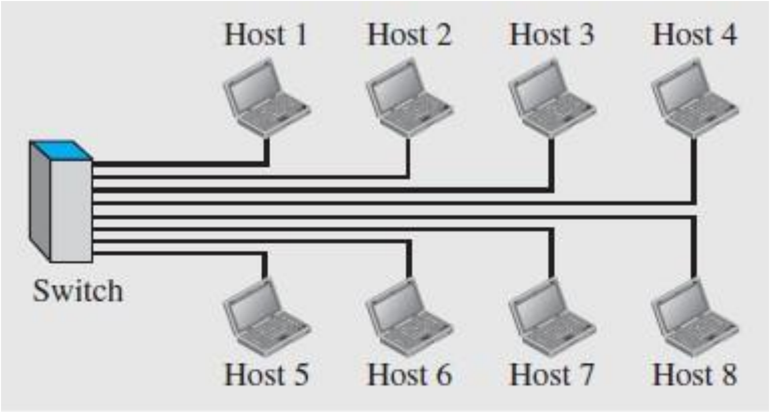
- A local area network (LAN) is usually privately owned and connects some hosts in a single office, building, or campus.
- Depending on the needs of an organization, a LAN can be as simple as two PCs and a printer in someone's home office, or it can extend throughout a company and include audio and video devices.
- Each host in a LAN has an identifier, an address that uniquely defines the host in the LAN.
- A packet sent by a host to another host carries both the source host's and the destination host's addresses.
- A LAN is normally limited in size, spanning an office, a building, or a campus.
- A LAN interconnects hosts

In the past, all hosts in a network were connected through a common cable, which meant that a packet sent from one host to another was received by all hosts. The intended recipient kept the packet; the others dropped the packet.



a. LAN with a common cable (past)

Today, most LANs use a smart connecting switch, which is able to recognize the destination address of the packet and guide the packet to its destination without sending it to all other hosts. The switch alleviates the traffic in the LAN and allows more than one pair to communicate with each other at the same time if there is no common source and destination among them.



b. LAN with a switch (today)

Wide Area Network

A wide area network (WAN) is also an interconnection of devices capable of communication.

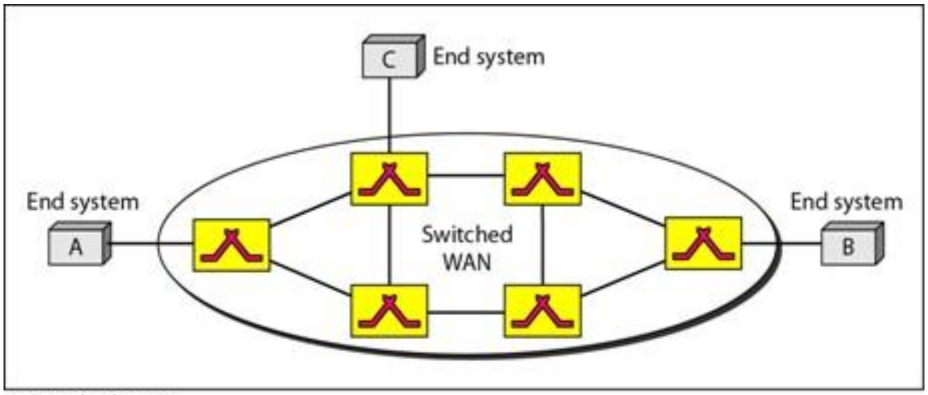
- WAN has a wider geographical span, spanning a town, a state, a country, or even the world.
- WAN interconnects connecting devices such as switches, routers, or modems.
- WAN is normally created and run by communication companies and leased by an organization that uses it.

Two distinct examples of WANs today:

- Point-to-point WANs and
- Switched WANs.

Switched WAN

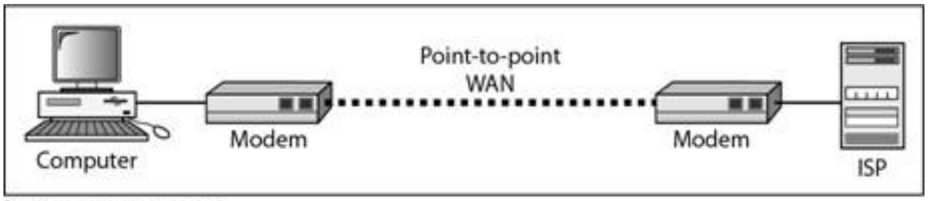
- A switched WAN is a network with more than two ends.
- A switched WAN is used in the backbone of global communication today.
- A switched WAN is a combination of several point-to-point WANs that are connected by switches.



a. Switched WAN

Point-to-Point WAN

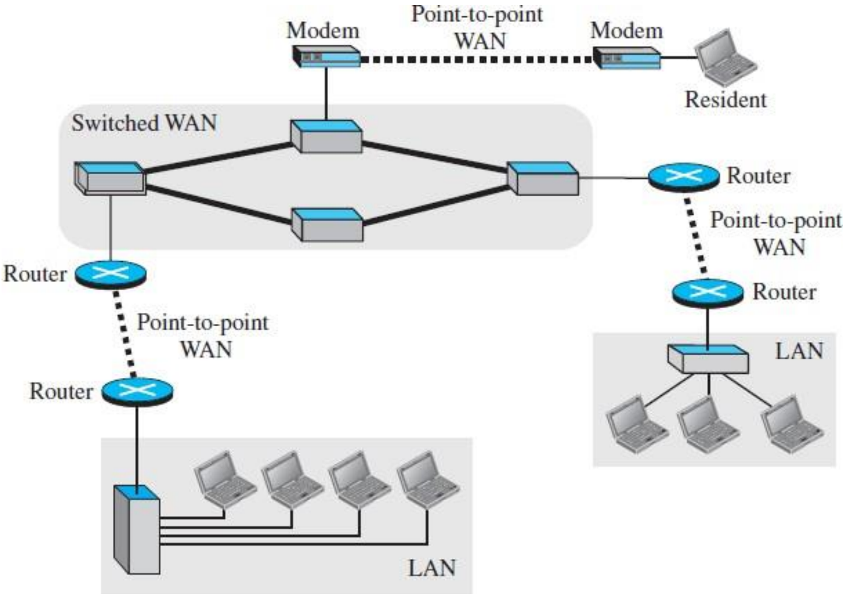
A point-to-point WAN is a network that connects two communicating devices through a transmission media (cable or air).



b. Point-to-point WAN

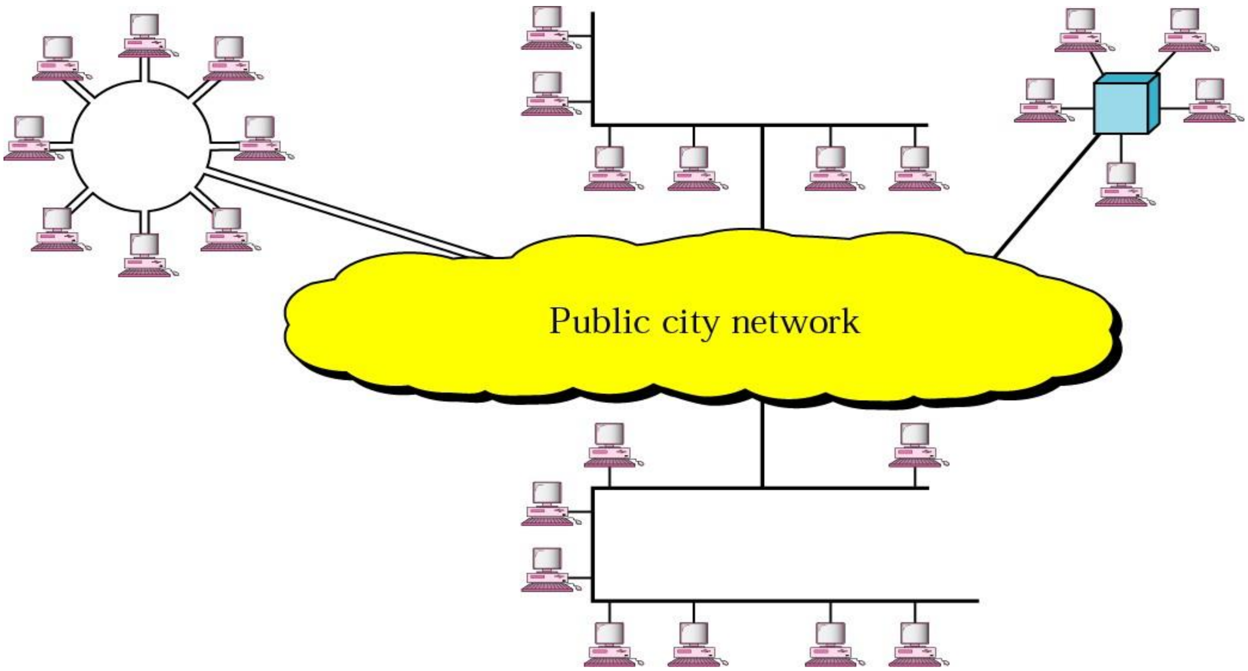
Difference between LANs and WANs

LANs	WANs
limited in size	wider geographical span
A LAN interconnects hosts	WAN interconnects connecting devices
privately owned by the organization	Created and run by communication companies



A heterogeneous network made of four WANs and three LANs

MAN: Metropolitan Area Network



Switching

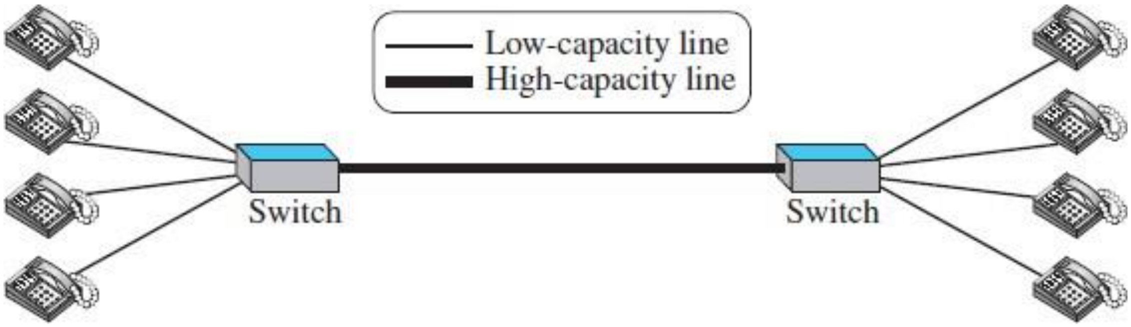
An internet is a **switched network** in which a switch connects at least two links together. A switch needs to forward data from a network to another network when required.

Types of switched networks

- Circuit-switched and
- Packet-switched networks.

Circuit-Switched Network

- In a **circuit-switched network,** a dedicated connection, called a circuit, is always available between the two end systems; the switch can only make it active or inactive.



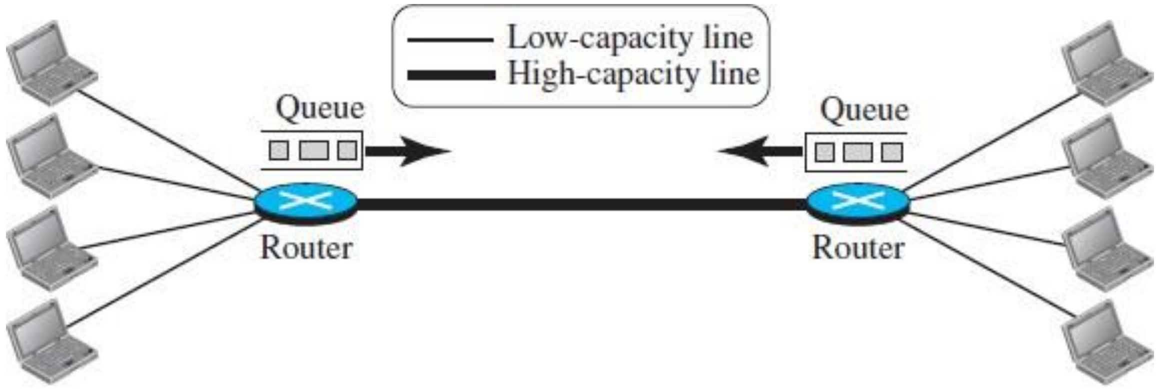
- Figure shows a very simple switched network that connects four telephones to each end, because circuit switching was very common in telephone networks in the past.
- The four telephones at each side are connected to a switch.
- The switch connects a telephone set at one side to a telephone set at the other side.
- The thick line connecting two switches is a high-capacity communication line that can handle four voice

communications at the same time; the capacity can be shared between all pairs of telephone sets.

- The switches used in this example have forwarding tasks but no storing capability.
- Circuit-switched network is efficient only when it is working at its full capacity.

Packet-Switched Network

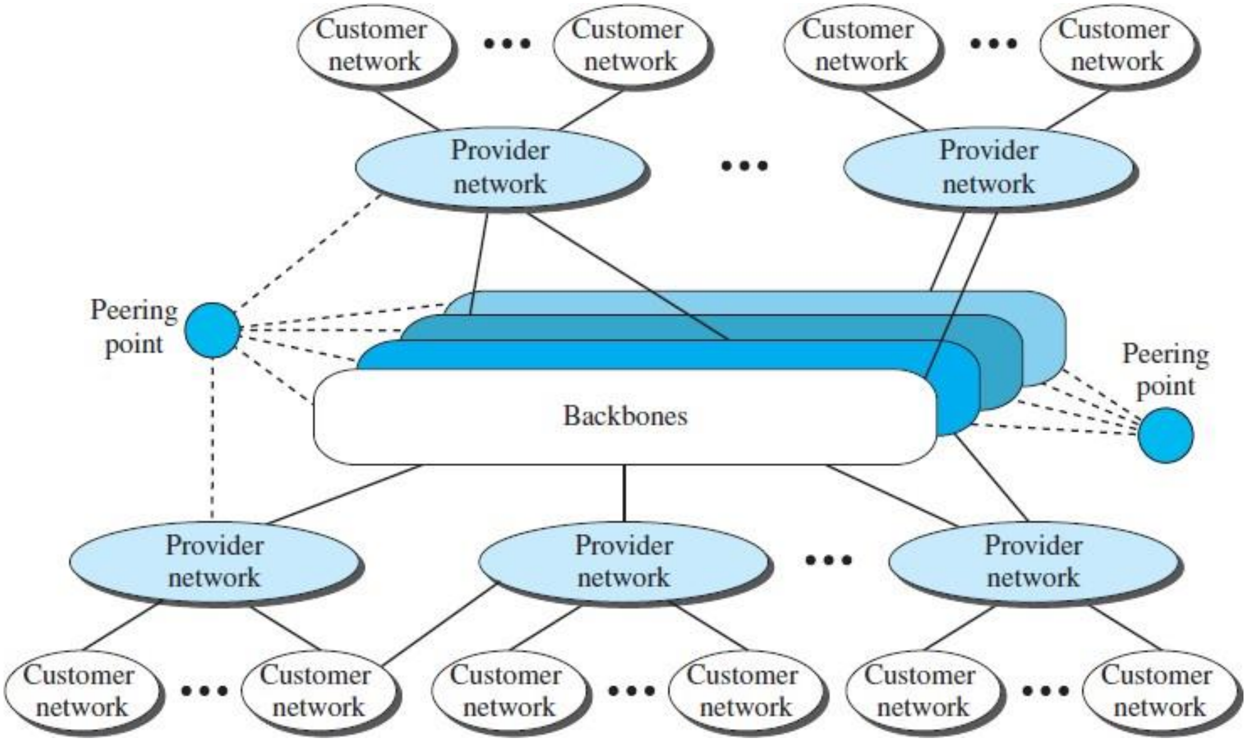
- In a computer network, the communication between the two ends is done in blocks of data called **packets.**
- This allows us to make the switches function for both storing and forwarding because a packet is an independent entity that can be stored and sent later.



- Figure shows a small packet-switched network that connects four computers at one site to four computers at the other site.
- A router in a packet-switched network has a queue that can store and forward the packet.
- Packet-switched network is more efficient than a circuit-switched network, but the packets may encounter some delays.

1.3.4 The Internet

- An internet is two or more networks that can communicate with each other. Internet is composed of thousands of interconnected networks.



- Figure shows a conceptual (not geographical) view of the Internet. The figure shows the Internet as several backbones, provider networks, and customer networks.
- At the top level, the *backbones* are large networks owned by some communication companies such as Sprint, Verizon (MCI), AT&T, and NTT.
- The backbone networks are connected through some complex switching systems, called *peering points*.
- At the second level, there are smaller networks, called *provider networks*.
- The provider networks are connected to backbones and sometimes to other provider networks. The *customer networks* are networks at the edge of the Internet that actually use the services provided by the Internet.
- Backbones and provider networks are also called **Internet Service Providers (ISPs).** The backbones are often referred to as *international ISPs;* the provider networks are often referred to as *national* or *regional ISPs*.

Accessing the Internet

The Internet today is an internetwork that allows any user to become part of it. The user, however, needs to be physically connected to an ISP. The physical connection is normally done through a point-to-point WAN.

1. ***Using Telephone Networks***

Today option for residences and small businesses to connect to the Internet is to change the voice line between the residence or business and the telephone center to a point-to-point WAN. This can be done in two ways.

- ***Dial-up service.*** The first solution is to add to the telephone line a modem that converts data to voice. The software installed on the computer dials the ISP and imitates making a telephone connection. Unfortunately, the dial-up service is very slow, and when the line is used for Internet connection, it cannot be used for telephone (voice) connection. It is only useful for small residences.

- ***DSL Service.*** Since the advent of the Internet, some telephone companies have upgraded their telephone lines to provide higher speed Internet services to residences or small businesses. The DSL service also allows the line to be used simultaneously for voice and data communication.

2. ***Using Cable Networks***

More and more residents over the last two decades have begun using cable TV services instead of antennas to receive TV broadcasting. The cable companies have been upgrading their cable networks and connecting to the Internet. It provides a higher speed connection, but the speed varies depending on the number of neighbors that use the same cable.

3. ***Using Wireless Networks***

Wireless connectivity has recently become increasingly popular. A household or a small business can use a combination of wireless and wired connections to access the Internet. With the growing wireless WAN access, a household or a small business can be connected to the Internet through a wireless WAN.

4. ***Direct Connection to the Internet***

A large organization or a large corporation can itself become a local ISP and be connected to the Internet. This can be done if the organization or the corporation leases a high-speed WAN from a carrier provider and connects itself to a regional ISP.

Example, a large university with several campuses can create an internetwork and then connect the internetwork to the Internet.

Important Questions:

1. Identify the five components of a data communications system.
2. What are the three criteria necessary for an effective and efficient network?
3. What are the advantages of a multipoint connection over a point-to-point one?
4. What are the two types of line configuration?
5. Categorize the four basic topologies in terms of line configuration.
6. What is the difference between half-duplex and full-duplex transmission modes?
7. Name the four basic network topologies, and cite an advantage of each type.
8. For *n* devices in a network, what is the number of cable links required for a mesh, ring, bus, and star topology?
9. What are some of the factors that determine whether a communication system is a LAN or WAN?
10. What is an internet? What is the Intranet?
11. Why are protocols needed?

PROTOCOL LAYERING

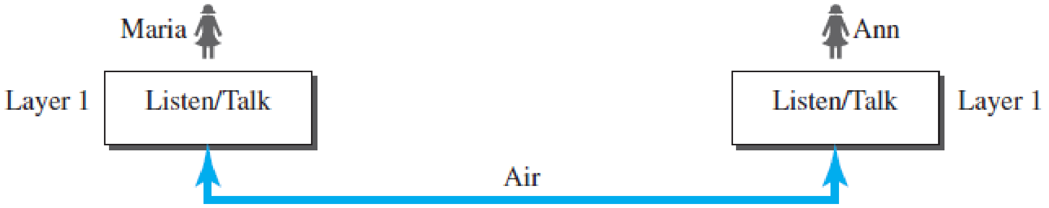
We defined the term *protocol* in Chapter 1. In data communication and networking, a protocol defines the rules that both the sender and receiver and all intermediate devices need to follow to be able to communicate effectively. When communication is simple, we may need only one simple protocol; when the communication is complex, we may need to divide the task between different layers, in which case we need a protocol at each layer, or **protocol layering.**

Scenarios

Let us develop two simple scenarios to better understand the need for protocol layering.

First Scenario

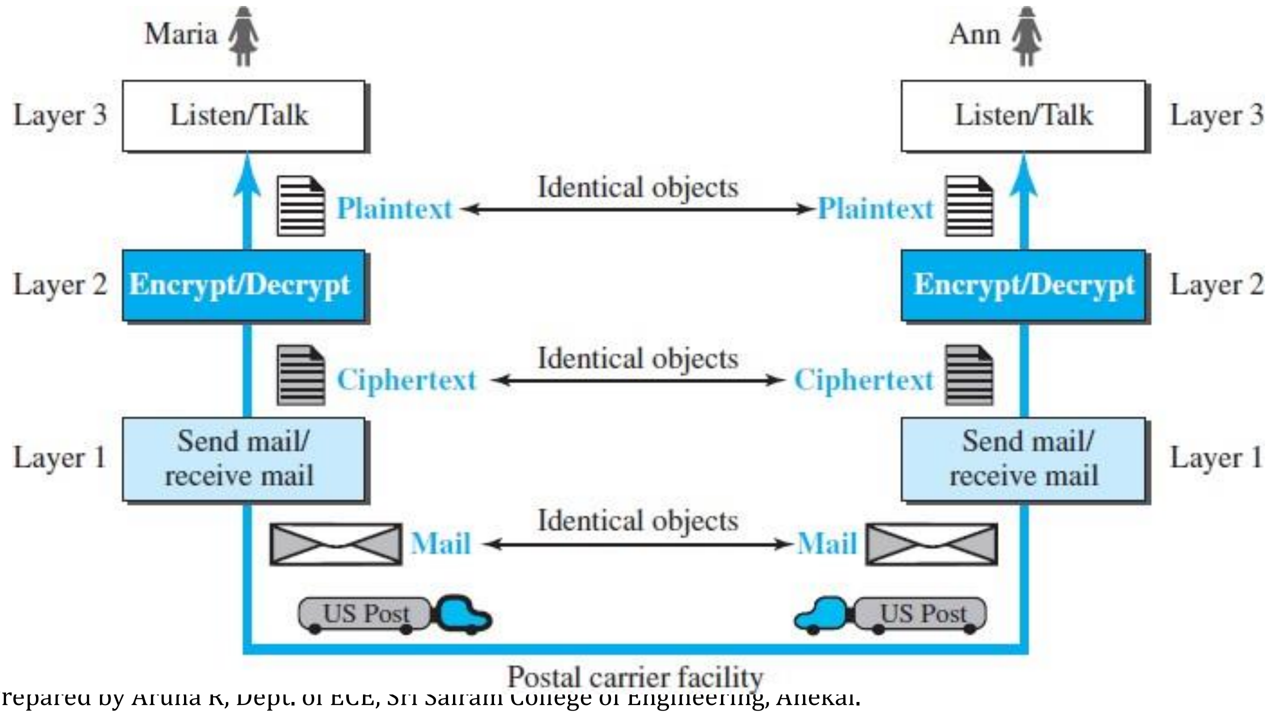
In the first scenario, communication is so simple that it can occur in only one layer. Assume Maria and Ann are neighbors with a lot of common ideas. Communication between Maria and Ann takes place in one layer, face to face, in the same language, as shown in Figure below.



A single-layer protocol

Second Scenario

- In the second scenario, the communication is happens from a far place, continue their communication and exchange ideas using regular mail through the post office.
- However, they do not want their ideas to be revealed by other people if the letters are intercepted. They agree on an encryption/decryption technique.
- The sender of the letter encrypts it to make it unreadable by an intruder; the receiver of the letter decrypts it to get the original letter.
- Now the communication takes place in three layers, as shown in Figure.
- Protocol layering enables us to divide a complex task into several smaller and simpler tasks.



Principles of Protocol Layering

Two principles of protocol layering.

First Principle

The first principle dictates that if we want bidirectional communication, need to make each layer so that it is able to perform two opposite tasks, one in each direction.

Example

- The third layer task is to listen (in one direction) and *talk* (in the other direction).
- The second layer needs to be able to encrypt and decrypt.
- The first layer needs to send and receive mail.

Second Principle

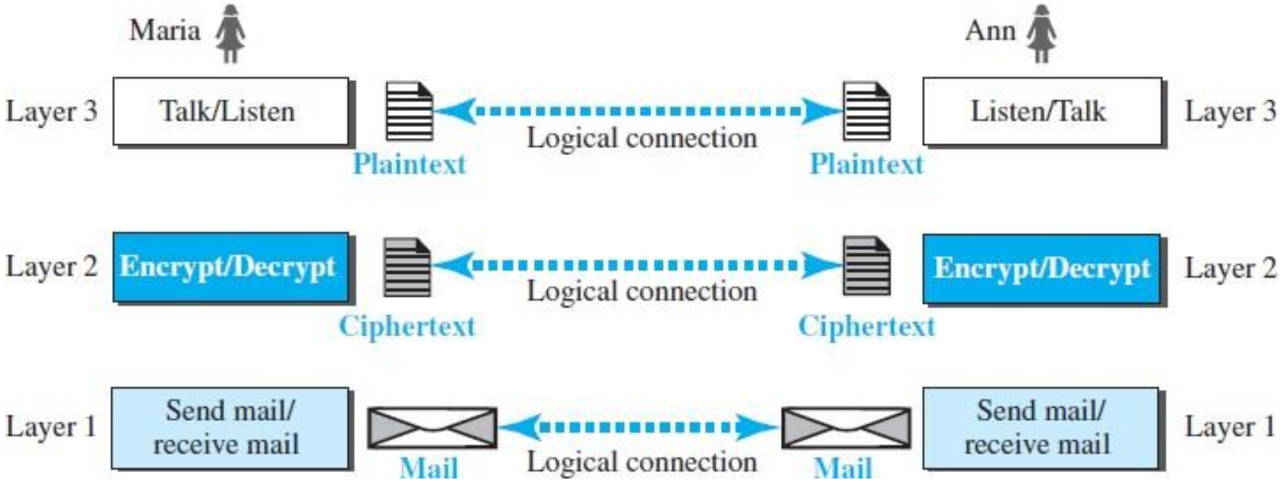
The second principle is need to follow in protocol layering is that the two objects under each layer at both sites should be identical.

Example

- The object under layer 3 at both sites should be a plain text letter.
- The object under layer 2 at both sites should be a cipher text letter.
- The object under layer 1 at both sites should be a piece of mail.

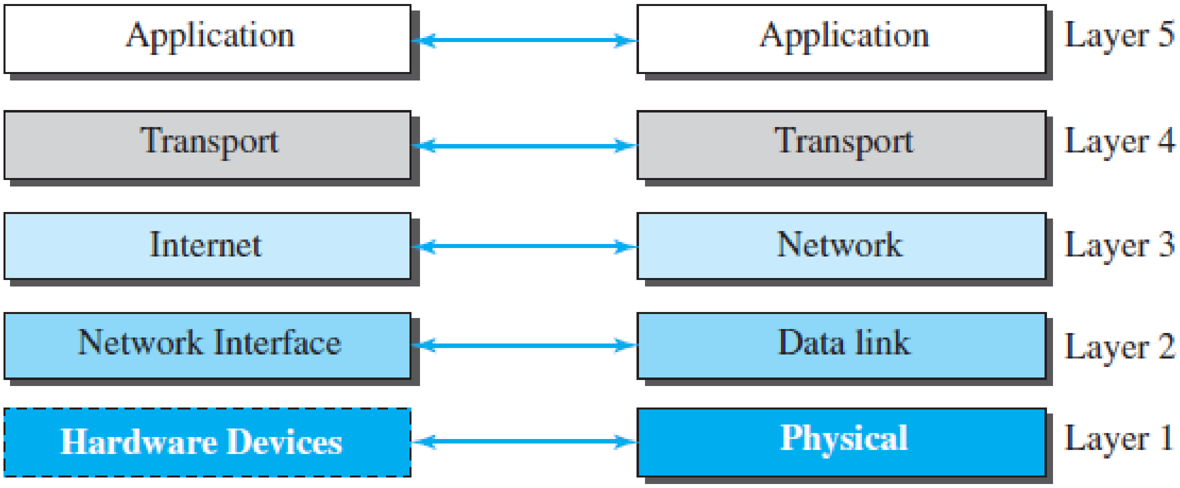
Logical Connections

- Logical connection between each layer as shown in Figure. This means that, layer-to-layer communication.
- Logical (imaginary) connection at each layer through which they can send the object created from that layer.



TCP/IP PROTOCOL SUITE: (Transmission Control Protocol/Internet Protocol).

- TCP/IP is a protocol suite (a set of protocols organized in different layers) used in the Internet today.
- It is a hierarchical protocol made up of interactive modules, each of which provides a specific functionality.
- The term *hierarchical* means that each upper level protocol is supported by the services provided by one or more lower level protocols.
- The original TCP/IP protocol suite was defined as four software layers built upon the hardware. However, TCP/IP is a five-layer model. Figure shows both configurations.

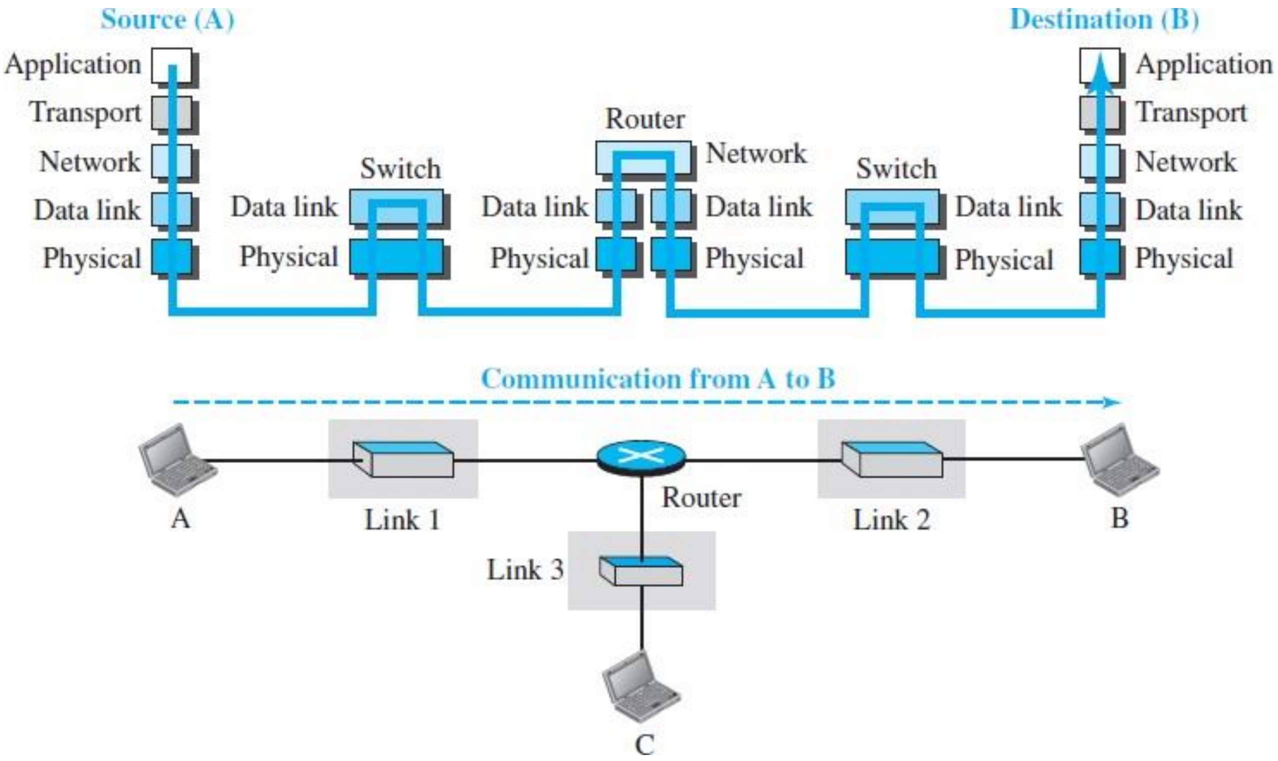


a. Original layers

b. Layers used in this book

Layered Architecture

To show how the layers in the TCP/IP protocol suite are involved in communication between two hosts, we assume that we want to use the suite in a small internet made up of three LANs (links), each with a link-layer switch. We also assume that the links are connected by one router, as shown in Figure.



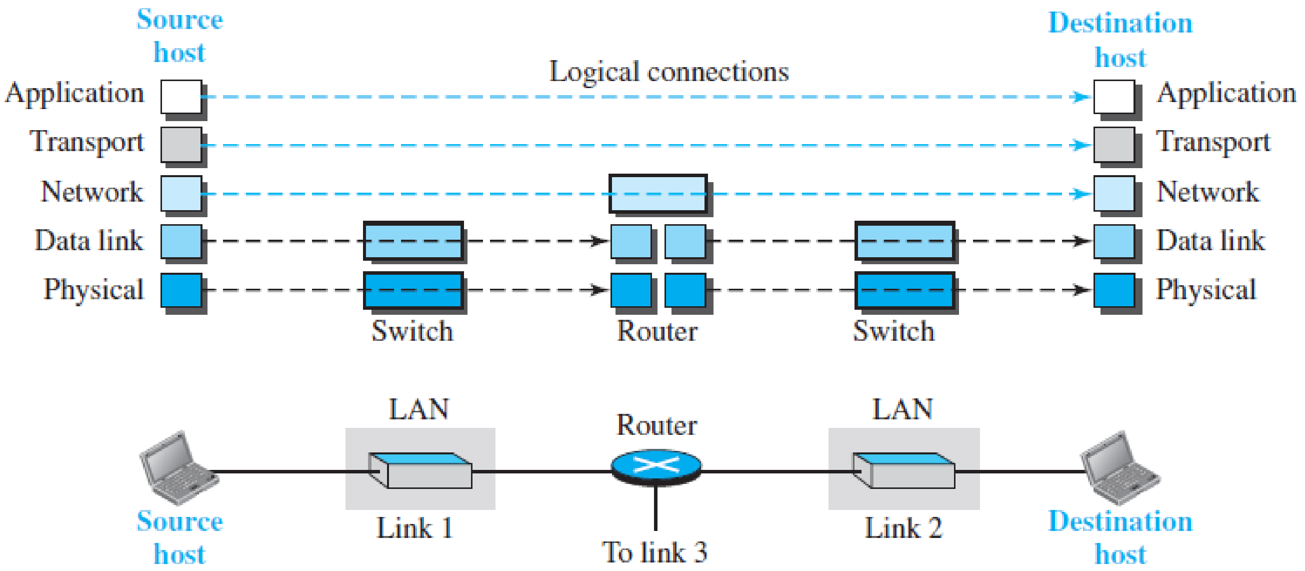
Example, if computer A communicates with computer B.

- The five communicating devices in this communication: source host (computer A), the link- layer switch in link 1, the router, the link-layer switch in link 2, and the destination host (computer B).
- Each device is involved with a set of layers depending on the role of the device in the internet.
- The two hosts are involved in all five layers; the source host needs to create a message in the application layer and send it down the layers so that it is physically sent to the destination host.

- The destination host needs to receive the communication at the physical layer and then deliver it through the other layers to the application layer.
- The router is involved in only three layers; network, Data link and Physical layer.
- A link-layer switch in a link, however, is involved only in two layers, data-link and physical.

Layers in the TCP/IP Protocol Suite

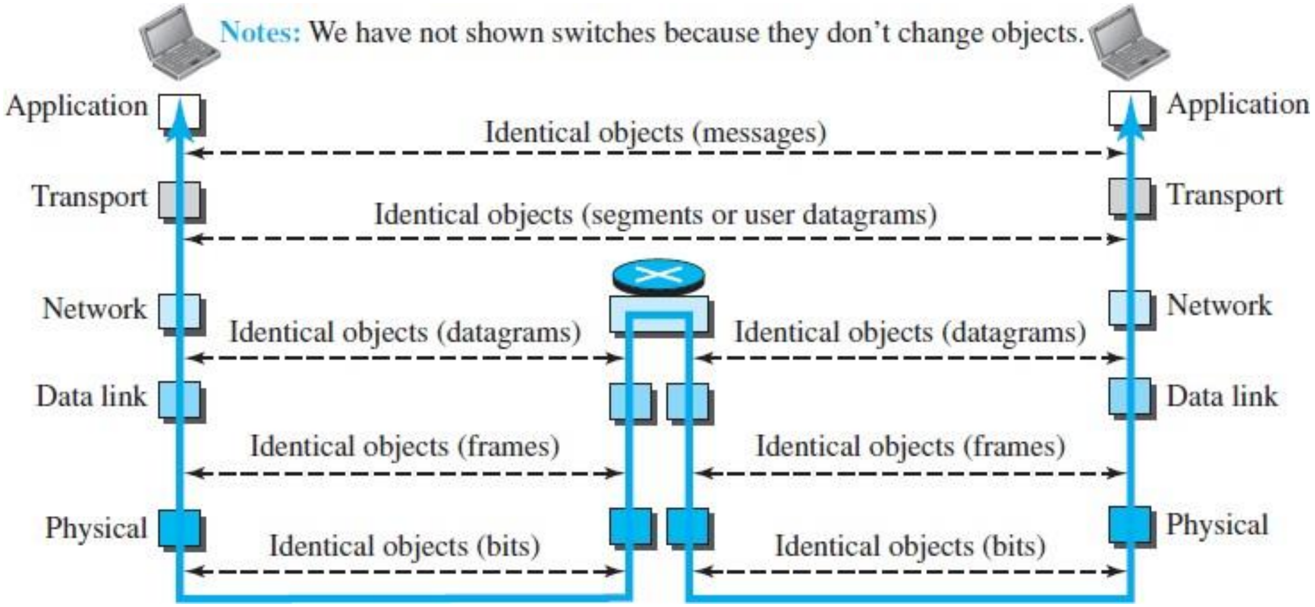
Figure shows logical connections in our simple internet. Using logical connections makes it easier for us to think about the duty of each layer.



The duty of the application, transport, and network layers is end-to-end. However, the duty of the data-link and physical layers is hop-to-hop, in which a hop is a host or router.

In other words, the domain of duty of the top three layers is the internet, and the domain of duty of the two lower layers is the link.

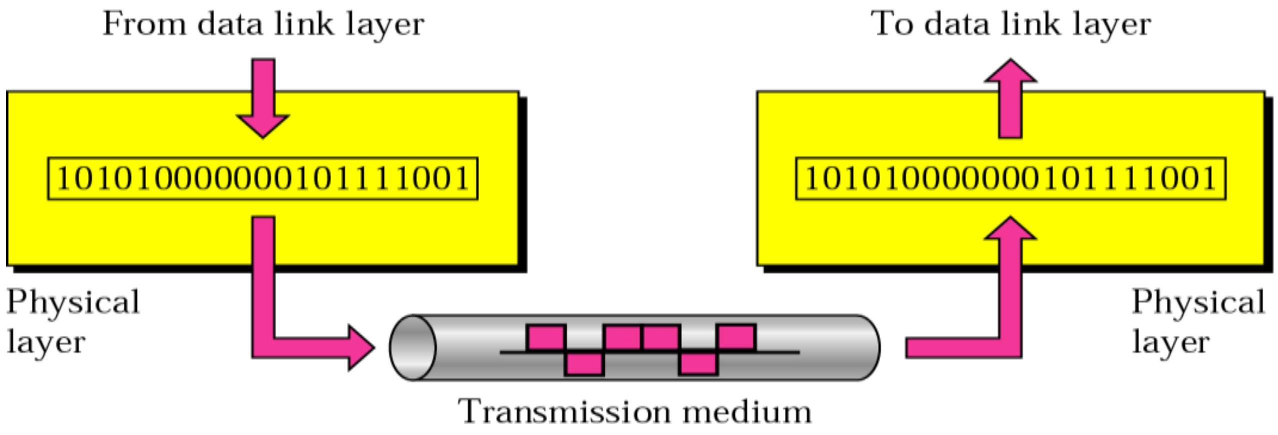
Figure shows the second principle for protocol layering, shows the identical objects below each layer related to each device. Note that the link between two hops does not change the object.


Notes: We have not shown switches because they don't change objects.
Application
Transport
Network
Data link
Physical
Application
Transport
Network
Data link
Physical
Identical objects (messages)
Identical objects (segments or user datagrams)
Identical objects (datagrams)
Identical objects (datagrams)
Identical objects (frames)
Identical objects (frames)
Identical objects (bits)
Identical objects (bits)

Description of Each Layer

i) Physical Layer:

- The physical layer is responsible for movements of individual bits from one hop (node) to the next
  - Mechanical and electrical specification, the procedures and functions

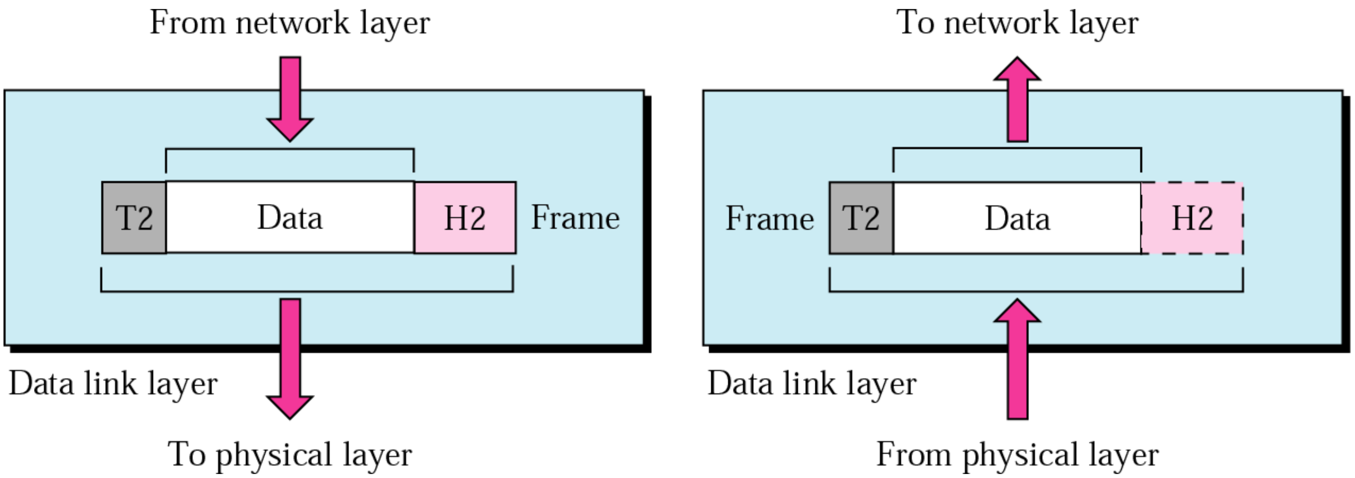


Physical Layer: Duties

- Physical characteristics of interfaces and media
- Representation of bits, encoding
- Transmission of Data rate and bit duration
- Synchronization of bits b/n sender & receiver
- Line configuration: Point-to-Point / Multi-Point
- Physical topology of n/w connection
- Transmission mode: simplex, half-duplex / full-duplex, analog/digital, FDM/TDM etc..

ii) Data Link Layer:

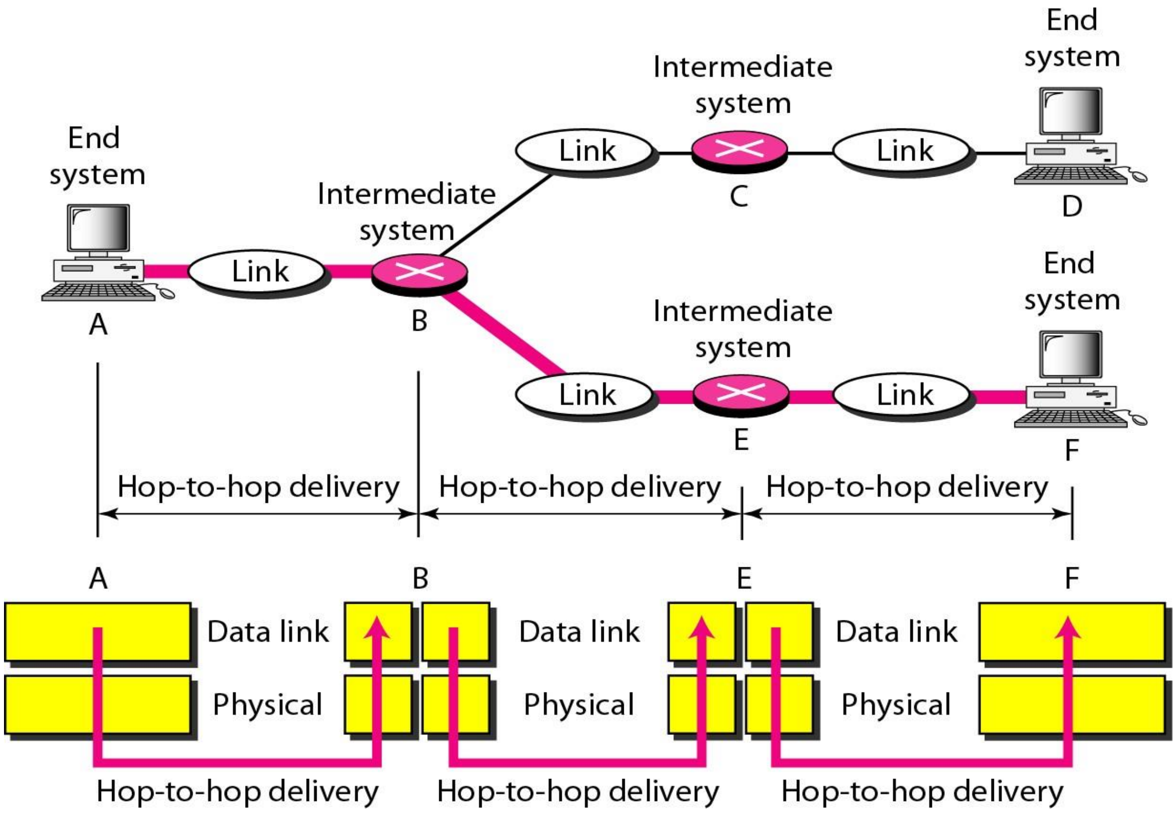
- The data link layer is responsible for moving frames from one hop (node) to the next
- Transform the physical layer to a reliable (error-free) link
- Bit patterns at the beginning (Header) and the end (Trailer) of the frame are attached to the data block



Data Link Layer: Duties

- Framing (manageable data units) and link management
- Physical addressing to identify sender/receiver

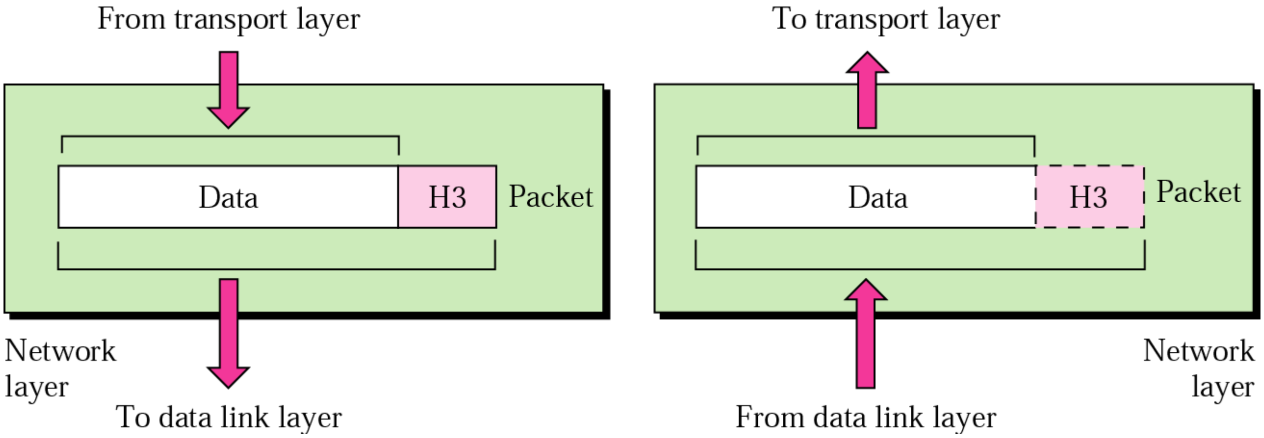
- ♣ Flow control mechanism to avoid flooding of the receiver
- ♣ Error control mechanism to detect and retransmit the lost frames
- ♣ Access control to determine which device has the control over the link



Hop-to-Hop Delivery

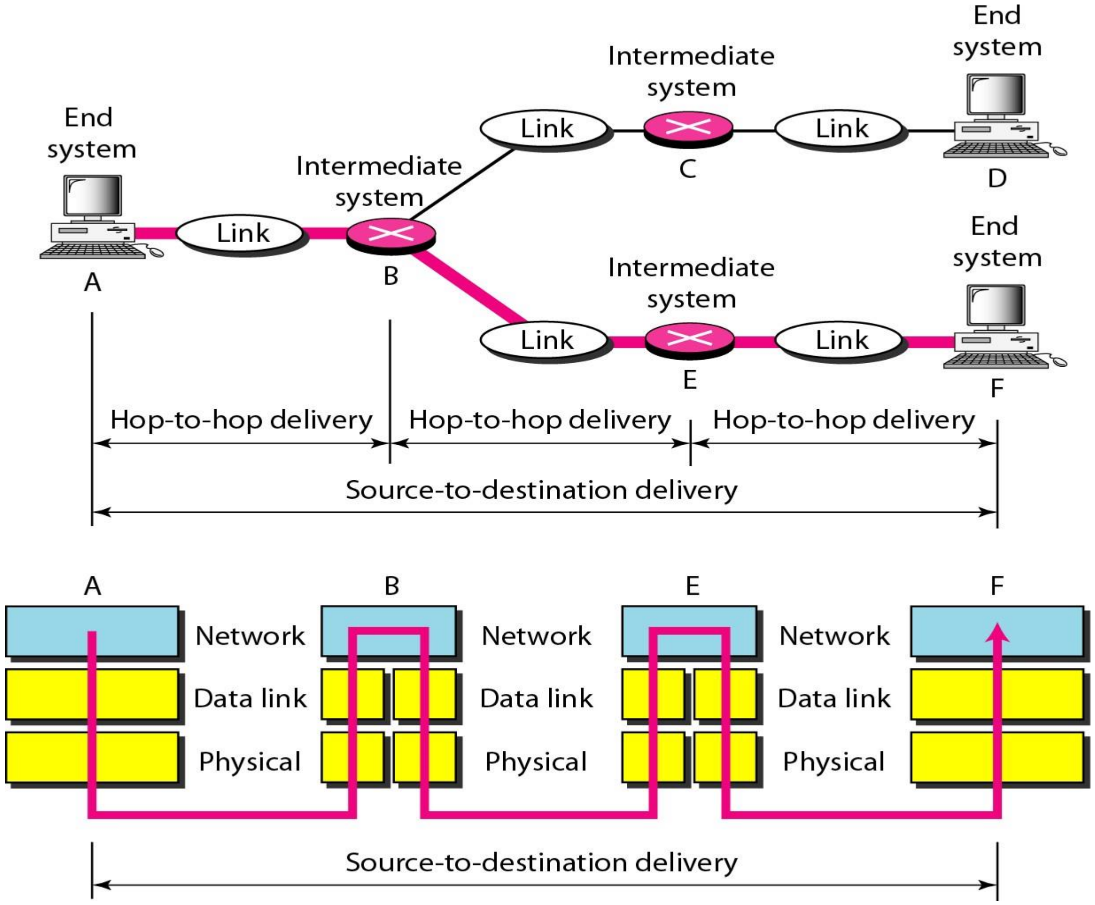
iii) Network Layer:

- ♣ The network layer is responsible for the delivery of packets from the source host to the destination host



Network Layer: Duties

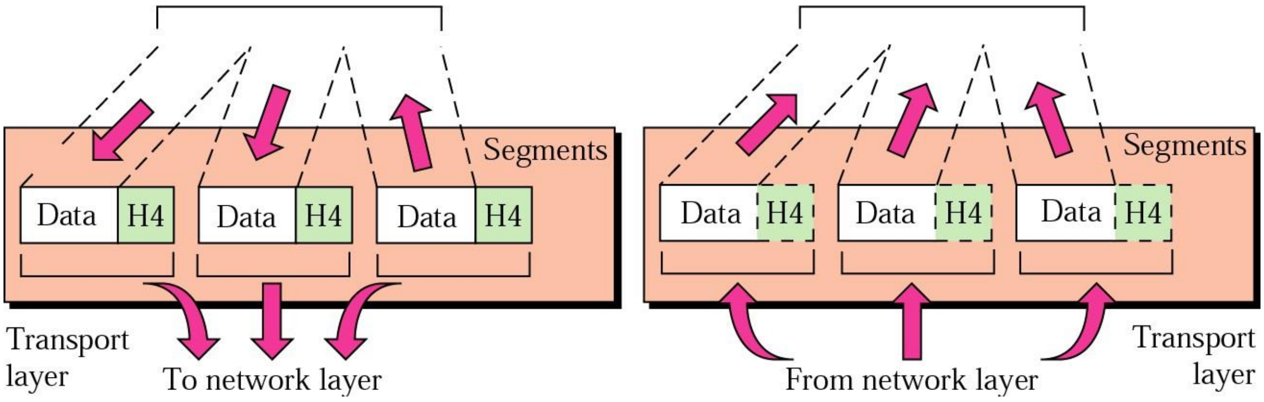
- ♣ Routing of packets from source to final destination
- ♣ Congestion control during traffic is high
- ♣ Logical address to be attached to specify the sender/receiver



Source-to-destination delivery

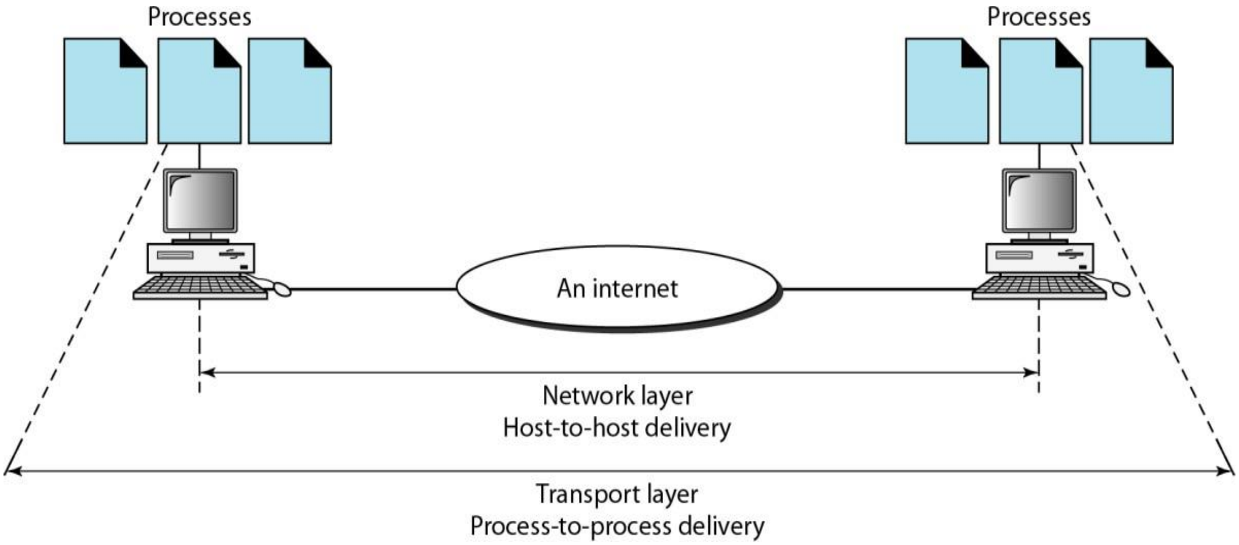
iv) Transport Layer:

♣ Transport layer is responsible for the delivery of a message from one process to another.



Transport Layer: Duties

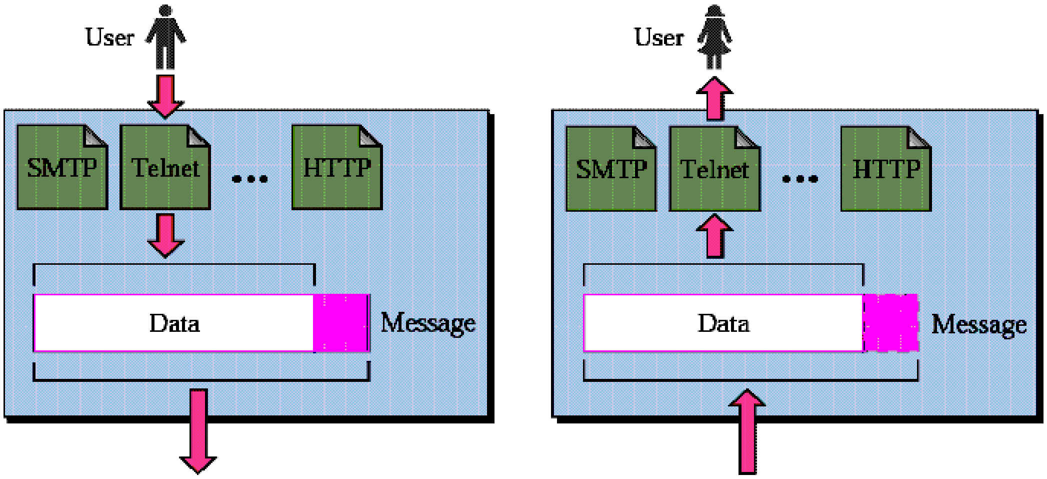
♣ Service-point (port) addressing: to ensure process-to-process delivery
♣ Segmentation and reassembly
♣ Connection control: connectionless / connection oriented service
♣ Flow control: control on transaction
♣ Error control: error free delivery



Reliable Process-to-Process Delivery of a Message

v) Application Layer:

The Application layer is responsible for providing services to the user.

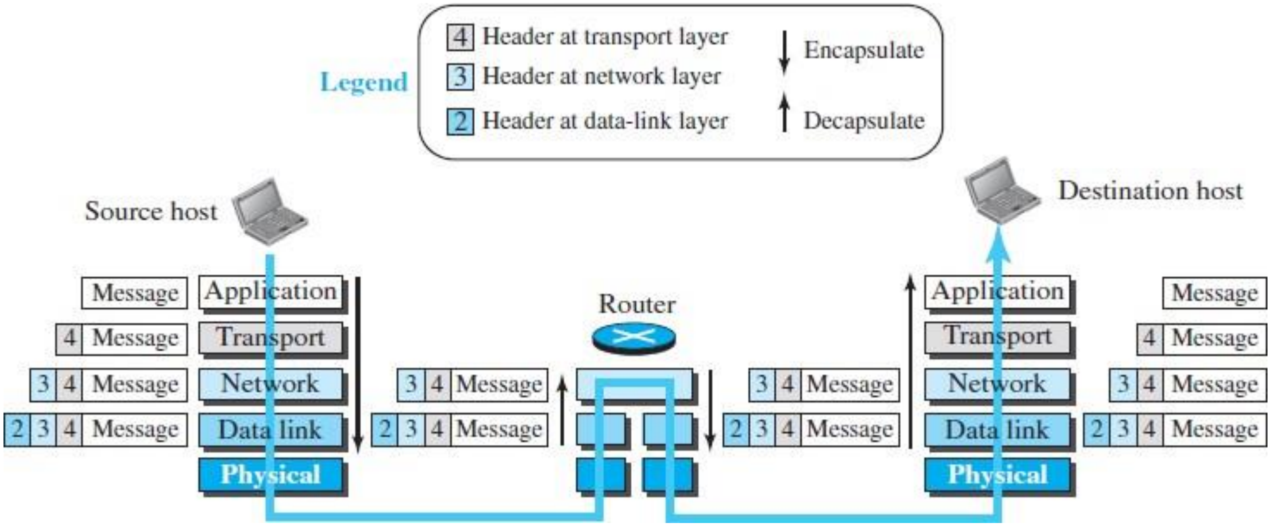


Application Layer: Duties

- ♣ Network virtual terminal (Remote access)
- ♣ File transfer, access and management
- ♣ Mail services
- ♣ Directory services

Encapsulation and Decapsulation

One of the important concepts in protocol layering in the Internet is encapsulation/decapsulation. Figure shows this concept for the small internet.



We have not shown the layers for the link-layer switches because no encapsulation/decapsulation occurs in this device. In Figure 2.8, we show the encapsulation in the source host, decapsulation in the destination host, and encapsulation and Decapsulation in the router.

Encapsulation at the Source Host

At the source, we have only encapsulation.

1. At the application layer, the data to be exchanged is referred to as a *message*. A message normally does not contain any header or trailer, but if it does, we refer to the whole as the message. The message is passed to the transport layer.

2. The transport layer takes the message as the payload, the load that the transport layer should take care of. It adds the transport layer header to the payload, which contains the identifiers of the source and destination application programs that want to communicate plus some more information that is needed for the end-toend delivery of the message, such as information needed for flow, error control, or congestion control. The result is the transport-layer packet, which is called the *segment* (in TCP) and the *user datagram* (in UDP). The transport layer then passes the packet to the network layer.

3. The network layer takes the transport-layer packet as data or payload and adds its own header to the payload. The header contains the addresses of the source and destination hosts and some more information used for error checking of the header, fragmentation information, and so on. The result is the network-layer packet, called a *datagram*. The network layer then passes the packet to the data-link layer.

4. The data-link layer takes the network-layer packet as data or payload and adds its own header, which contains the link-layer addresses of the host or the next hop (the router). The result is the link-layer packet, which is called a *frame*. The frame is passed to the physical layer for transmission.

Decapsulation and Encapsulation at the Router

At the router, we have both decapsulation and encapsulation because the router is connected to two or more links.

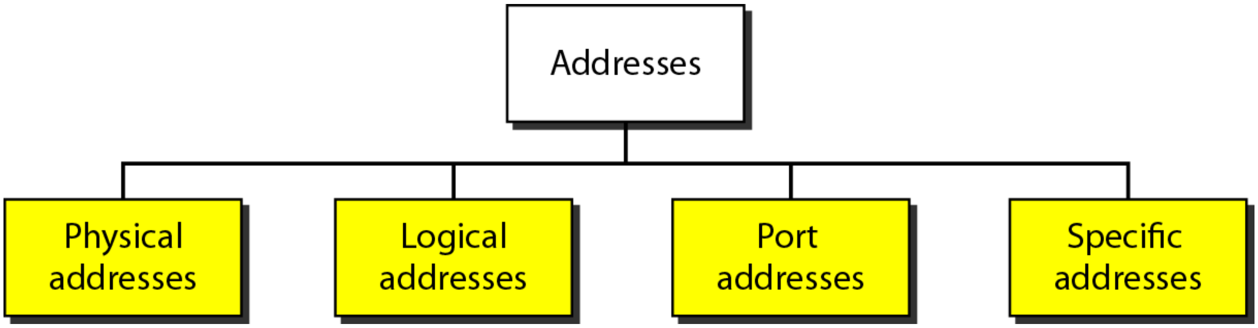
1. After the set of bits are delivered to the data-link layer, this layer decapsulates the datagram from the frame and passes it to the network layer.

2. The network layer only inspects the source and destination addresses in the datagram header and consults its forwarding table to find the next hop to which the datagram is to be delivered. The contents of the datagram should not be changed by the network layer in the router unless there is a need to fragment the datagram if it is too big to be passed through the next link. The datagram is then passed to the data-link layer of the next link.
3. The data-link layer of the next link encapsulates the datagram in a frame and passes it to the physical layer for transmission.

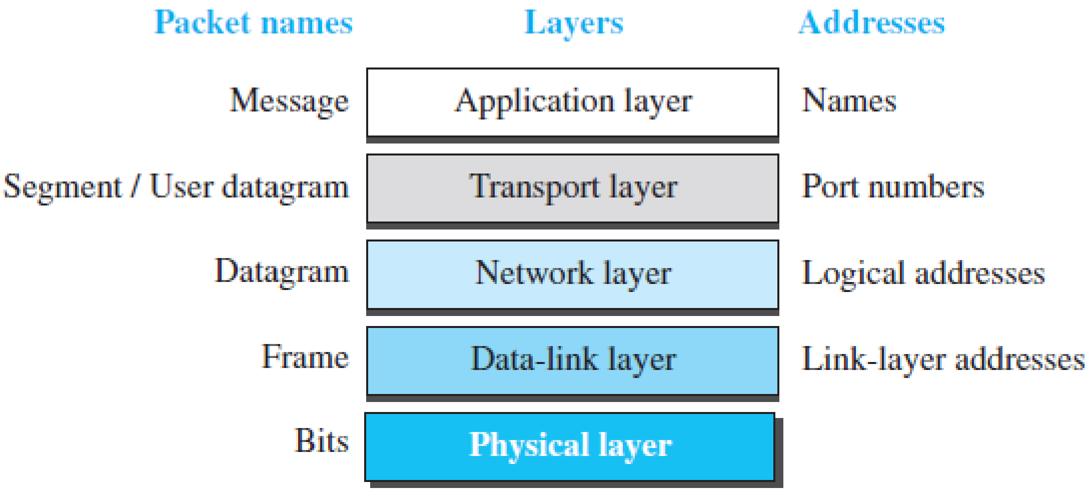
Decapsulation at the Destination Host
At the destination host, each layer only decapsulates the packet received, removes the payload, and delivers the payload to the next-higher layer protocol until the message reaches the application layer. It is necessary to say that decapsulation in the host involves error checking.

Addressing:
Four levels of addresses are used in an internet employing the TCP/IP Protocols:

a) Physical addresses
b) Logical addresses
c) Port addresses
d) Specific addresses

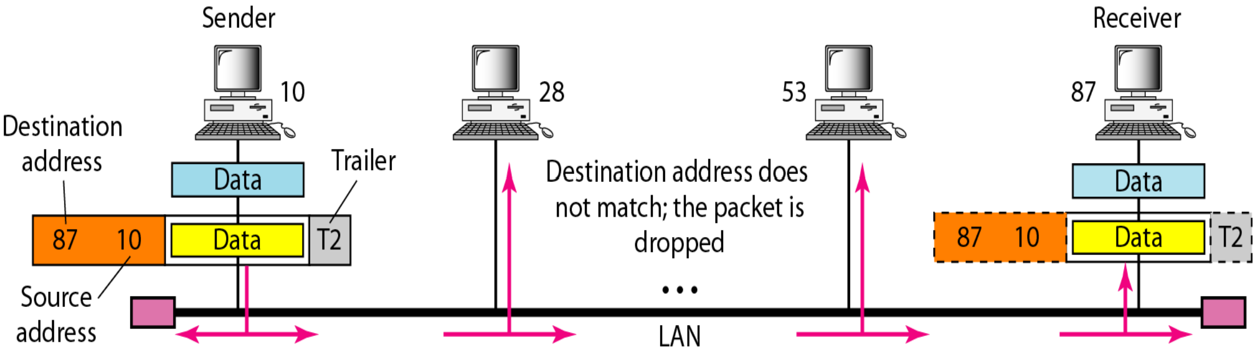


Each address is related to a specific layer in the TCP/IP architecture, as shown in the above fig.



Packet names	Layers	Addresses
Message	Application layer	Names
Segment / User datagram	Transport layer	Port numbers
Datagram	Network layer	Logical addresses
Frame	Data-link layer	Link-layer addresses
Bits	Physical layer	

a) Physical Address:
A node with physical address 10 sends a frame to a node with physical address 87. The two nodes are connected by a link (bus topology LAN). As the figure shows, the computer with physical address 10 is the sender, and the computer with physical address 87 is the receiver.

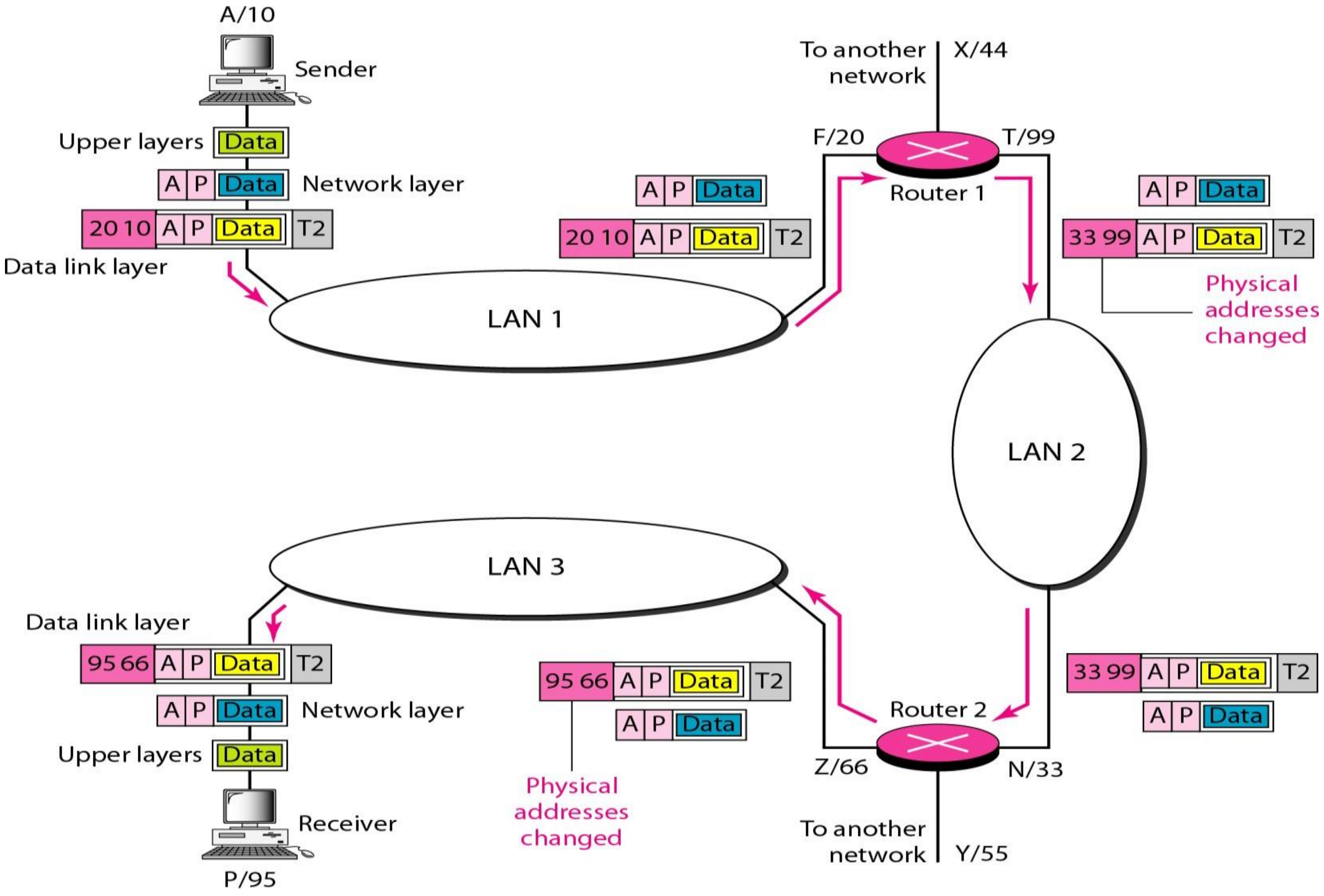


07:01:02:01:2C:4B

A 6-byte (12 hexadecimal digits) physical address.

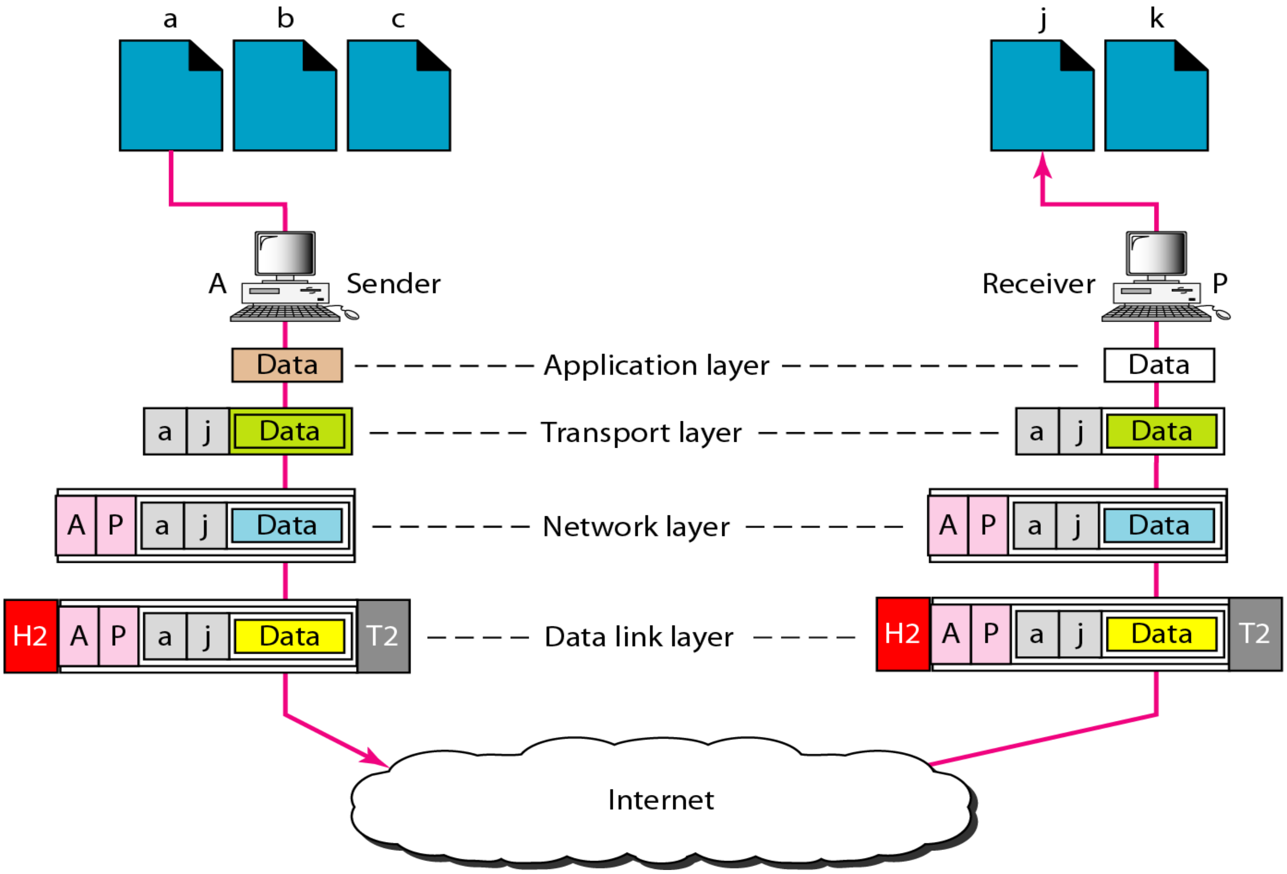
b) Logical (IP) Address

The physical addresses will change from hop to hop, but the logical addresses usually remain the same



c) Port Address

The physical addresses change from hop to hop, but the logical and port addresses usually remain the same



d) Specific Address

- ♣ Some application have user-friendly addresses that are designed for that specific address
- ♣ Example 1: e-mail address: kchung@kw.ac.kr
- ✸ Defines the recipient of an e-mail
- ♣ Example 2: URL (Universal Resource Locator) : www.google.com
- ✸ Used to find a document on the WWW

Difference between port address, a logical address and physical address

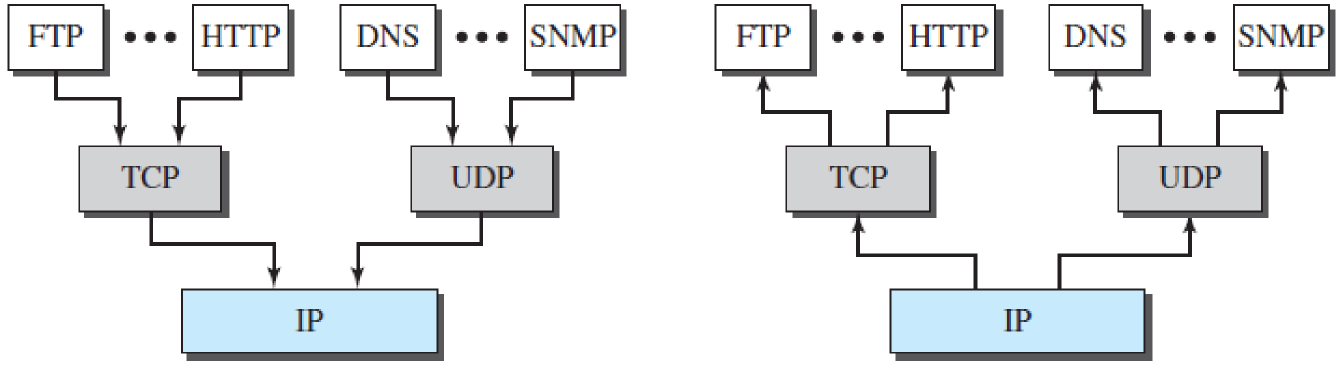
- ♣ Physical Address: Specifies the name of the Address of Source & Destination
- ♣ Logical Address: Specifies the Identification of the Device or Machine
- ♣ Port Address: Specifies the name of the File or Data

Multiplexing and Demultiplexing

Since the TCP/IP protocol suite uses several protocols at some layers, it has multiplexing at the source and demultiplexing at the destination.

- ➢ Multiplexing in this case means that a protocol at a layer can encapsulate a packet from several next-higher layer protocols (one at a time); demultiplexing means that a protocol can decapsulate and deliver a packet to several next-higher layer protocols (one at a time).
- ➢ Figure shows the concept of multiplexing and demultiplexing at the three upper layers.
- ➢ To be able to multiplex and demultiplex, a protocol needs to have a field in its header to identify to which protocol the encapsulated packets belong.
- ➢ At the transport layer, either UDP or TCP can accept a message from several application- layer protocols.

➢



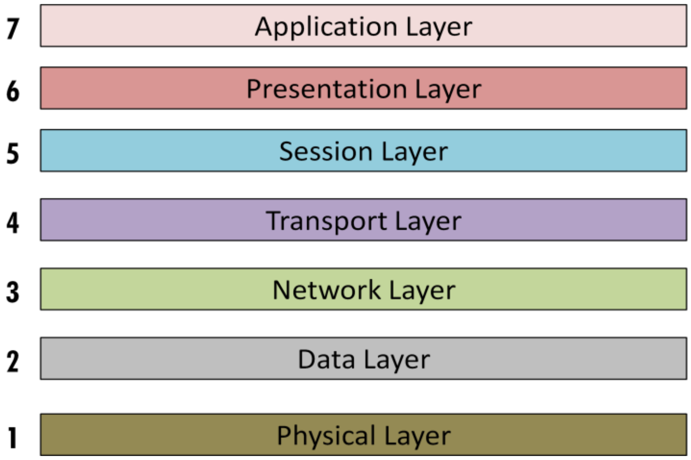
a. Multiplexing at source

b. Demultiplexing at destination

- At the network layer, IP can accept a segment from TCP or a user datagram from UDP. IP can also accept a packet from other protocols such as ICMP, IGMP, and so on.
- At the data-link layer, a frame may carry the payload coming from IP or other protocols such as ARP.

THE OSI MODEL

The OSI model shown in fig is based on the proposal developed by the International Standards Organization (ISO) as a first step towards international standardization of the protocols used in the various layers. The model is called the OSI (Open System Interconnection) reference model because it deals with connecting open systems, i.e., systems that are open for communication with other systems. The purpose of the OSI model is to show how to facilitate communication between different systems without requiring changes to the logic of the underlying hardware and software.



The OSI model is not a protocol; it is a model for understanding and designing a network architecture that is flexible, robust and interoperable. The OSI model is a layered framework for the design of network systems that allows communication between all types of computer systems. It consists of seven separate but related layers, each of which defines a part of the process of moving information across a network.

The principles that were applied to arrive at the seven layers are as follows:

♣ A layer should be created where a different level of abstraction is needed.

♣ Each layer should perform a well-defined function.

♣ The function of each layer should be chosen with an eye toward defining internationally standardized protocols.

- ♣ The layer boundaries should be chosen to minimize the information flow across the interfaces.
- ♣ The number of layers should be large enough that distinct functions need not be thrown together in the same layer out of necessity and small enough that the architecture does not become unwieldy.

Layered Architecture:

The OSI model is composed of seven layers: Physical, Data link, Network, Transport, Session, Presentation, Application layers. Below figure shows the layers involved when a message travels from A to B, it may pass through many intermediate nodes. These intermediate nodes involve only the first 3 layers of the OSI model.

Within a single machine, each layer calls upon the services of the layer just below it, layer 3 for ex. Uses the services provided by layer 2 & provides services for layer 4. Between machines, layer X on one machine communicates with layer X on another machine. This communication is governed by an agreed upon series of rules & Conventions called protocols. The processes on each machine that communicate at a given layer are called peer – to – peer processes. Communication between machines is therefore a peer – to –peer process using the protocols appropriate to a given layer.

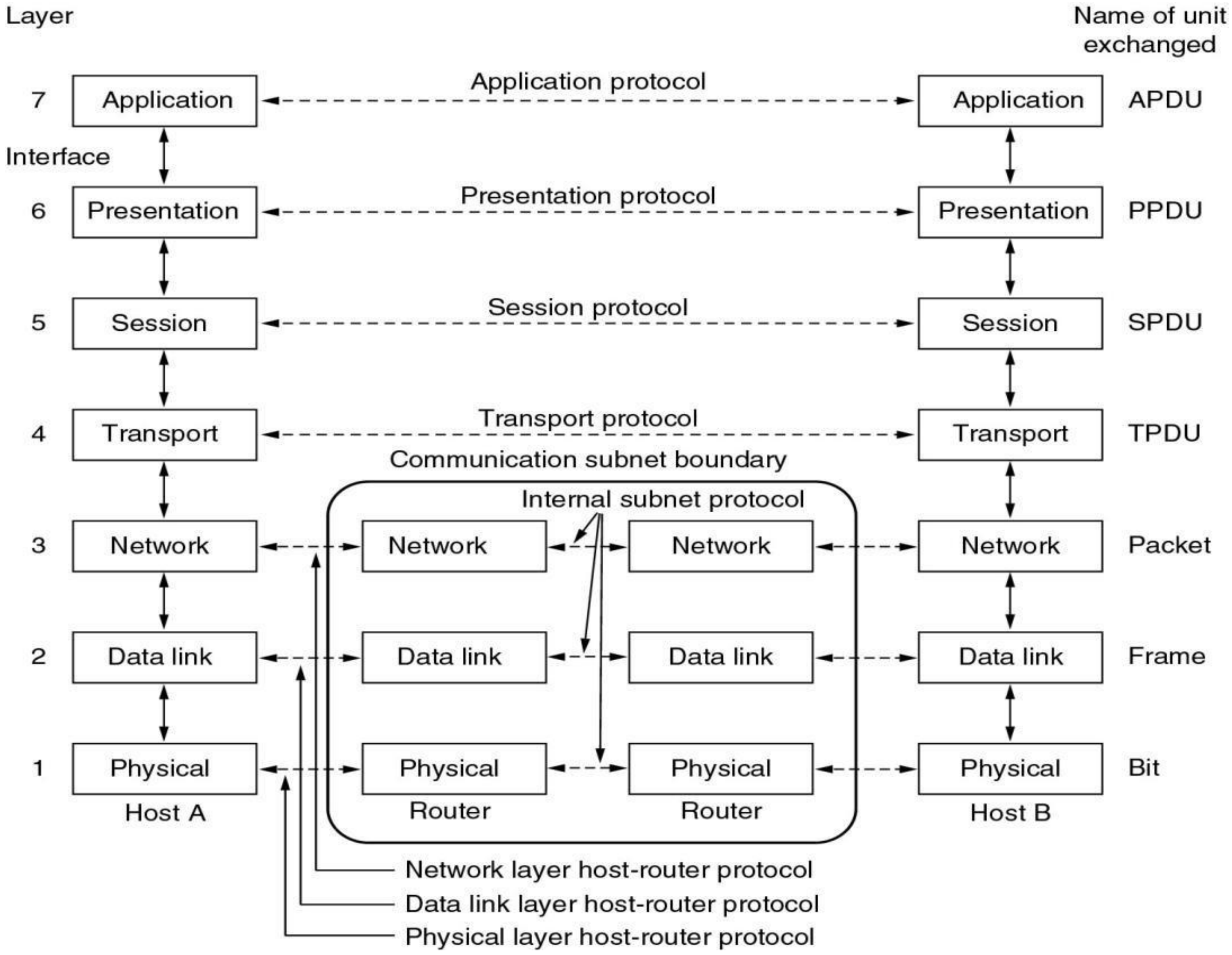
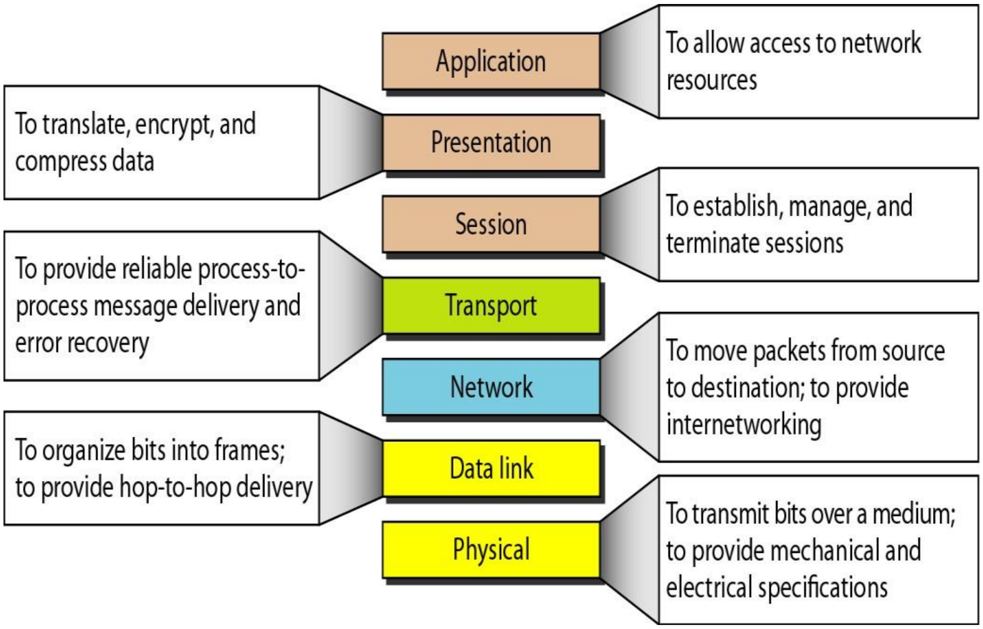
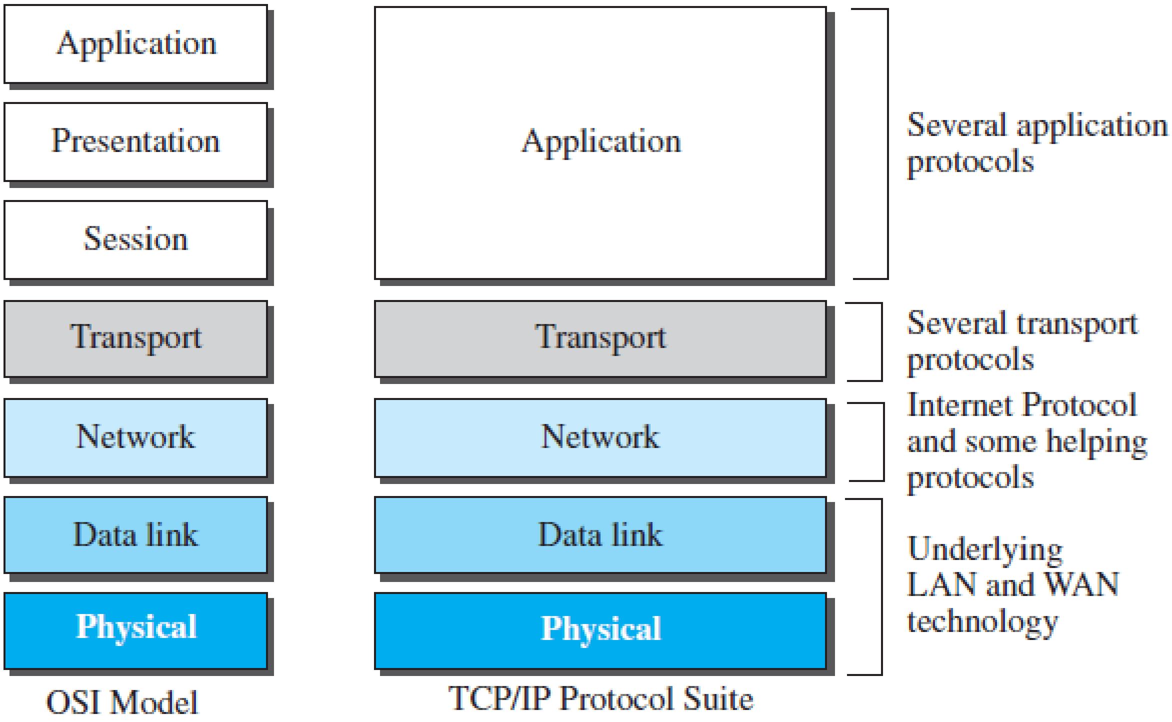


Fig: Interaction between layers in the OSI model


Application
To allow access to network resources
Presentation
To translate, encrypt, and compress data
Session
To establish, manage, and terminate sessions
Transport
To provide reliable process-to-process message delivery and error recovery
Network
To move packets from source to destination; to provide internetworking
Data link
To organize bits into frames; to provide hop-to-hop delivery
Physical
To transmit bits over a medium; to provide mechanical and electrical specifications

OSI versus TCP/IP

When we compare the two models, we find that two layers, session and presentation, are missing from the TCP/IP protocol suite. The application layer in the suite is usually considered to be the combination of three layers in the OSI model, as shown in Figure.



Two reasons were mentioned for this decision.

- First, TCP/IP has more than one transport-layer protocol. Some of the functionalities of the session layer are available in some of the transport-layer protocols.
- Second, the application layer is not only one piece of software. Many applications can be developed at this layer. If some of the functionalities mentioned in the session and presentation layers are needed for a particular application, they can be included in the development of that piece of software.

Lack of OSI Model's Success

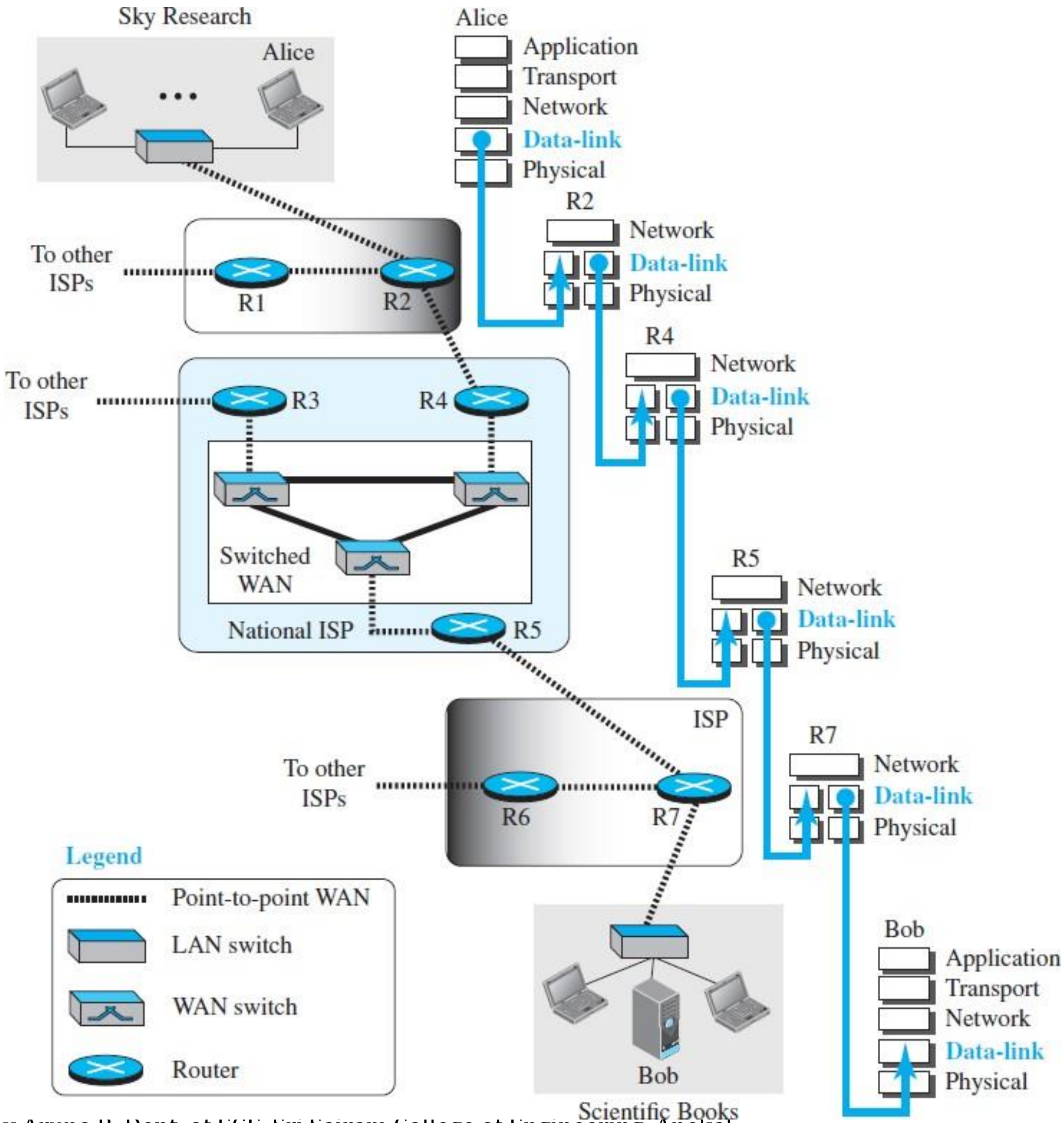
The OSI model appeared after the TCP/IP protocol suite. Most experts were at first excited and thought that the TCP/IP protocol would be fully replaced by the OSI model.

- First, OSI was completed when TCP/IP was fully in place and a lot of time and money had been spent on the suite; changing it would cost a lot. (Economic Loss)
- Second, some layers in the OSI model were never fully defined. (Protocols not defined for Presentation and Session Layers)
- Third, when OSI was implemented by an organization in a different application, high enough level of performance was not achieved to entice the Internet authority to switch from the TCP/IP protocol suite to the OSI model. (Expected Efficiency is low).

Introduction to Data-Link Layer

INTRODUCTION

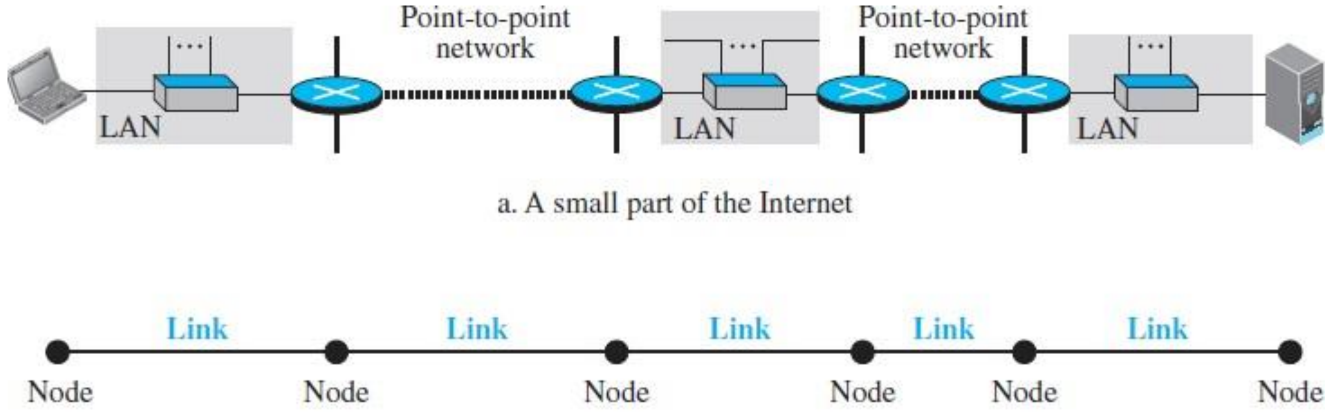
The Internet is a combination of networks glued together by connecting devices (routers or switches). If a packet is to travel from a host to another host, it needs to pass through these networks. Figure shows the same scenario, Communication at the data-link layer is made up of five separate logical connections between the data-link layers in the path.



The data-link layer at Alice's computer communicates with the data-link layer at router R2. The data-link layer at router R2 communicates with the data-link layer at router R4, and so on. Finally, the data-link layer at router R7 communicates with the data-link layer at Bob's computer. Only one data-link layer is involved at the source or the destination, but two data-link layers are involved at each router. The reason is that Alice's and Bob's computers are each connected to a single network, but each router takes input from one network and sends output to another network. Note that although switches are also involved in the data-link-layer communication, for simplicity we have not shown them in the figure.

Nodes and Links

Communication at the data-link layer is node-to-node. A data unit from one point in the Internet needs to pass through many networks (LANs and WANs) to reach another point. Theses LANs and WANs are connected by routers. It is customary to refer to the two end hosts and the routers as ***nodes*** and the networks in between as ***links***. Figure below is a simple representation of links and nodes when the path of the data unit is only six nodes.



a. A small part of the Internet

b. Nodes and links

The first node is the source host; the last node is the destination host. The other four nodes are four routers. The first, the third, and the fifth links represent the three LANs; the second and the fourth links represent the two WANs.

Services

- The datalink layer provides services to the network layer; it receives services from the physical layer.
- The duty scope of the data-link layer is node-to-node.
- It is responsible for delivering a datagram to the next node in the path, sending node needs to encapsulate the datagram received from the network in a frame, and the receiving node needs to decapsulate the datagram from the frame.

Framing

A packet at the data-link layer is normally called a frame.

Flow Control

The sending data-link layer at the end of a link is a producer of frames; the receiving data-link layer at the other end of a link is a consumer. If the rate of produced frames is higher than the rate of consumed frames, frames at the receiving end need to be buffered while waiting to be consumed (processed). Definitely, we cannot have an unlimited buffer size at the receiving side.

Different data-link-layer protocols use different strategies for flow control. Since flow control also occurs at the transport layer, with a higher degree of importance.

Error Control

Frame in a data-link layer needs to be changed to bits, transformed to electromagnetic signals, and transmitted through the transmission media. Frame is susceptible to error. The error needs first to be detected. After detection, it needs to be either corrected at the receiver node or discarded and retransmitted by the sending node. Since error detection and correction is an issue in every layer (node-to-node or host-to-host).

Congestion Control

Congestion control is considered an issue in the network layer or the transport layer because of its end-to-end nature.

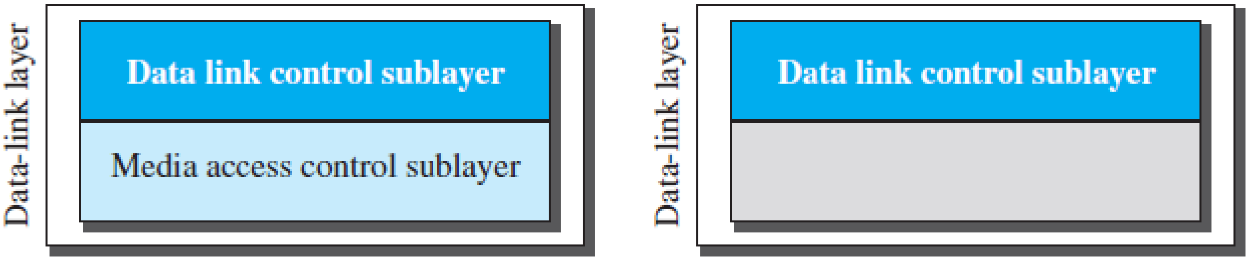
Two Categories of Links

- Point-to-point link
- Broadcast link.

In a point-to-point link, the link is dedicated to the two devices; in a broadcast link, the link is shared between several pairs of devices.

Two Sublayers

Data-link layer is divided into two sub-layers: **data link control (DLC)** and **media access control (MAC).** The media access control sublayer deals only with issues specific to broadcast links.

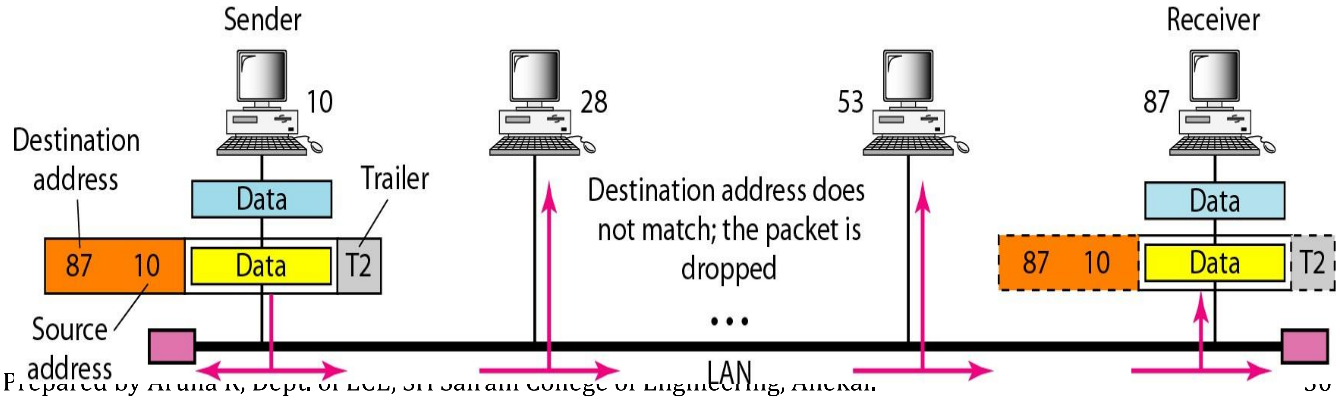

Data-link layer
Data link control sublayer
Media access control sublayer
Data-link layer
Data link control sublayer

a. Data-link layer of a broadcast link

b. Data-link layer of a point-to-point link

LINK-LAYER ADDRESSING

- A node with physical address 10 sends a frame to a node with physical address 87. The two nodes are connected by a link (bus topology LAN). As the figure shows, the computer with physical address 10 is the sender, and the computer with physical address 87 is the receiver.



Three Types of addresses

Some link-layer protocols define three types of addresses: unicast, multicast, and broadcast.

Unicast Address

Each host or each interface of a router is assigned a unicast address. Unicasting means one-to-one communication. A frame with a unicast address destination is destined only for one entity in the link.

Example:1

The unicast link-layer addresses in the most common LAN, Ethernet, are 48 bits (six bytes) that are presented as 12 hexadecimal digits separated by colons; for example, the following is a link- layer address of a computer.

07:01:02:01:2C:4B

A 6-byte (12 hexadecimal digits) physical address.

Multicast Address

Some link-layer protocols define multicast addresses. Multicasting means one-to-many communication. However, the jurisdiction is local (inside the link).

Example: 2

The multicast link-layer addresses in the most common LAN, Ethernet, are 48 bits (six bytes) that are presented as 12 hexadecimal digits separated by colons.

The second digit, however, needs to be an even number in hexadecimal. The following shows a multicast address:

A2:34:45:11:92:F1

Broadcast Address

Some link-layer protocols define a broadcast address. Broadcasting means one-to-all communication. A frame with a destination broadcast address is sent to all entities in the link.

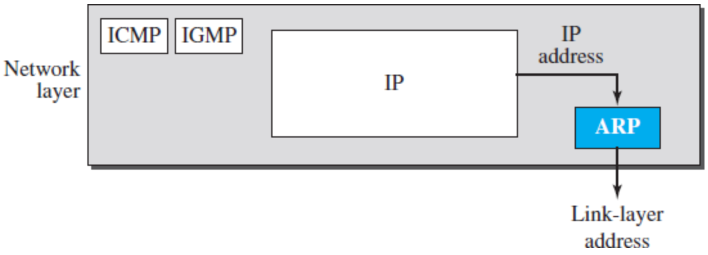
Example: 3

The broadcast link-layer addresses in the most common LAN, Ethernet, are 48 bits, all 1s, that are presented as 12 hexadecimal digits separated by colons. The following shows a broadcast address:

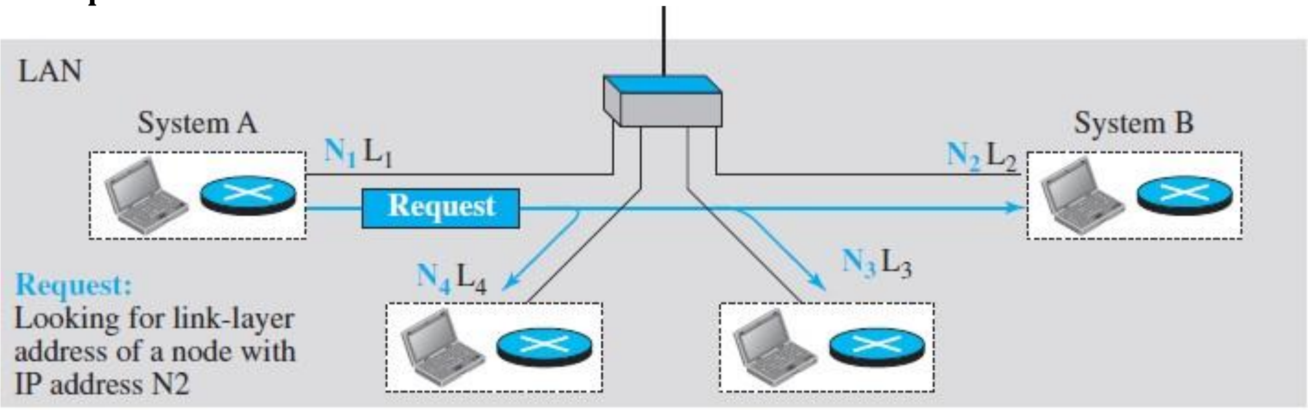
FF:FF:FF:FF:FF:FF

Address Resolution Protocol (ARP)

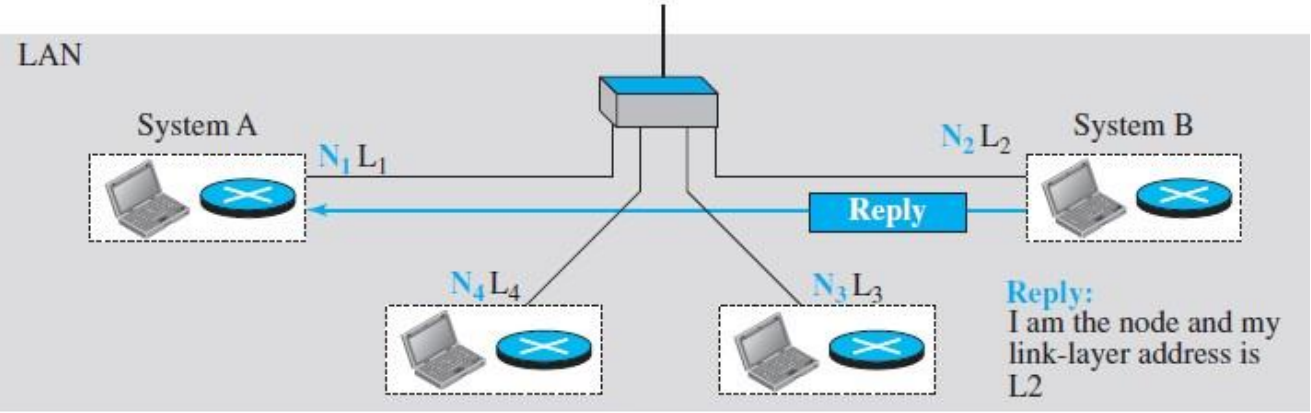
The ARP protocol is one of the auxiliary protocols defined in the network layer, as shown in Figure 9.6. It belongs to the network layer, because it maps an IP address to a logical-link address. ARP accepts an IP address from the IP protocol, maps the address to the corresponding link-layer address, and passes it to the data-link layer.



ARP operation



a. ARP request is broadcast



b. ARP reply is unicast

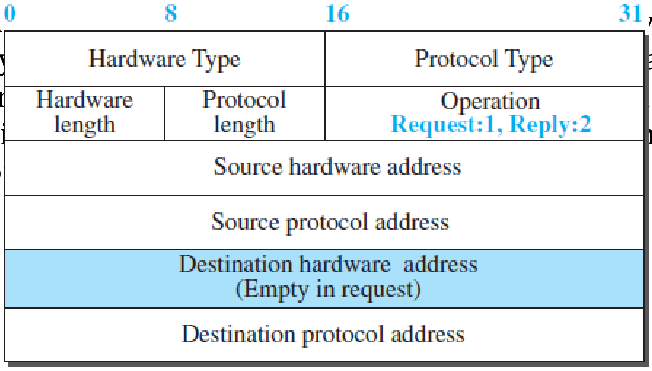
The system (A) has a packet that needs to be delivered to another system (B) with IP address **N2**. System A needs to pass the packet to its data-link layer for the actual delivery, but it does not know the physical address of the recipient. It uses the services of ARP by asking the ARP protocol to send a broadcast ARP request packet to ask for the physical address of a system with an IP address of **N2.**

This packet is received by every system on the physical network, but only system B will answer it, as shown in Figure b. System B sends an ARP reply packet that includes its physical address. Now system A can send all the packets it has for this destination using the physical address it received.

Packet Format

Figure shows the format of an ARP packet. The names of the fields are self explanatory. The hardware type field defines the type of the link-layer protocol; Ethernet is given the type 1. The protocol type field defines the network-layer

protocol: IPv4 protocol is (0800)16. The source h variable-length fields defining the link-layer and network-lay ardware address and destination protocol address fields defi dresses. An ARP packet is encapsulated directly into a data-l ow that the payload belongs to the ARP and not to the netwo



Data Link Control (DLC) DLC SERVICES

The data link control (DLC) deals with procedures for communication between two adjacent nodes—node-to-node communication—no matter whether the link is dedicated or broadcast. Data link control functions include *framing* and *flow and error control.*

Framing

Data transmission in the physical layer means moving bits in the form of a signal from the source to the destination. The physical layer provides bit synchronization to ensure that the sender and receiver use the same bit durations and timing.

Framing in the data-link layer separates a message from one source to a destination by adding a sender address and a destination address. The destination address defines where the packet is to go; the sender address helps the recipient acknowledge the receipt.

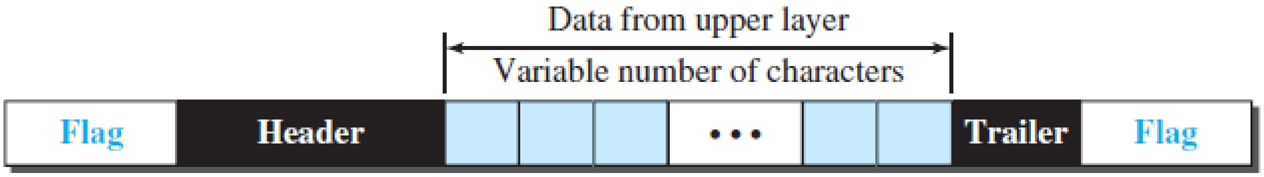
Frame Size

Frames can be of fixed or variable size.
In *fixed-size framing,* there is no need for defining the boundaries of the frames; the size itself can be used as a delimiter.

In variable-size framing, need a way to define the end of one frame and the beginning of the next. Historically, two approaches were used for this purpose: a character-oriented approach and a bit- oriented approach.

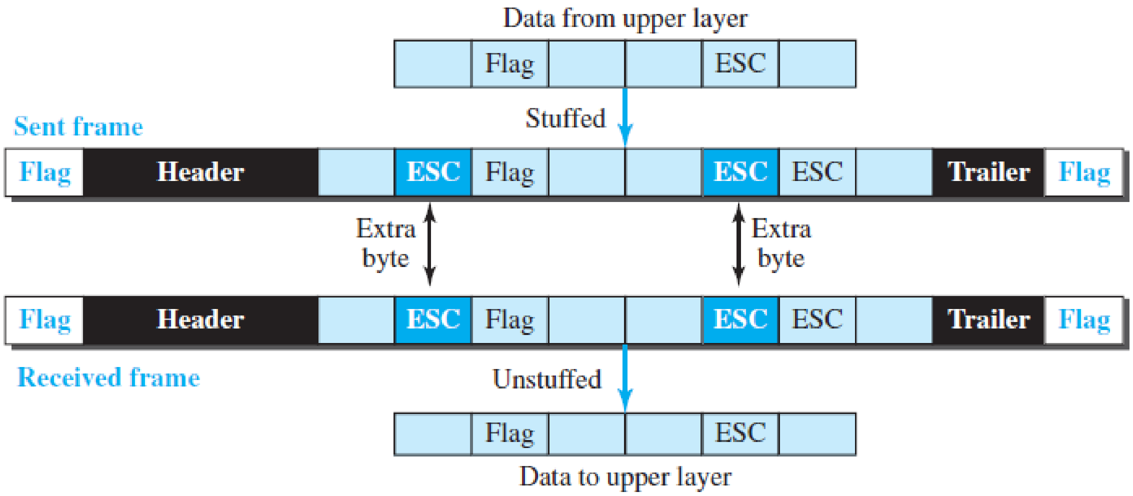
Character-Oriented Framing

In *character-oriented (or byte-oriented) framing,* data to be carried are 8-bit characters from a coding system such as ASCII. The header, which normally carries the source and destination addresses and other control information, and the trailer, which carries error detection redundant bits, are also multiples of 8 bits. To separate one frame from the next, an 8-bit (1-byte) flag is added at the beginning and the end of a frame. The flag, composed of protocol-dependent special characters, signals the start or end of a frame. Figure shows the format of a frame in a character- oriented protocol.



Character-oriented framing was popular when only text was exchanged by the data-link layers. The flag could be selected to be any character not used for text communication. Now, however, we send other types of information such as graphs, audio, and video; any character used for the flag could also be part of the information. If this happens, the receiver, when it encounters this pattern in the middle of the data, thinks it has reached the end of the frame. To fix this problem, a byte- stuffing strategy was added to character-oriented framing.

In **byte stuffing** (or character stuffing), a special byte is added to the data section of the frame when there is a character with the same pattern as the flag. The data section is stuffed with an extra byte. This byte is usually called the *escape character (ESC)* and has a predefined bit pattern. Whenever the receiver encounters the ESC character, it removes it from the data section and treats the next character as data, not as a delimiting flag. Figure shows the situation.



Bit-Oriented Framing

In bit-oriented framing, the data section of a frame is a sequence of bits to be interpreted by the upper layer as text, graphic, audio, video, and so on. However, in addition to headers (and possible trailers), we still need a delimiter to separate one frame from the other. Most protocols use a special 8-bit pattern flag, 01111110, as the delimiter to define the beginning and the end of the frame, as shown in Figure.

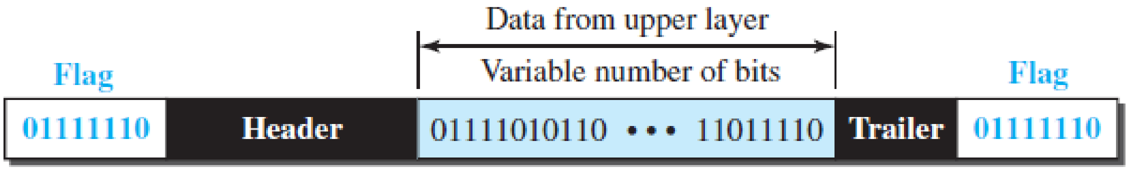
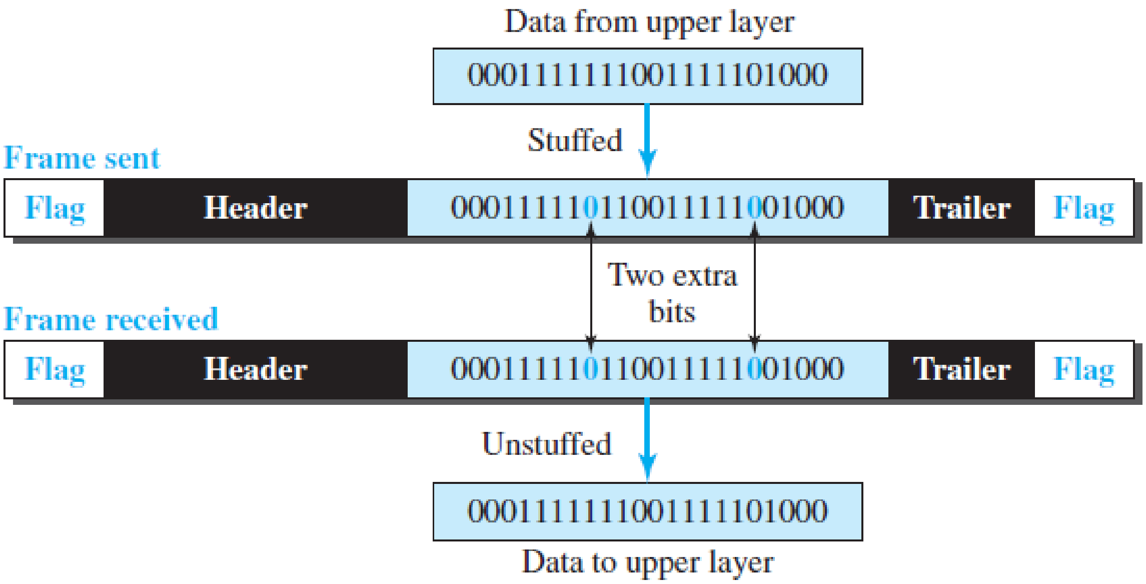


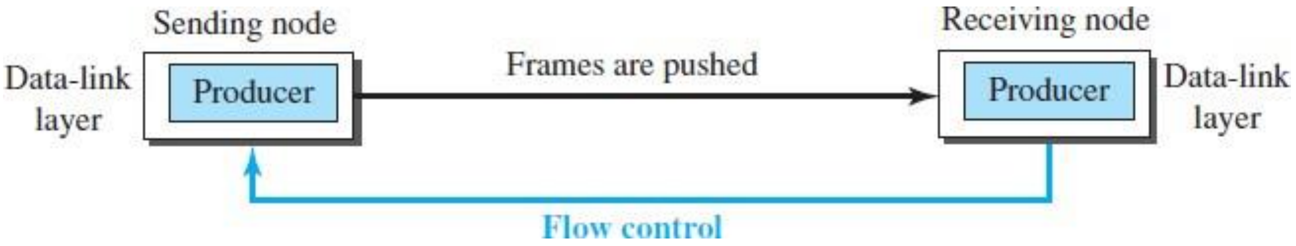
Figure below shows bit stuffing at the sender and bit removal at the receiver. Note that even if we have a 0 after five 1s, we still stuff a 0. The 0 will be removed by the receiver. This means that if the flaglike pattern 01111110 appears in the data, it will change to 011111010 (stuffed) and is not mistaken for a flag by the receiver. The real flag 01111110 is not stuffed by the sender and is recognized by the receiver.



Flow and Error Control

One of the responsibilities of the data-link control sublayer is flow and error control at the data- link layer.

Flow Control



The figure shows that the data-link layer at the sending node tries to push frames toward the data- link layer at the receiving node. If the receiving node cannot process and deliver the packet to its network at the same rate that the frames arrive, it becomes overwhelmed with frames. Flow control in this case can be feedback from the receiving node to the sending node to stop or slow down pushing frames.

Buffers

Although flow control can be implemented in several ways, one of the solutions is normally to use two *buffers*; one at the sending data-link layer and the other at the receiving data-link layer. A buffer is a set of memory locations that can hold packets at the sender and receiver. The flow control communication can occur by sending signals from the consumer to the producer. When the buffer of the receiving data-link layer is full, it informs the sending data-link layer to stop pushing frames.

Error Control

Error control at the data-link layer is normally very simple and implemented using one of the following two methods. In both methods, a CRC is added to the frame header by the sender and checked by the receiver.

- In the first method, if the frame is corrupted, it is silently discarded; if it is not corrupted, the packet is delivered to the network layer. This method is used mostly in wired LANs such as Ethernet.

- In the second method, if the frame is corrupted, it is silently discarded; if it is not corrupted, an acknowledgment is sent (for the purpose of both flow and error control) to the sender.

Connectionless and Connection-Oriented

A DLC protocol can be either connectionless or connection-oriented.

Connectionless Protocol

In a connectionless protocol, frames are sent from one node to the next without any relationship between the frames; each frame is independent. It means that there is no connection between frames. The frames are not numbered and there is no sense of ordering. Most of the data-link protocols for LANs are connectionless protocols.

Connection-Oriented Protocol

In this type of communication, the frames are numbered and sent in order. If they are not received in order, the receiver needs to wait until all frames belonging to the same set are received and then deliver them in order to the network layer. Connection oriented protocols are rare in wired LANs, but we can see them in some point-to-point protocols, some wireless LANs, and some WANs.

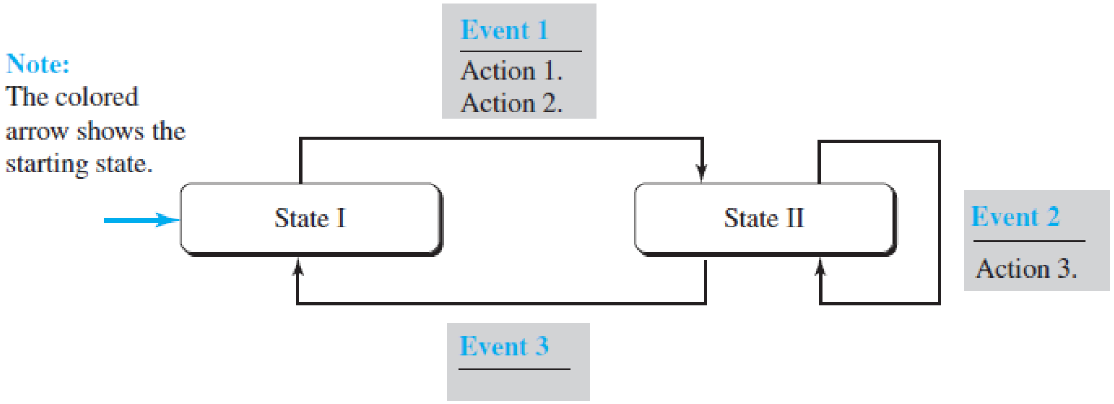
DATA-LINK LAYER PROTOCOLS

Traditionally four protocols have been defined for the data-link layer to deal with flow and error control:

- Simple protocol
- Stop-and-Wait protocol
- Go-Back-N and protocol
- Selective-Repeat protocol

The first two protocols still are used at the data-link layer, the last two have disappeared.

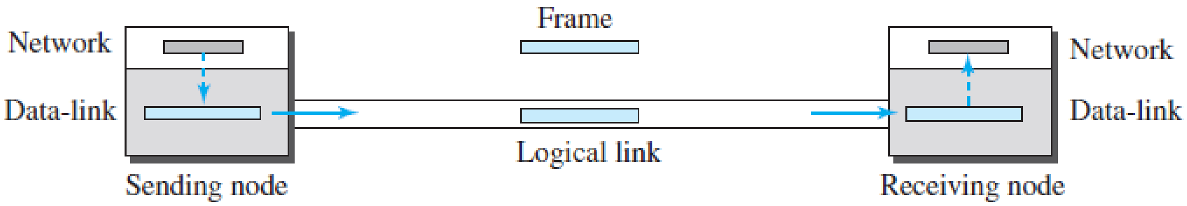
The behavior of a data-link-layer protocol can be better shown as a finite state machine (FSM). An FSM is thought of as a machine with a finite number of states. The machine is always in one of the states until an event occurs. Each event is associated with two reactions: defining the list of actions to be performed and determining the next state. One of the states must be defined as the initial state, the state in which the machine starts when it turns on.



The figure shows a machine with three states. There are only three possible events and three possible actions. The machine starts in state I. If event 1 occurs, the machine performs actions 1 and 2 and moves to state II. When the machine is in state II, two events may occur. If event 1 occurs, the machine performs action 3 and remains in the same state, state II. If event 3 occurs, the machine performs no action, but move to state I.

Simple Protocol

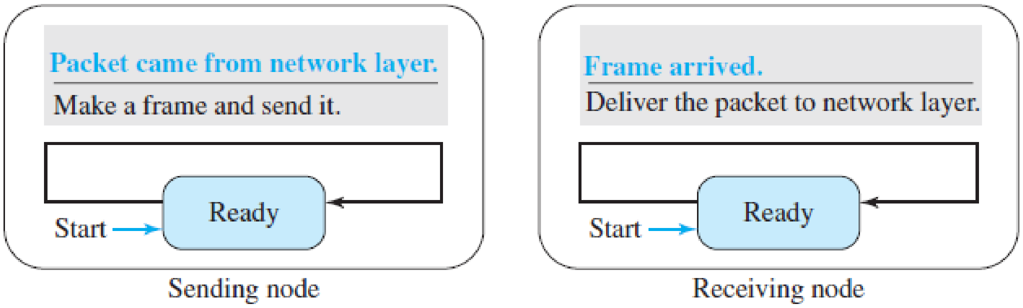
Our first protocol is a simple protocol with neither flow nor error control. Assume that the receiver can immediately handle any frame it receives. Figure shows the layout for this protocol.



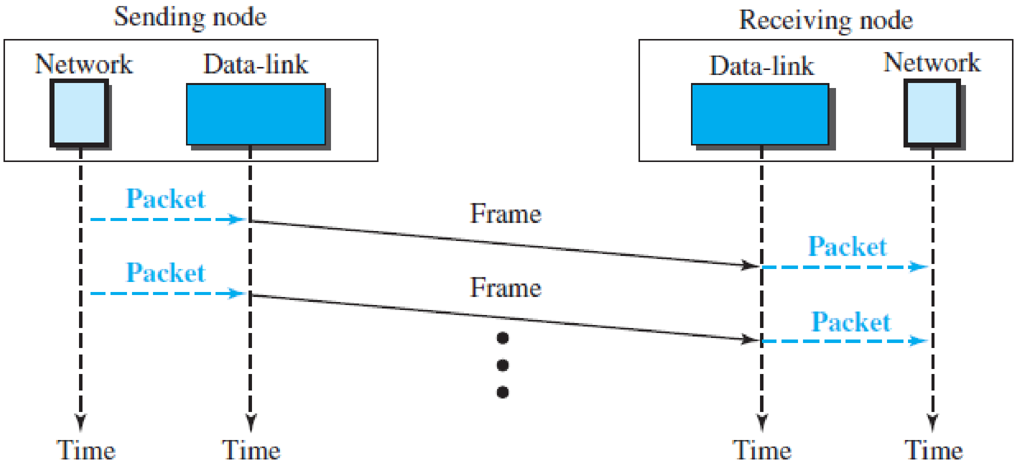
The data-link layer at the sender gets a packet from its network layer, makes a frame out of it, and sends the frame. The data-link layer at the receiver receives a frame from the link, extracts the packet from the frame, and delivers the packet to its network layer. The data-link layers of the sender and receiver provide transmission services for their network layers.

FSMs

- Each FSM has only one state, the *ready state*. The sending machine remains in the ready state until a request comes from the process in the network layer.
- When this event occurs, the sending machine encapsulates the message in a frame and sends it to the receiving machine.
- The receiving machine remains in the ready state until a frame arrives from the sending machine. When this event occurs, the receiving machine decapsulates the message out of the frame and delivers it to the process at the network layer.
- Figure shows the FSMs for the simple protocol.


Packet came from network layer.
Make a frame and send it.
Start
Ready
Sending node
Frame arrived.
Deliver the packet to network layer.
Start
Ready
Receiving node

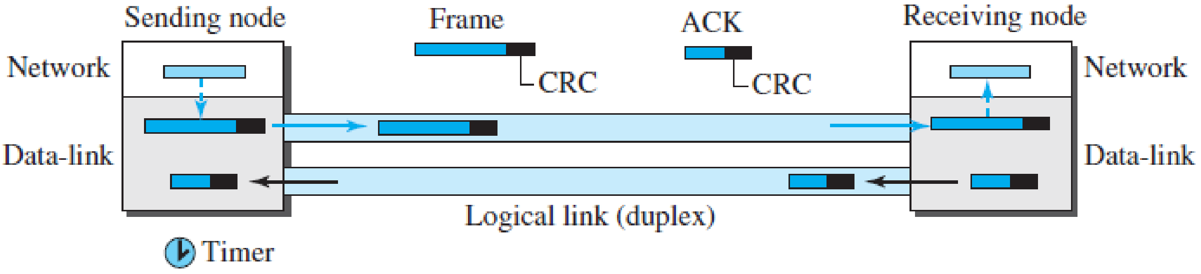
Example



Stop-and-Wait Protocol

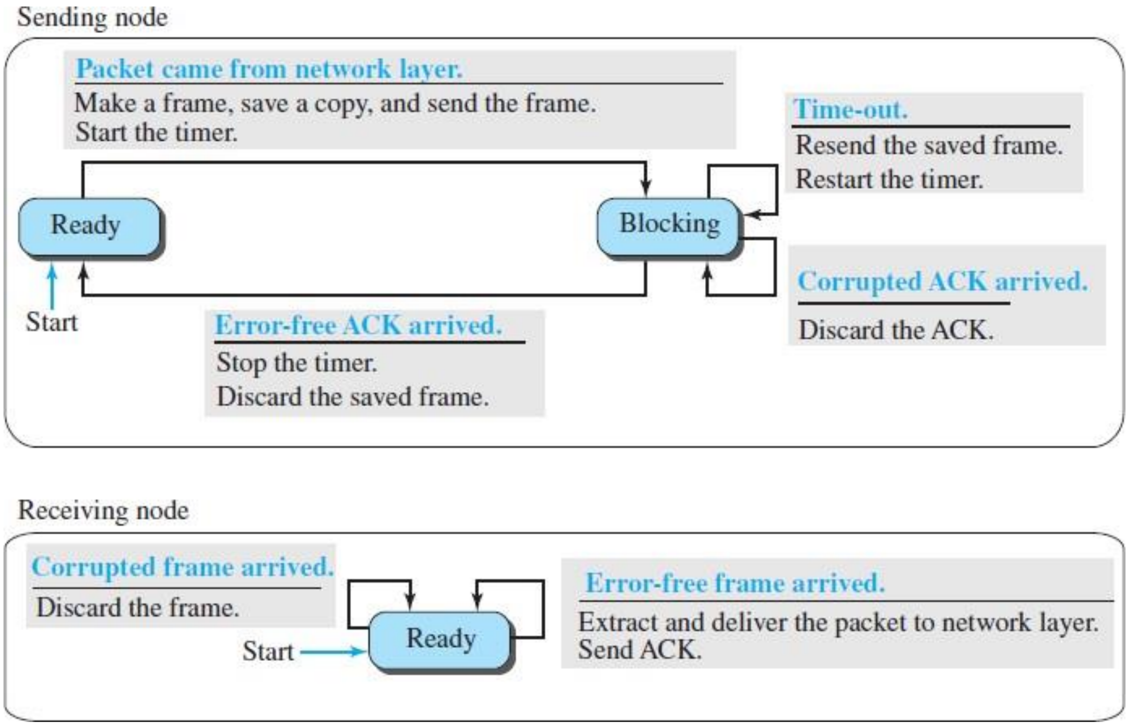
Stop-and-Wait protocol, which uses both flow and error control. In this protocol, the sender sends one frame at a time and waits for an acknowledgment before sending the next one. To detect corrupted frames, we need to add a CRC to each data frame. When a frame arrives at the receiver site, it is checked. If its CRC is incorrect, the frame is corrupted and silently discarded.

The silence of the receiver is a signal for the sender that a frame was either corrupted or lost. Every time the sender sends a frame, it starts a timer. If an acknowledgment arrives before the timer expires, the timer is stopped and the sender sends the next frame (if it has one to send). If the timer expires, the sender resends the previous frame, assuming that the frame was either lost or corrupted. This means that the sender needs to keep a copy of the frame until its acknowledgment arrives. When the corresponding acknowledgment arrives, the sender discards the copy and sends the next frame if it is ready. Figure shows the outline for the Stop-and-Wait protocol. Note that only one frame and one acknowledgment can be in the channels at any time.



FSMs

- Sender States
  The sender is initially in the ready state, but it can move between the ready and blocking state.
- Ready State.
  When the sender is in this state, it is only waiting for a packet from the network layer. If a packet comes from the network layer, the sender creates a frame, saves a copy of the frame, starts the only timer and sends the frame. The sender then moves to the blocking state.



- Blocking State. When the sender is in this state, three events can occur:
  - ♣ If a time-out occurs, the sender resends the saved copy of the frame and restarts the timer.
  - ♣ If a corrupted ACK arrives, it is discarded.

♣ If an error-free ACK arrives, the sender stops the timer and discards the saved copy of the frame. It then moves to the ready state.

- Receiver
  The receiver is always in the ready state. Two events may occur:

♣ If an error-free frame arrives, the message in the frame is delivered to the network layer and an ACK is sent.

- ♣ If a corrupted frame arrives, the frame is discarded.

Piggybacking

The two protocols we discussed in this section are designed for unidirectional communication, in which data is flowing only in one direction although the acknowledgment may travel in the other direction. Protocols have been designed in the past to allow data to flow in both directions. However, to make the communication more efficient, the data in one direction is piggybacked with the acknowledgment in the other direction. In other words, when node A is sending data to node B, Node A also acknowledges the data received from node B. Because piggybacking makes communication at the datalink layer more complicated, it is not a common practice.