UNIT-3

A. Wired LANs: Ethernet

3.1 IEEE STANDARDS

In 1985, the Computer Society of the IEEE started a project, called Project 802, to set standards to enable intercommunication among equipment from a variety of manufacturers. Project 802 does not seek to replace any part of the OSI or the Internet model. Instead, it is a way of specifying functions of the physical layer and the data link layer of major LAN protocols. The standard was adopted by the American National Standards Institute (ANSI).

The relationship of the 802 Standard to the traditional OSI model is shown in the figure. The IEEE has subdivided the data link layer into two sublayers: logical link control (LLC) and media access control (MAC). IEEE has also created several physical layer standards for different LAN protocols.

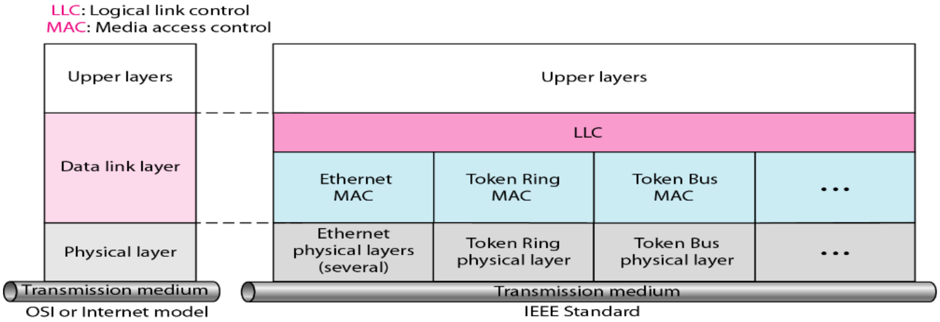


Figure 13.1 IEEE standard for LANs

3.1.1 Data Link Layer

The data link layer in the IEEE standard is divided into two sublayers: LLC and MAC.

Logical Link Control (LLC)

In IEEE Project 802, flow control, error control, and part of the framing duties are collected into one sublayer called the logical link control. Framing is handled in both the LLC sublayer and the MAC sublayer. The LLC is different from the media access control sublayer, which provides different protocols for different LANs. A single LLC protocol can provide interconnectivity between different LANs because it makes the MAC sublayer transparent.

Need for LLC: The purpose of the LLC is to provide flow and error control for the upper-layer protocols that actually demand these services. For example, if a LAN or several LANs are used in an isolated system, LLC may be needed to provide flow and error control for the application layer protocols. However, most upper-layer protocols such as IP, do not use the services of LLC.

Media Access Control (MAC)

IEEE Project 802 has created a sublayer called media access control that defines the specific access method for each LAN. For example, it defines CSMA/CD as the media access method for Ethernet LANs and the token-passing method for Token Ring and Token Bus LANs. Part of the framing function is also handled by the MAC layer. The MAC sublayer contains a number of distinct modules; each defines the access method and the framing format specific to the corresponding LAN protocol.

3.1.2 Physical Layer

The physical layer is dependent on the implementation and type of physical media used. IEEE defines detailed specifications for each LAN implementation. For example, although there is only one MAC sublayer for Standard Ethernet, there is a different physical layer specification for each Ethernet implementations.

3.2 STANDARD ETHERNET

The original Ethernet was created in 1976 at Xerox's Palo Alto Research Center (PARC). Since then, it has gone through four generations: Standard Ethernet (10 Mbps), Fast Ethernet (100 Mbps), Gigabit Ethernet (1 Gbps), and Ten-Gigabit Ethernet (10 Gbps), as shown in the figure 13.3.

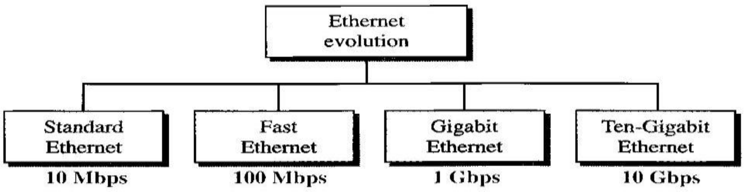


Figure 13.3 Ethernet evolution through four generations

3.2.1 MAC Sublayer

In Standard Ethernet, the MAC sublayer governs the operation of the access method. It also frames data received from the upper layer and passes them to the physical layer.

Frame Format

The Ethernet frame contains following seven fields. Ethernet does not provide any mechanism for acknowledging received frames, so that it is called unreliable medium. Acknowledgments must be implemented at the higher layers. The format of the MAC frame is shown in the figure.

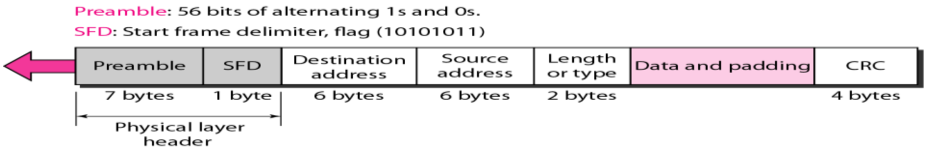


Figure 13.4 802.3 MAC frame

1. **Preamble**: It contains 7 bytes of alternating 0s and 1s that alerts the receiving system to the coming frame and enables it to synchronize its input timing. The pattern provides only an alert and a timing pulse.
2. **Start frame delimiter (SFD):** The second field (1 byte: 10101011) signals the beginning of the frame.
3. **Destination address (DA)**: The DA field is 6 bytes and contains the physical address of the destination station or stations to receive the packet.
4. **Source address (SA)**: The SA field is also 6 bytes and contains the physical address of the sender of the packet.
5. **Length or type**: This field is defined as a type field or length field. The original Ethernet used this field as the type field to define the upper-layer protocol using the MAC frame. The IEEE standard used it as the length field to define the number of bytes in the data field.
6. **Data:** This field carries data encapsulated from the upper-layer protocols. It is a minimum of 46 and a maximum of 1500 bytes.
7. **CRC**: The last field contains error detection information, in this case a CRC-32.

Frame Length

Ethernet has imposed restrictions on both the minimum and maximum lengths of a frame, as shown in fig.

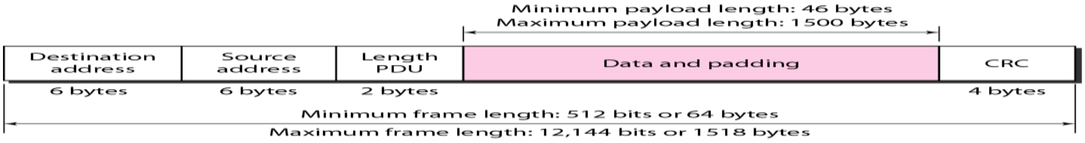

Minimum payload length: 46 bytes
Maximum payload length: 1500 bytes
Destination address
Source address
Length PDU
Data and padding
CRC
6 bytes
6 bytes
2 bytes
4 bytes
Minimum frame length: 512 bits or 64 bytes
Maximum frame length: 12,144 bits or 1518 bytes

Figure 13.5 Minimum and maximum lengths

The minimum length restriction is required for the correct operation of CSMA/CD. An Ethernet frame needs to have a minimum length of 512 bits or 64 bytes. Part of this length is the header and the trailer. If we count 18 bytes of header and trailer, then the minimum length of data from the upper layer is 64 - 18 = 46 bytes. If the upper-layer packet is less than 46 bytes, padding is added to make up the difference.

The standard defines the maximum length of a frame (without preamble and SFD field) as 1518 bytes. If we subtract the 18 bytes of header and trailer, the maximum length of the payload is 1500 bytes. The maximum length restriction has two historical reasons. First, memory was very expensive when Ethernet was designed: a maximum length restriction helped to reduce the size of the buffer. Second, the maximum length restriction prevents one station from monopolizing the shared medium, blocking other stations that have data to send.

Frame length: Minimum: 64 bytes (512 bits) Maximum: 1518 bytes (12,144 bits).

Addressing

Each station on an Ethernet network has its own network interface card (NIC). The NIC fits inside the station and provides the station with a 6-byte physical address. As shown in the figure, the Ethernet address is 6 bytes (48 bits), normally written in hexadecimal notation, with a colon between the bytes.

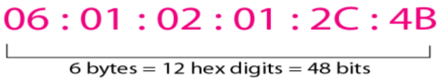

06 : 01 : 02 : 01 : 2C : 4B
6 bytes = 12 hex digits = 48 bits

Figure 13.6 Example of an Ethernet address in hexadecimal notation

Unicast, Multicast, and Broadcast Addresses:

The following figure shows how to distinguish a unicast address from a multicast address. **If the least significant bit of the first byte in a destination address is 0, the address is unicast; otherwise, it is multicast**.

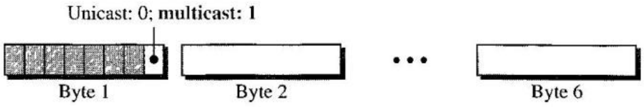


Figure 13.7 Unicast and multicast addresses

- A unicast destination address defines only one recipient; the relationship between the sender and the receiver is one-to-one.
- A multicast destination address defines a group of addresses; the relationship between the sender and the receivers is one-to-many.

- The broadcast address is a special case of the multicast address; the recipients are all the stations on the LAN.

The broadcast destination address is a special case of the multicast address in which all bits are 1s.

Access Method: CSMA/CD

Standard Ethernet uses 1-persistent CSMA/CD. In an Ethernet network, the round-trip time required for a frame to travel from one end of a maximum-length network to the other plus the time needed to send the jam sequence is called the slot time.

Slot time = round-trip time + time required to send the jam sequence

The slot time in Ethernet is defined in bits. It is the time required for a station to send 512 bits. This means that the actual slot time depends on the data rate. The choice of a 512-bit slot time was chosen to allow the proper functioning of CSMA/CD.

3.2.2 Physical Layer

The Standard Ethernet defines several physical layer implementations; four of the most common, are shown in figure.

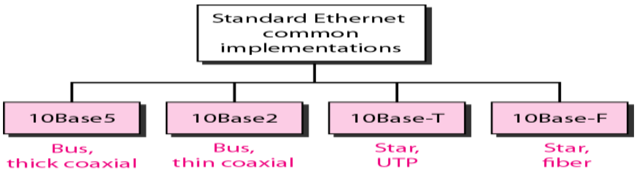


Figure 13.8 Categories of Standard Ethernet

10Base5: Thick Ethernet / Thicknet

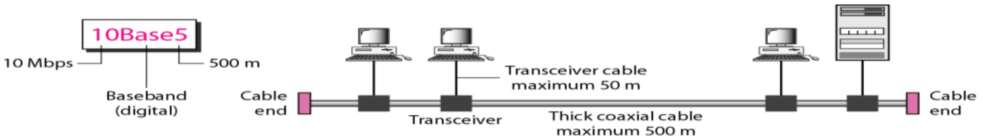


Figure 13.10 10Base5 implementation

10Base5 was the first Ethernet specification to use a bus topology with an external transceiver (transmitter/receiver) connected via a tap to a thick coaxial cable. The transceiver is responsible for transmitting, receiving, and detecting collisions.

The transceiver is connected to the station via a transceiver cable that provides separate paths for sending and receiving. This means that collision can only happen in the coaxial cable. The maximum length of the coaxial cable must not exceed 500 m, otherwise, there is excessive degradation of the signal. If a length of more than 500 m is needed, that can be connected using repeaters.

10Base2: Thin Ethernet / Cheapernet

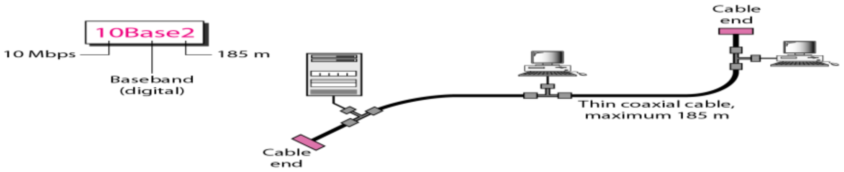


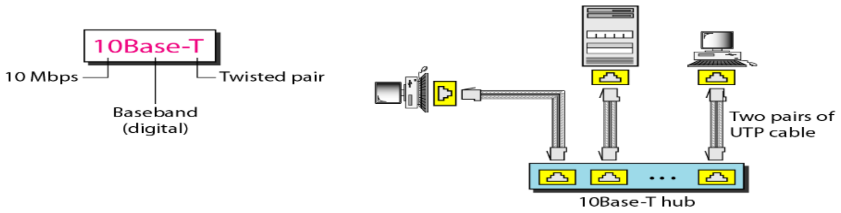
Figure 13.11 10Base2 implementation

10Base2 also uses a bus topology, but the cable is much thinner and more flexible. The cable can be bent to pass very close to the stations. In this case, the transceiver is normally part of the network interface card (NIC), which is installed inside the station.

The collision occurs in the thin coaxial cable. This implementation is more cost effective than 10Base5 because thin coaxial cable is less expensive than thick coaxial and the tee connections are much cheaper than taps. Installation is simpler because the thin coaxial cable is very flexible. However, the length of each segment cannot exceed 185 m (close to 200 m) due to the high level of attenuation in thin coaxial cable.

10Base- T: Twisted-Pair Ethernet

10Base-T uses a physical star topology. The stations are connected to a hub via two pairs of twisted cable



.

Figure 13.12 10Base-T implementation

Two pairs of twisted cable create two paths (one for sending and one for receiving) between the station and the hub. Any collision here happens in the hub. Compared to 10Base5 or 10Base2, we can see that the hub actually replaces the coaxial cable as far as a collision is concerned. The maximum length of the twisted cable here is defined as 100 m, to minimize the effect of attenuation in the twisted cable.

10Base-F: Fiber Ethernet

10Base-F uses a star topology to connect stations to a hub. The stations are connected to the hub using two fiber-optic cables.

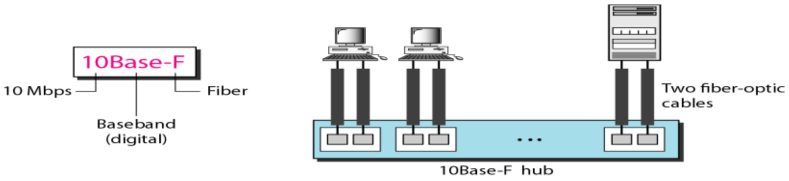


Figure 13.13 10Base-F implementation

3.3 FAST ETHERNET

Fast Ethernet was designed to compete with LAN protocols such as FDDI or Fiber Channel. IEEE created Fast Ethernet under the name 802.3u. Fast Ethernet is backward-compatible with Standard Ethernet, but it can transmit data 10 times faster at a rate of 100 Mbps. The goals of Fast Ethernet can be summarized as follows:

1. Upgrade the data rate to 100 Mbps.
2. Make it compatible with Standard Ethernet.
3. Keep the same 48-bit address.

4. Keep the same frame format.
5. Keep the same minimum and maximum frame lengths.

3.3.1 MAC Sublayer

A main consideration in the evolution of Ethernet from 10 to 100 Mbps was to keep the MAC sublayer untouched. A decision was made to drop the bus topologies and keep only the star topology. For the star topology, there are two choices:

- In the half-duplex approach, the stations are connected via a hub.
- In the full-duplex approach, the connection is made via a switch with buffers at each port.

The access method is the same (CSMA/CD) for the half-duplex approach but for full-duplex Fast Ethernet, there is no need for CSMA/CD. The implementations keep CSMA/CD for backward compatibility with Standard Ethernet.

Autonegotiation

A new feature added to Fast Ethernet is called autonegotiation. It allows a station or a hub a range of capabilities. Autonegotiation allows two devices to negotiate the mode or data rate of operation. It was designed particularly for the following purposes:

- To allow incompatible devices to connect to one another. For example, a device with a maximum capacity of 10 Mbps can communicate with a device with a 100 Mbps capacity.
- To allow one device to have multiple capabilities.
- To allow a station to check a hub's capabilities.

3.3.2 Physical Layer

The physical layer in Fast Ethernet is more complicated than Standard Ethernet. Some of the features of this layer are as follows.

Topology

Fast Ethernet is designed to connect two or more stations together. If there are only two stations, they can be connected point-to-point. Three or more stations need to be connected in a star topology with a hub or a switch at the center.

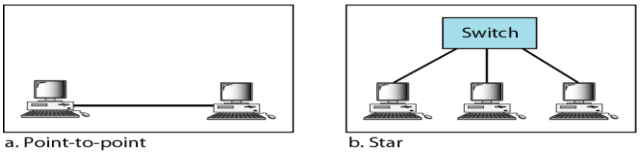


Figure 13.19 Fast Ethernet topology

Implementation

Fast Ethernet implementation at the physical layer can be categorized as either two-wire or four-wire. The two-wire implementation can be either category 5 UTP (100Base-TX) or fiber-optic cable (100Base-FX). The four-wire implementation is designed only for category 3 UTP (100Base-T4).

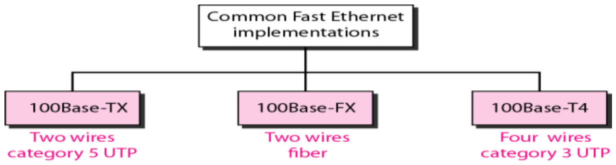


Figure 13.20 Fast Ethernet implementations

100Base-TX: uses <u>two pairs of twisted-pair cable</u> (either category 5 UTP or STP). For this implementation, the MLT-3 scheme was selected since it has good bandwidth performance. However, since MLT-3 is not a self-synchronous line coding scheme, 4B/5B block coding is used to provide bit synchronization by preventing the occurrence of a long sequence of 0s and ls.

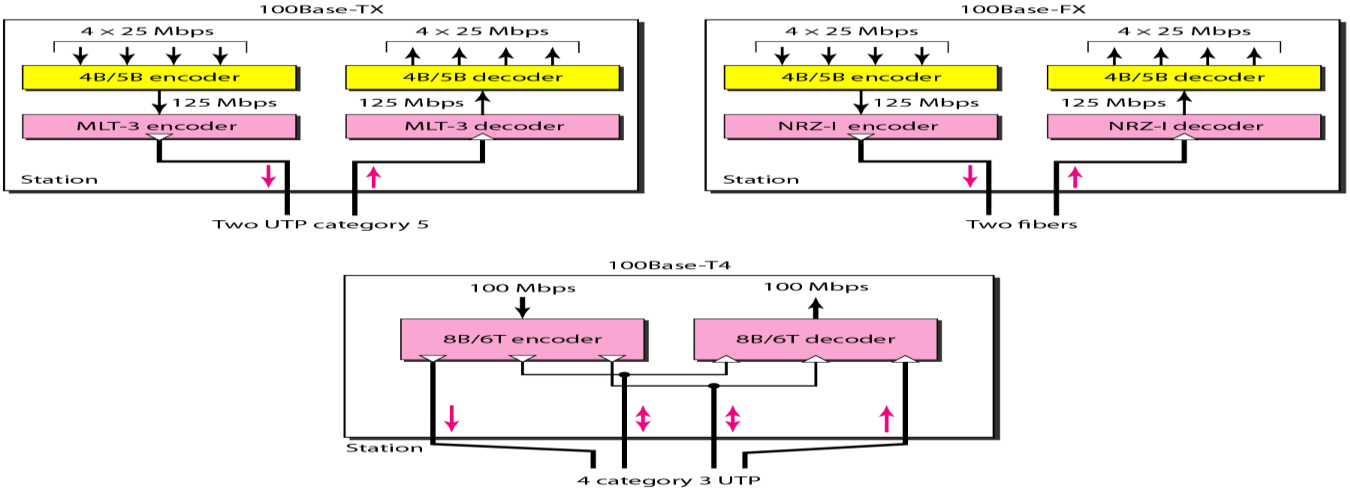
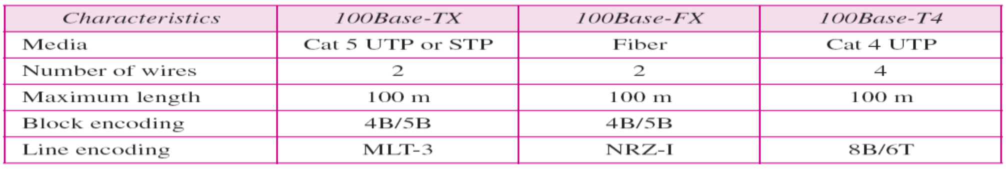


Figure 13.21 Encoding for Fast Ethernet implementation

100Base-FX: <u>uses two pairs of fiber-optic cables</u>. Optical fiber can easily handle high bandwidth requirements by using simple encoding schemes. The designers of 100Base-FX selected the NRZ-I encoding scheme for this implementation. However, NRZ-I has a bit synchronization problem for long sequences of 0s (or 1s, based on the encoding). To overcome this problem, the designers used 4B/5B block encoding as we described for 100Base-TX. The block encoding increases the bit rate from 100 to 125 Mbps, which can easily be handled by fiber-optic cable.

100Base-T4: <u>was designed to use category 3 or higher UTP</u>. The implementation uses four pairs of UTP for transmitting 100 Mbps. Encoding/decoding in 100Base-T4 is more complicated. As this implementation uses category 3 UTP, each twisted-pair cannot easily handle more than 25 Mbaud. In this design, one pair switches between sending and receiving. Three pairs of UTP category 3, however, can handle only 75 Mbaud (25 Mbaud) each. We need to use an encoding scheme that converts 100 Mbps to a 75 Mbaud signal. 8B/6T satisfies this requirement. In 8B/6T, eight data elements are encoded as six signal elements. This means that 100 Mbps uses only (6/8) x 100 Mbps, or 75 Mbaud.

Table 13.2 Summary of Fast Ethernet implementations



Characteristics	*100Base-TX*	*100Base-FX*	*100Base-T4*
Media	Cat 5 UTP or STP	Fiber	Cat 4 UTP
Number of wires	2	2	4
Maximum length	100 m	100 m	100 m
Block encoding	4B/5B	4B/5B	
Line encoding	MLT-3	NRZ-I	8B/6T

3.4 GIGABIT ETHERNET

The need for an even higher data rate resulted in the design of the Gigabit Ethernet protocol (1000 Mbps). The IEEE committee calls the Standard 802.3z. The goals of the Gigabit Ethernet design can be summarized as follows:

1. Upgrade the data rate to 1 Gbps.
2. Make it compatible with Standard or Fast Ethernet.
3. Use the same 48-bit address.
4. Use the same frame format.
5. Keep the same minimum and maximum frame lengths.
6. To support autonegotiation as defined in Fast Ethernet.

3.4.1 MAC Sublayer

Gigabit Ethernet has two distinctive approaches for medium access:

Full-Duplex Mode

In full- duplex mode, there is a central switch connected to all computers or other switches. In this mode, each switch has buffers for each input port in which data are stored until they are transmitted. There is no collision in this mode. This means that CSMA/CD is not used. Almost all implementations of Gigabit Ethernet follow the full-duplex approach.

In the full-duplex mode of Gigabit Ethernet, there is no collision; the maximum length of the cable is determined by the signal attenuation in the cable.

Half-Duplex Mode

In this case, a switch can be replaced by a hub, which acts as the common cable in which a collision might occur. The half-duplex approach uses CSMA/CD. The maximum length of the network in this approach is totally dependent on the minimum frame size. Three methods have been defined: traditional, carder extension, and frame bursting.

- **Traditional:** In the traditional approach, we keep the minimum length of the frame as in traditional Ethernet (512 bits). As it supports shorter slot time, so collision is detected 100 times earlier. This means that the maximum length of the network is 25 m. This length may be suitable if all the stations are in one room, but it may not even be long enough to connect the computers in one single office.
- **Carrier Extension:** To allow for a longer network, we increase the minimum frame length. The carrier extension approach defines the minimum length of a frame as 512 bytes (4096 bits). This means that the minimum length is 8 times longer. This method forces a station to add extension bits (padding) to any frame that is less than 4096 bits. In this way, the maximum length of the network can be increased 8 times to a length of 200 m.

- **Frame Bursting:** Carrier extension is very inefficient if we have a series of short frames to send; each frame carries redundant data. To improve efficiency, frame bursting was proposed. Instead of adding an extension to each frame, multiple frames are sent. However, to make these multiple frames look like one frame, padding is added between the frames so that the channel is not idle.

3.4.2 Physical Layer

The physical layer in Gigabit Ethernet is more complicated than that in Standard or Fast Ethernet. Some features of this layer are:

Topology

Gigabit Ethernet is designed to connect two or more stations. If there are only two stations, they can be connected point-to-point. Three or more stations need to be connected in a star topology with a hub or a switch at the center. Another possible configuration is to connect several star topologies or let a star topology be part of another as shown.

Implementation

Gigabit Ethernet can be categorized as either a two-wire or a four- wire implementation. The two-wire implementations use fiber-optic cable (1000Base-SX, short-wave, or 1000Base-LX, long-wave), or STP (1000Base-CX). The four-wire version uses category 5 twisted-pair cable (1000Base-T). In other words, we have four implementations, as shown in fig. 13.23.

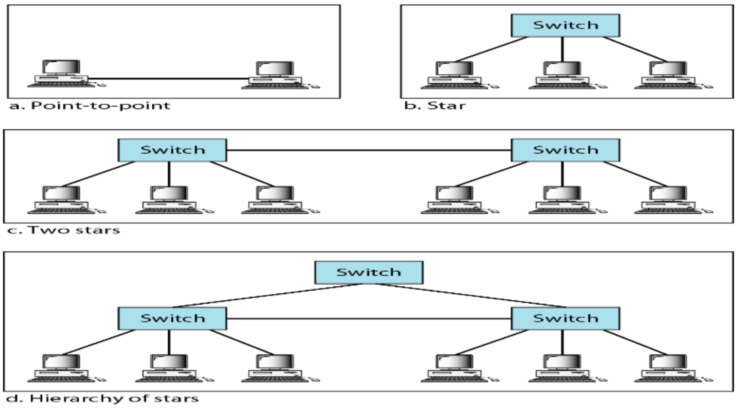


Figure 13.22 Topologies of Gigabit Ethernet

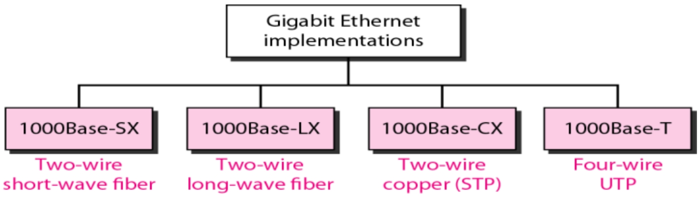


Figure 13.23 Gigabit Ethernet implementations

Encoding

The figure shows the encoding/decoding schemes for the four implementations.

Gigabit Ethernet cannot use the Manchester encoding scheme because it involves a very high bandwidth (2 GBaud). The two-wire implementations use an NRZ scheme, but NRZ does not self-synchronize properly. To synchronize bits, 8B/10B block encoding is used. This block encoding prevents long sequences of 0s or ls in the stream, but the resulting stream is 1.25 Gbps. in this implementation, one wire (fiber or STP) is used for sending and one for receiving.

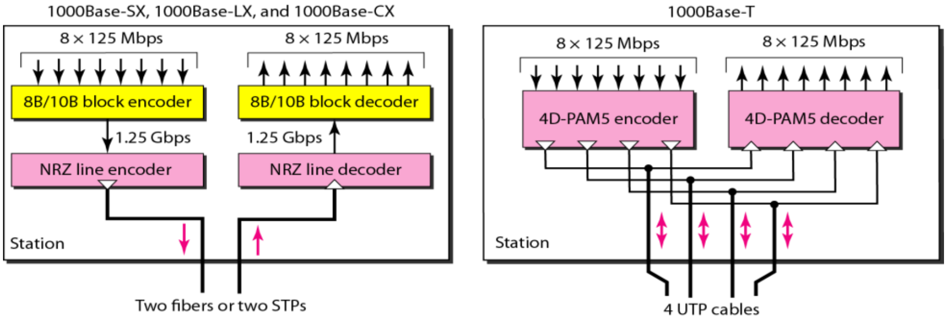
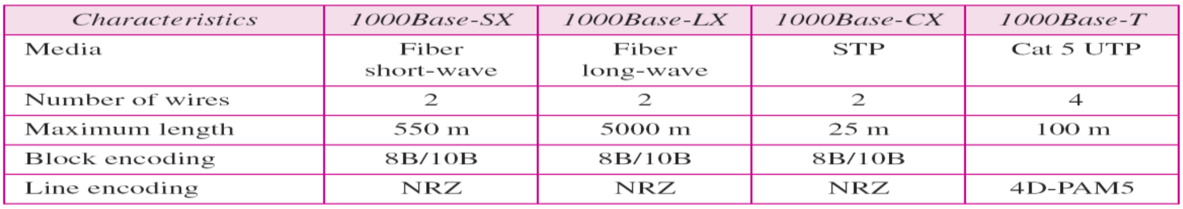


Figure 13.24 Encoding in Gigabit Ethernet implementations

In the four-wire implementation it is not possible to have 2 wires for input and 2 for output, because each wire would need to carry 500 Mbps, which exceeds the capacity for category 5 UTP. As a solution, 4D-PAM5 encoding is used to reduce the bandwidth. Thus, all four wires are involved in both input and output; each wire carries 250 Mbps, which is in the range for category 5 UTP cable.

Summary



Characteristics	*1000Base-SX*	*1000Base-LX*	*1000Base-CX*	*1000Base-T*
Media	Fiber short-wave	Fiber long-wave	STP	Cat 5 UTP
Number of wires	2	2	2	4
Maximum length	550 m	5000 m	25 m	100 m
Block encoding	8B/10B	8B/10B	8B/10B	
Line encoding	NRZ	NRZ	NRZ	4D-PAM5

3.5 Ten-Gigabit Ethernet

The IEEE committee created Ten-Gigabit Ethernet and called it Standard 802.3ae. The goals of the Ten-Gigabit Ethernet design can be summarized as follows:

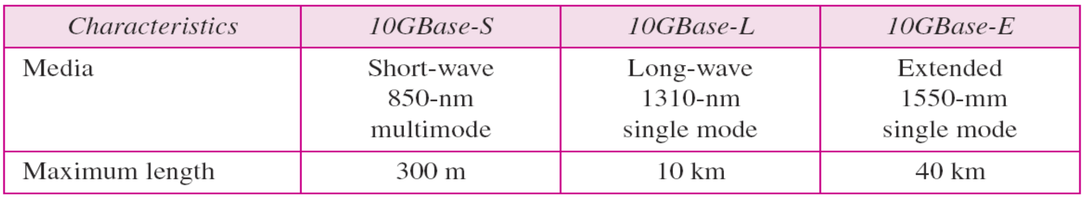
1. Upgrade the data rate to 10 Gbps.
2. Make it compatible with Standard, Fast, and Gigabit Ethernet.
3. Use the same 48-bit address.
4. Use the same frame format.
5. Keep the same minimum and maximum frame lengths.
6. Allow the interconnection of existing LANs into a MAN or a WAN.
7. Make Ethernet compatible with technologies such as Frame Relay and ATM.

MAC Sublayer

Ten-Gigabit Ethernet operates only in full duplex mode which means there is no need for contention; CSMA/CD is not used in Ten-Gigabit Ethernet.

Physical Layer

The physical layer in Ten-Gigabit Ethernet is designed for using fiber -optic cable over long distances. Three implementations are the most common: 10GBase-S, 10GBase-L, and 10GBase-E.



Characteristics	*10GBase-S*	*10GBase-L*	*10GBase-E*
Media	Short-wave 850-nm multimode	Long-wave 1310-nm single mode	Extended 1550-mm single mode
Maximum length	300 m	10 km	40 km

Table 13.4 Summary of Ten-Gigabit Ethernet implementations

B. Wireless LANs

3.6 IEEE 802.11: Architecture

The standard defines two kinds of services: the basic service set (BSS) and the extended service set (ESS).

Basic Service Set

IEEE 802.11 defines the basic service set (BSS) as the building block of a wireless LAN. A basic service set is made of stationary or mobile wireless stations and an optional central base station, known as the **access point (AP)**. The figure shows two sets in this standard. The BSS without an AP is a stand-alone network and cannot send data to other BSSs. It is called an ad hoc architecture. In this architecture, stations can form a network without the need of an AP; they can locate one another and agree to be part of a BSS. A BSS with an AP is sometimes referred to as an infrastructure network.

A BSS without an AP is called an ad hoc N/w; a BSS with an AP is called an infrastructure N/W.

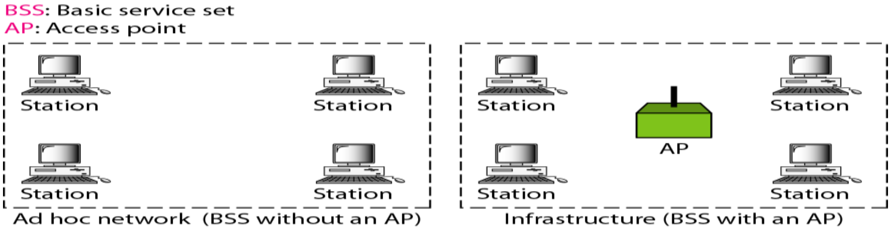


Figure 14.1 Basic service sets (BSSs)

Extended service set

An extended service set (ESS) is made up of two or more BSSs with APs. In this case, the BSSs are connected through a distribution system, which is usually a wired LAN. The distribution system connects the APs in the BSSs. IEEE 802.11 does not restrict the distribution system; it can be any IEEE LAN such as an Ethernet. Note that the extended service set uses two types of stations: mobile and stationary. The mobile stations are normal stations inside a BSS. The stationary stations are AP stations that are part of a wired LAN.

When BSSs are connected, the stations within reach of one another can communicate without the use of an AP. However, communication between two stations in two different BSSs usually occurs via two APs.

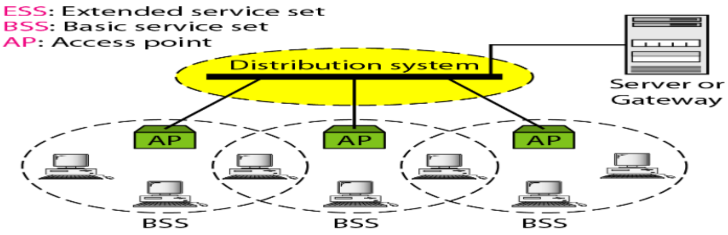


Figure 14.2 Extended service sets (ESSs)

Station Types

IEEE 802.11 defines three types of stations based on their mobility in a wireless LAN: **no-transition, BSS-transition, and ESS-transition mobility**.

- A station with no-transition mobility is either stationary (not moving) or moving only inside a BSS.
- A station with BSS-transition mobility can move from one BSS to another, but the movement is confined inside one ESS.
- A station with ESS-transition mobility can move from one ESS to another. However, IEEE 802.11 does not guarantee that communication is continuous during the move.

3.7 IEEE 802.11: MAC Sublayer

IEEE 802.11 defines two MAC sublayers: the distributed coordination function (DCF) and point coordination function (PCF). The figure shows the relationship between the two MAC sublayers, the LLC sublayer, and the physical layer.

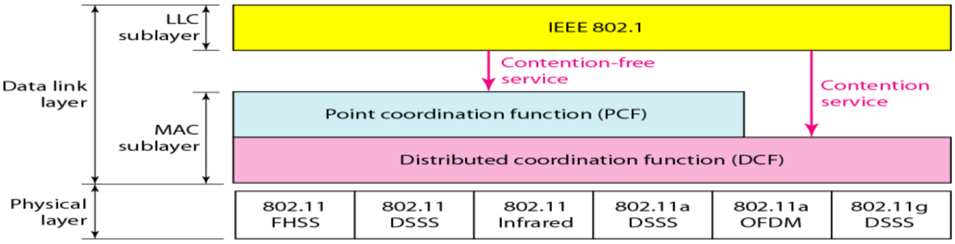


Figure 14.3 MAC layers in IEEE 802.11 standard

Distributed Coordination Function

DCF uses CSMA/CA as the access method. Wireless LANs cannot implement CSMA/CD for three reasons:

1. For collision detection a station must be able to send data and receive collision signals at the same time. This can mean costly stations and increased bandwidth requirements.
2. Collision may not be detected because of the hidden station problem.
3. The distance between stations can be great. Signal fading could prevent a station at one end from hearing a collision at the other end.

Process Flowchart: The figure shows the process flowchart for CSMA/CA as used in wireless LANs.

1. Before sending a frame, the source station senses the medium by checking the energy level at the carrier frequency.
   a. The channel uses a persistence strategy with back-off until the channel is idle.
   b. After the station is found to be idle, the station waits for a period of time called the distributed interframe space (DIFS); then the station sends a control frame called the request to send (**RTS**).

2. After receiving the RTS and waiting a period of time called the short interframe space (SIFS), the destination station sends a control frame, called the clear to send (**CTS**), to the source station. This control frame indicates that the destination station is ready to receive data.
3. The source station sends data after waiting an amount of time equal to SIFS.
4. The destination station, sends an acknowledgment to show that the frame has been received.

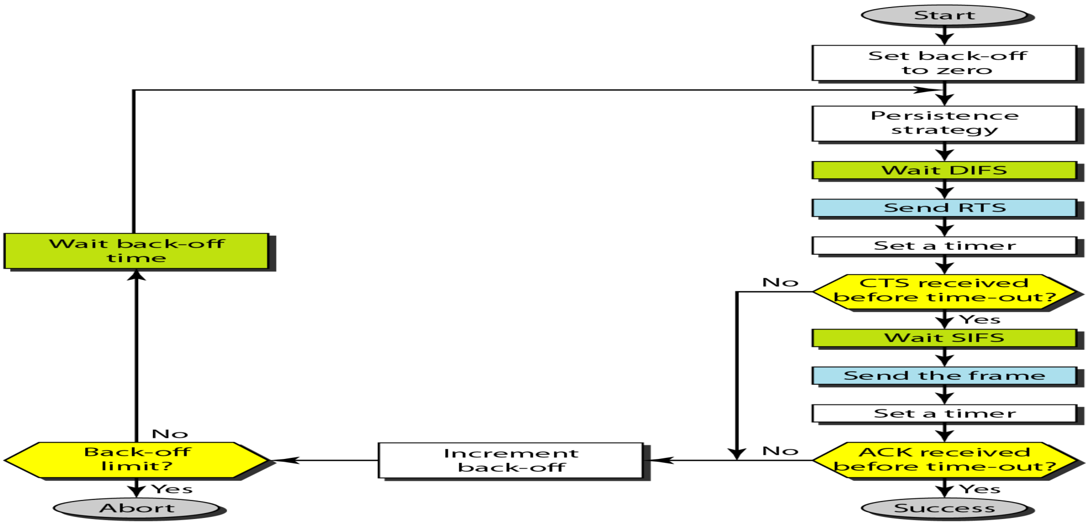


Figure 14.4 CSMA/CA flowchart

Frame Exchange Time Line: The figure shows the exchange of data and control frames in time.

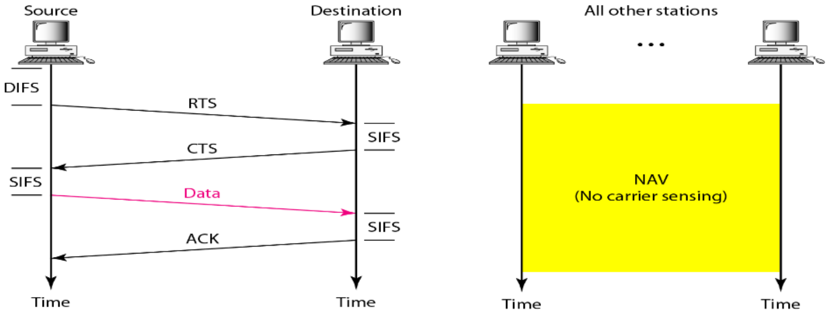


Figure 14.5 CSMA/CA and NAV

Network Allocation Vector: When a station sends an RTS frame, it includes the duration of time that it needs to occupy the channel. The stations that are affected by this transmission create a timer called a network allocation vector (**NAV**) that shows how much time must pass before these stations are allowed to check the channel for idleness.

Collision during Handshaking: What happens if there is collision during the time when RTS or CTS control frames are in transition, often called the handshaking period? Two or more stations may try to send RTS frames at the same time. These control frames may collide. However, because there is no mechanism for collision detection, the sender assumes there has been a collision if it has not received a CTS frame from the receiver. The back-off strategy is employed, and the sender tries again.

Point Coordination Function (PCF)

The point coordination function (PCF) is an optional access method that can be implemented in an infrastructure network (not in an ad hoc network). It is implemented on top of the DCF and is used mostly for time-sensitive transmission. PCF has a centralized, contention-free polling access method. The AP performs polling for stations that are capable of being polled. The stations are polled one after another, sending any data they have to the AP. To give priority to PCF over DCF, another set of interframe spaces has been defined: PIFS and SIFS. The SIFS is the same as that in DCF, but the PIFS (PCF IFS) is shorter than the DIFS.

Due to the priority of PCF over DCF, stations that only use DCF may not gain access to the medium. To prevent this, a repetition interval has been designed to cover both contention-free (PCF) and contention-based (DCF) traffic. The repetition interval, which is **repeated continuously**, starts with a special control frame, called a **beacon frame**. When the stations hear the beacon frame, they start their NAV for the duration of the contention-free period of the repetition interval. The figure shows an example of a repetition interval.

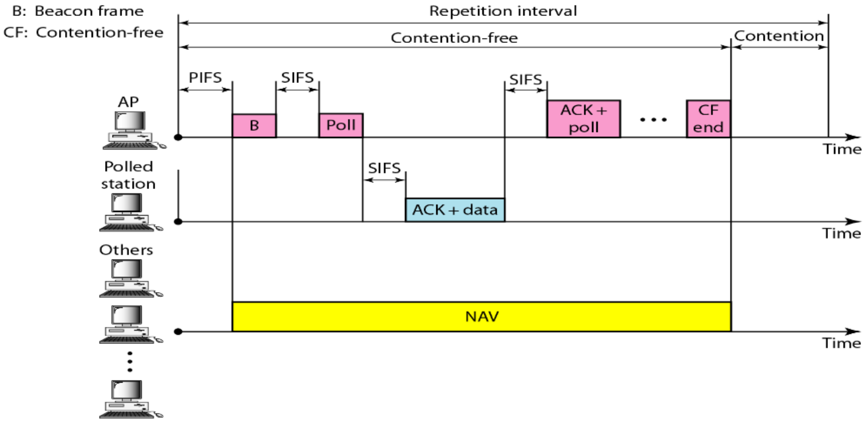


Figure 14.6 Example of repetition interval

During the repetition interval, the PC (point controller) can send a poll frame, receive data, send an ACK, receive an ACK, or do any combination of these. At the end of the contention-free period, the PC sends a CF end (contention-free end) frame to allow the contention-based stations to use the medium.

Fragmentation

The wireless environment is very noisy; a corrupt frame has to be retransmitted. The protocol, therefore, recommends fragmentation--the division of a large frame into smaller ones. It is more efficient to resend a small frame than a large one.

Frame Format

The MAC layer frame consists of nine fields, as shown.

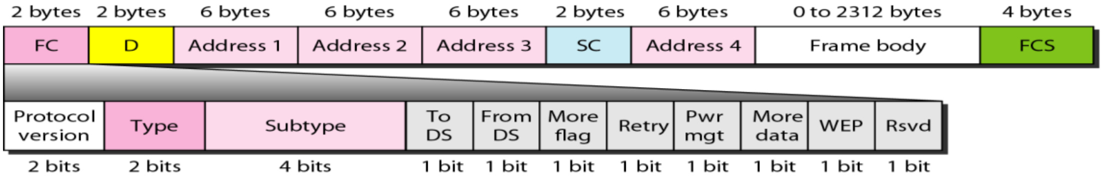
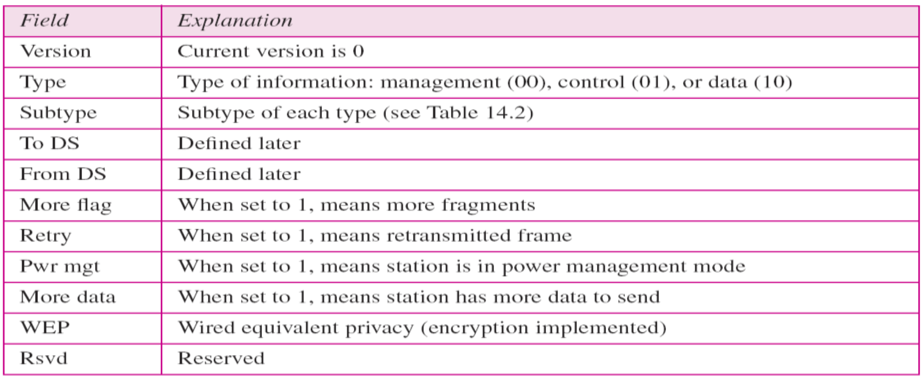


Figure 14.7 Frame format

1. **Frame control (FC):** The FC field is 2 bytes long and defines the type of frame and some control information. The table describes the subfields.



Field	*Explanation*
Version	Current version is 0
Type	Type of information: management (00), control (01), or data (10)
Subtype	Subtype of each type (see Table 14.2)
To DS	Defined later
From DS	Defined later
More flag	When set to 1, means more fragments
Retry	When set to 1, means retransmitted frame
Pwr mgt	When set to 1, means station is in power management mode
More data	When set to 1, means station has more data to send
WEP	Wired equivalent privacy (encryption implemented)
Rsvd	Reserved

2. **D**: this field defines the duration of the transmission that is used to set the value of NAV.
3. **Addresses**: There are four address fields, each 6 bytes long. The meaning of each address field depends on the value of the To DS and From DS subfields.
4. **Sequence control**: This field defines the sequence number of the frame to be used in flow control.
5. **Frame body**: This field, which can be between 0 and 2312 bytes, contains information based on the type and the subtype defined in the FC field.
6. **FCS**. The FCS field is 4 bytes long and contains a CRC-32 error detection sequence.

Frame Types

A wireless LAN defined by IEEE 802.11 has three categories of frames: management frames, control frames, and data frames.

1. **Management Frames:** It is used for the initial communication between stations and access points.
2. **Control Frames:** It is used for accessing the channel and acknowledging frames. The figure shows the format.

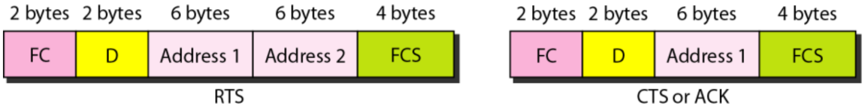
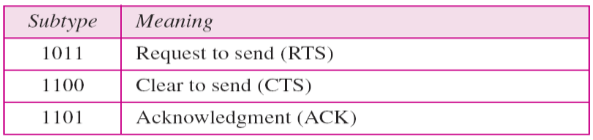


Figure 14.8 Control frames

For control frames the value of the type field is 01; the values of the subtype fields for frames are shown in the table.



Subtype	*Meaning*
1011	Request to send (RTS)
1100	Clear to send (CTS)
1101	Acknowledgment (ACK)

3. **Data Frames:** It is used for carrying data and control information.

3.8 IEEE 802.11: Addressing Mechanism

The IEEE 802.11 addressing mechanism specifies four cases, defined by the value of the two flags in the FC field, To DS and From DS. Each flag can be either 0 or 1, resulting in four different situations.

The interpretation of the four addresses (address 1 to address 4) in the MAC frame depends on the value of these flags, as shown in the table.

To DS	*From DS*	*Address 1*	*Address 2*	*Address 3*	*Address 4*
0	0	Destination	Source	BSS ID	N/A
0	1	Destination	Sending AP	Source	N/A
1	0	Receiving AP	Source	Destination	N/A
1	1	Receiving AP	Sending AP	Destination	Source

Case 1:00 in this case, To DS = 0 and From DS = 0. This means that the frame is not going to a distribution system (To DS = 0) and is not coming from a distribution system (From DS = 0). The frame is going from one station in a BSS to another without passing through the distribution system. The ACK frame should be sent to the original sender. The addresses are shown in figure.

Case 2:01 In this case, To DS = 0 and From DS = 1. This means that the frame is coming from a distribution system (From DS = 1). The frame is coming from an AP and going to a station. The ACK should be sent to the AP. The addresses are as shown in Figure 14.9. Note that address 3 contains the original sender of the frame (in another BSS).

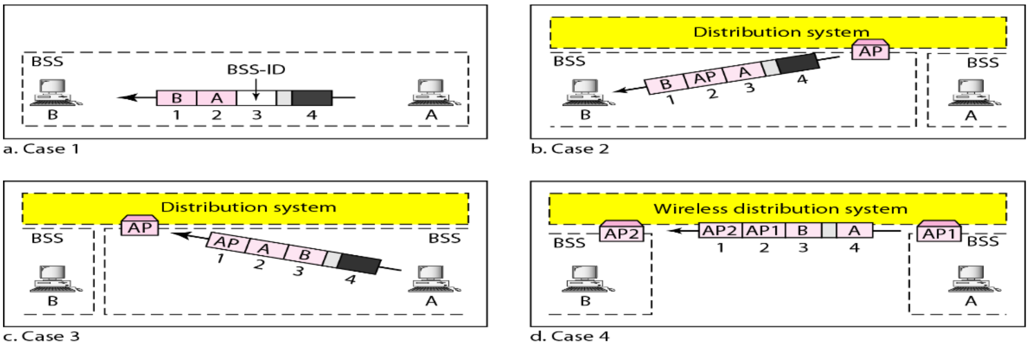


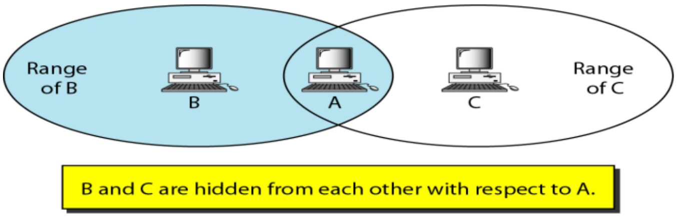
Figure 14.9 Addressing mechanisms

Case 3:10 In this case, To DS = 1 and From DS = 0. This means that the frame is going to a distribution system (To DS = 1). The frame is going from a station to an AP. The ACK is sent to the original station. The addresses are as shown in figure above. Note that address 3 contains the final destination of the frame.

Case 4:11 In this case, To DS = 1 and From DS = 1. This is the case in which the distribution system is also wireless. The frame is going from one AP to another AP in a wire- less distribution system. We do not need to define addresses if the distribution system is a wired LAN. Here, we need four addresses to define the original sender, the final destination, and two intermediate APs. Figure above shows the situation.

Hidden and Exposed Station Problems

Hidden Station Problem The figure below shows an example of the hidden station problem. Station B has a transmission range shown by the left oval (sphere in space); every station in this range can hear any signal transmitted by station B. Station C has a transmission range shown by the right oval (sphere in space); every station located in this range can hear any signal transmitted by C. Station C is outside the transmission range of B; likewise, station B is outside the transmission range of C. Station A, is in the area covered by both B and C; it can hear any signal transmitted by B or C.



B and C are hidden from each other with respect to A.

Figure 14.10 Hidden station problem

Assume that station B is sending data to station A. In the middle of this transmission, station C also has data to send to station A. However, station C is out of B's range and transmissions from B cannot reach C. Therefore C thinks the medium is free. Station C sends its data to A, which results in a collision at A, because this station is receiving data from both B and C. In this case, we say that stations B and C are hidden from each other with respect to A. Hidden stations can reduce the capacity of the network because of the possibility of collision.

Solution: use of the handshake frames (RTS and CTS). the figure shows that the RTS message from B reaches A, but not C. However, because both B and C are within the range of A, the CTS message, which contains the duration of data transmission from B to A reaches C. Station C knows that some hidden station is using the channel and refrains from transmitting until that duration is over.

The CTS frame in CSMA/CA handshake can prevent collision from a hidden station.

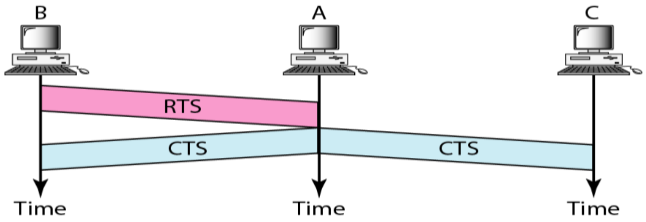
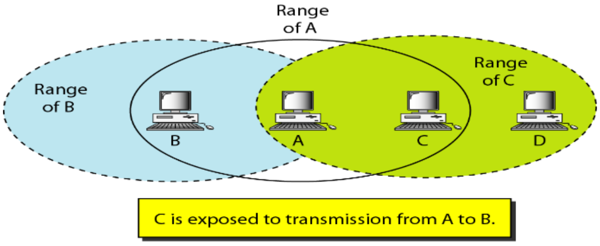


Figure 14.11 Use of handshaking to prevent hidden station problem

Exposed Station Problem Now consider a situation that is the inverse of the previous one: the exposed station problem. In this problem a station refrains from using a channel when it is, in fact, available. In the figure, station A is transmitting to station B. Station C has some data to send to station D, which can be sent without interfering with the transmission from A to B. However, station C is exposed to transmission from A; it hears what A is sending and thus suspend from sending. In other words, C is too conservative and wastes the capacity of the channel.



C is exposed to transmission from A to B.

Figure 14.12 Exposed station problem

The handshaking messages RTS and CTS cannot help in this case. Station C hears the RTS from A, but does not hear the CTS from B. Station C, after hearing the RTS from A, can wait for a time so that the CTS from B reaches A; it then sends an RTS to D to show that it needs to communicate with D. Both stations B and A may hear this RTS, but station A is in the sending state, not the receiving state. Station B, however, responds with a CTS. The problem is here. If station A has started sending its data, station C cannot hear the CTS from station D because of the collision; it cannot send its data to D. It remains exposed until A finishes sending its data as the figure shows.

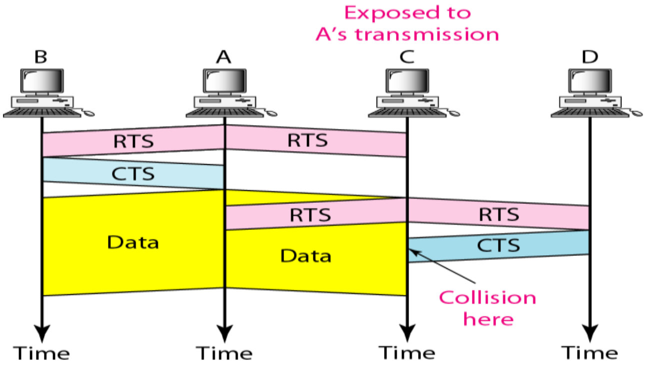


Figure 14.13 Use of handshaking in exposed station problem

3.9 IPv4 ADDRESSES

An **IPv4** address is a 32-bit address that **uniquely** and **universally** defines the connection of a device. They are unique in the sense that each address defines one, and only one, connection to the Internet. Two devices on the Internet can never have the same address at the same time. If a device operating at the network layer has m connections to the Internet, it needs to have m addresses.

3.9.1 Address Space

An address space is the total number of addresses used by the protocol. If a protocol uses N bits to define an address, the address space is 2^N because each bit can have two different values (0 or 1) and N bits can have 2^N values.

IPv4 uses 32-bit addresses, which means that the address space is 2^32 or 4,294,967,296 (more than 4 billion). This means that, theoretically, more than 4 billion devices could be connected to the Internet. But practically it is not true.

3.9.2 Notations

There are two prevalent notations to show an IPv4 address: binary notation and dotted decimal notation.

- **Binary Notation:** In binary notation, the IPv4 address is displayed as 32 bits. Each octet is often referred to as a byte. Example: 01110101 10010101 00011101 00000010
- **Dotted-Decimal Notation:** To make the IPv4 address more compact and easier to read, Internet addresses are usually written in decimal form with a decimal point (dot) separating the bytes.

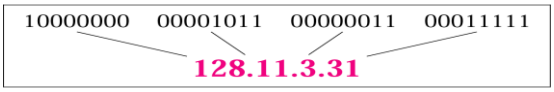


Figure 19.1 Dotted-decimal notation and binary notation for an IPv4 address

Classful Addressing

IPv4 addressing, uses the concept of classes. This architecture is called classful addressing. In classful addressing, the address space is divided into five classes: A, B, C, D, and E. Each class occupies some part of the address space. We can find the class of an address when given the address in binary or dotted-decimal notation.

- In binary notation, the first few bits can immediately tell us the class of the address.
- In decimal-dotted notation, the first byte defines the class. Both methods are shown in Fig19.2.

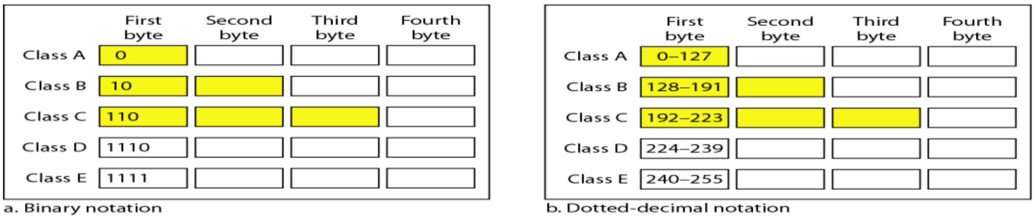


Fig19.2 **finding the classes in binary and dotted-decimal notation (127.0.0.1: loop back address)**

One problem with classful addressing is that each class is divided into a fixed number of blocks with each block having a fixed size as shown in Dotted-**decimal notation.**

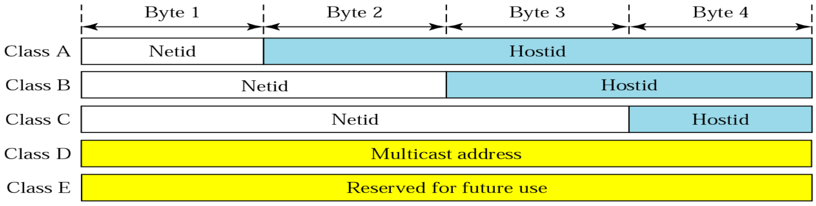


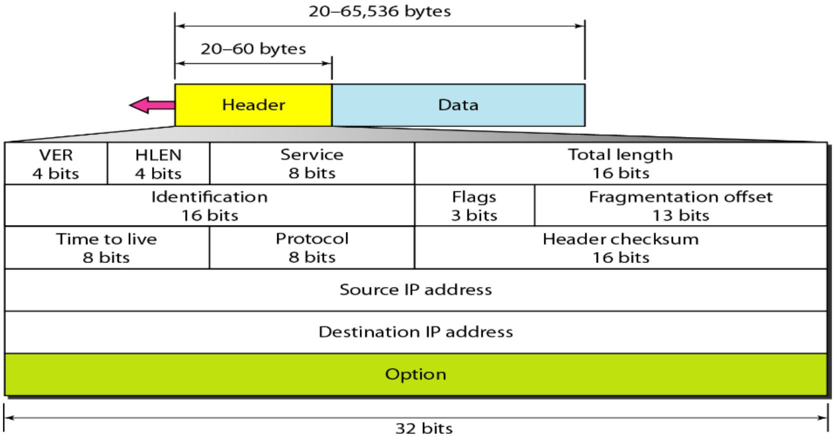
Figure 19.3 Netid and hostid

In classful addressing, an IP address in class A, B, or C is divided into netid and hostid. These parts are of varying lengths, depending on the class of the address. Figure 19.3 shows some netid and hostid bytes. The concept does not apply to classes D and E. In class A, one byte defines the netid, 3 bytes define the hostid. In class B, two bytes define the netid and two bytes define the hostid. In class C, three bytes define the netid and one byte defines the hostid.

Class A addresses were designed for large organizations with a large number of attached hosts or routers. Class B addresses were designed for midsize organizations with tens of thousands of attached hosts or routers. Class C addresses were designed for small organizations with a small number of attached hosts or routers.

A block in class A address is too large for almost any organization. This means most of the addresses in class A were wasted and were not used. A block in class B is also very large, probably too large for many of the organizations that received a class B block. A block in class C is probably too small for many organizations. Class D addresses were designed for multicasting. the class E addresses were reserved for future use.

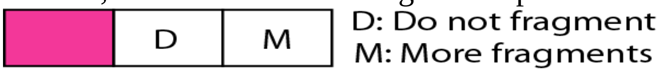
IPv4 Datagram Format



IPv4 is an unreliable and connectionless datagram protocol – a best effort delivery, Best effort means that IPv4 does its best to get a transmission through to its destination, but with no guarantees.

In IPv4 each datagram is handled independently, and each datagram can follow a different route to the destination. This implies that datagrams sent by the same source to the same destination could arrive out of order. Also, some could be lost or corrupted during transmission. Again, IPv4 relies on a higher-level protocol to take care of all these problems.

- **Version (VER)**: version of the IP protocol. Currently, the version is 4.
- **Header length (HLEN):** the total length of the datagram header in 4-byte words. Length is between 20 to 60 bytes.
- **Services**: service type or differentiated services.
- **Total length**: total length (header plus data) of the datagram in bytes. Total length of IP datagram is 65535 bytes.
- **Identification**: This 16-bit field identifies a datagram originating from the source host. The combination of the identification and source IPv4 address must uniquely define a datagram as it leaves the source host. used in fragmentation.
- **Flags**: used in fragmentation. This is a 3-bit field.
  - first bit: reserved (not used)
  - Second bit (do *not fragment)*: = 1 then, don't fragment the datagram. drops the packet if it is > MTU.
    =0 then, the datagram can be fragmented.
  - Third bit (*more fragment)*: =1, then there are more fragments after this one.
    =0, then this is the last fragmented packet



- **Fragmentation offset**: shows the relative position of this fragment with respect to the whole datagram. It is the offset of the data in the original datagram measured in units of 8 bytes. Used in fragmentation.
- **Time to live**: it is used to control the maximum number hops visited by the datagram. This field limits the lifetime of a datagram. This field was originally designed to hold a timestamp, which was decremented by each visited router. The datagram was discarded when the value became zero.
- **Protocol**: defines the higher-level protocol that uses the services of the IPV4 layer. An IPv4 datagram can encapsulate data from several higher-level protocols such as TCP, UDP, ICMP, and IGMP.
- **Checksum**: 1's compliment checksum.
- **Source address**: This 32-bit field defines the IPv4 address of the source.
- **Destination address**: This 32-bit field defines the IPv4 address of the destination.

Network Address Translation (NAT)

- How To save IP addresses;
    - For home users – who are connected to the Internet by dial up, dynamic addresses can be assigned to them for the connection period.
    - For business customers and many home users (ADSL), they want to stay connected continuously ➔ each user must have its own IP address ➔ total number of IP number an ISP can provide will not be enough to cover all customers (for example, class B block can support 65536 only).
- Solution is using NAT enabled router.
- NAT: enables a company to have large set of unique addresses internally (private addresses) and one address or a small set of addresses externally (public).

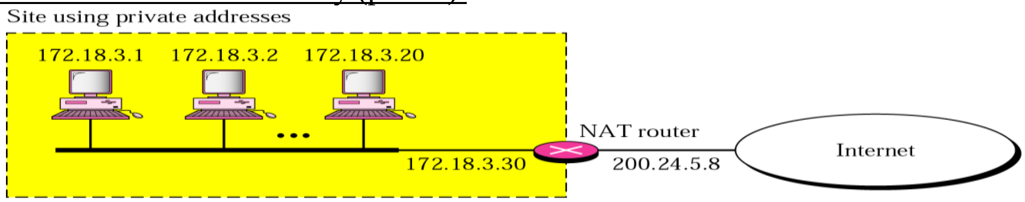


Figure 19.10 *A NAT implementation*

As Figure 19.10 shows, the private network uses private addresses. The router that connects the network to the global address uses one private address and one global address. The private network is transparent to the rest of the Internet; the rest of the Internet sees only the NAT router with the address 200.24.5.8.

Address Translation

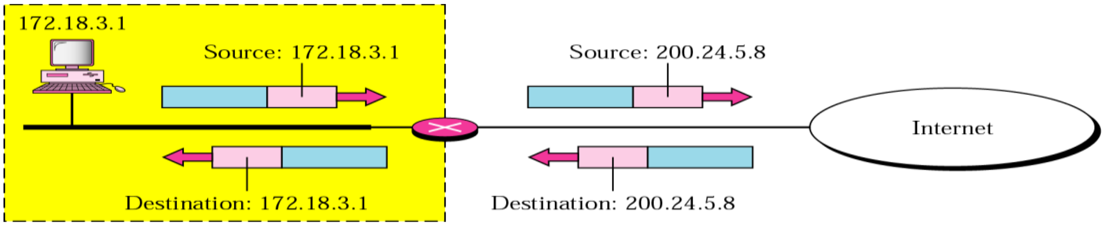


Figure 19.11 *Addresses in a NAT*

All the outgoing packets go through the NAT router, which replaces the *source address* in the packet with the global NAT address. All incoming packets also pass through the NAT router, which replaces the *destination address* in the packet (the NAT router global address) with the appropriate private address. Figure 19.11 shows an example of address translation.

Translation Table

how does the NAT router know the destination address for a packet coming from the Internet? There may be tens or hundreds of private IP addresses, each belonging to one specific host. The problem is solved if the NAT router has a translation table.

1.Using One IP Address In its simplest form, a translation table has only two columns: the private' address and the external address (destination address of the packet). When the router translates the source address of the outgoing packet, it also makes note of the destination address-where the packet is going. When the response comes back from the destination, the router uses the source address of the packet (as the external address) to find the private address of the packet. Figure 19.12 shows the idea.

2.Using a Pool of IP Addresses Since the NAT router has only one global address, only one private network host can access the same external host. To remove this restriction, the NAT router uses a pool of global addresses. For example, instead of using only one global address (200.24.5.8), the NAT router can use four addresses (200.24.5.8, 200.24.5.9, 200.24.5.10, and 200.24.5.11). In this case, four private network hosts can communicate with the same external host at the same time because each pair of addresses defines a connection.

3.Using Both IP Addresses and Port Numbers

- It make use of sender ip address and port no, NAT port and destination IP.
- The NAT router saves the **computer's private IP address** (source IP) and **port number** (source port) to the address translation table stored in the router. For transmitting data the router replaces the **sender IP address** with global IP address.
- The router replaces the sending computer's **source port** with a port number equal to the translation **table row number** where the router saved the sending computer's address information.
- When a packet arrives at the NAT router from the ISP router, the NAT router reads the **destination port number (NAT given port)** on the arriving packet and then uses it in the address translation table to extract the original computer **private IP address** and original **source port number.**
- **The destination port and the destination IP are replaced by the original values retrieved from the table.** The packet is then sent to the destination computer.

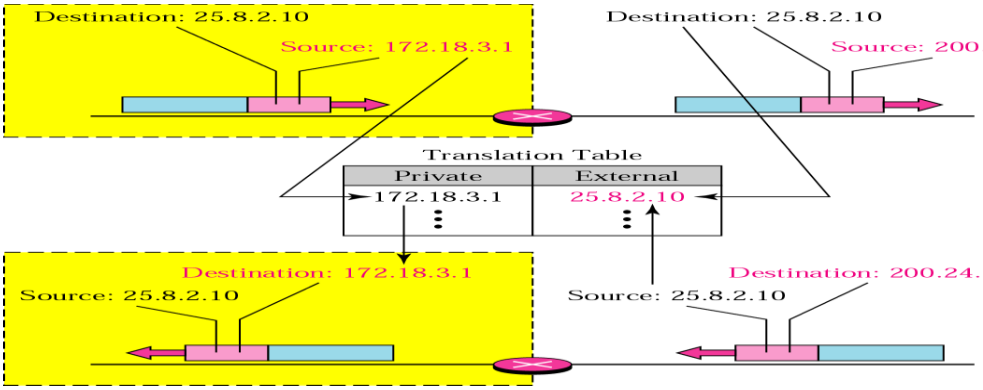


Figure 19.12 *NAT address translation*

Advantages of using NAT:

- No need to be allocated range of global addresses from ISP: just one global IP address is used for all devices ➔ **save IP address.**
- Can change addresses of devices in local network without notifying outside world.
- A change ISP without changing addresses of devices in local network.
- Can be used as firewall. A computer on an external network **cannot connect to your computer** unless your computer has initiated the contact. You can browse the Internet and connect to a site, and even download a file; but somebody else cannot use your IP address to connect to a port on your computer.

IPv6

IPv4 has some deficiencies that make it unsuitable for the fast-growing Internet.

1. Despite all short-term solutions, such as subnetting, classless addressing, and NAT, address depletion is still a long-term problem in the Internet.
2. The Internet must accommodate real-time audio and video transmission. This type of transmission requires minimum delay strategies and reservation of resources not provided in the IPv4 design.
3. The Internet must accommodate encryption and authentication of data for some applications. No encryption or authentication is provided by IPv4.

To overcome these deficiencies, IPv6 (Internetworking Protocol, version 6), also known as IPng (Internetworking Protocol, next generation) are used.

IPv6: Advantages:

1. **Larger address space**: An IPv6 address is 128 bits long, which is larger than 32-bit address of IPv4.
2. **Better header format**: IPv6 uses a new header format in which options are separated from the base header and inserted, when needed, between the base header and the upper-layer data. This simplifies and speeds up the routing process because most of the options do not need to be checked by routers.
3. **New options**: IPv6 has new options to allow for additional functionalities.
4. **Allowance for extension**. IPv6 is designed to allow the extension of the protocol if required by new technologies or applications.
5. **Support for resource allocation**. In IPv6, the type-of-service field has been removed, but a mechanism (called low *label)* has been added to enable the source to request special handling of the packet. This mechanism can be used to support traffic such as real-time audio and video.
6. **Support for more security**. The encryption and authentication options in IPv6 provide confidentiality and integrity of the packet.

IPv6 datagram format:

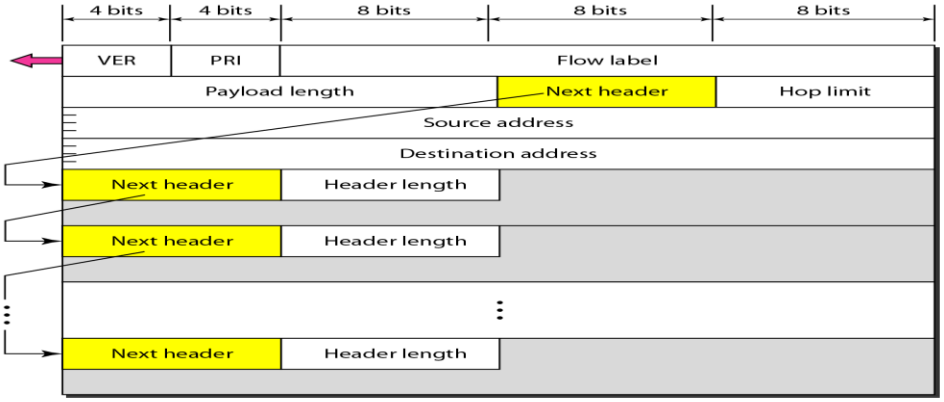
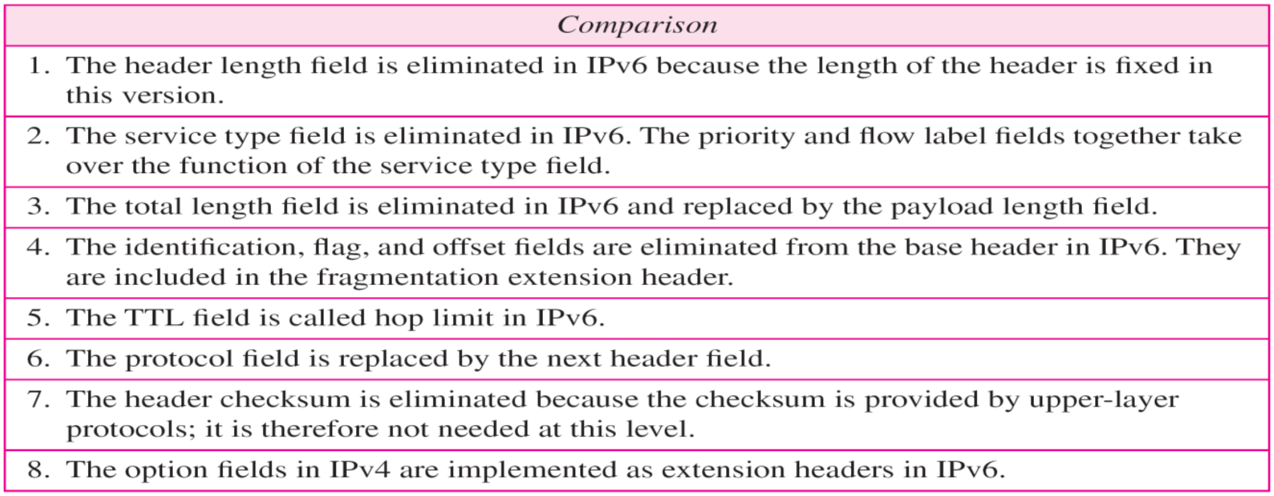


Figure 20.16 ***Format of an IPv6 datagram***

1. **Version**. This 4-bit field defines the version number of the IP. For IPv6, the value is 6.
2. **Priority.** The 4-bit priority field defines the priority of the packet with respect to traffic congestion.
3. **Flow label**. The flow label is a 3-byte (24-bit) field that is designed to provide special handling for a particular flow of data.
4. **Payload length**. The 2-byte payload length field defines the length of the IP datagram excluding the base header.
5. **Next header**. The next header is an 8-bit field defining the header that follows the base header in the datagram. The next header is either one of the optional extension headers used by IP or the header of an encapsulated packet such as UDP or TCP. Each extension header also contains this field. this field in version 4 is called the *protocol.*
6. **Hop limit**. This 8-bit hop limit field serves the same purpose as the TIL field in IPv4.
7. **Source address**. The source address field is a 16-byte (128-bit) Internet address that identifies the original source of the datagram.
8. **Destination address.** The destination address field is a 16-byte (128-bit) Internet address that usually identifies the final destination of the datagram.

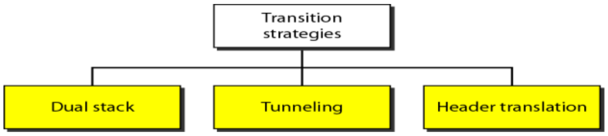
Table 20.9 ***Comparison between IPv4 and IPv6 packet headers***



Comparison
1. The header length field is eliminated in IPv6 because the length of the header is fixed in this version.
2. The service type field is eliminated in IPv6. The priority and flow label fields together take over the function of the service type field.
3. The total length field is eliminated in IPv6 and replaced by the payload length field.
4. The identification, flag, and offset fields are eliminated from the base header in IPv6. They are included in the fragmentation extension header.
5. The TTL field is called hop limit in IPv6.
6. The protocol field is replaced by the next header field.
7. The header checksum is eliminated because the checksum is provided by upper-layer protocols; it is therefore not needed at this level.
8. The option fields in IPv4 are implemented as extension headers in IPv6.

TRANSITION FROM IPv4 TO IPv6

Because of the huge number of systems on the Internet, the transition from IPv4 to IPv6 cannot happen suddenly. It takes a considerable amount of time before every system in the Internet can move from IPv4 to IPv6. The transition must be smooth to prevent any problems between IPv4 and IPv6 systems. Three strategies are:



1. ***Dual stack***

It is recommended that all hosts, before migrating completely to version 6, have a **dual** stack of protocols. In other words, a station must run IPv4 and IPv6 simultaneously until all the Internet uses IPv6. See Figure 20.19 for the layout of a dual-stack configuration.

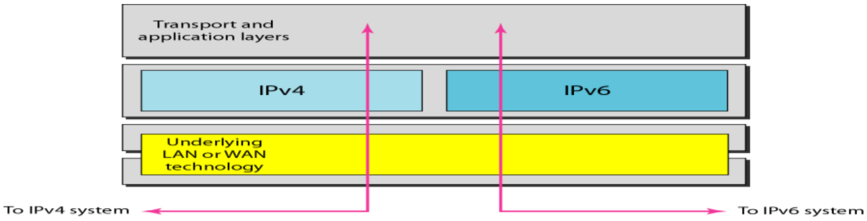

Transport and application layers
IPv4
IPv6
Underlying LAN or WAN technology
To IPv4 system
To IPv6 system

Figure 20.19 *Dual stack*

To determine which version to use when sending a packet to a destination, the source host queries the DNS. If the DNS returns an IPv4 address, the source host sends an IPv4 packet. If the DNS returns an IPv6 address, the source host sends an IPv6 packet.

Tunneling

Tunneling is a strategy used when two computers using IPv6 want to communicate with each other and the packet must pass through a region that uses IPv4. To pass through this region, the packet must have an IPv4 address. So the IPv6 packet is encapsulated in an IPv4 packet when it enters the region, and it leaves its capsule when it exits the region. It seems as if the IPv6 packet goes through a tunnel at one end and emerges at the other end. To make it clear that the IPv4 packet is carrying an IPv6 packet as data, the protocol value is set to 41. Tunneling is shown in Figure 20.20.

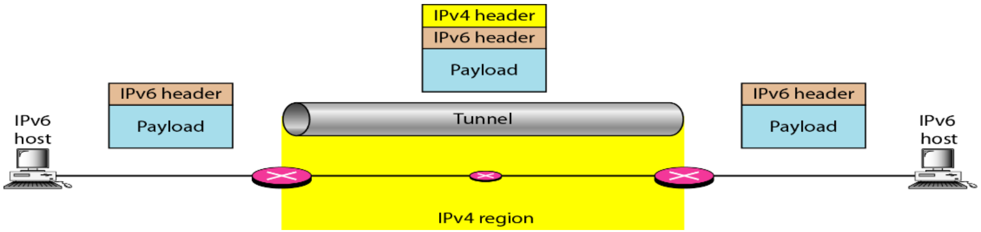


Figure 20.20 ***Tunneling strategy***

Header translation

Header translation is necessary when the majority of the Internet has moved to IPv6 but some systems still use IPv4. The sender wants to use IPv6, but the receiver does not understand IPv6. Tunneling does not work in this situation because the packet must be in the IPv4 format to be understood by the receiver. In this case, the header format must be totally changed through header translation. The header of the IPv6 packet is converted to an IPv4 header (see Figure 20.21). Header translation uses the mapped address to translate an IPv6 address to an IPv4 address.

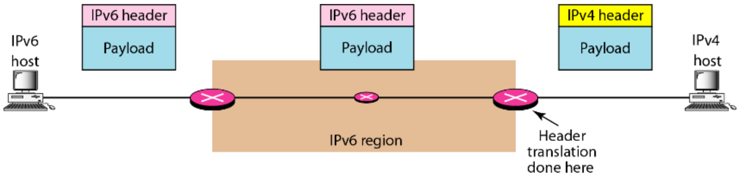


Figure 20.21 ***Header translation strategy***