Networking Devices

CHAPTER

4

4.1 Glossary

- *Bridge*: Network segments that typically use the same communication protocol use bridges to pass information from one network segment to the other.
- *Gateway*: When different communications protocols are used by networks, gateways are used to convert the data from the sender's
- *Hub*:Another name for a hub is a concentrator. Hubs reside in the core of the LAN cabling system. The hub connects workstations and sends every transmission to all the connected workstations.
- *Media Dependent Adapter*: A MDA is a plug-in module allowing selection among fiber-optic, twisted pair, and coaxial cable.
- *Media Filter*: When the electrical characteristics of various networks are different, media filter adapter connectors make the connections possible.
- *Multistation Access Unit*: MAUs are special concentrators or hubs for use in Token Ring networks instead of Ethernet networks.
- *Modems*: Modem is a device that *converts* digital signals to analog signals and analog signals to digital signals.
- *Network Interface Card*:NICs are printed circuit boards that are installed in computer workstations. They provide the physical connection and circuitry required to access the network.
- *Repeater*: Connectivity device used to regenerate and amplify weak signals, thus extending the length of the network. Repeaters perform no other action on the data.
- *Router*: Links two or more networks together, such as an Internet Protocol network. A router receives packets and selects the optimum path to forward the packets to other networks.
- *Switch*: A connection device in a network that functions much like a bridge, but directs transmissions to specific workstations rather than forwarding data to all workstations on the network.
- *Transceiver*: The name transceiver is derived from the combination of the words transmitter and receiver. It is a device that both transmits and receives signals and connects a computer to the network. A transceiver may be external or located internally on the NIC.

- *Firewall*: Firewall provides controlled data access. Firewalls can be hardware or software based and between networks. These are an essential part of a network's security strategy.

4.2 End Devices

In computer networks, the computers that we use on a daily basis are called *nodes* (also called *hosts* or end systems). They are called *hosts* because they host the application-level programs such as a Web browser or an electronic-mail program.

Sometimes, they are also called as *end systems* because they sit at the edge of the network connection. A node can be a computer or some other device, such as a printer. Every node has a unique network address, sometimes called a *Data Link Control* (DLC) address or *Media Access Control* (MAC) address.

An end device acts as the source (i.e., generates and sends messages) or as the destination (i.e., receives and consumes content) of the communication process.

In modern networks, a host can act as a client, a server, or both. Software installed on the host determines which role it plays on the network. Servers are hosts that have software installed that enables them to provide information and services, like e-mail or web pages, to other hosts on the network.

Some examples of end devices are:

- Computers, laptops, file servers, web servers.
- Network printers
- VoIP phones
- Security cameras
- Mobile handheld devices

4.3 Intermediary Devices

In addition to the end devices that people are familiar with, computer networks depends on intermediary devices to provide connectivity. These intermediary devices work behind the scenes to ensure that data flows across the network. Also, they connect the individual systems to the network and can connect multiple individual networks to form an *internetwork* (also called *Internet*). Examples of intermediary network devices are:

- Network Access Devices (hubs, switches, and wireless access points)
- Internetworking Devices (*routers*)
- Communication Servers and Modems
- Security Devices (*firewalls*)

The management of data as it flows through the network is also a role of the intermediary devices. These devices use the destination host address, along with information about the network interconnections, to determine the path that messages should take through the network. Processes running on the intermediary network devices perform these functions:

- Regenerate and retransmit data signals
- Maintain information about what pathways exist through the network and internetwork
- Notify other devices of errors and communication failures
- Direct data along alternate pathways when there is a link failure
- Classify and direct messages according to priorities

- Permit or deny the flow of data, based on security settings

The intermediate devices can be further classified by on their functionality as:

- *Connectivity Devices*: Connectivity devices are devices used to make physical network connections. They do *not make changes* to the data or transmission route. Connectivity devices operate at the physical layer of the OSI model.
- *Internetworking Devices*: Internetworking devices move data across a network. They *direct* data to specific locations within the network and/or *convert* data into alternative formats. Internetworking devices operate at OSI layers above the physical layer.

4.4 Connectivity Devices

4.4.1 Introduction

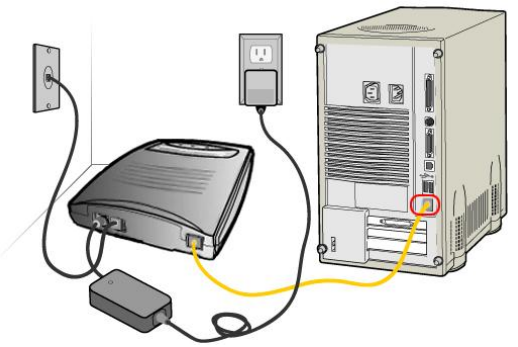
Connectivity devices are those devices used to make physical network connections. Connectivity devices operate at the physical layer of the Open Systems Interconnection Reference Model (OSI) model. The OSI model describes how computer services and procedures are standardized.

This standardization allows computers to share information and enables the interconnection of various networking connectivity devices regardless of vendor.

4.4.2 Network Interface Cards

A *network interface card* is a piece of computer hardware and its main functionality is to allow a computer to connect to a network. A network interface card is also called LAN *card*, *network adapter*, *network adapter boards, media access cards* or simply *NIC*.

Regardless of the name, they enable computers to communicate across a network. With this device, information packets can be transferred back and forth through a local area network (LAN). It acts a communication source for sending and receiving data on the network.

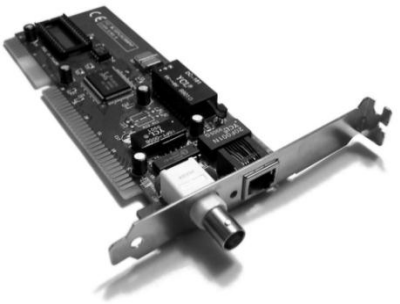


NIC provides physical access to a networking medium and often provides a low-level addressing system through the use of MAC addresses. It allows users to connect to each other either by using *cables* or *wirelessly*.

The network interface card (NIC) is an add-on component for a computer, much like a video card or sound card is. On most of the systems the NIC is integrated into the system board. On others it has to be installed into an expansion slot.

Most network interface cards have the *Ethernet* protocol as the language of the data that is being transferred back and forth. However, network interface cards do not all

necessarily need physical Ethernet or other cables to be functional. Some have wireless capabilities through including a small *built-in antenna* that uses radio waves to transmit information.



The computer must have a software driver installed to enable it to interact with the NIC. These drivers enable the operating system and higher-level protocols to control the functions of the adapter.

Each NIC has a unique *media access control* (MAC) address to direct traffic. This unique MAC address ensures that information is only being sent to a specific computer name and not to multiple ones if not intended to. Circled in the picture below is an example of an integrated network interface card.

The MAC (Media Access Layer) address, or hardware address, is a 12-digit number consisting of digits 0-9 and letters A-F. It is basically a hexadecimal number assigned to the card. The MAC address consists of two pieces: the first signifies which vendor it comes from, the second is the serial number unique to that manufacturer.

Example MAC addresses:

00-B0-D0-86-BB-F7 01-23-45-67-89-AB 00-1C-B3-09-85-15

The NIC performs the following functions:

- It translates data from the parallel data bus to a serial bit stream for transmission across the network.
- It formats packets of data in accordance with protocol.
- It transmits and receives data based on the hardware address of the card.

4.4.3 Transceivers

The term *transceiver* does not necessarily describe a separate network device but rather embedded in devices such as network cards.

Transceiver is a short name for *transmitter-receiver*. It is a device that both transmits and receives analog or digital signals. The term transceiver is used most frequently to describe the component in local-area networks (LANs) that actually applies signals onto the network wire and detects signals passing through the wire. For many LANs, the transceiver is built into the network interface card (NIC). Older types of networks, however, require an external transceiver.

The transceiver does not make changes to information transmitted across the network; it adapts the signals so devices connected by varying media can interpret them. A transceiver operates at the physical layer of the OSI model.

Technically, on a LAN the transceiver is responsible to place signals onto the network media and also detecting incoming signals traveling through the same cable. Given the

description of the function of a transceiver, it makes sense that that technology would be found with network cards (NICs).

4.4.4 Amplifiers and Repeaters

A repeater is an electronic device that receives a signal and retransmits it at a higher level or higher power, so that the signal can cover longer distances without degradation.

Transmitter sends a signal containing some information and after travelling some distance, usually, a signal get weakened (attenuated) due to energy loss in the medium. Therefore, it should be improved (or *amplified*). *Amplifier* is the circuit which magnifies the weak signal to a signal with more power.

Sometimes, this signal attenuation happens much before the arrival to the destination. In this case, signal is amplified and retransmitted with a power gain in one or more mid points. Those points are called *repeaters*. Therefore an amplifier is an essential part of a repeater.

Amplifier

Amplifier is an electronic circuit that increases the power of an input signal. There are many types of amplifiers ranging from voice amplifiers to optical amplifiers at different frequencies.

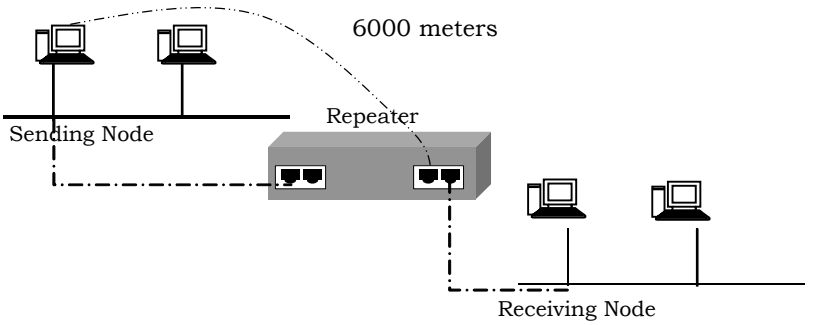
Repeater

The repeater is an electronic circuit that receives a signal and retransmits the same signal with a higher power. Therefore, a repeater consists of a signal receiver, an *amplifier* and a *transmitter*. Repeaters are often used in submarine communication cables as signal would be attenuated to just a random noise when travelling such a distance.

Different types of repeaters have different types of configurations depending on the transmission medium. If the medium is microwaves, repeater may consist of antennas and waveguides. If the medium is optical it may contain photo detectors and light emitters.

Difference between an Amplifier and a Repeater

1. Amplifier is used to magnify a signal, whereas repeater is used to receive and retransmit a signal with a power gain.
2. Repeater has an amplifier as a part of it.
3. Sometimes, amplifiers introduce some noise to the signal, whereas repeaters contain noise eliminating parts.



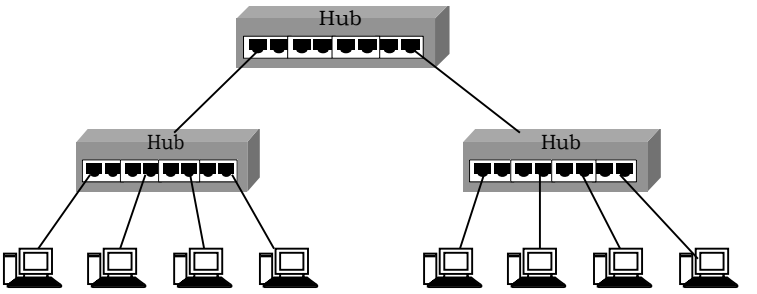
Thickwire can normally transmit a distance of 500 meters and this can be extended by introducing repeaters. *Thinwire* can normally transmit a distance of 185 meters, and can also be extended by using a repeater. This is the advantage to using a repeater. If a network layout exceeds the normal specifications of cable we can use repeaters to build network. This will allow for greater lengths when planning cabling scheme.

Repeaters *perform* no other action on the data. Repeaters were originally separate devices. Today a repeater may be a separate device or it may be incorporated into a hub. Repeaters operate at the physical layer of the OSI model.

4.4.5 Hubs

Hubs are commonly used to connect segments of a LAN. A hub contains multiple ports. When a packet arrives at one port, it is copied to the other ports so that all segments of the LAN can see all packets.

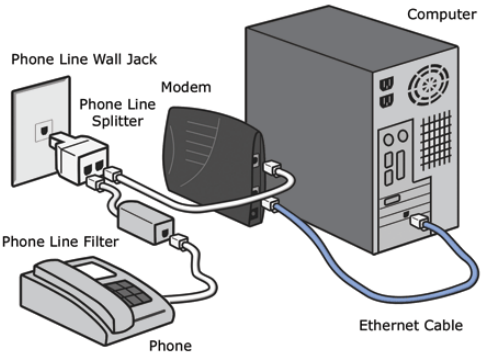
A *hub* contains multiple ports. When a packet arrives at one port, it is copied to all (broadcast) the ports of the hub. When the packets are copied, the destination address in the frame does not change to a *broadcast* address. It does this in a rudimentary way; it simply copies the data to all of the nodes connected to the hub.



The main function of the hub is to broadcast signals to different workstations in a LAN. General speaking, the term hub is used instead of repeater when referring to the device that serves as the center of a network.

4.4.6 Modems

Modem is a device that *converts* digital signals to analog signals and analog signals to digital signals. The word modem stands for *modulation* and *demodulation*. The process of converting digital signals to analog signals is called *modulation*. The process of converting analog signals to digital signals is called *demodulation*. Modems are used with computers to transfer data from one computer to another computer through telephone lines.



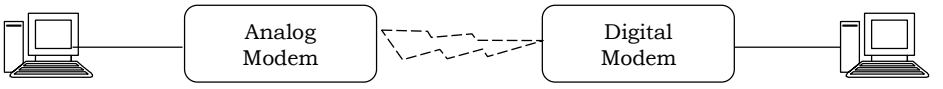
Types of Modem Connections

Modems have two types of connections and they are.

- Analog connection
- Digital connection

Analog Connection

The connection between the modem and the telephone line is called a *analog connection*. It converts digital signals from a computer to analogue signals that are then sent down the telephone line. A modem on the other end converts the analogue signal back to a digital signal the computer can understand. A workstation is connected to an analogue modem. The analogue modem is then connected to the telephone exchange analogue modem, which is then connected to the internet.



Digital Connection

The connection of modem to computer is called digital connection

Types of Modems

There are two types of modems:

- Internal modems
- External modems

Internal Modems

It fits into expansion slots inside the computer. It is directly linked to the telephone lines through the telephone jack. It is normally less inexpensive than external modem. Its transmission speed is also less external modem.

External Modems

It is the external unit of computer and is connected to the computer through serial port. It is also linked to the telephone line through a telephone jack. External modems are expensive and have more operation features and high transmission speed.

Advantages of Modems

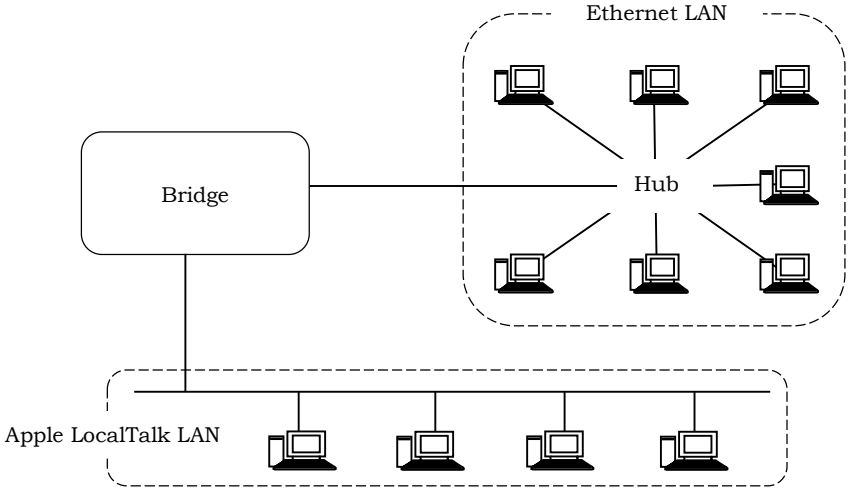
- Inexpensive hardware and telephone lines
- Easy to setup and maintain

Disadvantage of Modems

- Very slow performance

4.4 Internetworking Devices

4.4.1 Bridges



Bridge is a device which operates in both the physical and the data link layer of the OSI reference model. As a physical layer device, it *regenerates* the signal it receives. As a data link layer device, the bridge can check the physical (MAC) addresses (*source* and *destination*) contained in the frame.

Bridges can be used to divide a large network into *segments*. Bridges contain logic that allows them to keep the traffic for each *segment separate*. When a new frame enters to a bridge, the bridge not only regenerate the frame but it also checks the address of the destination and forwards the new copy only to the segment to which the destination address belongs.

A bridge device *filters* data traffic at a network boundary. Bridges reduce the amount of *traffic* on a LAN by dividing it into segments. Key features of a bridge are mentioned below:

- A bridge operates both in physical and data-link layer
- A bridge uses a table for *filtering/routing*
- A bridge does not *change* the physical (MAC) addresses in a *frame*

4.4.1.1 Why Use Bridges?

As an example, imagine for a moment that computers are people in a room. Everyone is glued to 1 spot and can't move around. If *Ram* wants to talk to *Mary*, he shouts out "*Hey Mary*" and *Mary* responds; and a conversation occur as a result.

On a small scale this works quite well. The Internet (as we know it today) is not just 2 or a few people talking directly to each other. The internet is literally billions of devices. If they were all placed into the same room (network-segment); imagine what would happen if *Ram* wanted to talk to *Mary*. *Ram* would yell "*Hey Mary*!" and Ram's voice would be lost in the crowd. Building a room to fit billions of people is equally ridiculous.

For this reason, networks are separated into smaller segments (smaller rooms) which allow devices who are in the same segment (room) to talk directly to each other's. But, for the devices outside the segment we need some sort of device (router) to pass messages from one room to the next room. But the vast number of segments (rooms) means we need some sort of addressing scheme so the various routers in the middle know how to get a message from *Ram* to *Mary*.

Segmenting a large network with an interconnect device (*bridge*) has many *advantages*. Among these are *reduced* collisions (in an Ethernet network), contained *bandwidth* utilization, and the ability to filter out unwanted packets. Bridges were created to allow network administrators to segment their networks transparently. What this means is that individual stations need not know whether there is a bridge separating them or not. It is up to the bridge to make sure that packets get properly forwarded to their destinations. This is the fundamental principle underlying all of the bridging behaviours we will discuss.

4.4.1.2 Types of Bridges

Several different types of bridges are available for internetworking LANs.

1. *Transparent Basic Bridge* [*Transparent Forwarding Bridge*]: Places incoming frame onto all outgoing ports *except* original incoming port.
2. *Transparent Learning Bridge*: Stores the origin of a frame (from which port) and later uses this information to place frames to that port.
3. *Transparent Spanning Bridge*: Uses a subset of the LAN topology for a loop-free operation.
4. *Source Routing Bridge*: Depends on routing information in frame to place the frame to an outgoing port.

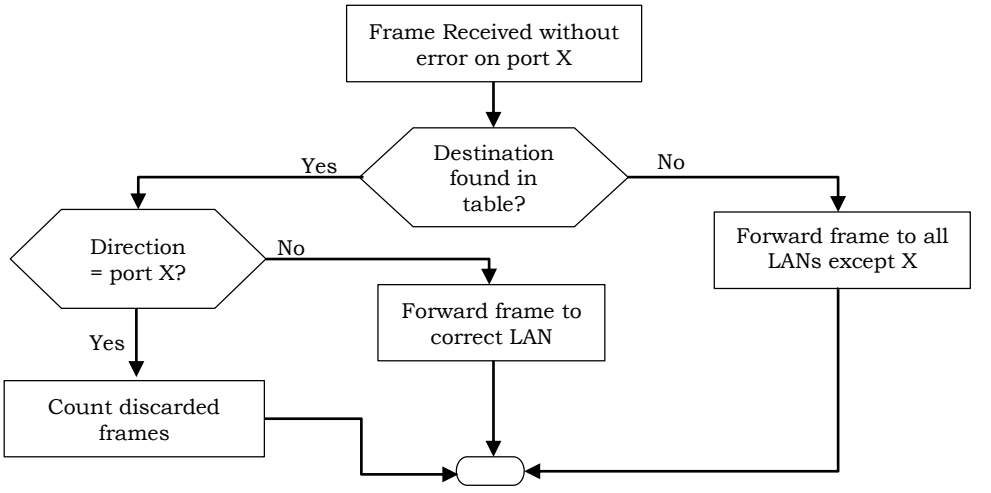
4.4.1.2.1 Transparent Basic Bridges [Transparent Forwarding Bridge]

The simplest type of bridge is called the *transparent basic bridge*. It is called *transparent* because the nodes using a bridge are unaware of its presence. This bridge receives traffic coming in on each port and stores the traffic until it can be transmitted on the outgoing ports. It will not forward the traffic from the port from which it was received.

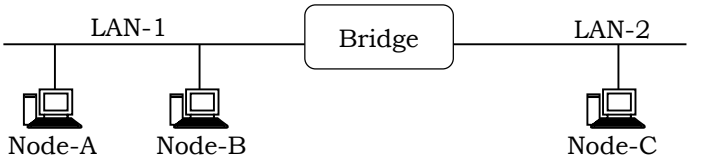
The bridge does not make any conversion of the traffic. The bridge forwards (*receive* and *subsequently transmit*) frames from one LAN to another. Obviously, the bridge forwards all frames like a *repeater*.

Transparent Bridge Forwarding

If the destination address is present in the forwarding database (table) already created, the packet is forwarded to the port number to which the destination host is attached. If it is not present, forwarding is done on all parts (*flooding*). This process is called *bridge forwarding*.



Bridge forwarding operation is explained with the help of flowchart.



In the figure above, consider three nodes A, B, and C. Assume each node sends frames to all other nodes. The source addresses A, B are observed to be on network LAN-1, while the address of node C will be observed to be on network LAN-2.

Basic functions of the bridge forwarding are mentioned below.

1. If the source address is *not* present in the forwarding table, the bridge *adds* the source address and corresponding interface to the table. It then checks the destination address to determine if it is in the table.
2. If the destination address is listed in the table, it determines if the destination address is on the same LAN as the source address. If it is, then the bridge *discards* the frame since all the nodes have already received the frame.
3. If the destination address is listed in the table but is on a different LAN than the source address, then the frame is forwarded to that LAN.
4. If the destination address is not listed in the table, then the bridge forwards the frame to all the LANs except the one that which originally received the frame. This process is called *flooding*.

In some bridges, if the bridge has not accessed an address in the forwarding table over a period of time, the address is removed to free up memory space on the bridge. This process is referred to as *aging*.

Packets with a source A and destination B are received and discarded, since the node B is directly connected to the LAN-1, whereas packets from A with a destination C are forwarded to network LAN-2 by the bridge.

4.4.1.2.2 Transparent Bridge Learning

To learn which addresses are in use, and which ports (interfaces on the bridge) are closest to, the bridge observes the headers of received frames. By examining the MAC source address of each received frame, and recording the port on which it was received, the bridge may learn which addresses belong to the computers connected via each port. This is called *learning*.

The learned addresses are stored in the *interface address table* (*database*) associated with *each* port (*interface*). Once this table has been setup, the bridge examines the destination address of all received frames; it then scans the interface tables to see if a frame has been received from the same address (i.e. a packet with a source address matching the current destination address).

At the time of installation of a transparent bridge, the table is empty. When a packet is encountered, the bridge checks its source address and build up a table by associating a source address with a port address to which it is connected. The flowchart explains the learning process.

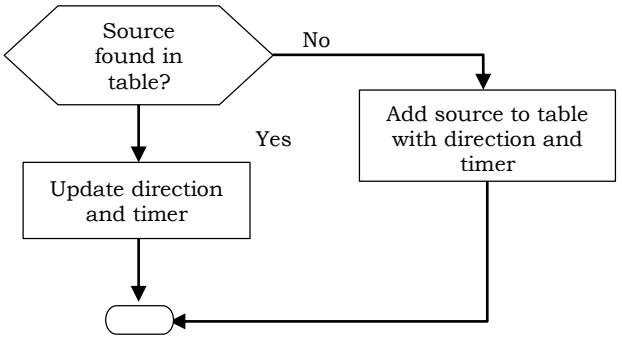
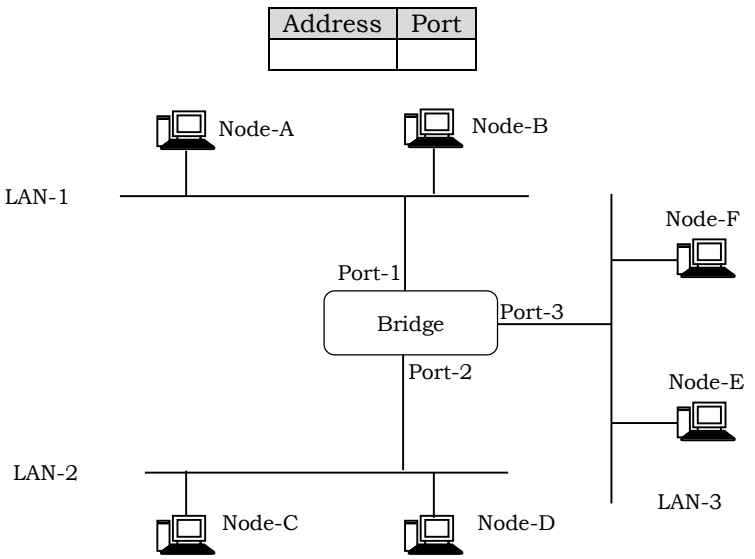


Table Building

The table building up operation is illustrated in figure. Initially the table is empty.



1. When node A sends a frame to node D, the bridge does not have any entry for either D or A. The frame goes out from all three ports. The frame floods the

network. However, by looking at the source address, the bridge learns that node A must be located on the LAN connected to port 1.

This means that frame destined for A (in future), must be sent out through port 1. The bridge adds this entry to its table. The table has its first entry now.

Address	Port
A	1

2. When node E sends a frame to node A, the bridge has an entry for A, so it forwards the frame only to port 1. There is no flooding. Also, it uses the source address of the frame (E in this case), to add a second entry to the table.

Address	Port
A	1
E	3

3. When node B sends a frame to C, the bridge has no entry for C, so once again it floods the network and adds one more entry to the table.

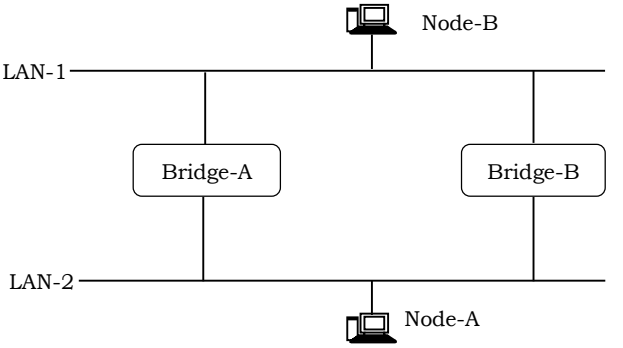
Address	Port
A	1
E	3
B	1

4. The process of learning continues as the bridge forwards frames.

Loop Problem

Forwarding and learning processes work without any problem as long as there is no redundant bridge in the system. On the other hand, redundancy is desirable from the viewpoint of reliability, so that the function of a failed bridge is taken over by a redundant bridge.

The existence of redundant bridges creates the so-called loop problem as shown figure. Assuming that after initialization tables in both the bridges are empty let us consider the following steps:



- *Step* 1: Node A sends a frame to node B. Both the bridges forward the frame to LAN 1 and update the table with the source address of A.
- *Step* 2: Now there are two copies of the frame on LAN-1. The copy sent by Bridge-A is received by Bridge-B and vice versa. As both the bridges have no information about node B, both will forward the frames to LAN-2.
- *Step* 3: Again both the bridges will forward the frames to LAN-1 because of the lack of information of the node B in their database and again Step-2 will be repeated, and so on.

So, the frame will continue to *loop* around the two LANs indefinitely.

4.4.1.2.3 Transparent Spanning Bridges

As seen in previous section, redundancy creates loop problem in the system and it is undesirable. To prevent loop problem, the IEEE (Institute of Electrical and Electronics Engineers) specification requires that the bridges use a special topology. Such a topology is known as *spanning tree* (a graph where there is no loop) topology.

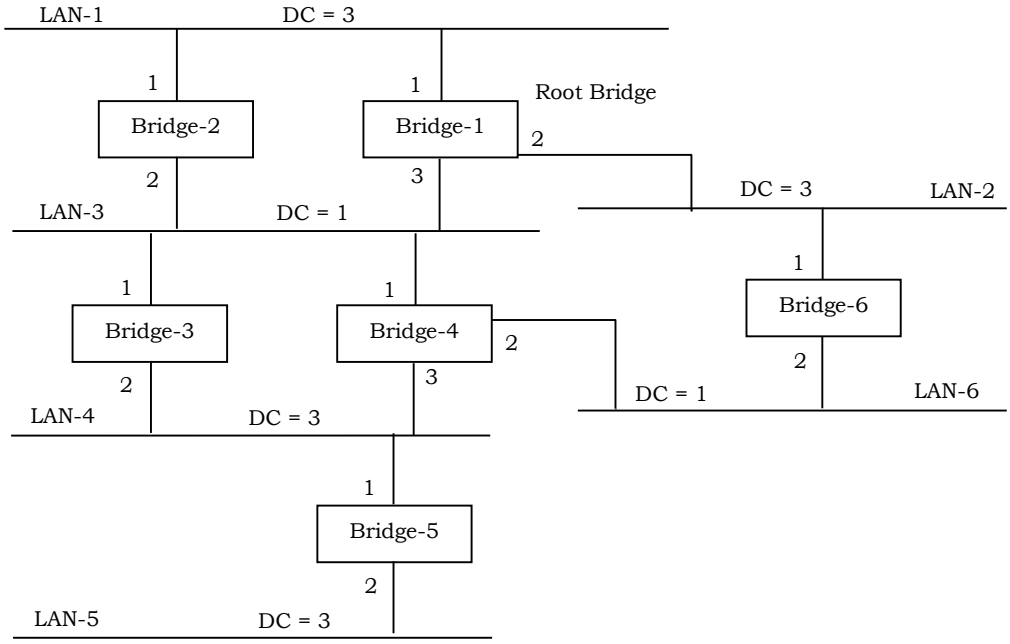
The methodology for setting up a spanning tree is known as *spanning tree algorithm*. *Spanning tree algorithm* creates a tree out of a graph. Without changing the physical topology, a logical topology is created that overlay on the physical by using the following steps:

1. Select a bridge as *root-bridge*, which has the smallest ID.
2. Select root ports for all the bridges, except for the root bridge, which has least-cost path (say, minimum number of hops) to the root bridge.
3. Choose a *designated* bridge, which has least-cost path to the *root-bridge*, in each LAN.
4. Select a port as *designated port* that gives least-cost path from the *designated bridge* to the *root* bridge.
5. Mark the designated port and the root ports as *forwarding* ports and the remaining ones as *blocking* ports.

An Example

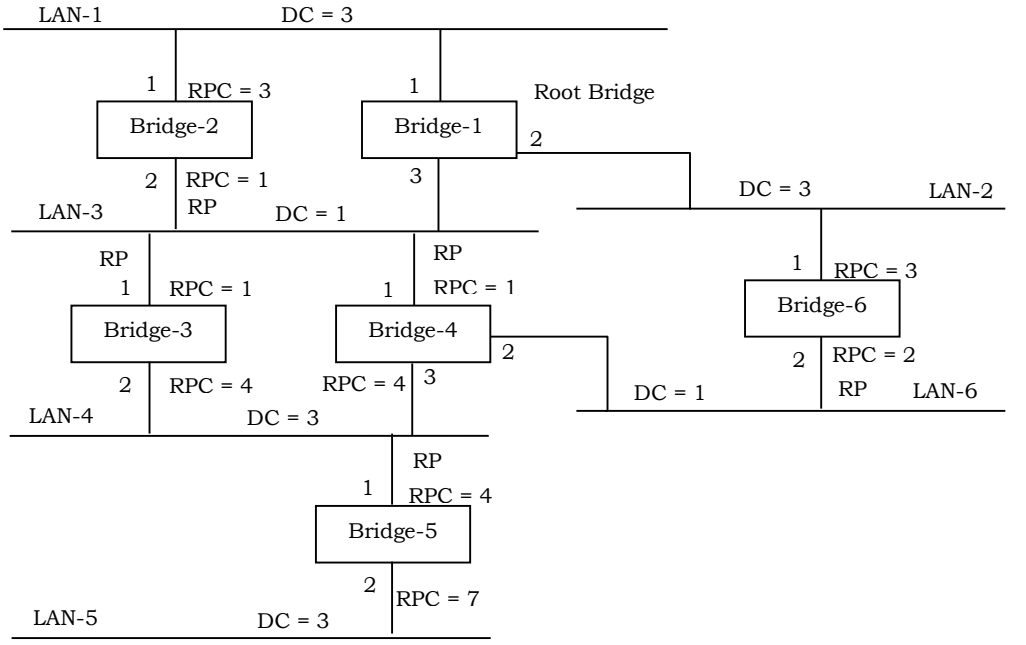
Let us walk through the below example for running the spanning tree algorithm on. Note that some of the LAN segments have a cost 3 times that of others. The following convention is used for the remaining discussion:

- DC means designated cost for a LAN segment
- Bridge-# means bridge number
- A number around a bridge is a port number



Step 1 of the algorithm is already shown in the first picture: Bridge 1 is chosen as the *root bridge* since all the bridges are assumed to have the same priority. The tie is broken by choosing the bridge with the smallest ID number.

Next, we determine the root path cost (RPC) for each port on each bridge *other than* the *root* bridge. Then each bridge other than the root chooses its port with the lowest RPC as the root port (RP). Ties are broken by choosing the *lowest-numbered* port. The root port is used for all control messages from the root bridge to this particular bridge.



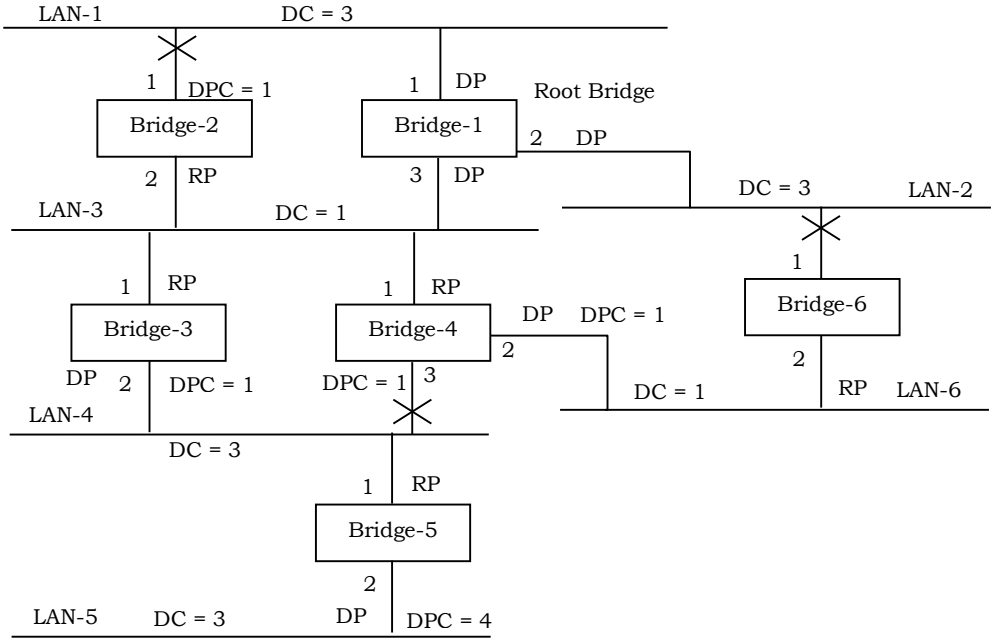
Sample RPC calculation: Consider port 1 of Bridge-5. Between it and root bridge we have to traverse at least LAN-3 and LAN-4, with costs 1 and 3 respectively. Total cost is 4. Thus RPC = 4 for port 1 of Bridge-5.

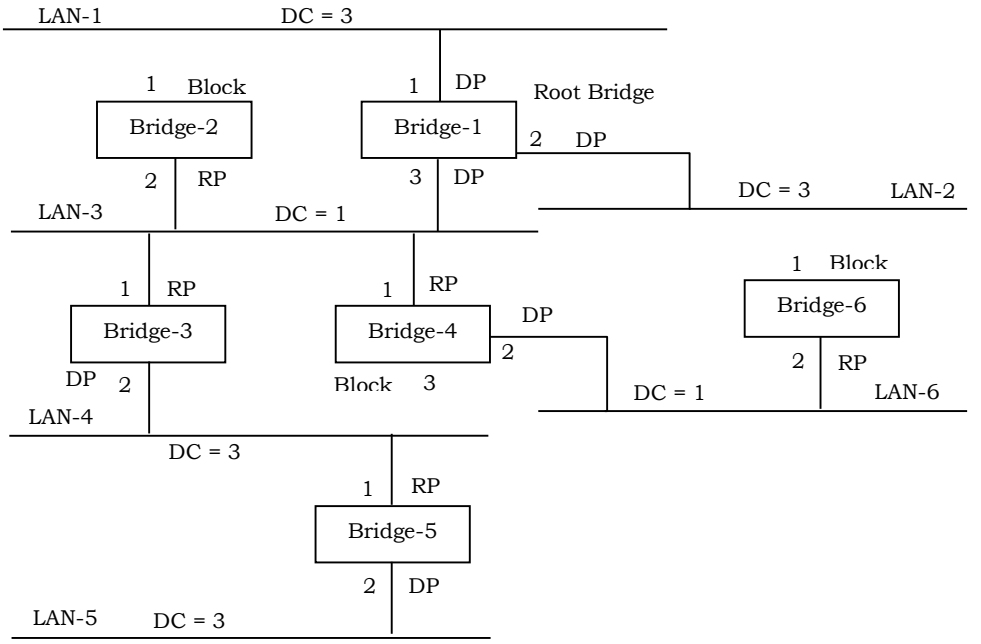
Next, step 3 of the algorithm is to select a designated bridge and a designated port on this bridge for each LAN segment. This is the bridge that gives the least cost (DPC, designated port cost) for getting between this LAN segment and the root bridge. The port on this bridge by which we attach this LAN segment is called the *designated port* (DP). If there is a tie for the lowest DPC, the bridge with the smallest ID number is chosen.

The root bridge is always the designated bridge for the LAN segments directly attached to it. The ports by which the root bridge attaches to the LAN segments are thus designated ports. We assume that no LAN segment attaches to the root bridge by more than 1 port. Since a root port cannot be chosen as a designated port, do not waste time even considering root ports as possible designated ports.

In the drawing on the next page, we see that LAN-1, LAN-2, and LAN-3 are directly attached to the root bridge via ports 1, 2, and 3 respectively on the root bridge. Thus we only need to consider LAN-4, LAN-5, and LAN-6. LAN-4 could use either port 2 on Bridge-3 or port 3 on Bridge-4 as its designated port. The DPC for each is 1 since anything sent from LAN-4 through such a port goes across LAN-3 to the root bridge and the cost of LAN-3 is just 1.

Since we have a tie for the DP we choose the one on the lowest number bridge. That means that Bridge-3 is the designated bridge and its port 2 is the designated port for LAN-3. For LAN-5 there is only one port that could be chosen, so the designated port for LAN-5 is port 2 on Bridge-5 and the designated bridge is Bridge-5. There is no choice for LAN-6 either as one port is a root port. Thus the designated port for S6 is the other one: port 2 on Bridge-4.

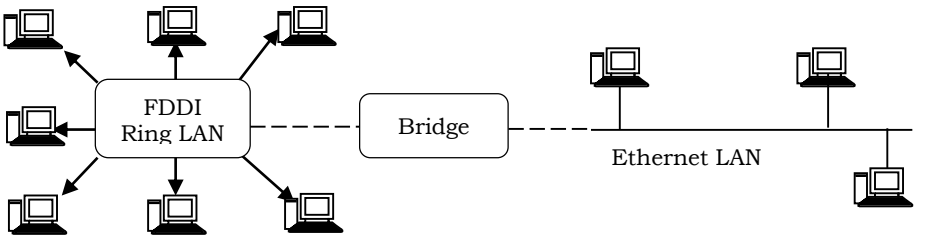




Finally, in step 4 each port that is not a root port or designated port is set to be in a blocking state so that no traffic can flow through it. The blocked ports are X-ed out

above. This, then, produces our spanning tree (no loops). To better see the spanning tree, the picture can be redrawn as shown on the next page, with the root bridge as the root of the tree.

4.4.1.2.4 Translational Bridges

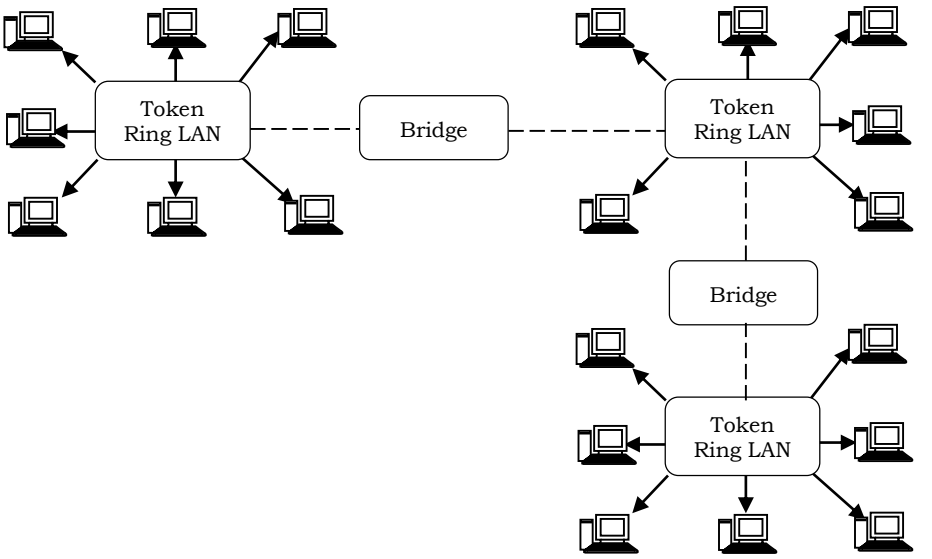


Translational bridges are a type of transparent bridge that connects LANs that use different protocols at the data link and physical layers, for example, FDDI (Fiber Distributed Data Interface) and Ethernet.

4.4.1.2.5 Source Routing Bridges

In source routing bridges, the routing operation is determined by the source host and the frame specifies which route the *frame* to follow. A host can discover a route by sending a *discovery* frame, which spreads through the entire network using all possible paths to the destination.

Each frame gradually gathers addresses as it goes. The destination responds to each frame and the source host chooses an appropriate route from these responses. For example, a route with minimum *hop-count* can be chosen. Whereas transparent bridges do not modify a frame, a source routing bridge adds a routing information field to the frame. Source routing approach provides a shortest path at the cost of extra burden on the network.



Source route bridging is used in token ring networks. A source route bridge links two or more rings together. There are fundamental characteristics in how a source route bridge transmits a frame between rings. A source route bridge does not create and

maintain forwarding tables. The decision to forward or drop a frame is based on information provided in the frame.

The destination station is responsible for maintaining routing tables that define a route to all workstations on the network. The source workstation is responsible for determining the path of a frame to its destination. If no route information is available, then the source station has the ability to perform route discovery to learn the potential paths that can be taken.

4.4.2 Switches

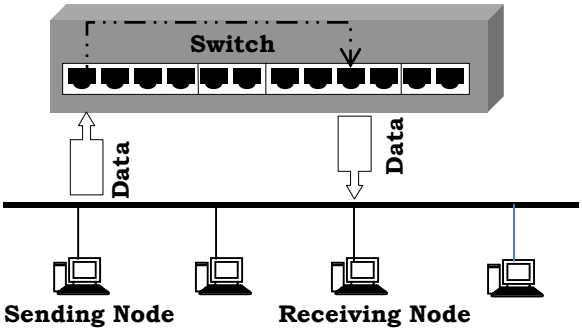
Switch is a device that filters and forwards packets between LAN segments. Switch works at the layer 2 of the OSI model. The main purpose of the switch is to concentrate connectivity while making data transmission more efficient. Think of the switch as something that combines the connectivity of a hub with the traffic regulation of a bridge on each port. Switches makes decisions based on MAC addresses.

A switch is a device that performs switching. Specifically, it forwards and filters OSI layer 2 datagrams (chunk of data communication) between ports (connected cables) based on the MAC addresses in the packets.

As discussed earlier, a hub forwards data to all ports, regardless of whether the data is intended for the system connected to the port. This mechanism is inefficient; and switches tries to address this issue to some extent. This is different from a hub in that it only forwards the datagrams to the ports involved in the communications rather than all ports connected. Strictly speaking, a switch is not capable of routing traffic based on IP address (layer 3) which is necessary for communicating between network segments or within a large or complex LAN.

4.4.2.1 How a Switch works?

Rather than forwarding data to all the connected ports, a switch forwards data only to the port on which the destination system is connected. It looks at the Media Access Control (MAC) addresses of the devices connected to it to determine the correct port.



A MAC address is a unique number that is stamped into every NIC. By forwarding data only to the system to which the data is addressed, the switch decreases the amount of traffic on each network link dramatically.

4.4.2.2 Switching Methods

We can specify one of *four* possible forwarding methods for each port in a switch:

1. Cut-through
2. Fragment-free
3. Store-and-forward
4. Adaptive

4.4.2.2.1 Store and Forward Switching

In *store* and *forward* switching, Switch copies each of the complete Ethernet frame into the switch memory and computes a Cyclic Redundancy Check (CRC) for errors. If a Cyclic Redundancy Check (CRC) error is found, the Ethernet frame is dropped and if there is no Cyclic Redundancy Check (CRC) error, the switch forwards the Ethernet frame to the destination device. Store and Forward switching can cause delay in switching since Cyclic Redundancy Check (CRC) is calculated for each Ethernet frame.

4.4.2.2.2 Cut-through Switching

In *cut-through* switching, the switch copies into its memory only the destination MAC address (first 6 bytes of the frame) of the frame before making a switching decision. A switch operating in cut-through switching mode reduces delay because the switch starts to forward the Ethernet frame as soon as it reads the destination MAC address and determines the outgoing switch port. Problem related with cut-through switching is that the switch may forward bad frames.

4.4.2.2.3 Fragment-Free Switching

Fragment-free switching is an advanced form of cut-through switching. The switches operating in cut-through switching read only up to the destination MAC address field in the Ethernet frame before making a switching decision. The switches operating in fragment-free switching read at least 64 bytes of the Ethernet frame before switching it to avoid forwarding Ethernet runt frames (Ethernet frames smaller than 64 bytes).

4.4.2.2.4 Adaptive switching

Adaptive switching mode is a user-defined facility to maximize the efficiency of the switch. Adaptive switching starts in the default switch forwarding mode we have selected. Depending on the number of errors (say, CRC errors) at that port, the mode changes to the *best* of the other two switching modes.

4.4.3 Routers

4.4.3.1 What is Router?

Routers are *physical* devices that join multiple *networks* together. Technically, a router is a Layer 3 device, meaning that it connects two or more networks and that the router operates at the network layer of the OSI model.

Routers maintain a table (called *routing table*) of the available routes and their conditions and use this information along with distance and cost algorithms to

determine the best route for a given packet. Typically, a packet may travel through a number of network points with routers before arriving at its destination.

The purpose of the router is to examine incoming packets (layer 3), chose the best path for them through the network, and then switches them to the proper outgoing port. Routers are the most important traffic controlling devices on large networks.

Routers are networking devices that forward data packets between networks using headers and *forwarding tables* to determine the best path to forward the packets. Routers also provide interconnectivity between *like* and *unlike* media (networks which use different protocols).

4.4.3.2 Understanding Concepts of Routers

As an example, assume that we want to send a postcard just based on person names (with minimum information). For example, *Bill Gate*s [USA], *Sachin Tendulkar* [India] or *Albert Einstein* [USA] it would be routed to them due to their fame; no listing of the street address or the city name would be necessary. The postal system can do such routing to famous personalities, depending on the name alone.

In an Internet, a similar discussion is possible: *reach* any *website* anywhere in the world without knowing where the site is currently located. Not only that, it is possible to do so very efficiently, within a matter of a few seconds.

4.4.3.2.1 What is Network Routing?

How is this possible in a communication network, and how can it be done so quickly? The answer to this question is *Network routing*. *Network routing* is the ability to send a unit of information from source to destination by finding a path through the network, and by doing efficiently and quickly.

4.4.3.2.2 What is Addressing?

First, we start with a key and necessary factor, called *addressing*. In many ways, addressing in a network has similarities to postal addressing in the postal system. So, we will start with a brief discussion of the postal addressing system to relate them.

A typical postal address that we write on a postcard has several components—the name of the person, followed by the street address with the house number (*house address*), followed by the city, the state name, and the postal code. If we take the

processing view to route the postcard to the right person, we essentially need to consider this address in the reverse order of listing, i.e., start with the postal code, then the city or the state name, then the house address, and finally the name of the person.

You may notice that we can reduce this information somewhat; that is, you can just use the postal code and leave out the name of the city or the name of the state, since this is redundant information. This means that the information needed in a postal address consists of three main parts: the postal code, the street address (with the house number), and the name.

A basic routing problem in the postal network is as follows:

1. The postcard is first routed to the city or the geographical region where the postal code is located.
2. Once the card reaches the postal code, the appropriate delivery post office for the address specified is identified and delivered to.
3. Next, the postman or postwoman delivers the postcard at the address, without giving much consideration to the name listed on the card.
4. Rather, once the card arrives at the destination address, the residents at this address take the responsibility of handing it to the person addressed.

The routing process in the postal system is broken down to three components:

- How to get the card to the specific postal code (and subsequently the post office),
- How the card is delivered to the destination address, and
- Finally, how it is delivered to the actual person at the address.

If we look at it in another way, the place where the postcard originated in fact does not need to know the detailed information of the street or the name to start with; the postal code is sufficient to determine to which geographical area or city to send the card. So, we can see that postal routing uses address hierarchy for routing decisions.

An advantage of this approach is the decoupling of the routing decision to multiple levels such as the postal code at the top, then the street address, and so on. An important requirement of this hierarchical view is that there must be a way to divide the complete address into multiple distinguishable parts to help with the routing decision.

Now, consider an electronic communication network; for example, a critical communication network of the modern age is the Internet. Naturally, the first question that arises is: how does addressing work for routing a unit of information from one point to another, and is there any relation to the postal addressing hierarchy that we have just discussed? Second, how is service delivery provided? In the next section, we address these questions.

4.4.3.2.3 Addressing and Internet Service: An Overview

In many ways, Internet addressing has similarities to the postal addressing system. The addressing in the Internet is referred to as *Internet Protocol* (IP) *addressing*. An IP address defines *two* parts: one part that is similar to the postal code and the other part that is similar to the house address; in Internet terminology, they are known as the *netid* and the *hostid*, to identify a network and a host address, respectively.

A host is the end point of communication in the Internet and where a communication starts. A host is a generic term used for indicating many different entities; the most common ones are a web-server, an email server, desktop, laptop, or any computer we use for accessing the Internet. A *netid* identifies a contiguous block of addresses.

4.4.3.2.4 Network Routing: An Overview

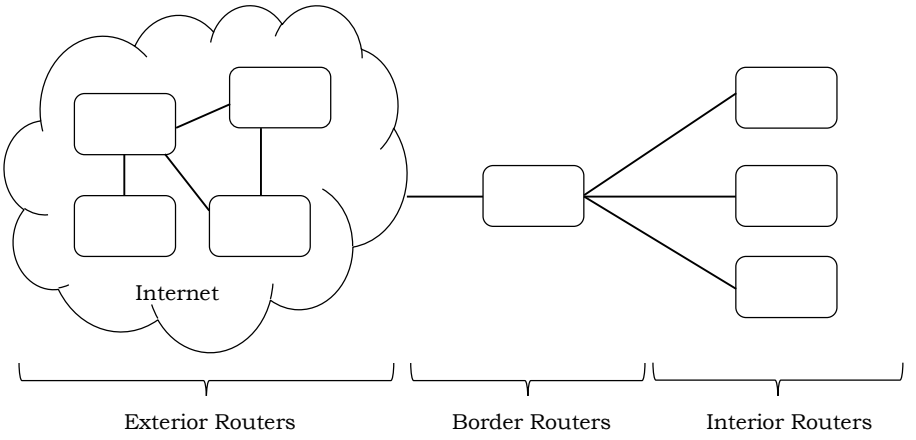
In the previous section, we provided a broad overview of addressing and transfer mechanisms for data in Internet communication services. Briefly, we can see that eventually packets are to be routed from a source to a destination. Such packets may need to traverse many cross-points, similar to traffic intersections in a road transportation network. Cross-points in the Internet are known as *routers*.

A router's functions are to read the destination address marked in an incoming IP packet, to consult its internal information to identify an outgoing link to which the packet is to be forwarded, and then to forward the packet. Similar to the number of lanes and the speed limit on a road, a network link that connects two routers is limited by how much data it can transfer per unit of time, commonly referred to as the band-width or capacity of a link; it is generally represented by a data rate, such as 1.54 megabits per second (Mbps). A network then carries traffic on its links and through its routers to the eventual destination; traffic in a network refers to packets generated by different applications, such as web or email.

Note: For more about IP Addressing and routing, refer *IP Addressing* and *Routing Protocols* chapters.

4.4.3.3 Types of Routers

Depending on the role that routers perform, routers can be classified in many different ways.



4.4.3.3.1 Interior Routers

Interior routers work within networks. These routers handle packets travelling between nodes on the same Intra-network. An interior router is used to divide a large network into more easily manageable subnetworks. It can keep one part of a network secure from another and it can allow different technologies, for example, Ethernet and token ring, to be used in the same network.

4.4.3.3.2 Border Routers

Border routers exist on one network and their function is to connect that network with outside networks, including the Internet. They discover routes between the interior network and others and they handle incoming and outgoing traffic.

4.4.3.3.3 Exterior Routers

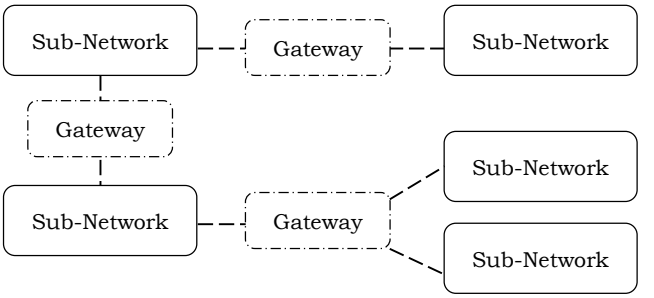
Exterior routers are most common on the Internet. They do not exist on a particular network but rather in the space between networks where data passes through on its way to its destination. Exterior routers do not store routes to particular hosts; but they store routes to other *routers*. Their primary role is to receive packets and then forward them in the direction of their destination.

4.4.4 Gateways

The term *gateway* is used in networking to describe the *gate* to the Internet. The *gateway* controls traffic that travels from the inside network to the Internet and provides security from traffic that wants to enter the inside network from the Internet.

A network gateway is an internetworking system which joins two networks that use different base protocols. A network gateway can be implemented completely in software, completely in hardware, or as a combination of both. Depending on the types of protocols they support, network gateways can operate at any level of the OSI model.

Since a gateway (by definition) appears at the edge of a network, related capabilities like firewalls tend to be integrated with it. On home networks, a router typically serves as the network gateway although ordinary computers can also be configured to perform equivalent functions.



As mentioned earlier, the Internet is not a single network but a collection of networks that communicate with each other through gateways. A gateway is defined as a system that performs relay functions between networks, as shown in figure above. The different networks connected to each other through gateways are often called *subnetworks*, because they are a smaller part of the larger overall network.

With TCP/IP, all interconnections between physical networks are through gateways. An important point to remember for use later is that gateways route information packets based on their destination network name, not the destination machine. Gateways are completely transparent to the user.

4.4.1 Default Gateway

The default gateway is needed only for systems that are part of an internetwork (in the above figure, note that two subnetworks connected to same gateway). Data packets with a destination IP address not on the local subnet are forwarded to the default gateway. The default gateway is normally a computer system or router connected to the local subnet and other networks in the internetwork.

If the default gateway becomes unavailable, the system cannot communicate outside its own subnet, except for with systems that it had established connections with prior to the failure.

4.4.2 Multiple Gateways

If the default gateway becomes unavailable, data packets cannot reach their destination. *Multiple gateways* can be used to solve this problem.

4.4.3 Difference between Gateway and Router

4.4.3.1 Gateway

The *key difference* between gateway and router is, gateway it is defined as a network node that allows a network to interface with another network with different protocols. A router is a device that is capable of sending and receiving data packets between computer networks, also creating an overlay network.

Gateways and routers are two words are often confused due to their similarities. Both gateways and routers are used to regulate traffic into more separate networks. However, these are two different technologies and are used for different purposes.

The term gateway can be used to define two different technologies: gateway and default gateway. These two terms should not be confused. In terms of communications network, gateway it is defined as a network node that allows a network to interface with another network with different protocols. In simple terms, gateway allows two different networks to communicate with each other. It contains devices such as impedance protocol translators, rate converters, or signal translators to allow system interoperability.

A protocol translation/mapping gateway interconnects networks that have different network protocol technologies. Gateways acts as a network point that acts as an entrance to another network. The gateway can also allow the network to connect the computer to the internet. Many routers are available with the gateway technology, which knows where to direct the packet of data when it arrives at the gateway. Gateways are often associated with both routers and switches.

Default gateway is a computer or a computer program that is configured to perform the tasks of a traditional gateway. These are often used by ISP or computer servers that act as gateway between different systems. When getting an internet connection, an ISP usually provides a device that allows the user to connect to the Internet; these devices are called *modems*. In organizational systems a computer is used as a node to connect the internal networks to the external networks, such as the Internet.

4.4.3.2 Router

A router is a device that is capable of sending and receiving data packets between computer networks, also creating an overlay network. The router connects two or more data line, so when a packet comes in through one line, the router reads the address information on the packet and determines the right destination, it then uses the information in its routing table or routing policy to direct the packet to the next network. On the internet, routers perform *traffic directing* functions. Routers can also be wireless as well as wired.

The most common type of routers is small office or home routers. These are used for passing data from the computer to the owner's cable or DSL modem, which is connected to the internet. Other routers are huge enterprise types that connect large businesses to powerful routers that forward data to the Internet.

When connected in interconnected networks, the routers exchange data such as destination addresses by using a dynamic routing protocol. Each router is responsible for building up a table that lists the preferred routes between any two systems on the

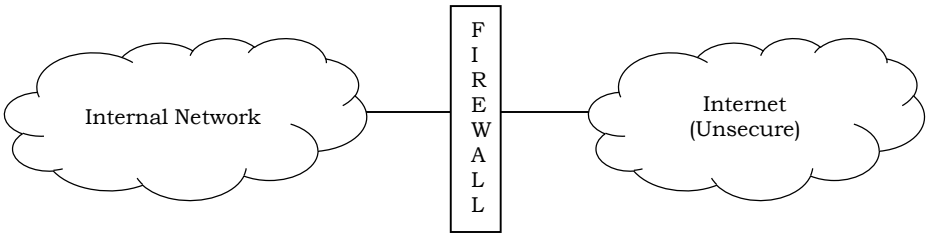
interconnected networks. Routers can also be used to connect two or more logical groups of computer devices known as subnets. Routers can offer multiple features such as a DHCP server, NAT, Static Routing, and Wireless Networking.

These days' routers are mostly available with built-in gateway systems make it easier for users with them not having to buy separate systems.

4.4.5 Firewalls

The term firewall was derived from *civil engineering* and intended to *prevent* the*spread* of fire from one *room* to another. From the computer security perspective, the Internet is an unsafe environment; therefore *firewall* is an excellent metaphor for network security.

A firewall is a system designed to prevent unauthorized access to or from a private network. Firewalls can be implemented in either hardware or software form, or a combination of both. Firewalls prevent unauthorized users from accessing private networks. A firewall sits between the two networks, usually a private network and a public network such as the Internet.

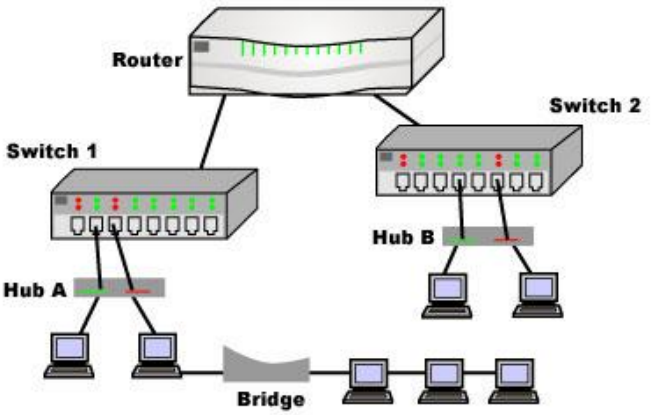


Connecting a computer or a network of computers may become targets for malicious software and hackers. A firewall can offer the security that makes a computer or a network less vulnerable.

Note: For more details, refer *Firewalls* section in *Network Security* chapter.

4.4.6 Differences between Hubs, Switches, and Routers

Today most routers have something combining the features and functionality of a router and switch/hub into a single unit. So conversations regarding these devices can be a bit misleading — especially to someone new to computer networking.



The functions of a router, hub and a switch are all quite different from one another, even if at times they are all integrated into a single device. Let's start with the hub and

the switch since these two devices have similar roles on the network. Each serves as a central connection for all of your network equipment and handles a data type known as frames. Frames carry the data. When a frame is received, it is amplified and then transmitted on to the port of the destination PC. The big difference between these two devices is in the method in which frames are being delivered.

In a hub, a frame *broadcasts* to every one of its ports. It doesn't matter that the frame is only destined for one port. The hub cannot distinguish which port a frame should be sent to. Broadcasting it on every port ensures that it will reach its intended destination. This places a lot of traffic on the network and can lead to poor network response times.

Additionally, a 10/100Mbps hub must share its bandwidth with each and every one of its ports. So, when only one PC is broadcasting, it will have access to the maximum available bandwidth. If, however, multiple PCs are broadcasting, then that bandwidth will need to be divided among all of those systems, which will degrade performance.

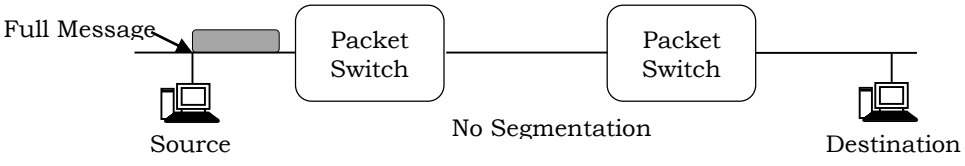
A switch, however, keeps a record of the *MAC* addresses of all the devices connected to it. With this information, a switch can identify which system is sitting on which port. So, when a frame is received, it knows exactly which port to send it to, without significantly increasing network response times. And, unlike a hub, a 10/100Mbps switch will allocate a full 10/100Mbps to each of its ports. So regardless of the number of PCs transmitting, users will always have access to the maximum amount of bandwidth. It's for these reasons why a switch is considered to be a much better choice than a hub.

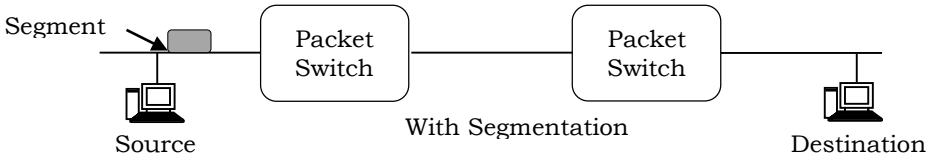
Routers are completely different devices. Where a hub or switch is concerned with transmitting frames, a router's job, as its name implies, is to route packets to other networks until that packet ultimately reaches its destination. One of the key features of a packet is that it not only contains data, but the destination address of where it's going.

A router is typically connected to at least two networks, commonly two Local Area Networks (LANs) or Wide Area Networks (WAN) or a LAN and its ISP's network, for example, your PC or workgroup and EarthLink. Routers are located at gateways, the places where two or more networks connect. Using headers and forwarding tables, routers determine the best path for forwarding the packets. Router use protocols such as ICMP to communicate with each other and configure the best route between any two hosts.

Problems and Questions with Answers

Question 1: In modern packet-switched networks, the source host segments long, application-layer messages (for example, an image or a music file) into smaller packets and sends the packets into the network. The receiver then reassembles the packets back into the original message. We refer to this process as *message segmentation*. Figure shows the end-to-end transport of a message with and without message segmentation. Consider a message that is 9.0106 bits long that is to be sent from source to destination in figure. Suppose each link in the figure is 1.5 Mbps. Ignore propagation, queuing, and processing delays.





A) Consider sending the message from source to destination without message segmentation. How long does it take to move the message from the source host to the first packet switch? Keeping in mind that each switch uses store-and-forward packet switching, what is the total time to move the message from source host to destination host?

B) Now suppose that the message is segmented into 5,000 packets, with each packet being 1,500 bits long. How long does it take to move the first packet from source host to the first switch? When the first packet is being sent from the first switch to the second switch, the second packet is being sent from the source host to the first switch. At what time will the second packet be fully received at the first switch?

C) How long does it take to move the file from source host to destination host when message segmentation is used? Compare this result with your answer in part (A) and comment.

Answer:

A) Time to send message from source host to first packet switch $=\frac{9\times10^6}{1.5\times10^6}$sec = 6 sec. With store-and-forward switching, the total time to move message from source host to destination host = 6 sec × 3 hops = 18 sec.

B) Time to send 1st packet from source host to first packet switch = $\frac{1.5\times10^3}{1.5\times10^6}$ sec = 1 msec.

Time at which second packet is received at the first switch = 1.5×10^6 time at which first packet is received at the second switch = 2 × 1 msec = 2 msec.

C) Time at which 1st packet is received at the destination host = 1 msec × 3 hops = 3 msec . After this, every 1msec one packet will be received; thus time at which last (5000^{th}) packet is received = 3 msec + 4999 × 1 msec = 5.002 sec.

It can be seen that delay in using message segmentation is significantly less (more than $\frac{1}{3}$ rd).

Question 2: For the following statement, indicate whether the statement is True or False.

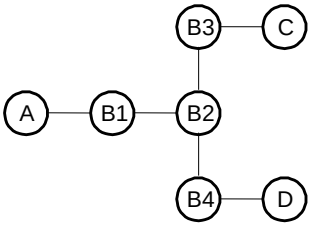
Switches exhibit lower latency than routers.

Answer: True. No routing table look-up, no delays associated with storing data *queuing*, bits flow through the switch essentially as soon as they arrive.

Question 3: Packet switches have queues while circuit switches do not. Is it true or false?

Answer: False. Routers have queues; switches do not, even though the packet switch must have more memory than a circuit switch to receive a full packet before it can forward it on.

Question 4: Consider the arrangement of learning bridges shown in the following figure. Assuming all are initially empty, give the forwarding tables for each of the bridges B1-B4 after the following transmissions:



D sends to C; A sends to D; C sends to A

Answer: When D sends to C, all bridges see the packet and learn where D is. However, when A sends to D, the packet is routed directly to D and B3 does not learn where A is. Similarly, when C sends to A, the packet is routed by B2 towards B1 only, and B4 does not learn where C is.

The forwarding table for Bridge B1:

Destination	Next Hop
A	A-Interface
C	B2-Interface
D	B2-Interface

The forwarding table for Bridge B2:

Destination	Next Hop
A	B1-Interface
C	B3-Interface
D	B4-Interface

The forwarding table for Bridge B3:

Destination	Next Hop
C	C-Interface
D	B2-Interface

The forwarding table for Bridge B4:

Destination	Next Hop
A	B2-Interface
D	D-Interface

Question 5: Which type of bridge observes network traffic flow and uses this information to make future decisions regarding frame forwarding?
A) Remote B) Source routing C) Transparent D) Spanning tree

Answer: C

Question 6: Learning network addresses and converting frame formats are the function of which device?
A) Switch B) Hub C) MAU D) Bridge

Answer: D

Question 7: The device that can operate in place of a hub is a:
A) Switch B) Bridge C) Router D) Gateway

Answer: A

Question 8: Which of the following is NOT true with respective to a transparent bridge and a router?
A) Both bridge and router selectively forward data packets

B) A bridge uses IP addresses while a router uses MAC addresses
C) A bridge builds up its routing table by inspecting incoming packets
D) A router can connect between a LAN and a WAN.

Answer: B. Bridge is the device which work at data link layer whereas router works at network layer. Both selectively forward packets, build routing table and connect between LAN and WAN but since bridge works at data link it uses MAC addresses to route whereas router uses IP addresses.

CHAPTER

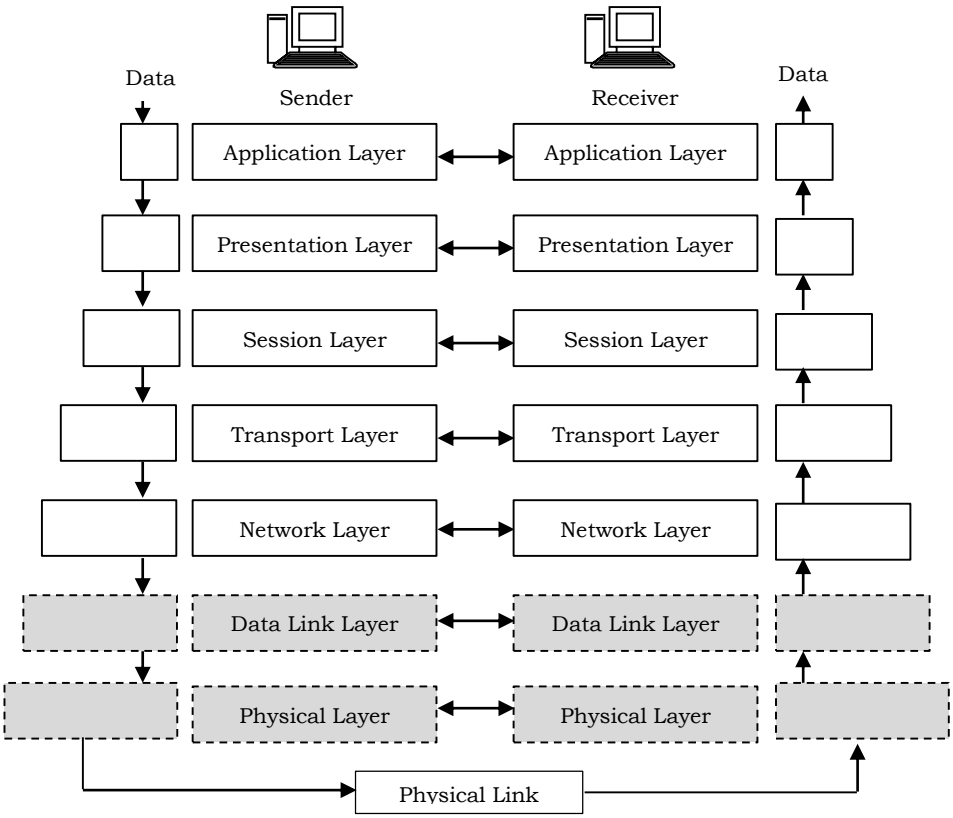
LAN Technologies

5

5.1 Introduction

The bottom two layers of the Open Systems Interconnection (OSI) model deal with the physical structure of the network and the means by which network devices can send information from one device on a network to another.

The data link layer controls how data packets are sent from one node to another.



5.2 Types of Network Links

There are two types of network links: *point-to-point* links, and *broadcast* links.

5.2.1 Broadcasting Network Links

Broadcast is a method of sending a signal where multiple nodes may hear a single sender node. As an example, consider a conference room with full of people. In this conference room, a single person starts saying some information loudly.

During that time, some people may be sleeping, and may not hear what person is saying. Some people may not be sleeping, but not paying attention (they are able to hear the person, but choose to ignore). Another group of people may not only be awake, but be interested in what is being said. This last group is not only able to hear the person speaking, but is also listening to what is being said.

In this example, we can see that a single person is broadcasting a message to all others that may or may not be able to hear it, and if they are able to hear it, may choose to listen or not.

5.2.1.1 Simplex Broadcasting Network

Radio and TV stations are a good examples of everyday life *broadcast networks*. In this case the radio/TV stations are a type of communications called *Simplex*. In a simplex type of communication, data is only expected to flow in one direction.

5.2.1.2 Half-Duplex Broadcasting Network

Conference-room meetings are another everyday example of a broadcast network. In this example, everyone may speak to everyone else, but when more than one person speaks, interference (collision) from multiple conversations may make it impossible to listen to more than one conversation even though we can hear both conversations. In this conference-room example, we can see parties are able to share access to a common media (human voice as sound through the air.) They compete for access to speak, but for the most part, only one person speaks at a time for everyone to hear. This is an example of a type of communications called *half-duplex*.

5.2.1.3 Full-Duplex Broadcasting Network

Let us consider the singing competition where we can see a group of singers attempting to sing in Harmony. They can each speak separately on their own, but if they speak on different topics, the conveyed information for any of them may be lost by each other. This is an example of another type of communication called *full-duplex*.

This means that they are not only able to speak, but listen at the same time they are speaking. All of them will speak and listen at the same time. How is this possible? In order to sing in harmony, each singer must be able to hear the frequencies being used by the other singers, and strive to create a frequency with their voice that matches the desired frequency to create that harmony.

This feed-back of each singer to listen to the collective, and possibly key into a specific singer's voice is used by them as they sing to create the exact frequency needed, and ensure their timing is the same as the rest of the singers. All members are able to hear all other members, and speak at the same time. They are all acting as a *full-duplex* communications in a broadcast network.

5.2.2 Point-to-Point Network Links

Point-to-Point is a method of communication where one node speaks to another node. A woman in a restaurant whispers to her husband a message. Nobody else in the restaurant knows what was said. The conversation was only between them.

5.2.2.1 Simplex Point-to-Point Network

An example of a very simple *simplex* point-to-point network could be a doorbell (the circuit.) When the doorbell button is depressed at the front door, a signal is passed to bell which performs its functions to announce the button has been depressed. The bell does not send a message to button. The message travels only in one direction and takes place between the button and the bell.

5.2.2.2 Half-Duplex Point-to-Point Network

As an example, let us assume that we have a couple who are openly affectionate, sat on a bench in a park, and holding hands under a blanket.

Also, assume that this couple has their own code in holding hands for speaking to each other. For example, 3 squeezes maps to *I Love You* and 4 squeezes of the hand maps to *I Love You Too*. The wife squeezes her husband's hand 3 times. He gets this message, and smiles (acknowledging the receipt of the message) and then returns a new message of 4 squeezes. She smiles (acknowledging her receipt of the message she felt.) If both parties attempted to squeeze each other's hands at the same time, then the number of squeezes may be confused. So we can see each party may speak through squeezing each other's hands, but only one may speak at a time.

This conversation takes place only between these two people. Here we see point-to-point and *half-duplex*.

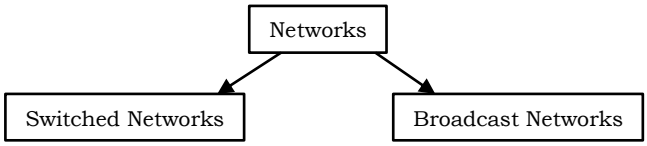
5.2.2.3 Full-Duplex Point-to-Point Network

Data can travel in both directions simultaneously. There is no need to switch from transmit to receive mode like in half duplex. Full-duplex network operates like a two-way, two-lane street. Traffic can travel in both directions at the same time.

5.3 Medium Access Control Techniques

As we have seen, networks can be divided into two types:

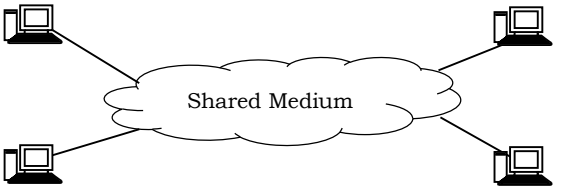
1) S*witched* communication network (also called *point-to-point*, *peer-to-peer,* and *switched*): *peer-to-peer* communication is performed with the help of transmission lines such as multiplexers and switches.
2) *Broadcast* communication network: In this we have a medium which is shared by a number of nodes. *Broadcast* is a method of sending a signal where multiple nodes may hear a single sender node.



A point-to-point link consists of a single sender on one end of the link, and a single receiver at the other end of the link. Many link-layer protocols have been designed for

point-to-point links; PPP (point-to-point protocol) and HDLC (High-level Data Link Control) are two such protocols.

Now, let us consider a different kind of scenario in which we have a medium which is shared by a number of users.



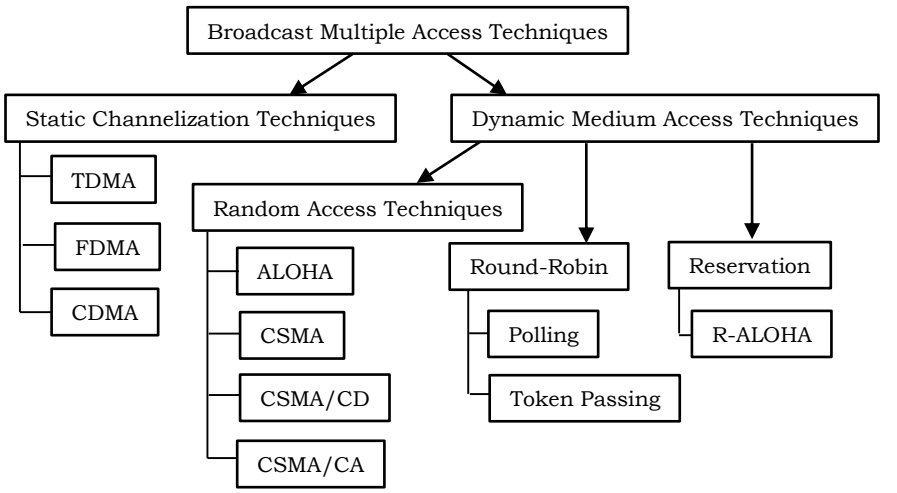
Any user can broadcast the data into the network. Now whenever it is broadcasted obviously there is a possibility that several users will try to broadcast simultaneously. This problem can be addressed with medium access control techniques.

Now question arises how different users will send through the shared media. It is necessary to have a protocol or technique to regulate the transmission from the users. That means, at a time only one user can send through the media and that has to be decided with the help of Medium Access Control (MAC) techniques. Medium access control techniques determines the next user to talk (i.e., transmit into the channel).

A good example is something we are familiar with - a classroom - where teacher(s) and student(s) share the same, single, broadcast medium. As humans, we have evolved a set of protocols for sharing the broadcast channel ("Give everyone a chance to speak." "Don't speak until you are spoken to." "Don't monopolize the conversation." "Raise your hand if you have question." "Don't interrupt when someone is speaking." "Don't fall asleep when someone else is talking.").

Similarly, computer networks have protocols called *multiple access* protocols. These protocols control the nodes data transmission onto the shared broadcast channel.

There are various ways to classify multiple access protocols. Multiple access protocols can be broadly divided into four types; random, round-robin, reservation and channelization. These four categories are needed in different situations. Among these four types, channelization technique is static in nature. We shall discuss each of them one by one.



5.4 Random Access Techniques

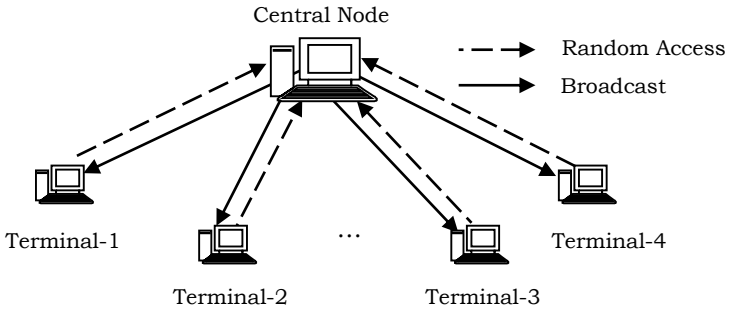
Random access method is also called *contention-based* access. In this method, no station is assigned to control another. Random MAC techniques can be further divided into four different types; ALOHA, CSMA, CSMA/CD and CSMA/CA.

When each node has a fixed flow of information to transmit (for example, a data file transfer), reservation based access methods are useful as they make an efficient use of communication resources. If the information to be transmitted is bursty in nature, the reservation-based access methods are not useful as they waste communication resources.

Random-access methods are useful for transmitting short messages. The random access methods give freedom for each *node* to get access to the network whenever the user has information to send.

5.4.1 ALOHA

Aloha protocol was developed by *Abramson* at *University of Hawaii*. In the *Hawaiian* language, Aloha means *affection*, *peace*, and *compassion*. University of Hawaii consists of a number of islands and obviously they cannot setup wired network in these islands. In the University of Hawaii, there was a centralized computer and there were terminals distributed to different islands. It was necessary for the central computer to communicate with the terminals and for that purpose *Abramson* developed a protocol called *Aloha*.



Central node and terminals (stations) communicate by using a wireless technique called *packet radio*. Each of these stations can transmit by using *uplink* frequency which is *random* access shared by all the terminals. After receiving the data, the central node retransmits by using a *downlink* frequency and that will be received by all terminals.

There are two different types of ALOHA:

1. Pure ALOHA
2. Slotted ALOHA

5.4.1.1 Pure Aloha

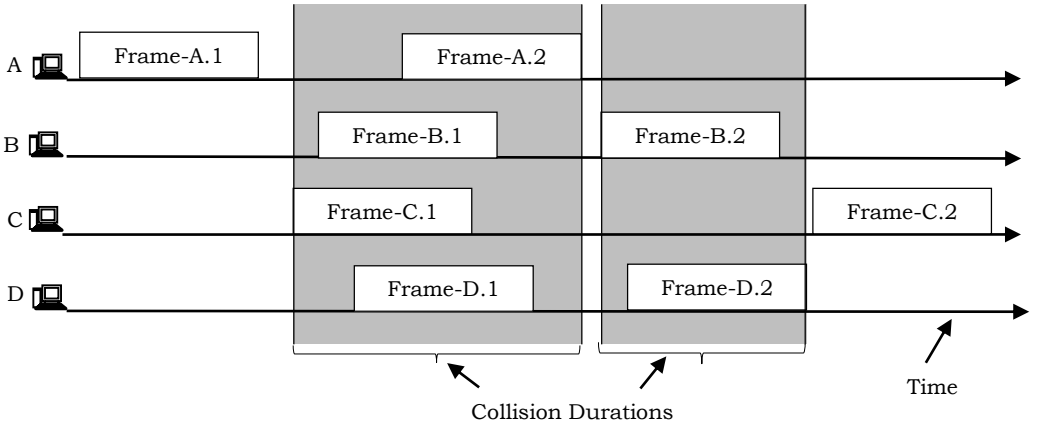
The first version of protocol given by *Amdrason* works like this:

1. If a node has data to send, send the data
2. If the message collides with another transmission, try resending later
3. In case of collision, sender waits random time before retrying

This simple version is also called *pure ALOHA*. Note that, in Pure ALOHA, sender does not check whether the channel is busy before transmitting.

5.4.1.1.1 Frames in Pure ALOHA

Pure ALOHA assumes all frames have the same length. A shared communication system like ALOHA requires a method for handling collisions. Collisions will occur when two or more systems try to send data at the same time. In the ALOHA system, a node transmits whenever data is available to send. If another node transmits at the same time, a collision occurs, and the frames that were transmitted are lost. However, a node can listen to broadcasts on the medium, even its own, and determine whether the frames were transmitted.



As shown in diagram, whenever two frames try to occupy the channel at the same time, there will be a collision and both will be damaged. If first bit of a new frame overlaps with just the last bit of a frame almost finished, both frames will be totally destroyed and both will have to be retransmitted.

5.4.1.1.2 Pure ALOHA Protocol

Pure ALOHA uses two different frequencies for data transfers. The central node broadcasts packets to everyone on the outbound (also called *downlink*) channel, and the terminals sends data packets to the central node on the inbound (also called *uplink*) channel.

If data was received correctly at the central node, a short acknowledgment packet was sent to the terminal; if an acknowledgment was not received by a terminal after a short wait time, it would automatically retransmit the data packet after waiting a randomly selected time interval. This acknowledgment mechanism was used to detect and correct for collisions created when two terminals both attempted to send a packet at the same time.

- In pure ALOHA, the stations transmit frames whenever they have data to send.
- When two or more stations transmit at the same time, there will be a collision and the frames will get destroyed.
- In pure ALOHA, whenever any station transmits a frame, it expects the acknowledgement from the receiver.
- If acknowledgement is not received within specified time, the station assumes that the frame has been destroyed.