Unit 3 Circuit Switching and Packet Switching

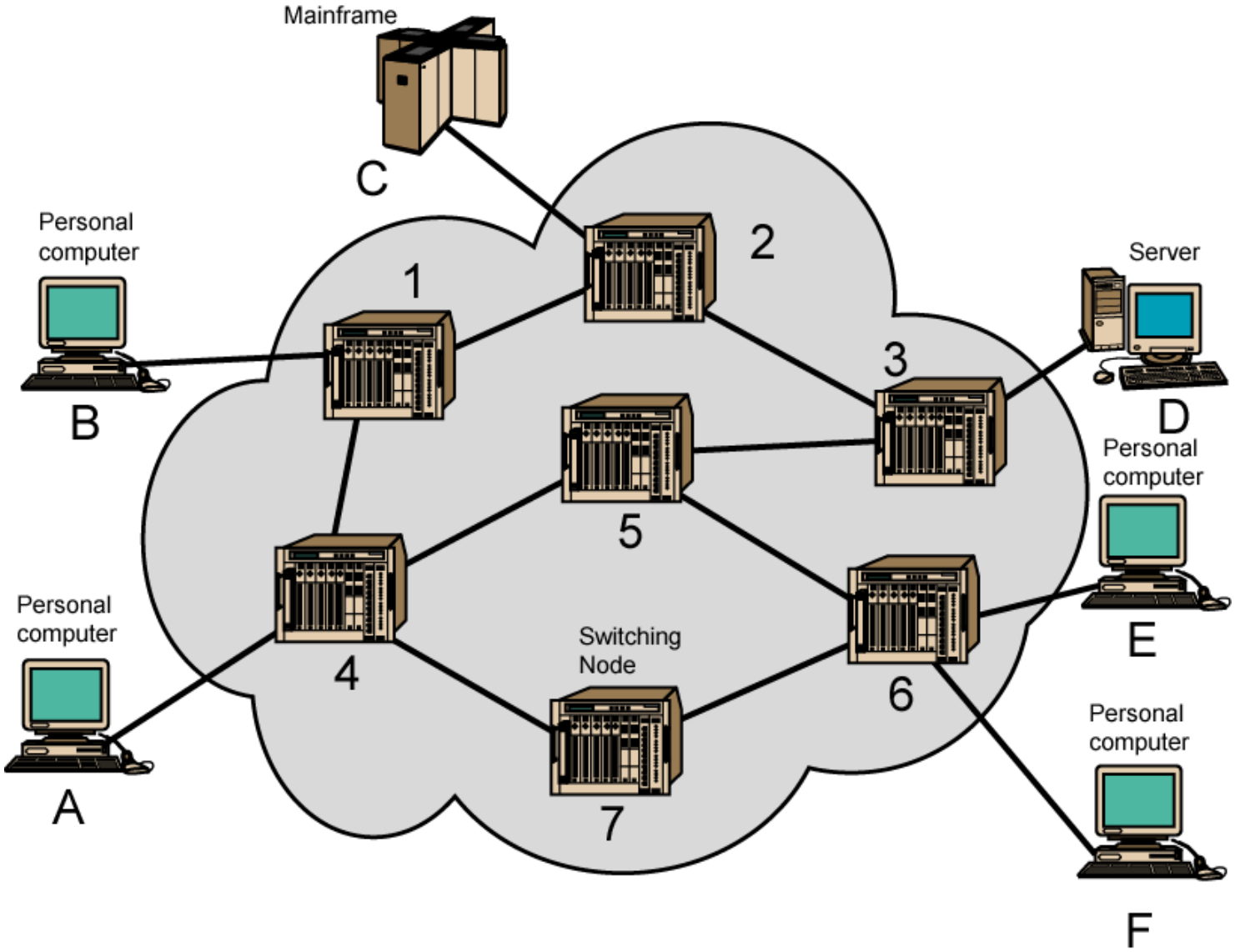
Overview

- Networks are used to **interconnect many devices.**
  - Since the invention of the telephone, **circuit switching** has been the dominant technology for voice communications.
  - Since 1970, **packet switching** has evolved substantially for digital data communications.
  - It was designed to provide a more efficient facility than circuit switching for bursty data(**data burst** is the broadcast of a relatively high-bandwidth transmission over a short period.) traffic.
    - Two types of packet switching:
      - Datagram (such as today's Internet)
      - Virtual circuit (such as Frame Relay, ATM)

Switched Communications Networks

- Long distance transmission between stations (called "end devices") is typically done over a network of **switching nodes**.
- **Switching nodes** do not concern with content of data. Their purpose is to provide a switching facility that will **move the data from node to node until they reach their destination (the end device).**
- A collection of nodes and connections forms a communications network.
- In a switched communications network, data entering the network from a station are **routed** to the destination by being switched from node to node.

Simple Switching Network



Switching Nodes

- Nodes may connect to other nodes, or to some stations.
- Network is usually partially connected
  - —However, some redundant connections are desirable for reliability
- Two different switching technologies
  - —Circuit switching
  - —Packet switching

Circuit Switching

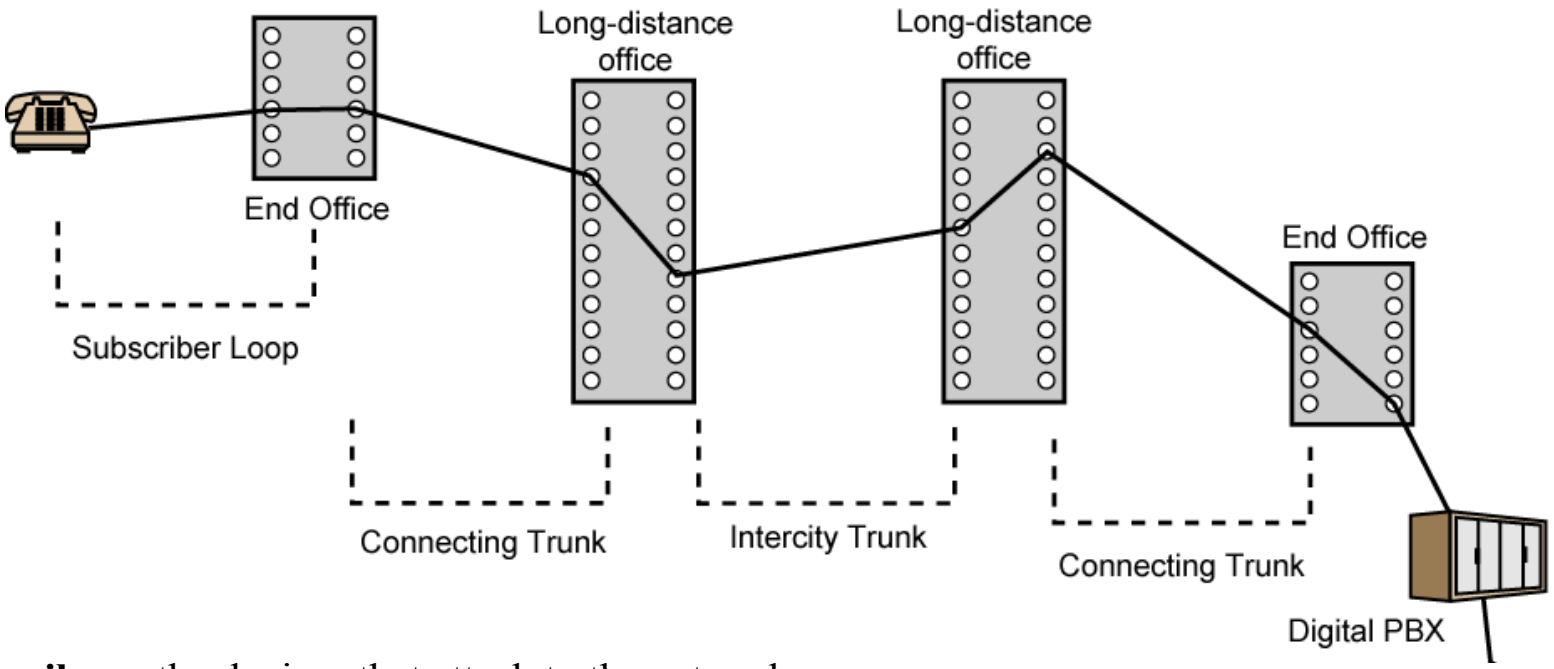
- **Circuit switching:**
  - There is a dedicated communication path between two stations (end-to-end)
  - The path is a connected sequence of links between network nodes.
  - On each physical link, a logical channel is dedicated to the connection.
- Communication via circuit switching has three phases:
  - Circuit establishment (link by link)
    - Routing & resource allocation
  - Data transfer
  - Circuit disconnect
    - Deallocate the dedicated resources
- The switches must know **how to find the route** to the destination and **how to allocate bandwidth (channel)** to establish a connection.

Circuit Switching Properties

- Inefficiency
  - Channel capacity is dedicated for the whole duration of a connection
  - If no data, capacity is wasted
- Delay
  - Long initial delay: circuit establishment takes time
- Developed for voice traffic (public telephone network) but can also applied to data traffic.
  - For voice connections, the resulting circuit will enjoy a high percentage of utilization because most of the time one party or the other is talking.

PSTN (public switched telephone **network**) is the world's collection of interconnected voice-oriented public telephone **networks**, both commercial and government-owned.

Public Circuit Switched Network



Subscribers: the devices that attach to the network.

Subscriber loop: the link between the subscriber and the network.

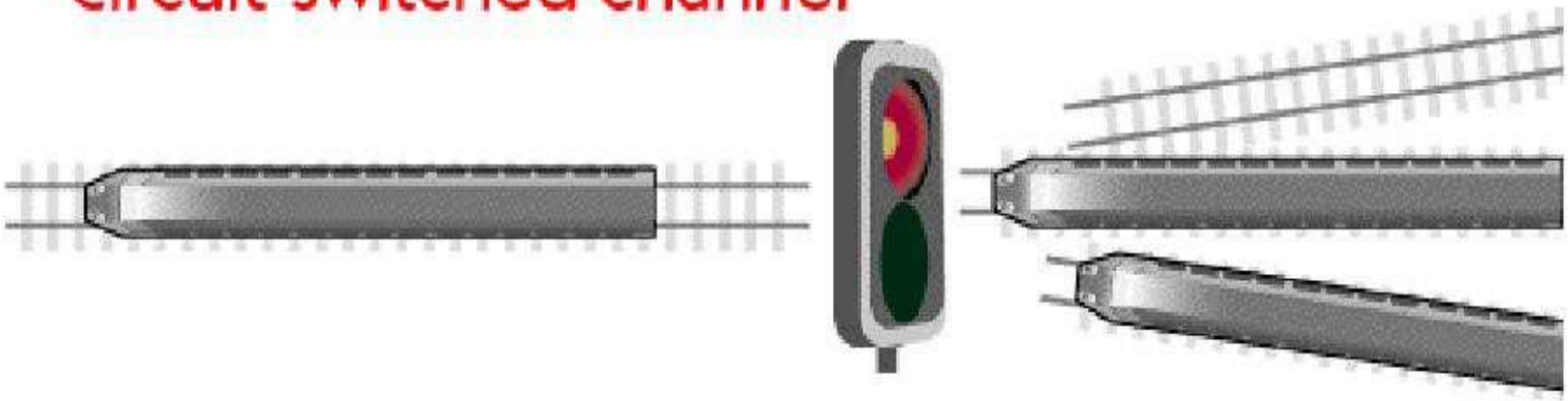
Exchanges: the switching centers in the network.

End office: the switching center that directly supports subscribers.

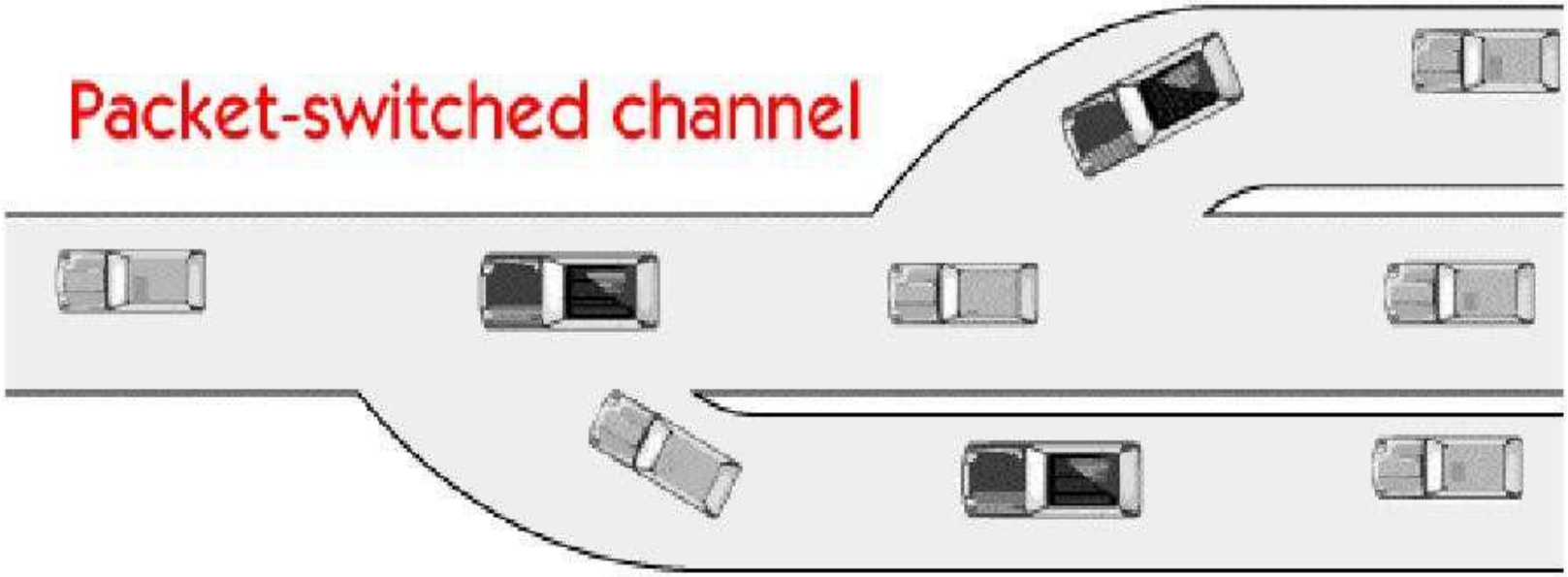
Trunks: the branches between exchanges. They carry multiple voice-frequency circuits.

A **trunk** is a communications line or link designed to carry multiple signals simultaneously to provide **network** access between two points

Circuit-switched channel



Packet-switched channel



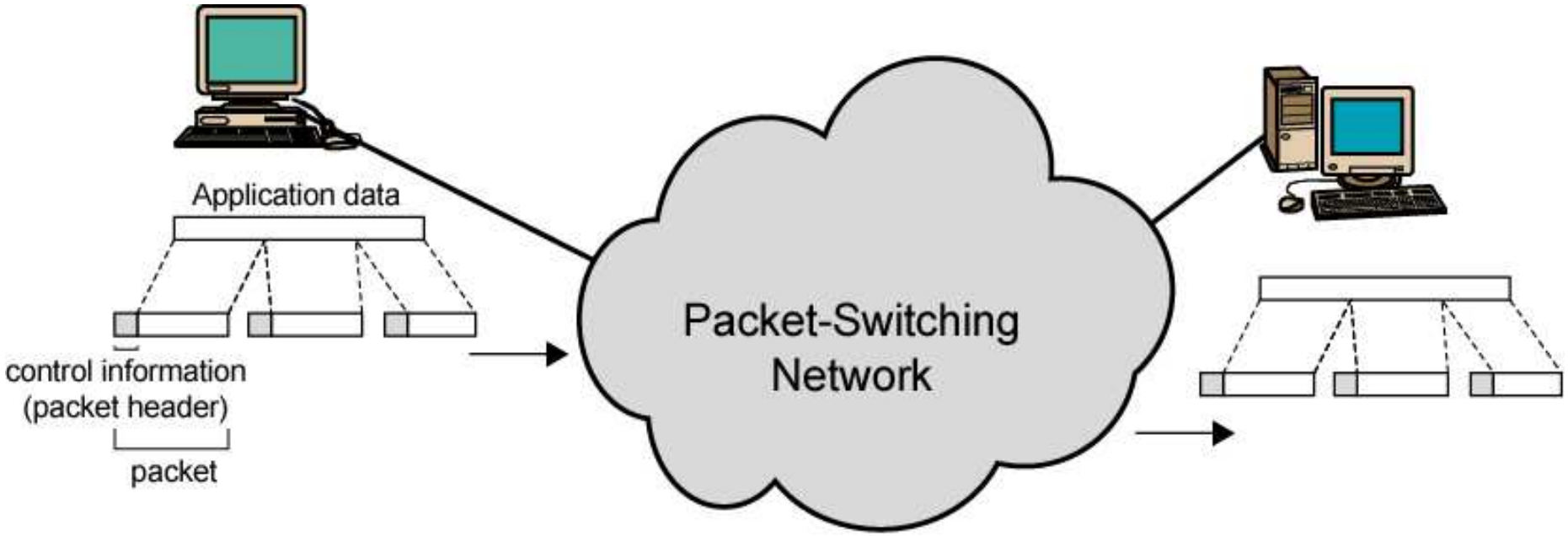
Packet Switching Principles

- Problem of circuit switching
  - —designed for **voice service**
  - —Resources dedicated to a particular call
  - —For data transmission, much of the time the connection is idle (say, web browsing)
  - —Data rate is fixed
    - Both ends must operate at the same rate during the entire period of connection
- Packet switching is designed to address these problems.

Basic Operation

- Data are transmitted in short packets
  - Typically at the order of 1000 bytes
  - Longer messages are split into series of packets
  - Each packet contains a portion of user data plus some control info
- Control info contains at least
  - Routing (addressing) info, so as to be routed to the intended destination
  - Recall the content of an IP header!
- **store and forward**
  - On each switching node, packets are received, stored briefly (buffered) and passed on to the next node.

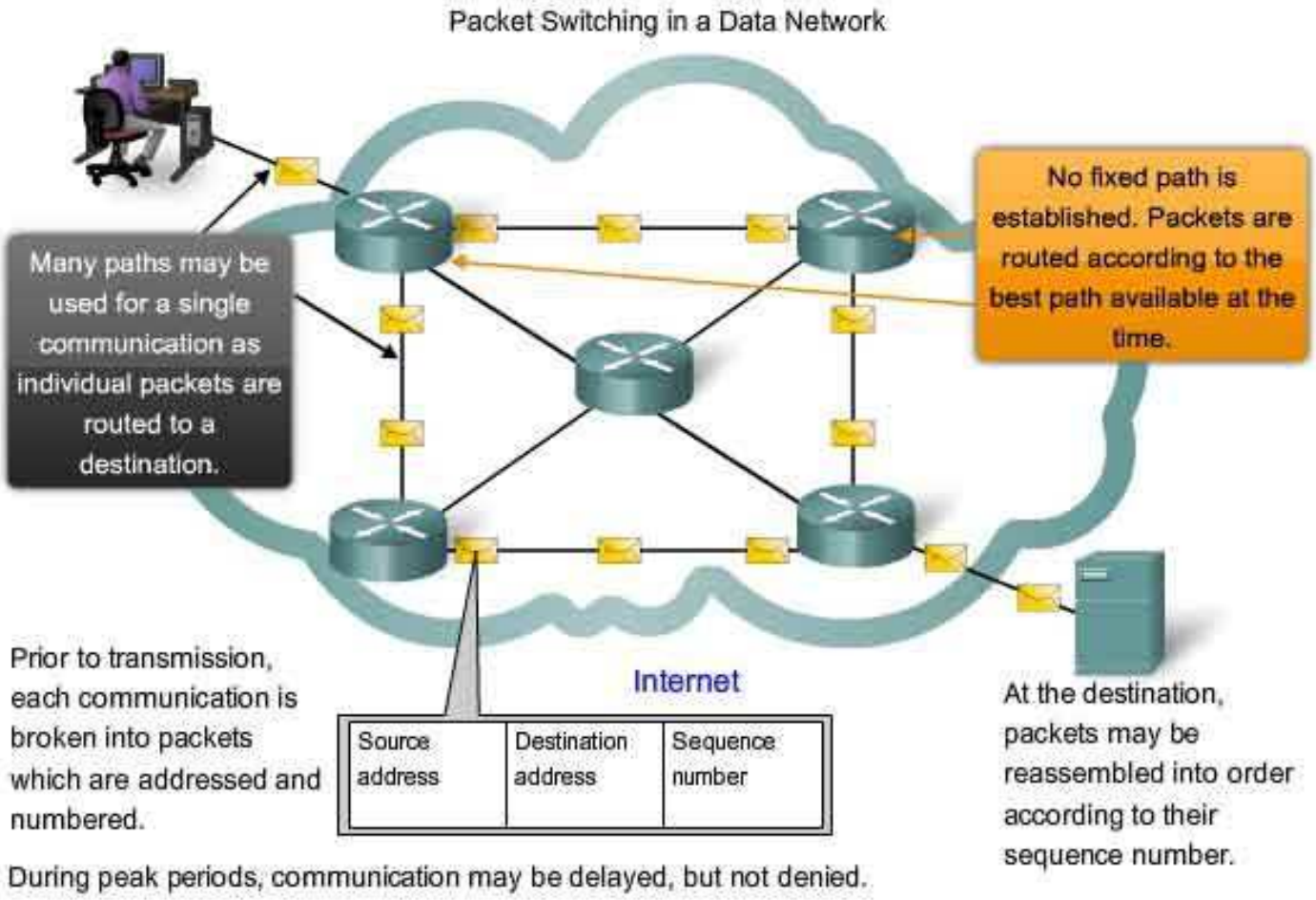
Use of Packets



Advantages of Packet Switching

- **In circuit-switching**, a connection could be blocked if there lacks free resources.
- **On a packet-switching network**, even with heavy traffic, packets are still accepted, by delivery delay increases.
- **Priorities can be used**
  - On each node, **packets with higher priority can be forwarded first.**
  - They will experience less delay than lower-priority packets.

Packet Switching



Packet Switching Technique

- A station breaks long message into packets
- Packets are sent out to the network sequentially, one at a time
- How will the network handle this stream of packets as it attempts to route them through the network and deliver them to the intended destination?
  - —Two approaches
    - **Datagram** approach
    - **Virtual circuit** approach

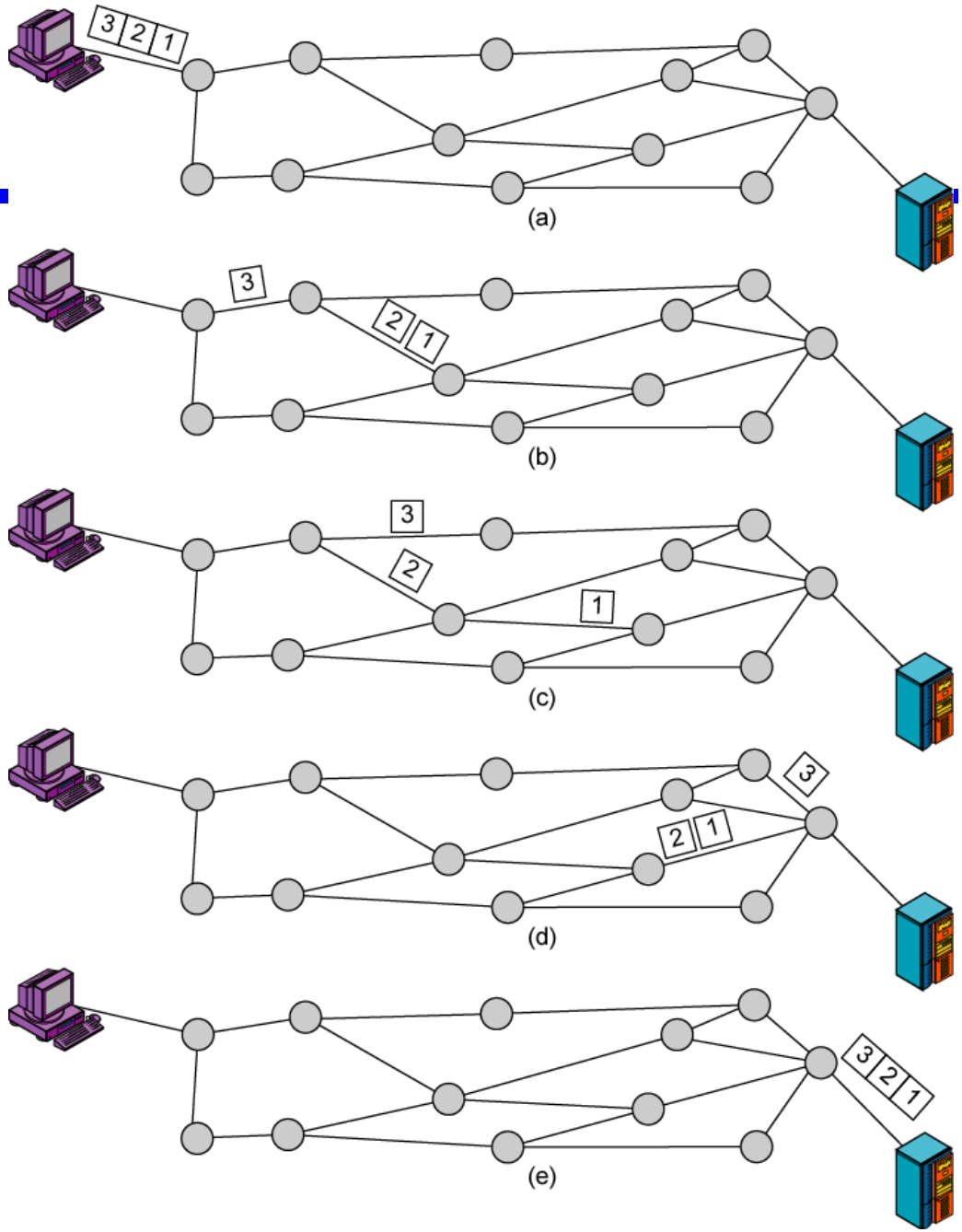
Differences between Circuit Switching and Packet Switching

Circuit switching	Packet switching
1. Call set up is required.	1. Call setup is not required.
2. Dedicated connection between two Hosts.	2. No dedicated connection between two Hosts.
3. Connection/Communication is lost, if any link in the path between the Hosts is broken.	3. Connection/Communication could continue between the Hosts since data have many routes between the Hosts.
4. Information take the same route between the connected Hosts	4. Information could take different routes to reach the destination Host.
5. Information always arrives in order.	5. Information could arrive out of order to the destination
6. Bandwidth available is fixed.	6. Bandwidth available is variable.
7. Congestion is call based.	7. Congestion is packet based.
8. Bandwidth utilization is partial.	8. Bandwidth utilization is full.
9. It does not uses store-and-forward transmission.	9. It uses store-and forward transmission.
10. It is Transparent.	10. Not transparent.
11. Charging is time based.	11. Charging is packet based.

Datagram

- Each packet is treated independently, with no reference to packets that have gone before.
  - Each node chooses the next node on a packet's path.
- Packets can take any possible route.
- Packets may arrive at the receiver out of order.
- Packets may go missing.
- It is up to the receiver to re-order packets and recover from missing packets.
- Example: Internet

Datagram



Virtual Circuit

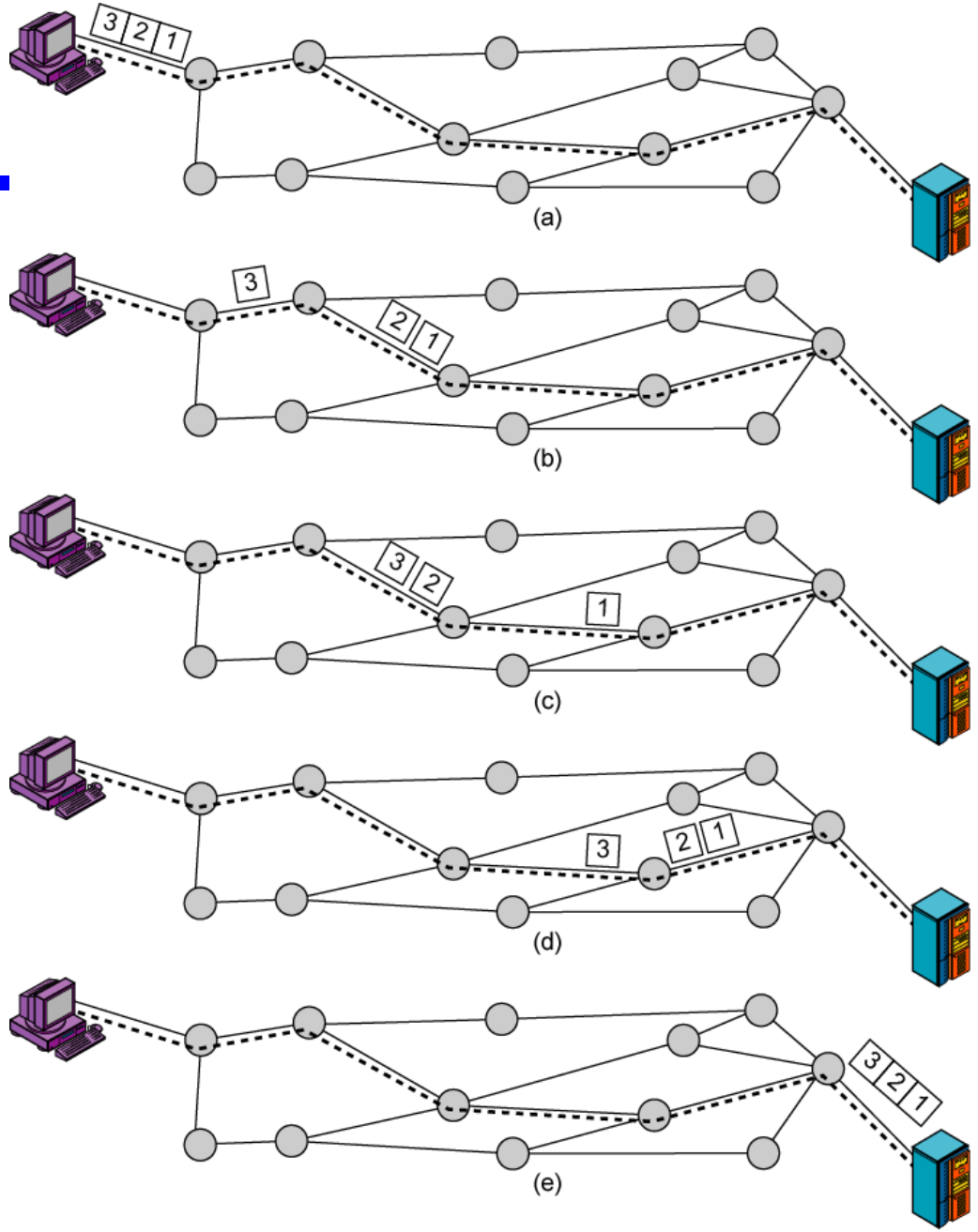
- In virtual circuit, a preplanned route is established before any packets are sent, then all packets follow the same route.
- Each packet contains a **virtual circuit identifier** instead of destination address, and each node on the preestablished route knows where to forward such packets.
  - —The node need not make a routing decision for each packet.
- A **virtual circuit identifier** (VCID) is a type of numeric **identifier** used to distinguish between different **virtual circuits** in a connection-oriented **circuit**-switched telecommunication network.

Virtual Circuit

A route between stations is set up prior to data transfer.

All the data packets then follow the same route.

But there is no dedicated resources reserved for the virtual circuit! Packets need to be stored-and-forwarded.



Virtual Circuits v Datagram

- Virtual circuits
  - Network can provide sequencing (packets arrive at the same order) and error control (retransmission between two nodes).
  - Packets are forwarded more quickly
    - Based on the virtual circuit identifier
    - No routing decisions to make
  - Less reliable
    - If a node fails, all virtual circuits that pass through that node fail.
- Datagram
  - No call setup phase
    - Good for bursty data, such as Web applications
  - More flexible
    - If a node fails, packets may find an alternate route
    - Routing can be used to avoid congested parts of the network

Kinds of Virtual Circuits

- A **switched virtual circuit** (SVC) is a type of **virtual circuit** in telecommunication and computer networks that is used to establish a temporary connection between two different network nodes until completion of a data transfer session, after which the connection is terminated.
- A **permanent virtual circuit** (PVC) is a continuously dedicated **virtual circuit.**

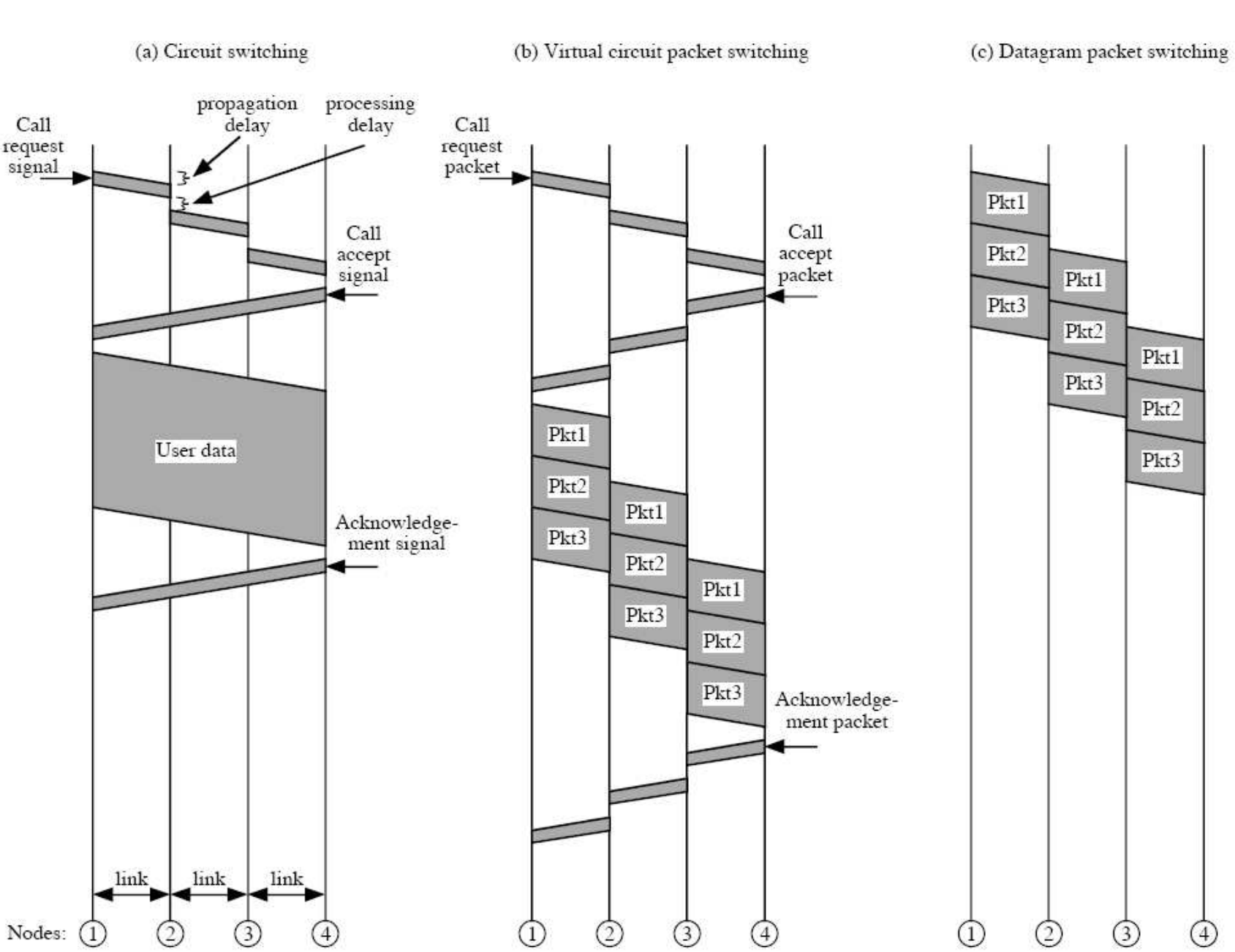
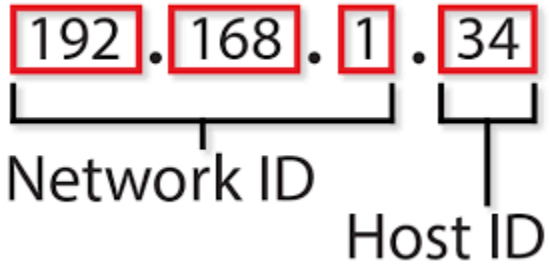


Figure 10.15 Event Timing for Circuit Switching and Packet Switching

Comparison of communication switching techniques

Circuit Switching	Datagram Packet Switching	Virtual Circuit Packet Switching
Dedicated transmission path	No dedicated path	No dedicated path
Continuous transmission of data	Transmission of packets	Transmission of packets
Fast enough for interactive	Fast enough for interactive	Fast enough for interactive
Messages are not stored	Packets may be stored until delivered	Packets stored until delivered
The path is established for entire conversation	Route established for each packet	Route established for entire conversation
Call setup delay; negligible transmission delay	Packet transmission delay	Call setup delay; packet transmission delay
Busy signal if called party busy	Sender may be notified if packet not delivered	Sender notified of connection denial
Overload may block call setup; no delay for established calls	Overload increases packet delay	Overload may block call setup; increases packet delay
Electromechanical or computerized switching nodes	Small switching nodes	Small switching nodes
User responsible for message loss protection	Network may be responsible for individual packets	Network may be responsible for packet sequences
Usually no speed or code conversion	Speed and code conversion	Speed and code conversion
Fixed bandwidth	Dynamic use of bandwidth	Dynamic use of bandwidth
No overhead bits after call setup	Overhead bits in each packet	Overhead bits in each packet

IPv4



An Internet Protocol **address** (**IP address**) is a numerical label assigned to each device participating in a computer network that uses the Internet Protocol for communication.

An **IP address** serves two main functions: host or network interface identification and location addressing.

Internet Protocol Version 4 (**IPv4**) is the fourth revision of the Internet Protocol and a widely used protocol in data communication over different kinds of networks.

IPv4 is a connectionless protocol used in **packet-switched layer networks, such as Ethernet.**

An internet protocol is the set of rules that govern how packets are transmitted over a network. IPv5 is a version of internet protocol (IP) that was never formally adopted as a standard. The v5 stands for version 5 of internet protocol. Computer networks use version 4, typically called IPv4, or a newer version of IP called IPv6.

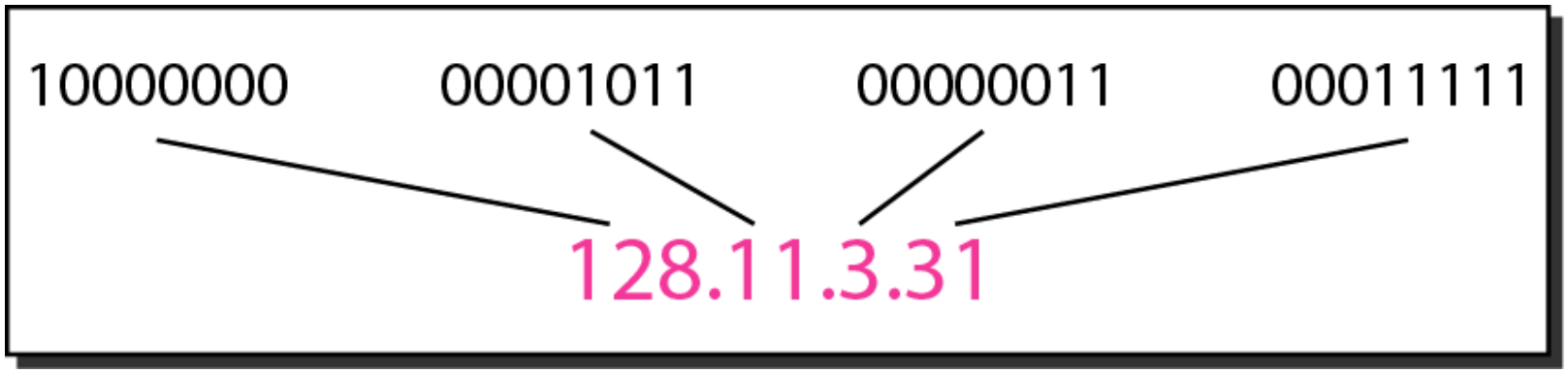
For IPv4, this pool is **32**-bits (232) in size and contains **4,294,967,296** IPv4 addresses.

The IPv6 address space is 128-bits (2128) in size, containing 340,282,366,920,938,463,463,374,607,431,768,211,456 IPv6 addresses

IPv6 is the sixth revision to the Internet Protocol and the successor to **IPv4**.

It functions similarly to **IPv4** in that it provides the unique, numerical IP addresses necessary for Internet-enabled devices to communicate. However, it does sport one major difference: it utilizes 128-bit addresses.

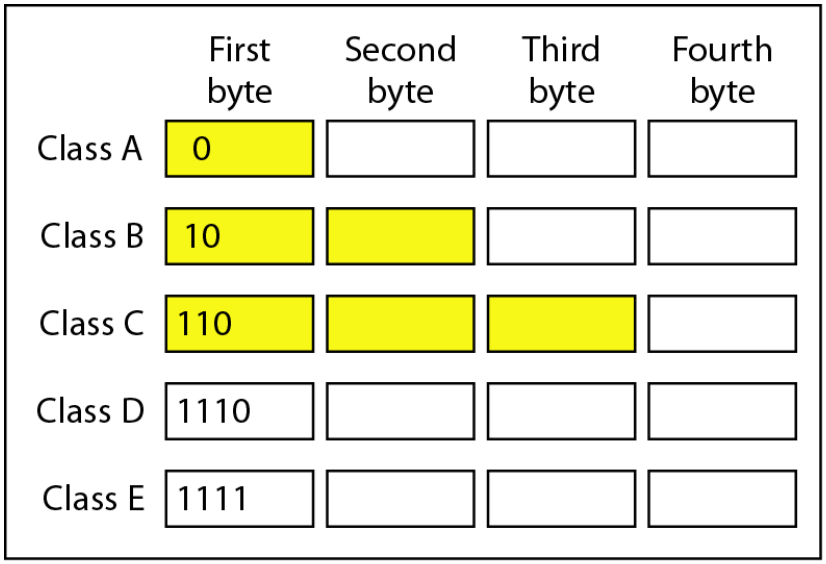
Dotted-decimal notation and binary notation for an IPv4 address



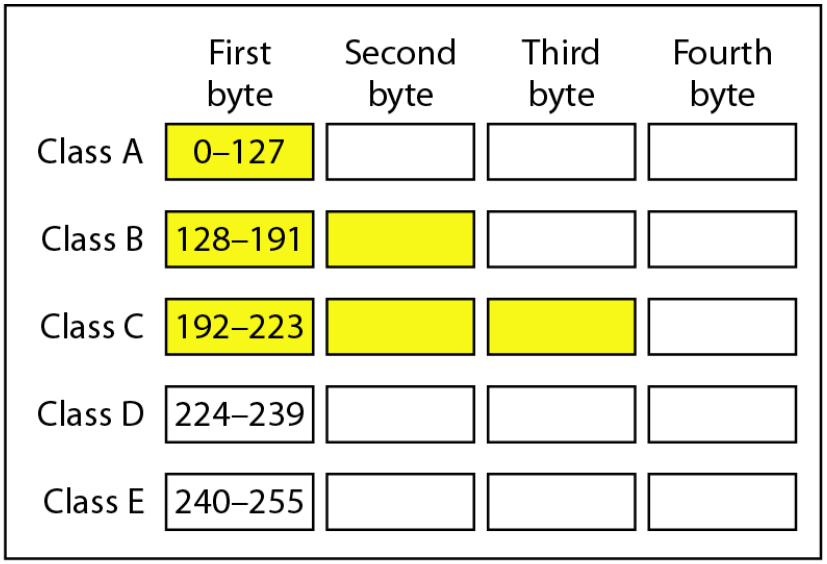
Note

In classful addressing, the address space is divided into five classes: A, B, C, D, and E.

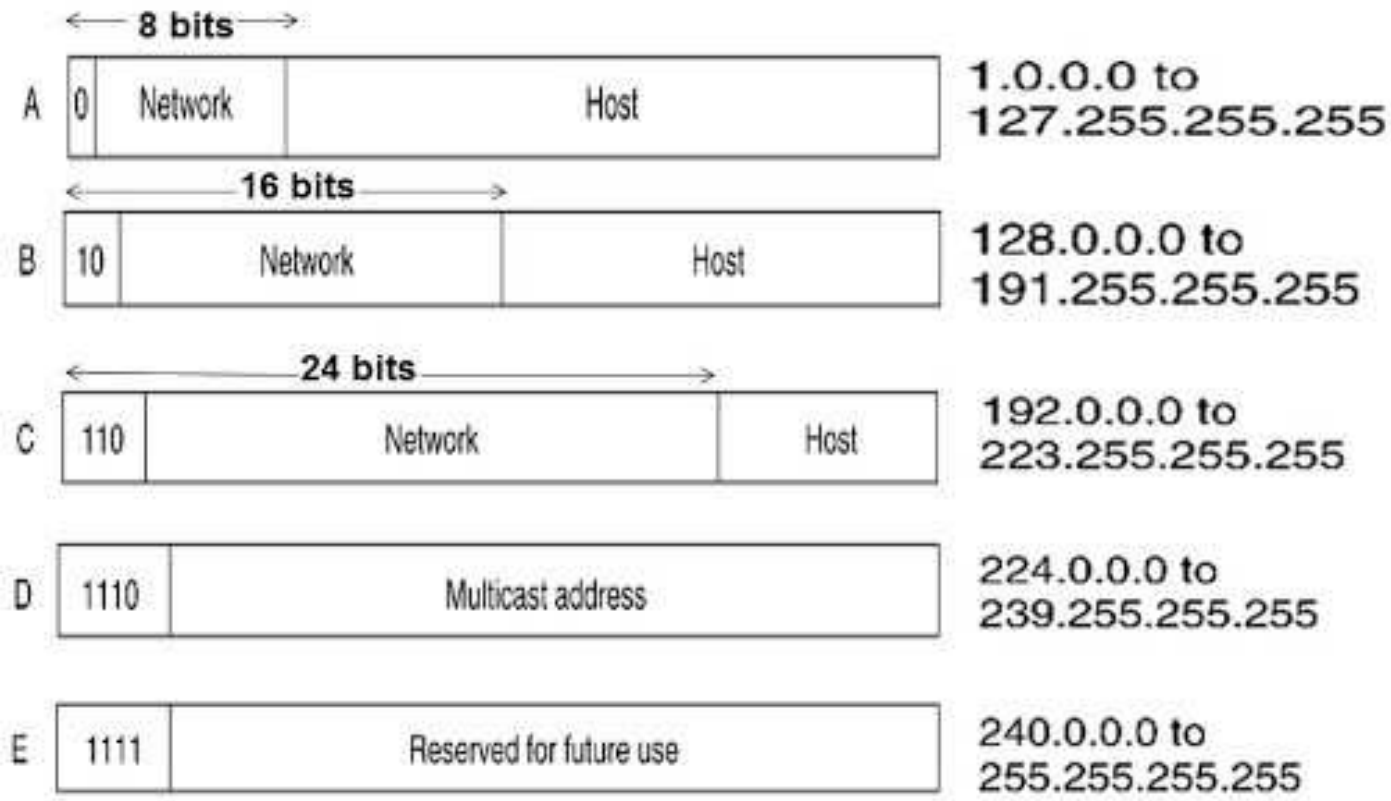
Figure *Finding the classes in binary and dotted-decimal notation*



a. Binary notation



b. Dotted-decimal notation


8 bits
A
0
Network
Host
1.0.0.0 to
127.255.255.255
16 bits
B
10
Network
Host
128.0.0.0 to
191.255.255.255
24 bits
C
110
Network
Host
192.0.0.0 to
223.255.255.255
D
1110
Multicast address
224.0.0.0 to
239.255.255.255
E
1111
Reserved for future use
240.0.0.0 to
255.255.255.255

Example

Change the following IPv4 addresses from binary notation to dotted-decimal notation.

a. 10000001 00001011 00001011 11101111

b. 11000001 10000011 00011011 11111111

Solution

We replace each group of 8 bits with its equivalent decimal number and add dots for separation.

a. 129.11.11.239

b. 193.131.27.255

Example

Change the following IPv4 addresses from dotted-decimal notation to binary notation.

a. 111.56.45.78

b. 221.34.7.82

Solution

We replace each decimal number with its binary equivalent.

a. 01101111 00111000 00101101 01001110

b. 11011101 00100010 00000111 01010010

Example

Find the error, if any, in the following IPv4 addresses.

a. 111.56.045.78

b. 221.34.7.8.20

c. 75.45.301.14

d. 11100010.23.14.67

Solution

a. There must be no leading zero (045).

b. There can be no more than four numbers.

c. Each number needs to be less than or equal to 255.

d. A mixture of binary notation and dotted-decimal notation is not allowed.

Table 19.1 *Number of blocks and block size in classful IPv4 addressing*

Class	*Number of Blocks*	*Block Size*	*Application*
A	128	16,777,216	Unicast
B	16,384	65,536	Unicast
C	2,097,152	256	Unicast
D	1	268,435,456	Multicast
E	1	268,435,456	Reserved

Example

Find the class of each address.

a. 0**0000001** 00001011 00001011 11101111

b. 110**00001** 10000011 00011011 11111111

c. 14.23.120.8

d. 252.5.15.111

Solution

a. The first bit is 0. This is a class A address.

b. The first 2 bits are 1; the third bit is 0. This is a class C address.

c. The first byte is 14; the class is A.

d. The first byte is 252; the class is E.

SUBNET MASK

A **subnet mask** is **used to** divide an IP address into two parts. One part identifies the host (computer), the other part identifies the **network** to which it belongs.

To better understand how IP addresses and **subnet masks** work, look at an IP (Internet Protocol) address and see how it is organized.

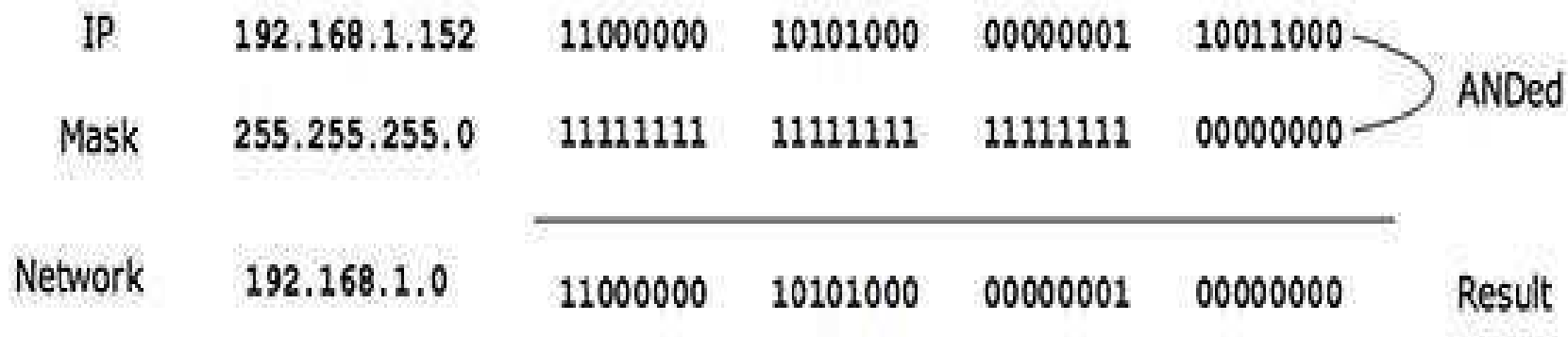
The 32-bit IP address contains information about the host and its network. It is very necessary to distinguish both.

For this, routers use Subnet Mask, which is as long as the size of the network address in the IP address. Subnet Mask is also 32 bits long. If the IP address in binary is ANDed with its Subnet Mask, the result yields the Network address.

SUBNET MASK

For example, say the IP Address is 192.168.1.152 and the Subnet Mask is 255.255.255.0 then:

This way the Subnet Mask helps extract the Network ID and the Host from an IP Address. It can be identified now that 192.168.1.0 is the Network number and 192.168.1.152 is the host on that network.



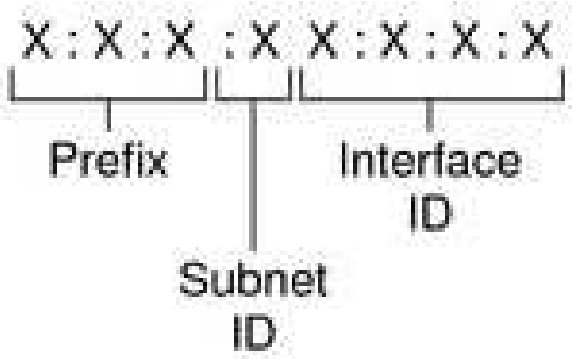
Ipv6

- Internet Protocol Version 6 (IPv6) is a network layer protocol that enables data communications over a packet switched network.
- Packet switching involves the sending and receiving of data in packets between two nodes in a network.
- The working standard for the IPv6 protocol was published by the Internet Engineering Task Force (IETF) in 1998.
- The IETF specification for IPv6 is RFC 2460.
- IPv6 is often referred to as the "next generation Internet" because of its expanded capabilities and its growth through recent large scale deployments.

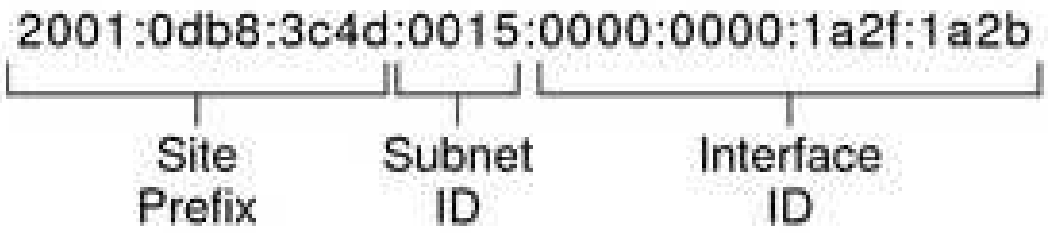
Ipv6

- In 2004, Japan and Korea were acknowledged as having the first public deployments of IPv6.
- The explosive growth in mobile devices including mobile phones, notebook computers, and wireless handheld devices has created a need for additional blocks of IP addresses.
- IPv4 currently supports a maximum of approximately 4.3 billion unique IP addresses.
- IPv6 supports a theoretical maximum of 2128 addresses (340,282,366,920,938,463,463,374,607,431,768,211,456 to be exact!).

Ipv6



Example:



Ipv6

Address Structure

An IPv6 address is made of 128 bits divided into eight 16-bits blocks. Each block is then converted into 4-digit Hexadecimal numbers separated by colon symbols.

For example, given below is a 128 bit IPv6 address represented in binary format and divided into eight 16-bits blocks:

0010000000000001 0000000000000000 0011001000111000
1101111111100001 0000000001100011 0000000000000000
0000000000000000 1111111011111011

Address Structure

Each block is then converted into Hexadecimal and separated by ':' symbol: 2001:0000:3238:DFE1:0063:0000:0000:FEFB

Even after converting into Hexadecimal format, IPv6 address remains long. IPv6 provides some rules to shorten the address. The rules are as follows:

Rule.1: Discard leading Zero(es):

In Block 5, 0063, the leading two 0s can be omitted, such as (5th block):

2001:0000:3238:DFE1:63:0000:0000:FEFB

Rule.2: If two of more blocks contain consecutive zeroes, omit them all and replace with double colon sign ::, such as (6th and 7th block):

2001:0000:3238:DFE1:63::FEFB

Consecutive blocks of zeroes can be replaced only once by :: so if there are still blocks of zeroes in the address, they can be shrunk down to a single zero, such as (2nd block):

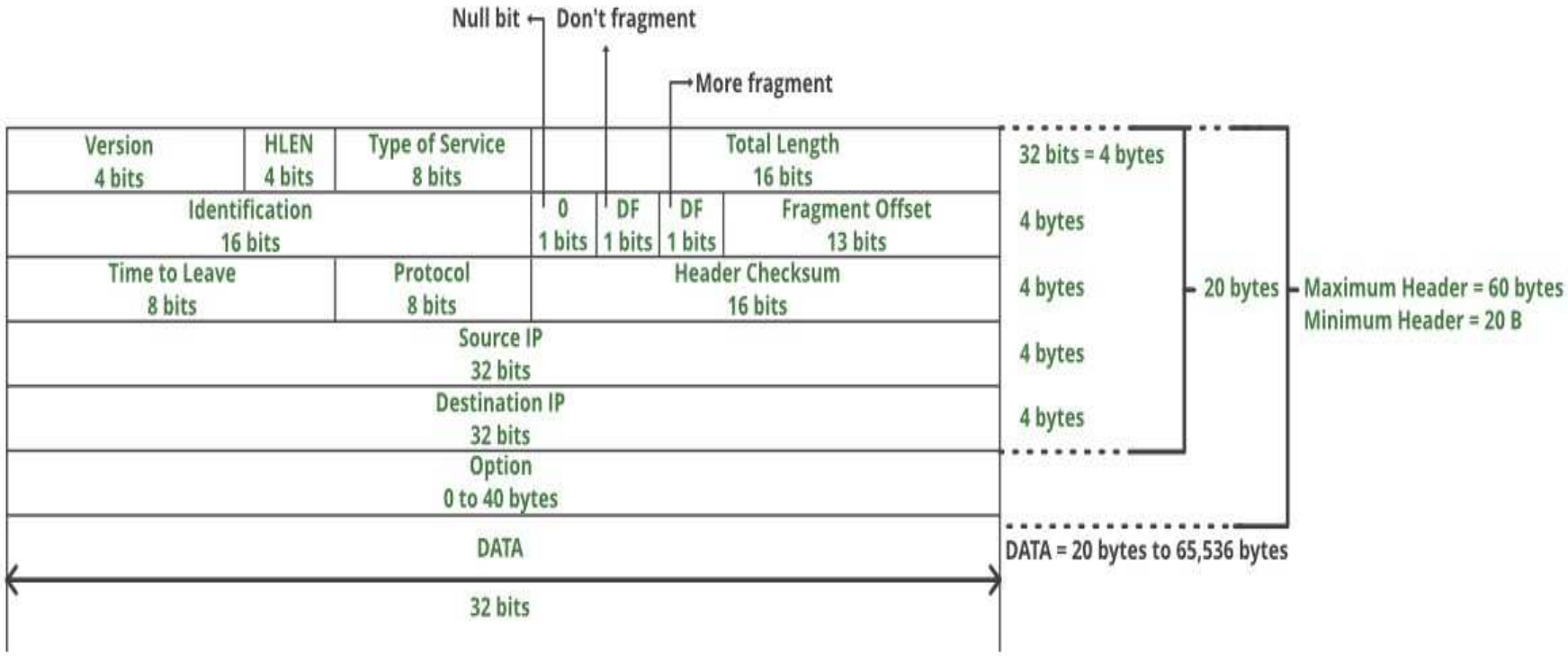
2001:0:3238:DFE1:63::FEFB

IPv6 Address Format

X:X:X:X:X:X:X:X
where X = 0000 ... FFFF (hex)

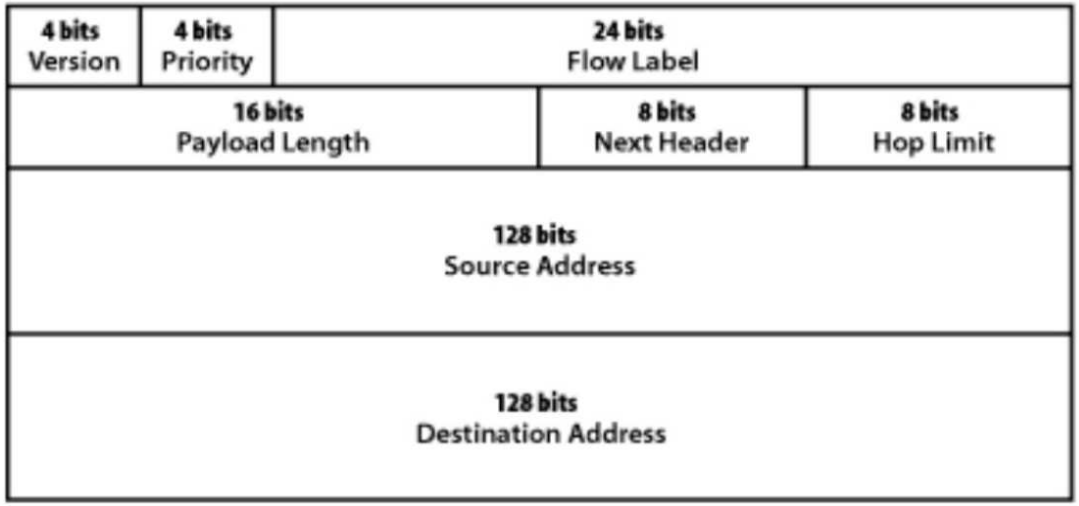
- 2001:0DB8:0000:0000:0008:8000:0000:417A
- 2001:DB8:0:0:8:8000:0:417A
- 2001:DB8::8:8000:0:417A
- 2001:DB8:0:0:8:8000::417A
- 2001:db8::8:8000:417A

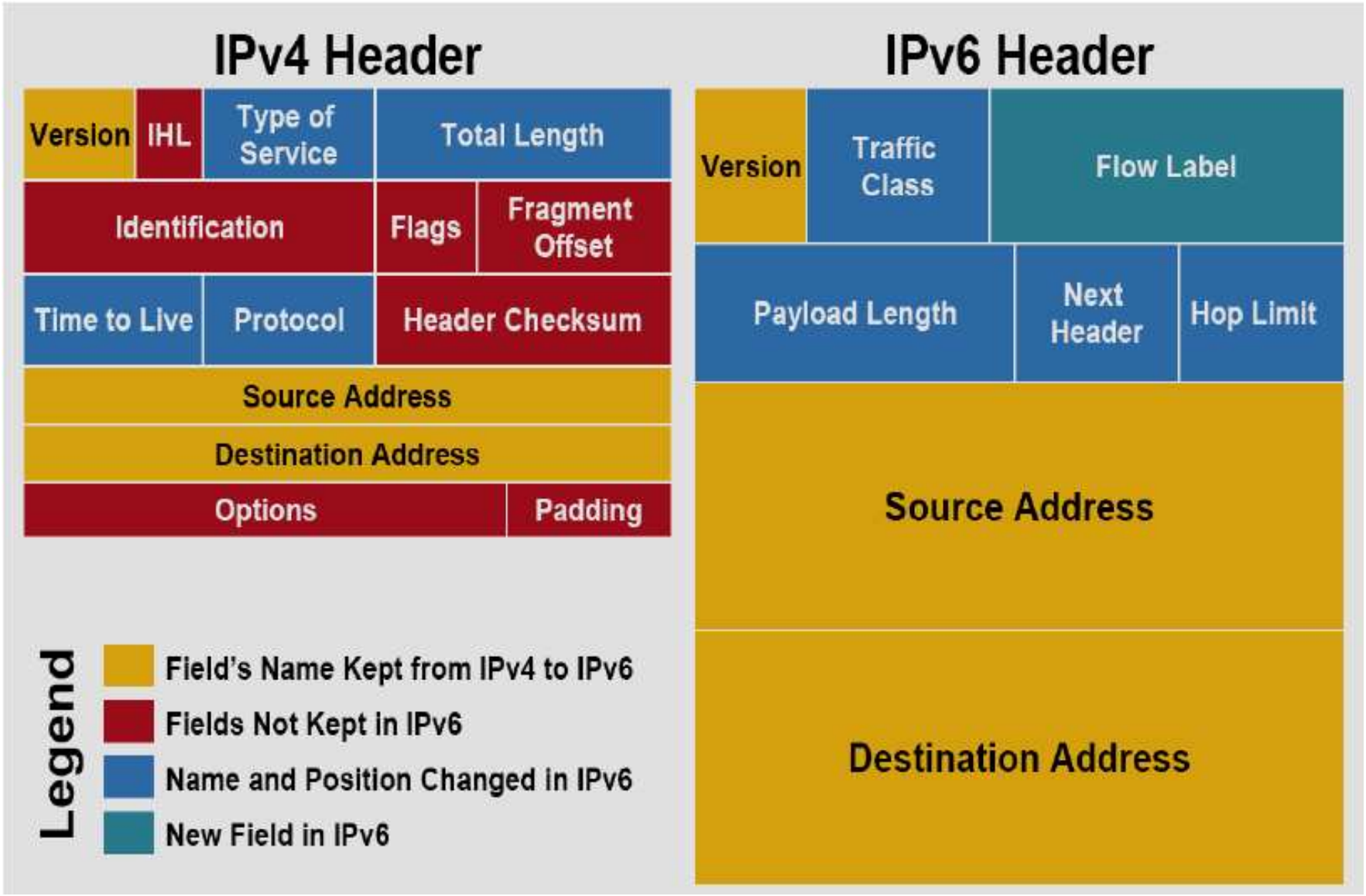
IPV4 Frame Format



IPv6 Address Scheme :

- An IP packet contains several types of information, as illustrated.



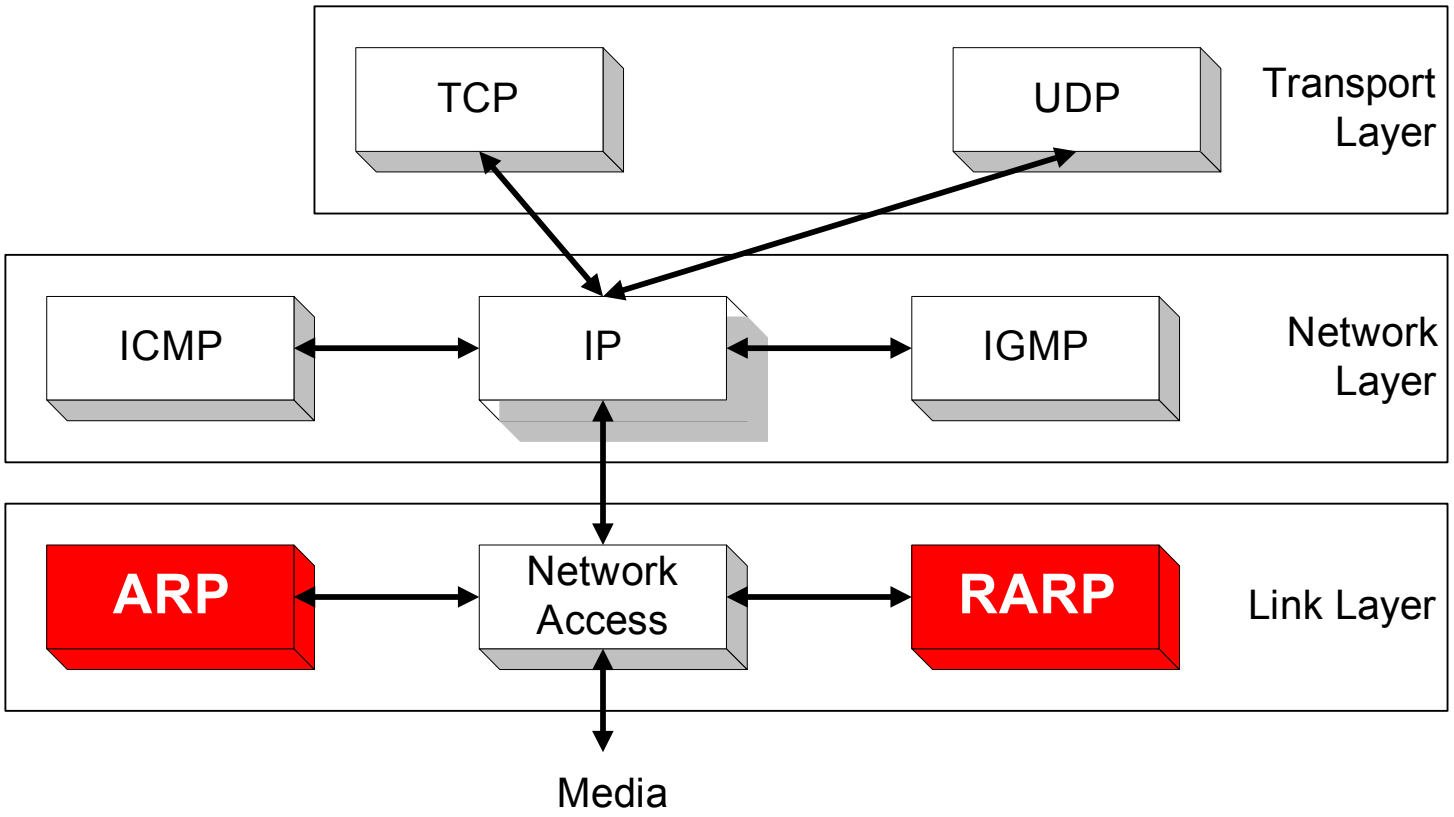

IPv4 Header
Version
IHL
Type of Service
Total Length
Identification
Flags
Fragment Offset
Time to Live
Protocol
Header Checksum
Source Address
Destination Address
Options
Padding
IPv6 Header
Version
Traffic Class
Flow Label
Payload Length
Next Header
Hop Limit
Source Address
Destination Address
Legend
Field's Name Kept from IPv4 to IPv6
Fields Not Kept in IPv6
Name and Position Changed in IPv6
New Field in IPv6

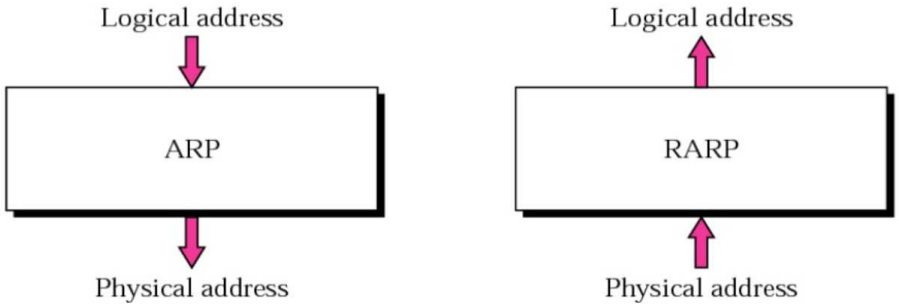
2. Difference b/w IPv4 and IPv6

S.no	IPv4	IPv6
1.Address size	32-bit	128-bit
2.Address format	Dotted decimal notation 192.149.252.76	Hexadecimal notation 3FFE:F200:AB00:0123: 0234:890A:XC01:SD98:
3.Prefix notation	192.149.0.0/24	3FFE:F200:AB00:0123::/48
4.No. of Addresses	2^{32} = 4 billions +	2^{128} =340,282,366,920, 938,463,374,607, 431,768,211,456
5.datagram format		
6. Headers	IPv4 has a 20 octets i.e. 160 bits, containig info. essential to routing and delivery. it consists of 12 fields	IPv6 has 8 fields with a total size of 40 octets i.e. 320 bits.

Address Resolution Protocol (ARP)

Overview


TCP
UDP
Transport Layer
ICMP
IP
IGMP
Network Layer
ARP
Network Access
RARP
Link Layer
Media


Logical address
ARP
Physical address
Logical address
RARP
Physical address

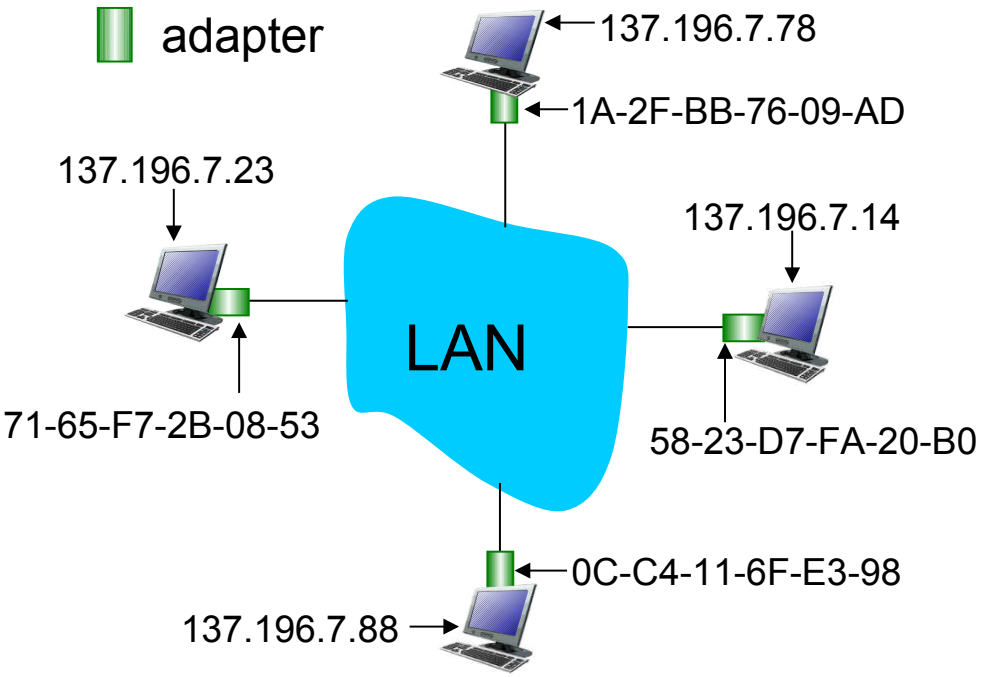
IP and LAN addresses

- The Internet is based on 32bit IP addresses
- Applications only deal with IP addresses
- But all Internet devices connect to a physical link via hardware Network Interface Card (NIC) that has an address.
- Data link protocols (Ethernet, Frame Relay) have different addresses

Nature of MAC addresses

- Hardare address allocation administered by IEEE
- Manufacturer buys portion of hardware address space (to assure uniqueness)
- Analogy of Internet Addresses:
  - **Hardware address**: like Social Security Number
  - **IP address**: like postal address
- Hardware: flat address ➜ portable
  - assigned once, un-chageable, goes with you, move from one physical location to another it doesn't change
- IP: hierarchical address -> *not* portable
  - address assigned based on physical location, i.e.,

IP and MAC Addresses



- IP addresses are "generally" known – i.e., application can find it in DNS database.
- How do we find a device's hardware address?
- Use a "dynamic binding" procedure - an address resolution process that finds hardware address for an IP address.

ADDRESS MAPPING

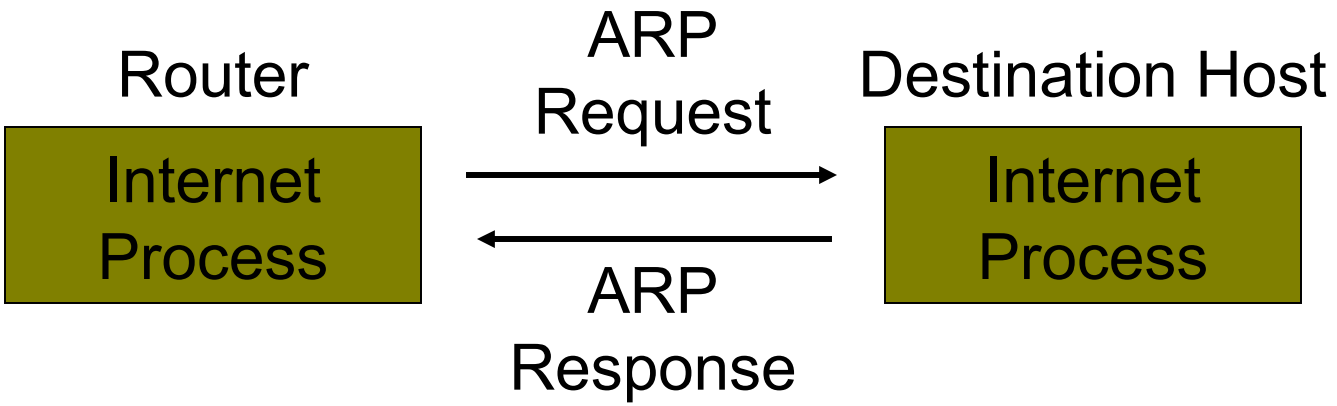
The delivery of a packet to a host or a router requires two levels of addressing:
***logical* and *physical*.**
We need to be able to map a logical address to its corresponding physical address and vice versa.
These can be done using either *static* or *dynamic* mapping.

ADDRESS MAPPING

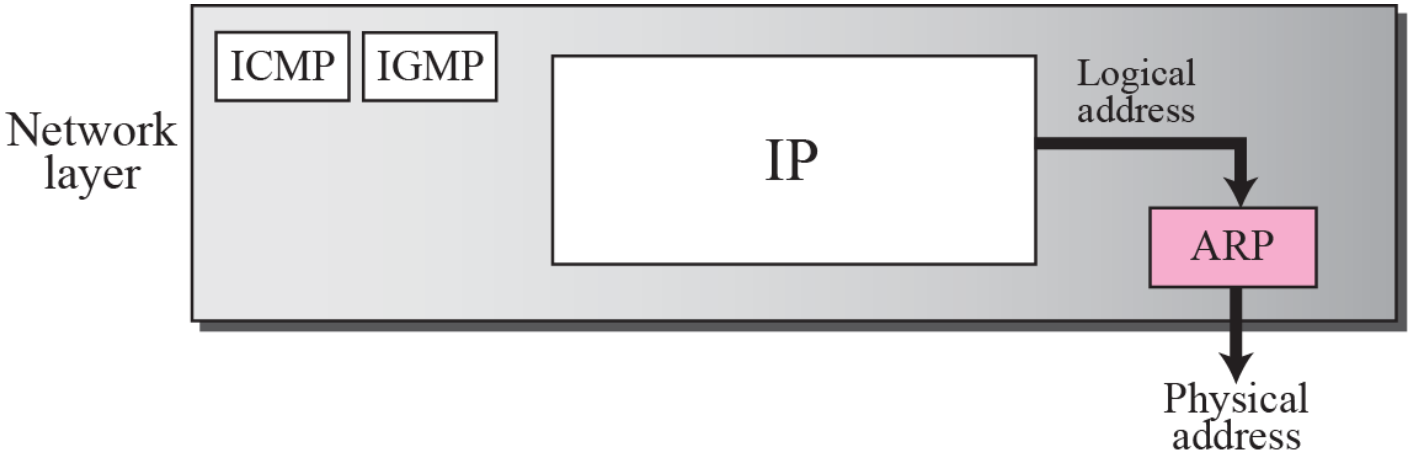
- Anytime a host or a router has an IP datagram to send to another host or router, it has the logical (IP) address of the receiver.
- But the IP datagram must be encapsulated in a frame to be able to pass through the physical network.
- This means that the sender needs the physical address of the receiver.
- A mapping corresponds a logical address to a physical address.
- ARP accepts a logical address from the IP protocol, maps the address to the corresponding physical address and pass it to the data link layer.

Address Resolution Protocol

- ARP Requests and Responses are sent between the internet layer processes on the router and the destination host

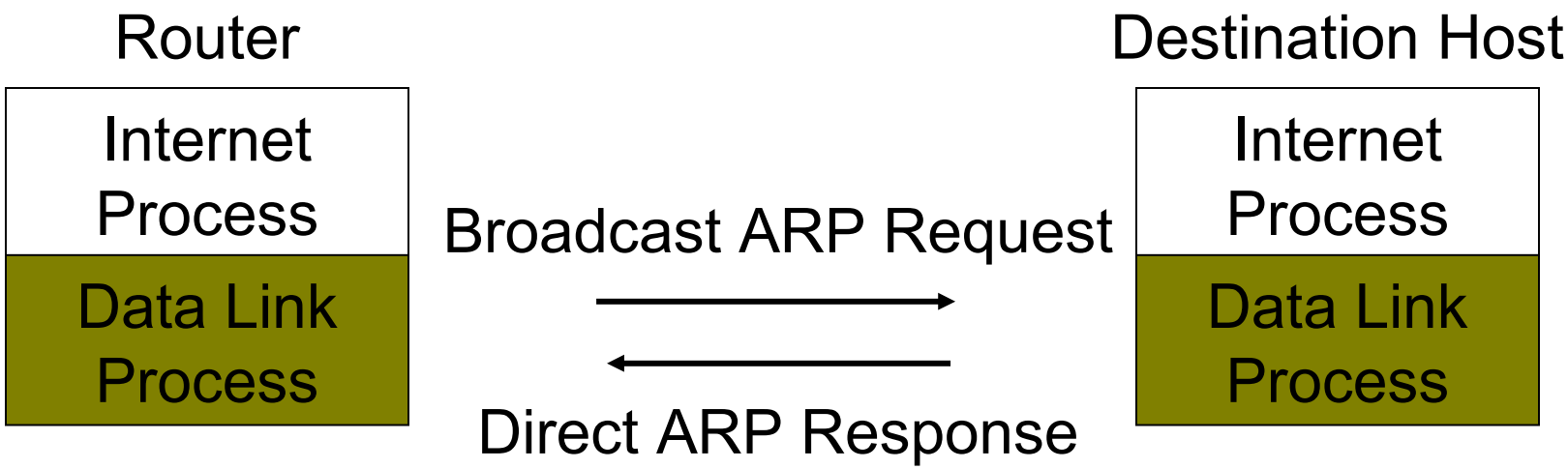


Position of ARP in TCP/IP protocol suite



Address Resolution Protocol

- However, the data link processes deliver these ARP packets
  - Router broadcasts the ARP Request
  - Destination host sends ARP response to the subnet source address found in the broadcast frame

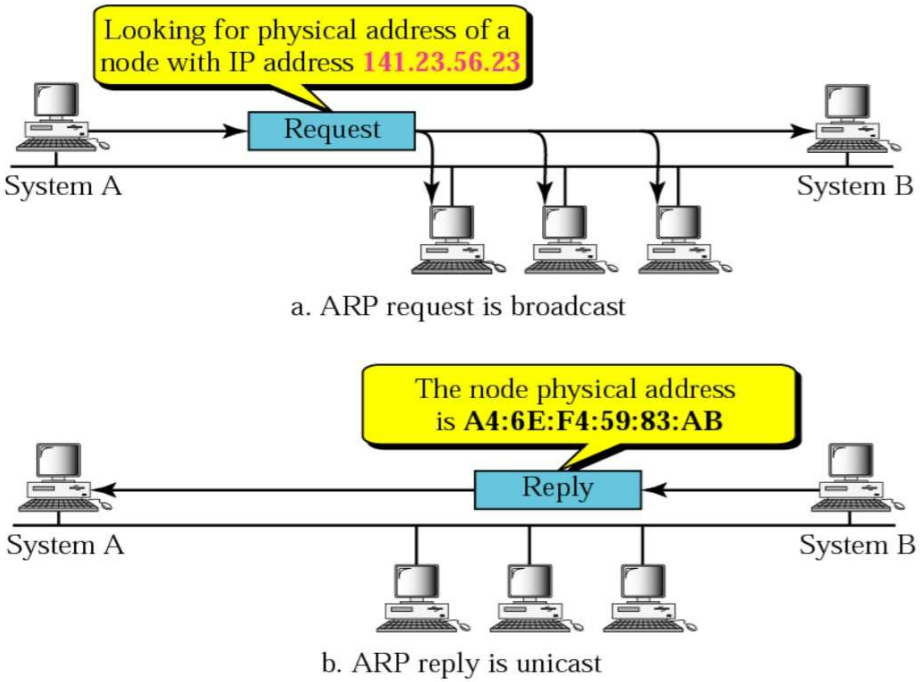


Address Resolution Protocol

ARP associates an IP address with its physical address.

On a typical physical network, such as a LAN, each device on a link is identified by a physical or station address that is usually imprinted on the NIC.

Logical address to physical address translation can be done statically (not practical) or dynamically (with ARP)



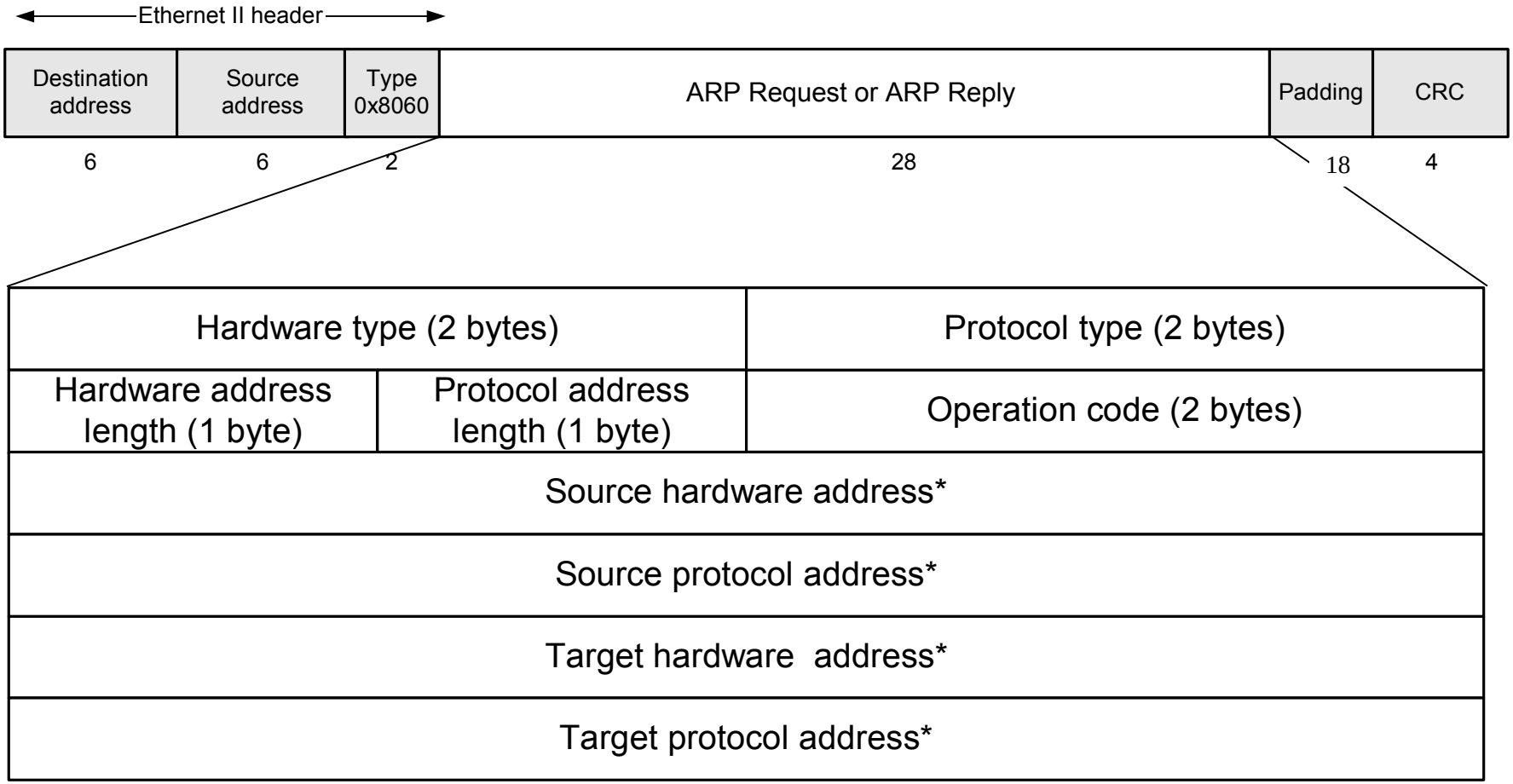
ARP Description

- Allows device A to find device B's hardware address
- **Technique:** broadcast query and obtain unicast response
- Query: sent as a "hardware" broadcast (link layer broadcast)
  - limited broadcast: ARP only used to map addresses within a single physical/segment network, never across multiple (IP) networks
  - Query contains A's hardware address and B's IP address
- Response: sent as a unicast to A's hardware address
  - contains B's hardware address

ARP Operation

- A wants to send datagram to B
  - A starts with B's IP address
  - A knows B is on the local network (resolved by A using network prefix)
- A broadcasts ARP query packet, containing B's IP address
  - dest address in Ethernet frame = FF-FF-FF-FF-FF-FF
  - source address in Ethernet frame = A's hardware address
  - all nodes on data link/single segment network, receive ARP query
  - Query (ARP packet) contains A's hardware address and B's IP address
- B receives A's ARP query packet, recognizes its IP address, replies to A with its (B's) hardware address
  - frame sent to A's hardware address (unicast) containing B's hardware address

ARP Packet Format



* Note: The length of the address fields is determined by the corresponding address length fields

ARP Packet Format(1)

- Hardware Type: Hardware Type field in the Address Resolution Protocol (ARP) **Message specifies the type of hardware used for the local network transmitting the Address Resolution Protocol (ARP) message.** Ethernet is the common Hardware Type and he value for Ethernet is 1. The size of this field is 2 bytes.
- Protocol Type: Each protocol is assigned a number used in this field. IPV4 is 2048 (0x0800 in Hexa).
- Hardware Address Length: Hardware Address Length in the Address Resolution Protocol (ARP) Message is length in bytes of a hardware MAC Address. Ethernet MAC Address are 6 bytes long.

ARP Packet Format(2)

- Protocol Address Length: Length in bytes of a logical address (IPv4 Address). IPv4 Addresses are 4 bytes long.
- Opcode: Opcode field in the Address Resolution Protocol (ARP) Message specifies the nature of the ARP message. 1 for ARP request and 2 for ARP reply.
- Sender Hardware Address: Layer 2 (MAC Address) address of the device sending the message.
- Sender Protocol Address: The protocol address(IPv4 Address) of the device sending the message
- Target Hardware Address: Layer 2 (MAC Address) of the intended receiver. This field is ignored in requests.
- Target Protocol Address: The protocol address(IPv4 Address) of the intended receiver.

ARP Caching

- Because the mapping of IP addresses to media access control (MAC) addresses occurs at each hop (Layer 3 device) on the network for every datagram sent over an internetwork, performance of the network could be compromised.
- To minimize broadcasts and limit wasteful use of network resources, Address Resolution Protocol (ARP) caching was implemented.
- ARP caching is the method of storing network addresses and the associated data-link addresses in memory for a period of time as the addresses are learned.

ARP Caching(1)

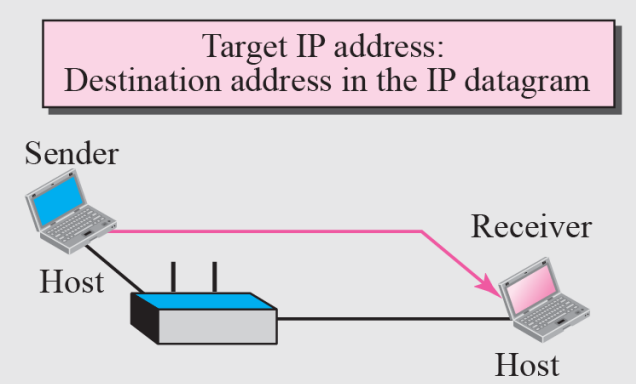
- This minimizes the use of valuable network resources to broadcast for the same address each time a datagram is sent.
- The cache entries must be maintained because the information could become outdated, so it is critical that the cache entries are set to expire periodically.
- Every device on a network updates its tables as addresses are broadcast.

ARP Caching(2)

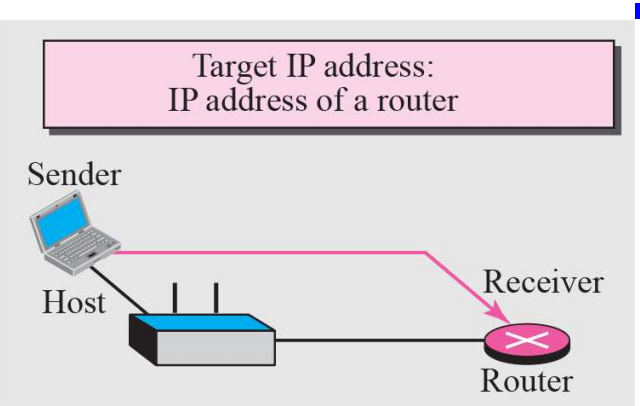
- There are **static ARP cache entries and dynamic ARP cache entries**.
- **Static entries** are manually configured and kept in the cache table on a permanent basis. Static entries are best for devices that have to communicate with other devices usually in the same network on a regular basis.
- **Dynamic entries** are added by Cisco software, kept for a period of time, and then removed.

Four cases using ARP

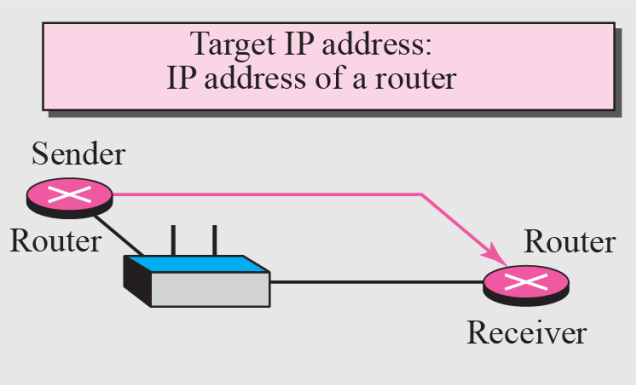
Case 1: A host has a packet to send to a host on the same network.



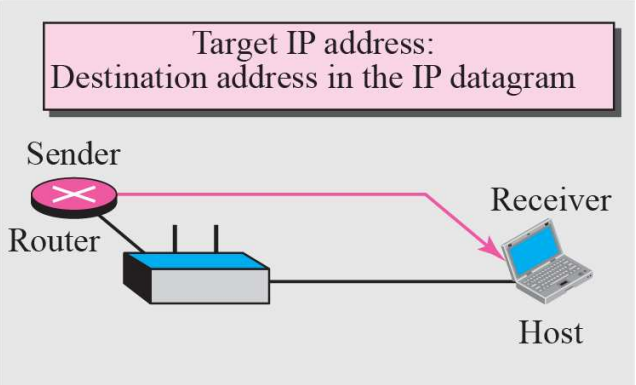
Case 2: A host has a packet to send to a host on another network.



Case 3: A router has a packet to send to a host on another network.



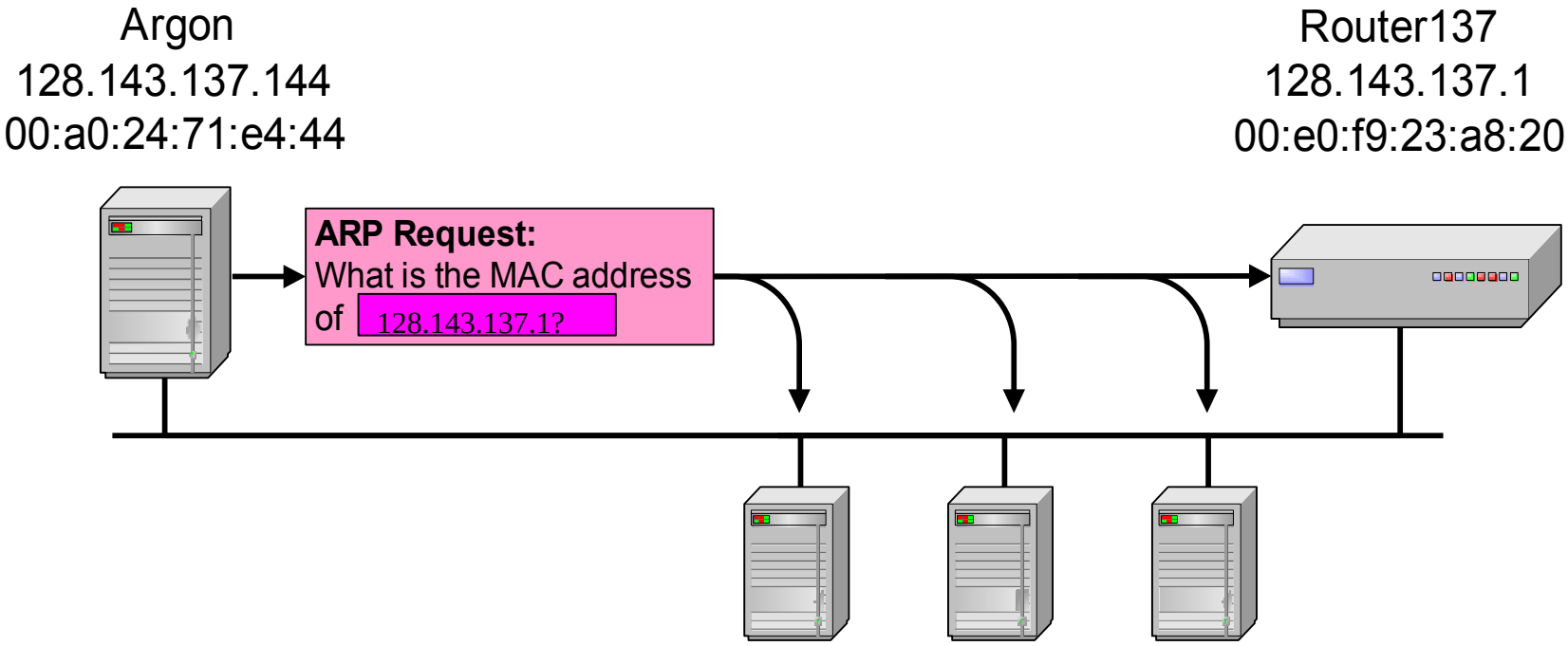
Case 4: A router has a packet to send to a host on the same network.



Address Translation with ARP

ARP Request:

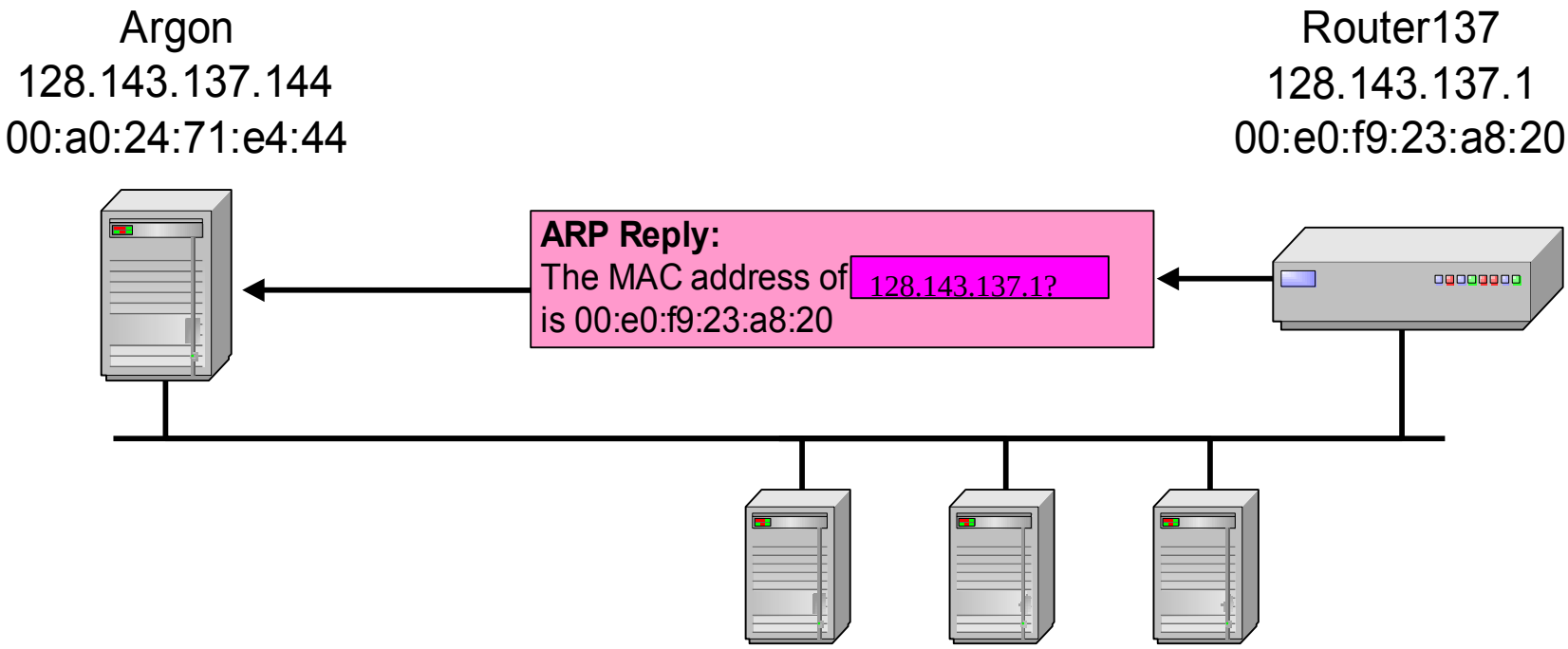
Argon broadcasts an ARP request to all stations on the network: **"What is the hardware address of 128.143.137.1?"**



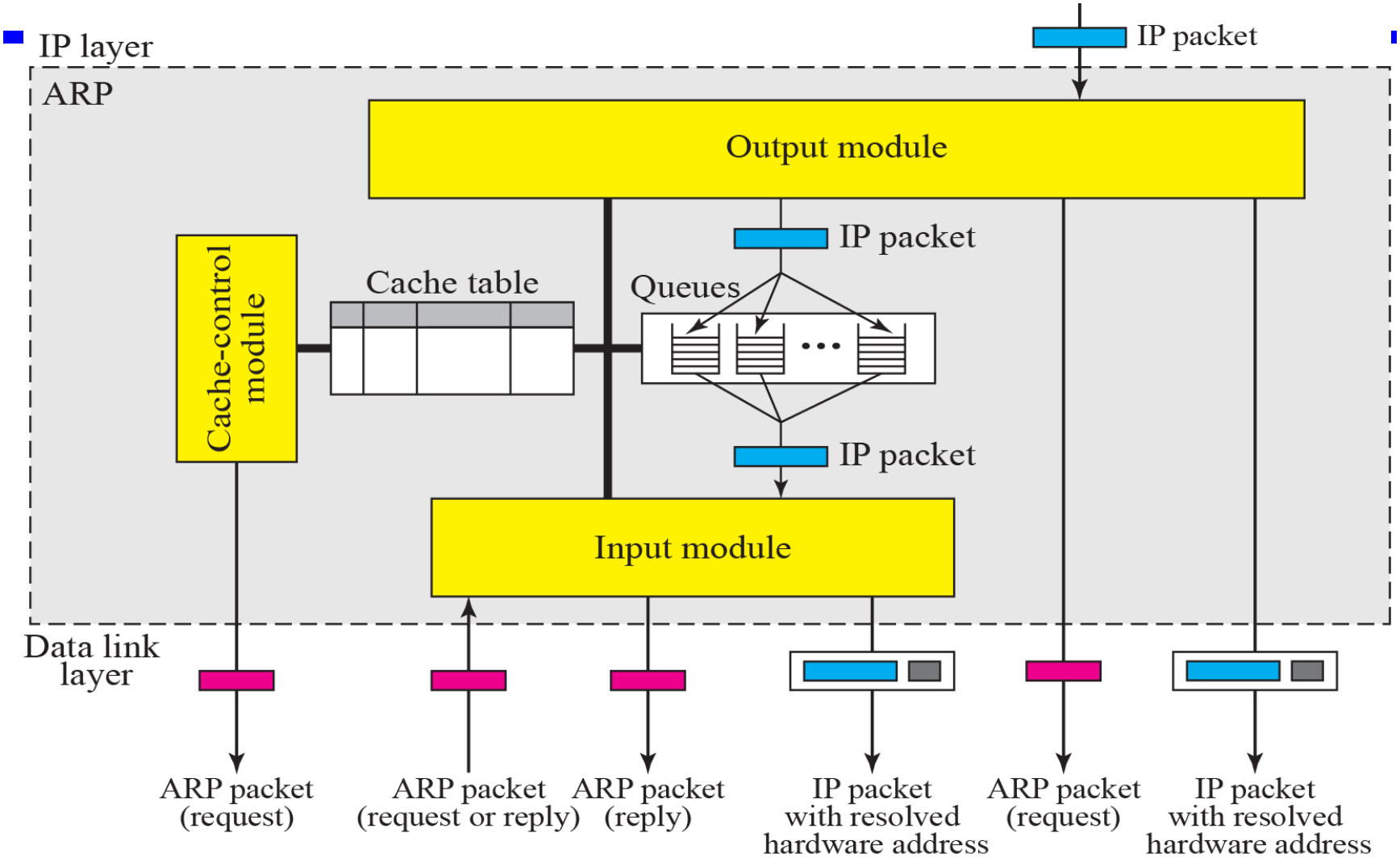
Address Translation with ARP

ARP Reply:

Router 137 responds with an ARP Reply which contains the hardware address



ARP components



ARP Cache (table)

- Contents of the ARP Cache:

  (128.143.71.37) at 00:10:4B:C5:D1:15 [ether] on eth0
  (128.143.71.36) at 00:B0:D0:E1:17:D5 [ether] on eth0
  (128.143.71.35) at 00:B0:D0:DE:70:E6 [ether] on eth0
  (128.143.136.90) at 00:05:3C:06:27:35 [ether] on eth1
  (128.143.71.34) at 00:B0:D0:E1:17:DB [ether] on eth0
  (128.143.71.33) at 00:B0:D0:E1:17:DF [ether] on eth0

- ARP is "plug-and-play":
  - nodes create their ARP tables *without intervention from net administrator*

Things to know about ARP

- What happens if an ARP Request is made for a non-existing host?

  Several ARP requests are made with increasing time intervals between requests. Eventually, ARP gives up.

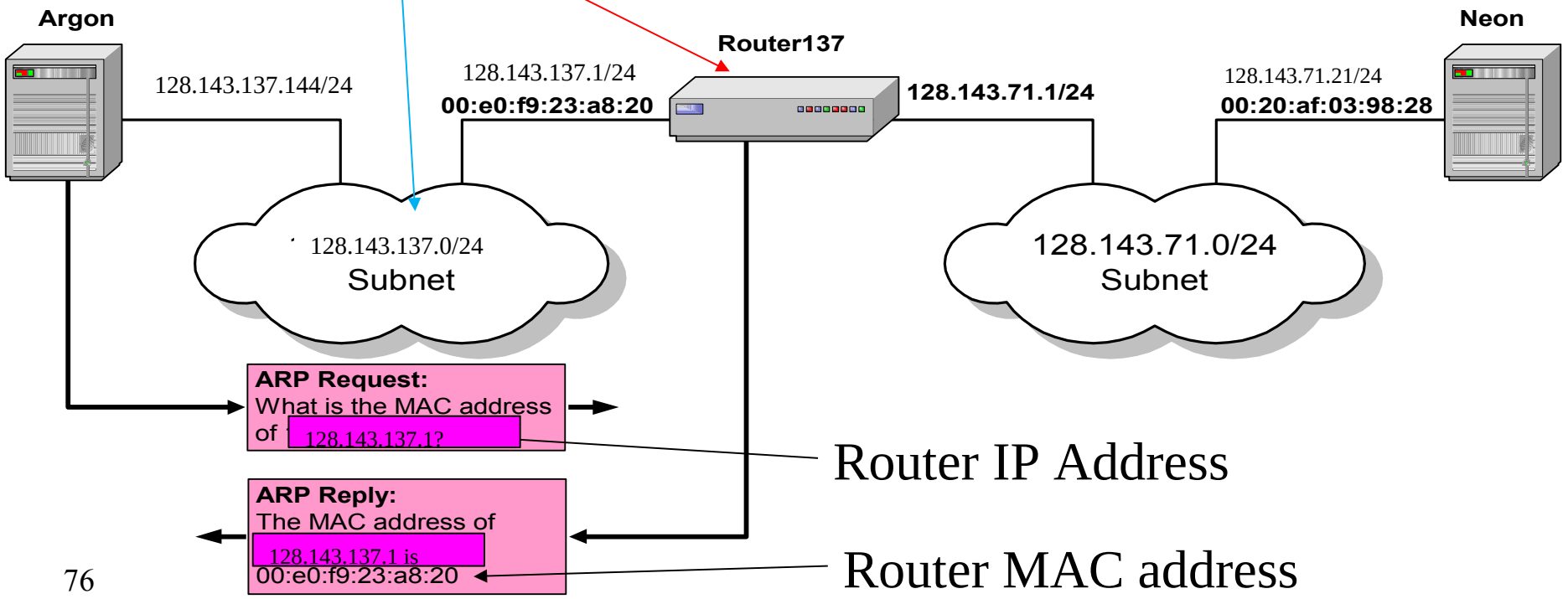
- What if a host sends an ARP request for its own IP address? Know as gratuitous ARP

  No response hopefully

  This is useful for detecting if an IP address has already been assigned (via DHCP).

ARP in our Example

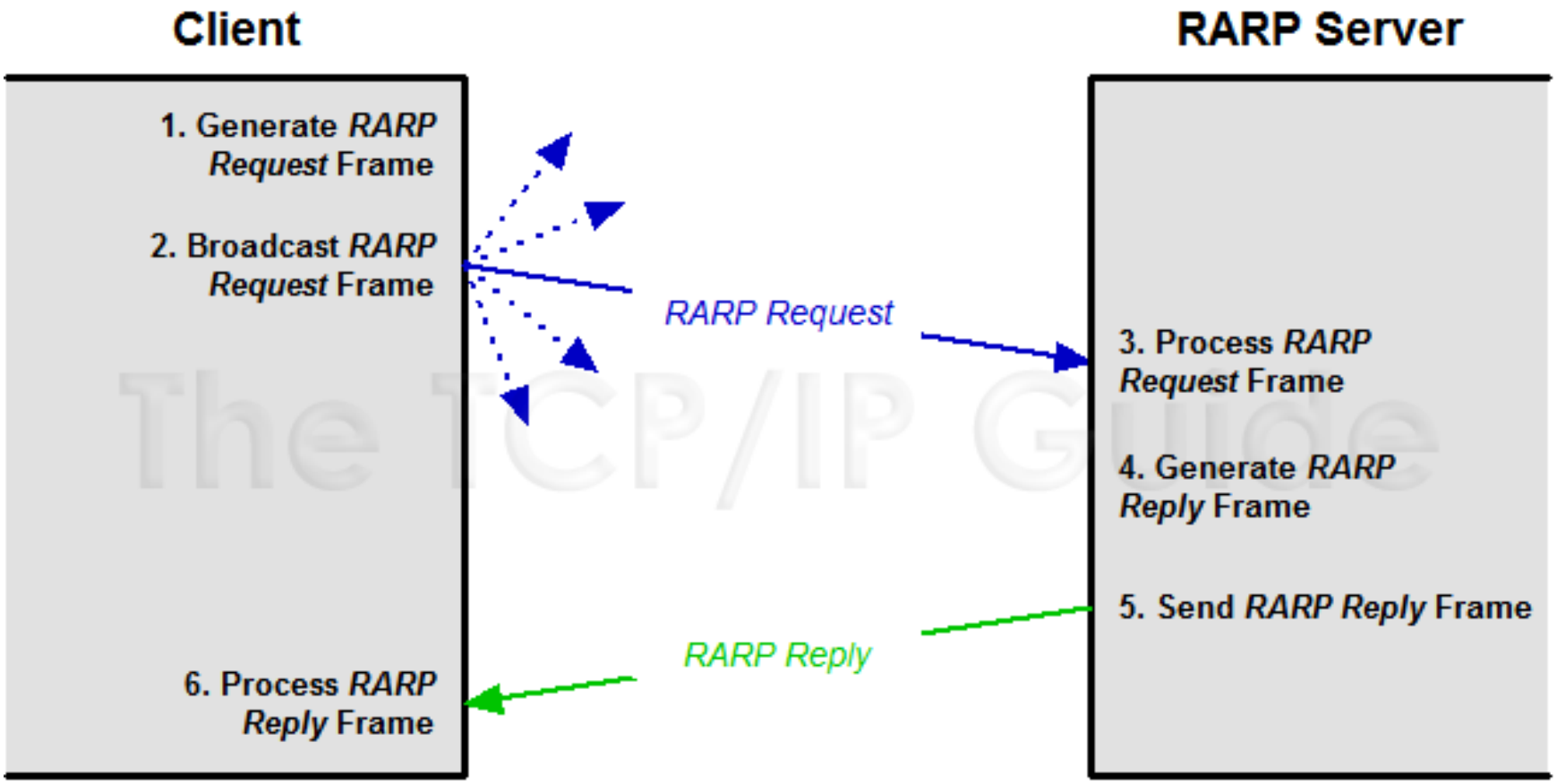
- **ARP:** Router responds to ARP Request from host Argon that arrives on one of its connected networks for the MAC address corresponding to the IP address of its interface on that connected network. Argon realizes that it needs to use router to reach Neon as the two hosts are on different IP networks.
- Router responds with its MAC address and then transfers the datagram to the next segment.



Proxy Arp

- Allow devices on two different IP subnetworks to share a single IP network prefix
  - Source believes destination is on same IP network
- Setup router to respond to the ARP broadcast requests for destinations on different subnet
  - router masquerades as destination for ARP request sent by source on a subnet
  - the two devices are unaware that they are on different subnets, subnet mask indicates that they have the same network prefix.
- Masquerades: router responds to broadcast ARP Request from source host that arrives on one of its connected networks for a destination host that is on one of its other connected networks.

Reverse Address Resolution Protocol (RARP) Operation



RARP uses a simple request/reply exchange to allow a device to obtain an IP address.

Reverse Address Resolution Protocol (RARP) Operation

- **Source Device Generates RARP Request Message:** The source device generates an *RARP Request* message.
- Thus, it uses the value 3 for the *Opcode* in the message.
- It puts its own data link layer address as both the *Sender Hardware Address* and also the *Target Hardware Address*.
- It leaves both the *Sender Protocol Address* and the *Target Protocol Address* blank, since it doesn't know either.
- **Source Device Broadcasts RARP Request**

Reverse Address Resolution Protocol (RARP) Operation

- **Local Devices Process RARP Request Message:** The message is received by each device on the local network and processed. Devices that are not configured to act as RARP servers ignore the message.

- **RARP Server Generates RARP Reply Message:** Any device on the network that is set up to act as an RARP server responds to the broadcast from the source device. It generates an *RARP Reply* using an *Opcode* value of 4.
- It sets the *Sender Hardware Address* and *Sender Protocol Address* to its own hardware and IP address of course, since it is the sender of the reply.

  It then sets the *Target Hardware Address* to the hardware address of the original source device.
- It looks up in a table the hardware address of the source, determines that device's IP address assignment, and puts it into the *Target Protocol Address* field.

Reverse Address Resolution Protocol (RARP) Operation

- **RARP Server Sends RARP Reply Message:** The RARP server sends the *RARP Reply* message unicast to the device looking to be configured.
- **Source Device Processes RARP Reply Message:** The source device processes the reply from the RARP server.

  It then configures itself using the IP address in the *Target Protocol Address* supplied by the RARP server.

Disadvantages of RARP

It uses destination address of all 1s to reach the RARP Server.
Such broadcasts are not forwarded by Routers so RARP server is needed to each network.

It provides only IP Address.

Not used in TCP/IP Version 6.

To get around this problem, bootstrap protocol called BOOTP was invented.

Unlike RARP, it uses UDP messages which are forwarded over routers.

16.1 BOOTP

The Bootstrap Protocol (BOOTP) is a client/server protocol that configures a diskless computer or a computer that is booted for the first time. BOOTP provides the

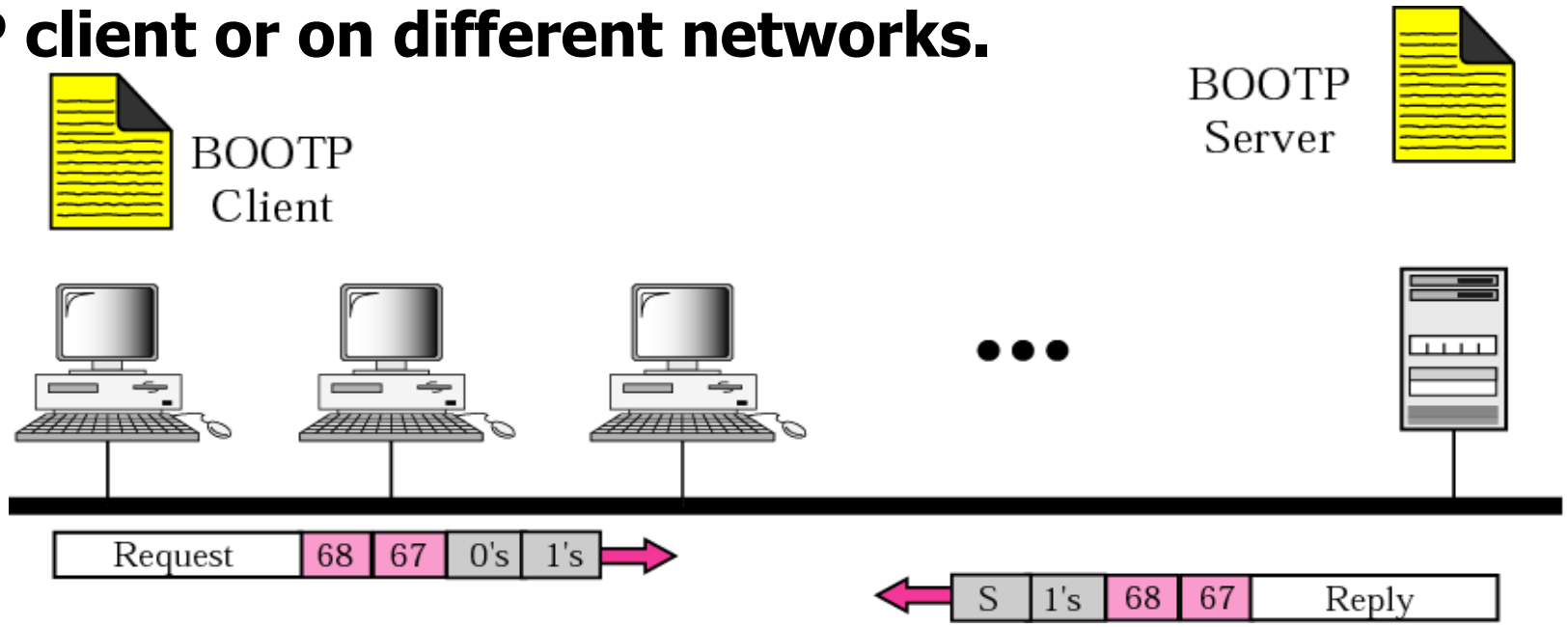
- *IP address*
- *net mask*
- *the address of a default router*
- *the address of a name server.*

A diskless node (or diskless workstation) is a workstation or personal computer without disk drives, which employs network booting to load its operating system from a server. (A computer may also be said to *act as a diskless node*, if its disks are unused and network booting is used.)

BOOTP is static. When a client workstation asks for the above info, it is retrieved from a fixed table. Every time the client asks for the info, it gets the same results.

Figure 16.1 *Client and server on the same network*

e BOOTP server can be on the same network as the OTP client or on different networks.



BOOTP places its packet inside a UDP packet (note that BOOTP is an application layer program).

- **The BOOTP server issues a passive open command on UDP port number 67 and waits for a client.**

- **A booted client issues an active open command on port number 68. The message is encapsulated in a UDP user datagram and then in an IP packet.**

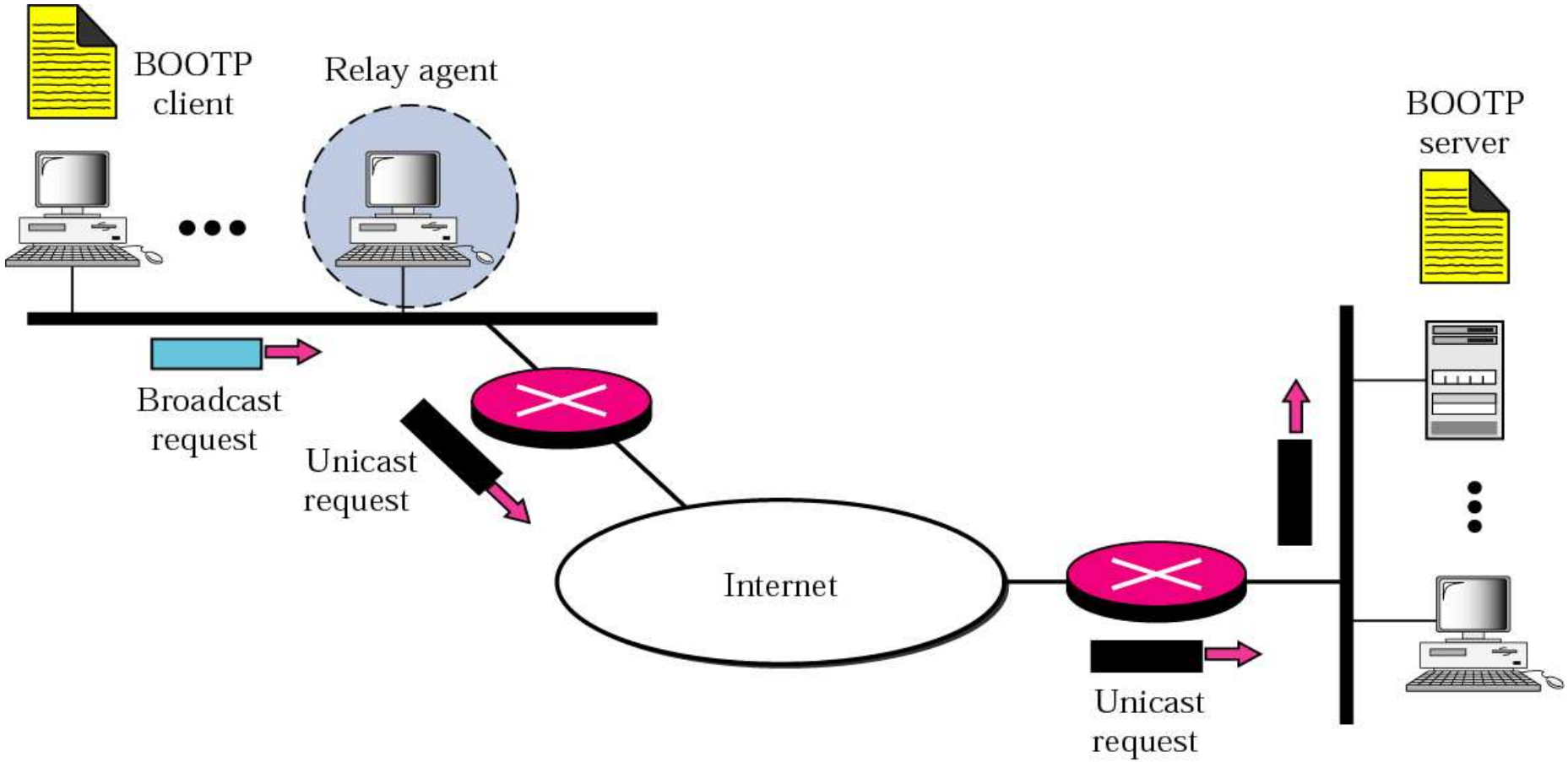
- **In the IP packet, the source address is all 0s and the destination address is all 1s.**

- **Server responds with a UDP datagram source port 67 and destination port 68.**

- **Can also bypass ARP since server also knows the MAC address of the client.**

Client and server on two different networks

When client and server are on different networks, we need a relay agent, because client does not know IP address of server, and a limited broadcast address gets dumped by the local router. Relay agent knows the IP addr of the server.

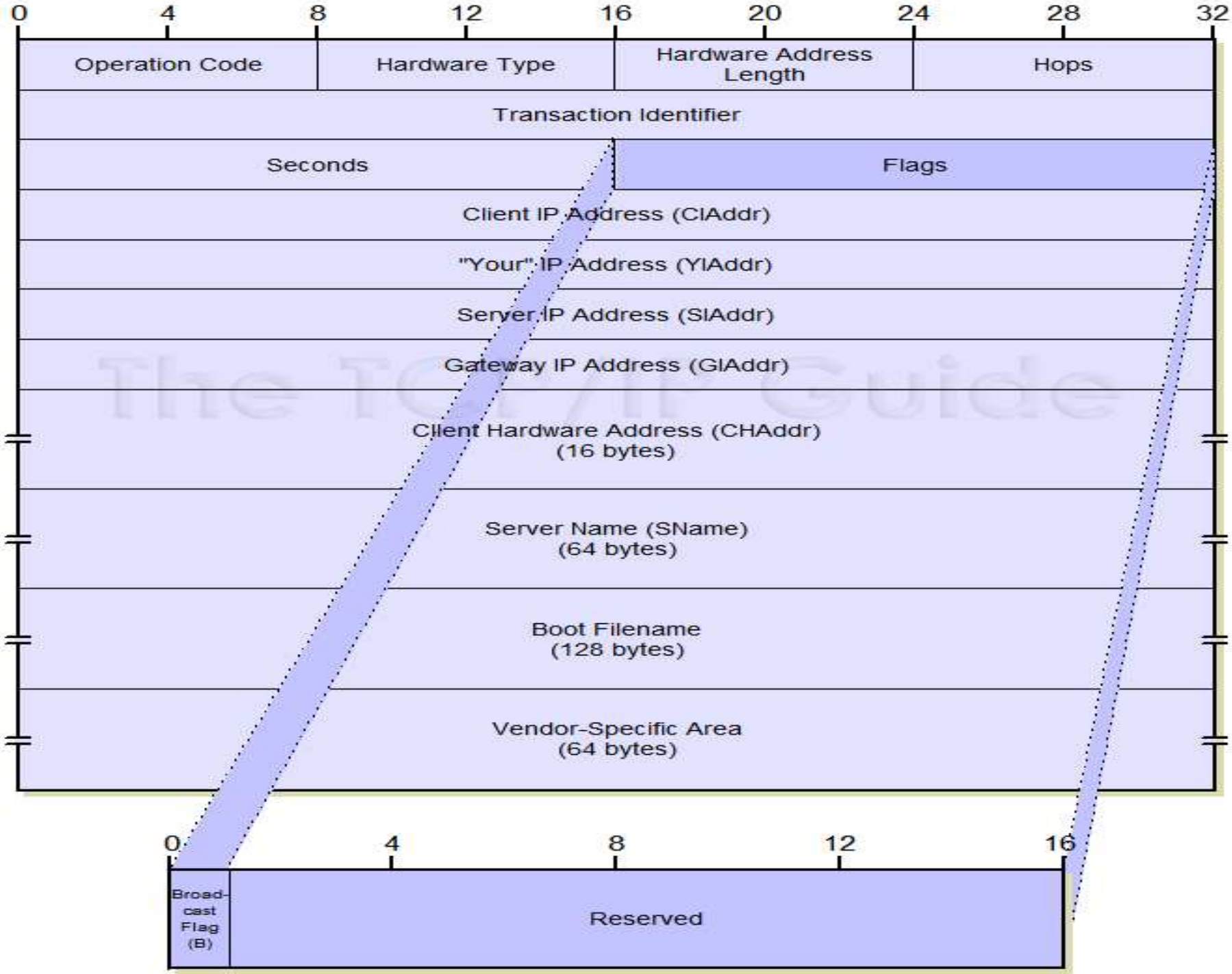


Disadvantages of BOOTP

- It requires manual configuration of tables mapping IP address to Ethernet Address.

- When a new host is added to a LAN, it cannot use BOOTP until an administrator has assigned it an IP Address and entered its (Ethernet address, IP Address) into the BOOTP Configuration tables by hand.

- To eliminate this error prone step, BOOTP was extended and given a new name DHCP(Dynamic Host Configuration Protocol.)


0
4
8
12
16
20
24
28
32
Operation Code
Hardware Type
Hardware Address Length
Hops
Transaction Identifier
Seconds
Flags
Client IP Address (CIAddr)
"Your" IP Address (YIAddr)
Server IP Address (SIAddr)
Gateway IP Address (GIAddr)
Client Hardware Address (CHAddr)
(16 bytes)
Server Name (SName)
(64 bytes)
Boot Filename
(128 bytes)
Vendor-Specific Area
(64 bytes)
0
4
8
12
16
Broad-cast Flag (B)
Reserved

Operation Code: **Specifies the type of message. A value of 1 indicates a request (*BOOTREQUEST* message) while a value of 2 is a reply (*BOOTREPLY* message).**

Hardware Type: This field specifies the type of hardware used for the local network, and is used in exactly the same way as the equivalent field (*HRD*) in the Address Resolution Protocol (ARP) message format. Some of the most common values for this field:

HType **Field Value**	**Hardware Type**
1	Ethernet (10 Mb)
6	IEEE 802 Networks
7	ARCNET
15	Frame Relay
16	Asynchronous Transfer Mode (ATM)
17	HDLC
18	Fibre Channel
19	Asynchronous Transfer Mode (ATM)
20	Serial Line

Hardware Address Length: Specifies how long hardware addresses are in this message. For Ethernet or other networks using IEEE 802 MAC addresses, the value is 6. This too is the same as the field with a similar name (*HLN*) in the ARP field format.

Hops: Set to 0 by a client before transmitting a request and used by BOOTP relay agents to control the forwarding of BOOTP messages.

Transaction Identifier: A 32-bit identification field generated by the client, to allow it to match up the request with replies received from BOOTP servers.

Seconds: According to RFC 951, the client enters into this field the number of seconds "elapsed since [the] client started trying to boot". This is supposed to provide information to BOOTP servers to help them decide which requests to respond to first.

- **Transaction ID :** set by the client and used to match a reply with the request
- **Number of seconds :** indicating the number of seconds elapsed since the time the client started to boot
- **Your IP address :** client address filled by server (in the client message)
- **Server IP address** : in a reply message
- **Gateway IP address** : IP address of a router in a reply message

Flags: In the original BOOTP standard (RFC 951), this was an empty two-byte field. RFC 1542 changed this to a *Flags* field, which at present contains only one flag. The structure of the field is thus as follows:

Subfield Name	Size (bytes)	Description
B	1/8 (1 bit)	***Broadcast Flag:*** A client that doesn't know its own IP address at the time it sends its BOOTP request sets this flag to 1. This serves as an immediate indicator to the BOOTP server or relay agent that receives the request that it definitely should send its reply by broadcast.
Reserved	1 7/8 (15 bits)	***Reserved:*** Set to zero and not used.

Client IP Address: If the client has a current IP address that it plans to keep using, it puts it in this field.

"Your" IP Address: The IP address that the server is assigning to the client. This may be different than the IP address currently used by the client.

Server IP Address: The IP address of the BOOTP server sending a *BOOTREPLY* message.

Gateway IP Address: This field is used to route BOOTP messages when BOOTP relay agents facilitate the communication of BOOTP requests and replies between a client and a server on different subnets or networks. To understand the name, remember that the old TCP/IP term for "router" is "gateway"; BOOTP relay agents are typically routers.

Client Hardware Address: The hardware (layer two) address of the client sending a *BOOTREPLY*.

Server Name: The server sending a *BOOTREPLY* may optionally put its name in this field.
This can be a simple text "nickname" or a fully-qualified DNS domain name (such as "myserver.organization.org").

Boot Filename: Contains the full directory path and file name of a boot file that can be downloaded by the client.

Vendor-Specific Area: Originally created to allow vendors to customize BOOTP to the needs of different types of hardware, this field is now also used to hold additional vendor-independent configuration information.

- **Server name :** the domain name of the server in a reply packet
- **Boot filename :** the full pathname of the boot in a reply packet (128-byte)
- **Options** : used in a reply message (64-byte)
  - carrying either additional information (such as the network mask or default router address) or some specific vendor information

Table 16.1 Options for BOOTP

Description	*Tag*	*Length*	*Value*
Padding	0		
Subnet mask	1	4	Subnet mask
Time offset	2	4	Time of the day
Default routers	3	Variable	IP addresses
Time servers	4	Variable	IP addresses
DNS servers	6	Variable	IP addresses
Print servers	9	Variable	IP addresses
Host name	12	Variable	DNS name
Boot file size	13	2	Integer
Vendor specific	128–254	Variable	Specific information
End of list	255		

16.2 DHCP

The Dynamic Host Configuration Protocol (DHCP) provides static and dynamic address allocation that can be manual or automatic.

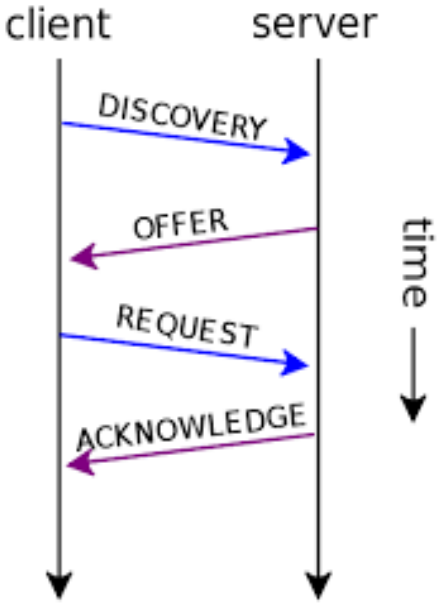
The topics discussed in this section include:

Static Address Allocation
Dynamic Address Allocation
Manual and Automatic Configuration
Packet Format
Transition States
Exchanging Messages

DHCP

The **DHCP** client requests an IP address by broadcasting a DHCPDiscover message to the local subnet.
The client is offered an address when a **DHCP** server responds with a DHCPOffer message containing an IP address and configuration information for lease to the client.

DHCP client starts by broadcasting the DHCP DISCOVER packet.
The broadcast is received by the DHCP Server(s), which in turn replies with the DHCP OFFER message.
The DHCP OFFER message contains the IP address offered by the server and the time period for which the IP address is allocated [The IP address provided may be random, or may be based on some policy configured by the admin].



The DHCP client may receive multiple DHCP OFFER messages, however it chooses only one DHCP OFFER message based on the policy configured in the DHCP Client. Usually its on the first come first serve basis. However we can configure the DHCP Client to choose the DHCP OFFER having the longest lease time or some preferred subnet [If IP from different subnet are offered]. The DHCP client now replies with the DHCP REQUEST message.

The DHCP REQUEST message is a broadcast message. When other DHCP servers receive this message, they withdraw any offers that they might have made to the client and return the offered address to the pool of available addresses. The intended DHCP server on receiving the message

DHCP basics

Bootp is static, but DHCP is dynamic (but it can also be static).

DHCP has a pool of available addresses. When a request arrives, DHCP pulls out the next available address and assigns it to the client for a negotiable time period.

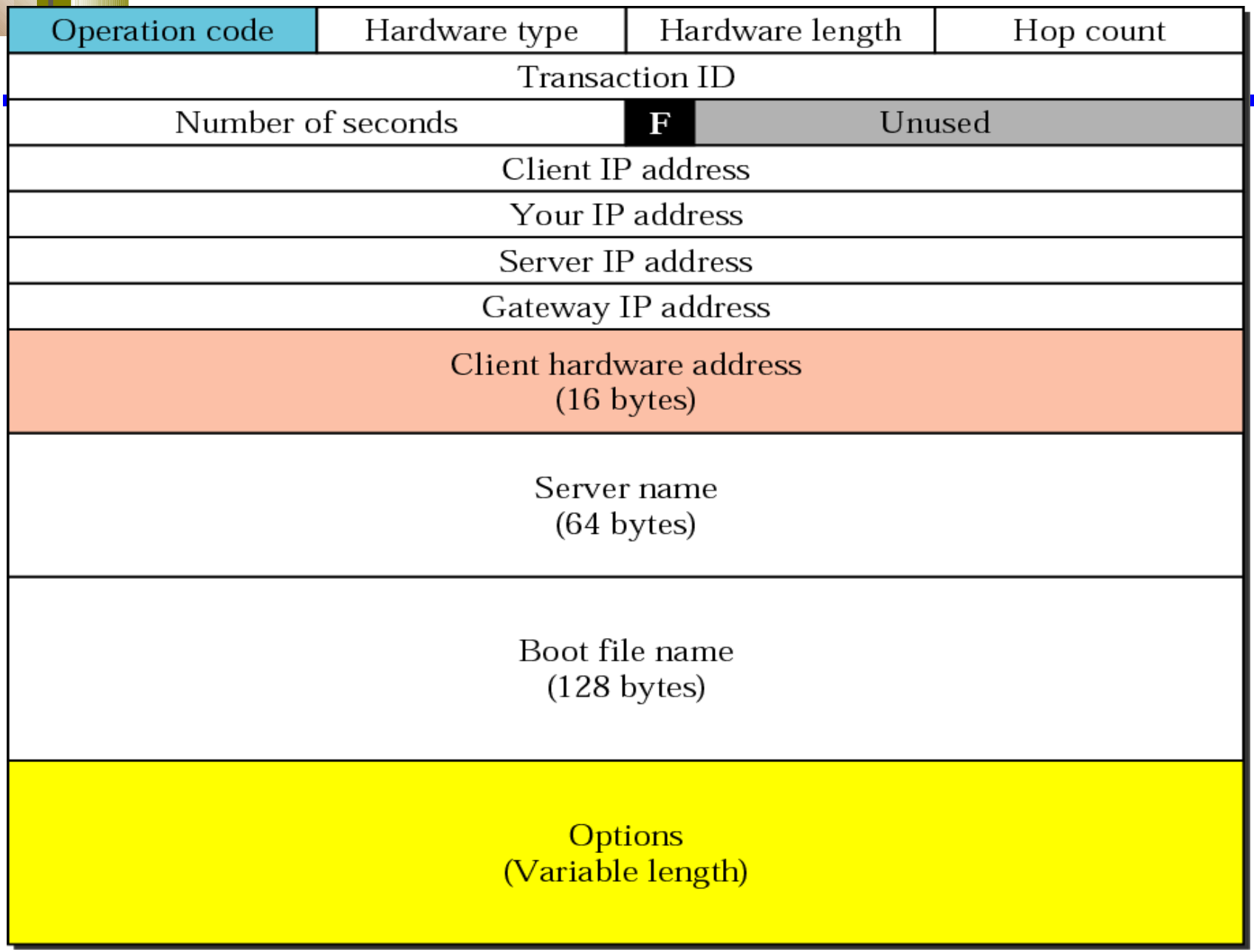
When a request comes in from a client, the DHCP server first consults the static table.

DHCP is great when devices and IP addresses change.

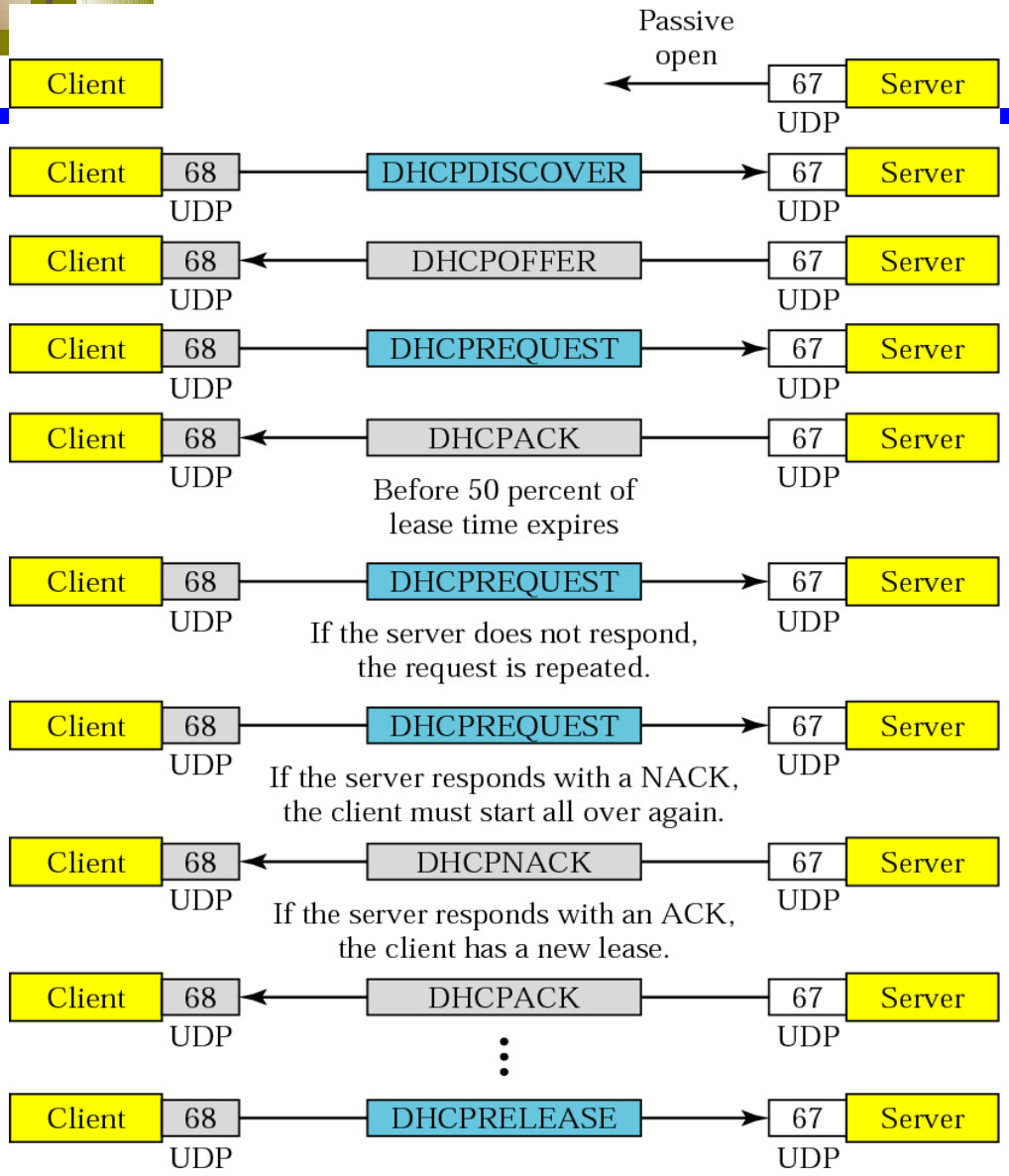
The DHCP packet format is almost identical to the BOOTP packet format (in order to be compatible with BOOTP).

Only difference is 1-bit flag.

DHCP packet


Operation code
Hardware type
Hardware length
Hop count
Transaction ID
Number of seconds
F
Unused
Client IP address
Your IP address
Server IP address
Gateway IP address
Client hardware address
(16 bytes)
Server name
(64 bytes)
Boot file name
(128 bytes)
Options
(Variable length)

Exchanging messages



Discover: client tries to find out what servers are out there.

Offer: those servers that can provide this service respond

Request: client selects one offer and makes a request

ACK: server acks the request

When 50% of the lease period is expired, client asks for a renewal.

If ACK received, reset timer. If NAK, go back to intializing state.

Unicast Routing

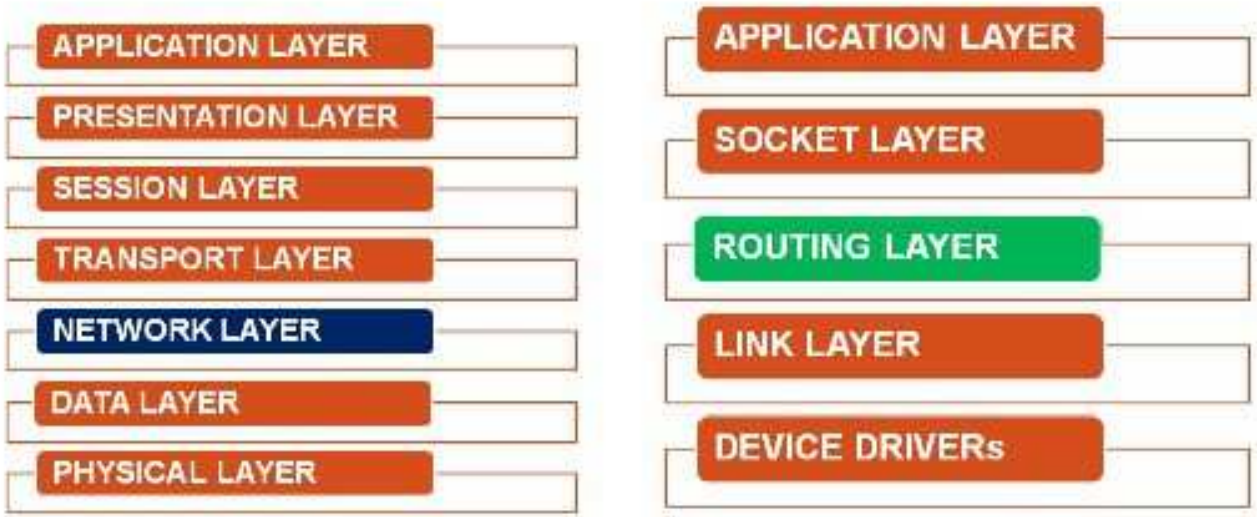
What is routing?

- **Routing** is the process of selecting a path in a network along which the packets shall be sent to a destination

- **Routing** consists of
  - A Router
  - A set of routing protocols
  - A routing information base (RIB)
  - One or more routing algorithms

At which layer is routing done?

- Generally routing is done at **network layer**
- **Multilayer layer routing** and **Cross layer routing** is also prevalent nowadays
- **Firewalls** are often integrated with routers



Why is routing required?

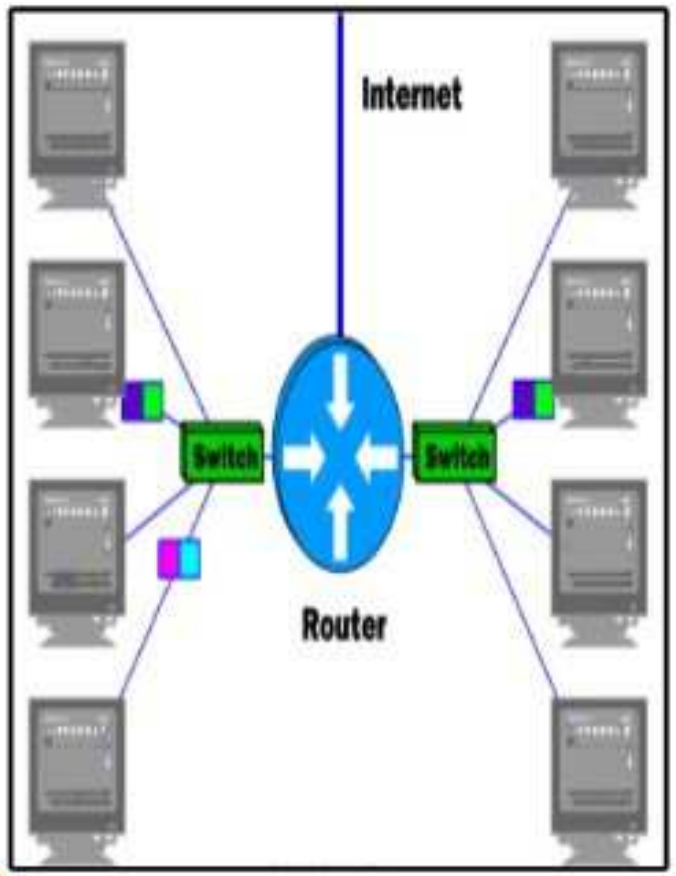
- For **practical limitation** of **physical connections**
- For efficiently managing the **network traffic**
- For **efficient usage** of network resources
- For catering to different **types of services**
- For **congestion control**

Router, Switch and Hub

- The basic difference is varying ***intelligence***

HUB	• Least expensive and complicated. No intelligence • Just directs incoming packets from one port to other
SWITCH	• More expensive and intelligent • Knows which port is carrying the traffic from which host/interface
ROUTER	• Most expensive and intelligent, Most complicated • Learns about its neighboring conditions, manipulates data traffic

How does a router work?



One

- Packet arrives at the router for ***delivery***
- Router reads **destination *IP address***

Two

- Router searches '***Routing Table***'
- Determines **next hop** of the packet

Three

- Router ***forwards*** the packet to next hop
- Packet is said to be 'routed'

Delivery

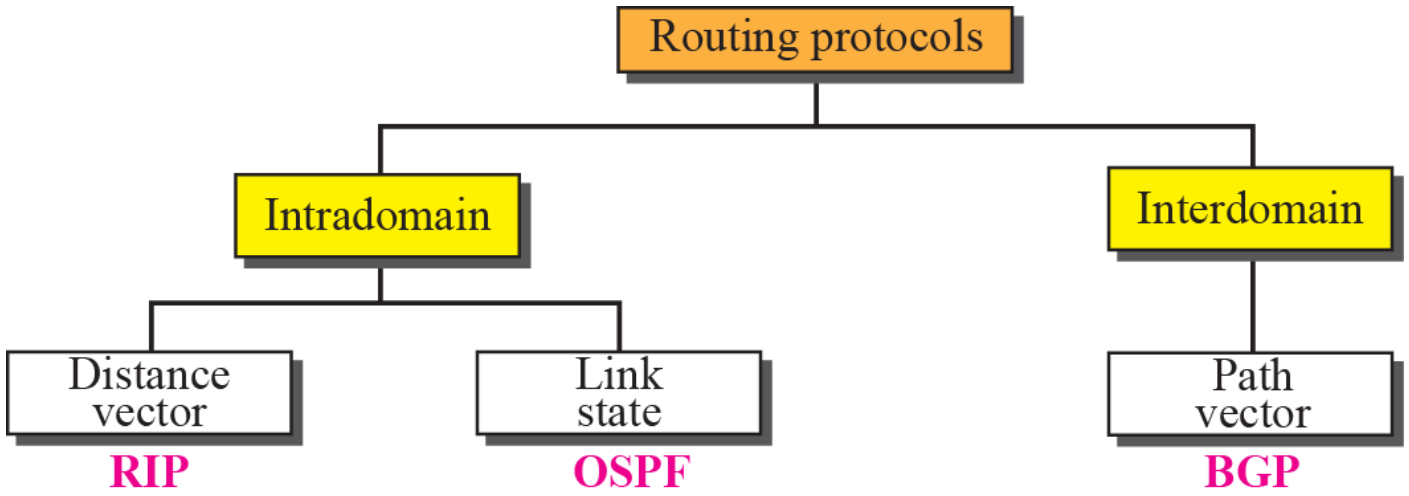
- The **network layer** supervises the handling of packets by the underlying **physical network**

- Every packet undergoes at least one **“Direct Delivery”** and one or more **“Indirect Delivery”**

Routing

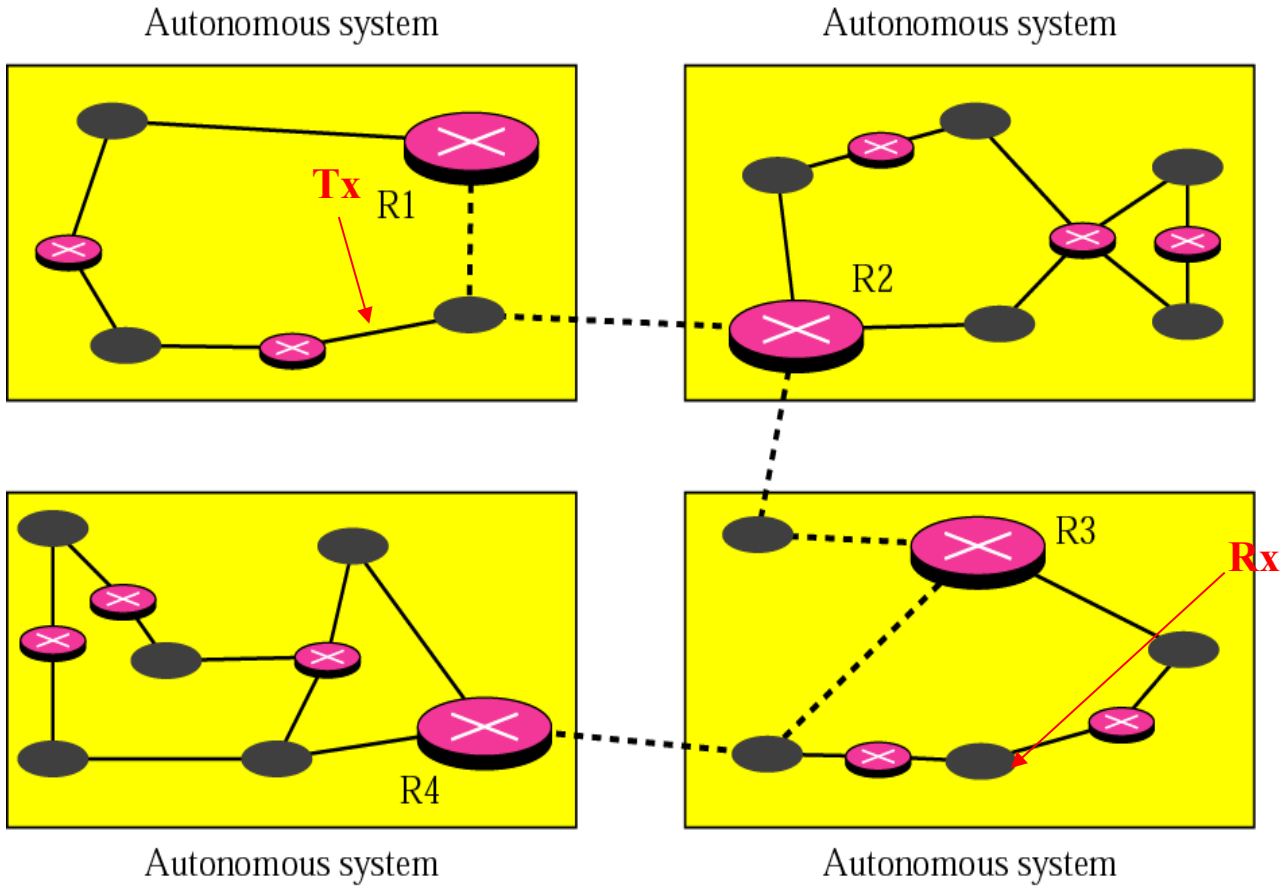
- Routing is the process of transferring data across an internetwork from a source host to a destination host.
- Routing can be understood in terms of two processes: host routing and router routing .
- Host routing occurs when the sending host forwards a packet. Based on the destination network address, the sending host must decide whether to forward the packet to the destination or to a router.
- Router routing occurs when a router receives a packet that is to be forwarded. The packet is forwarded between routers (when the destination network is not directly attached to the router) or between a router and the destination host (when the destination

Popular routing protocols



- **RIP – Routing Information Protocol – treats each network the same (assigns the same cost for each network)**
- **OSPF – Open Shortest Path First protocol – assigns a cost for passing through a network based on the type of service required – routes through the network can have different cost – each router would have several tables**
- **BGP – Border Gateway Protocol – is an exterior routing protocol that uses a policy that defines what paths should be chosen**

Autonomous System



Routing Protocols vs. Algorithms

- **Routing protocols** allow routers to share info with one another dynamically - as the Internet makes changes, the routing protocols allow routers to inform other routers
  - Routers communicate to their neighboring routers
  - Routing protocols implement the procedures for combining info received from other routers

- **Routing Algorithms** – decision making analysis – the "brains" – using the info provided

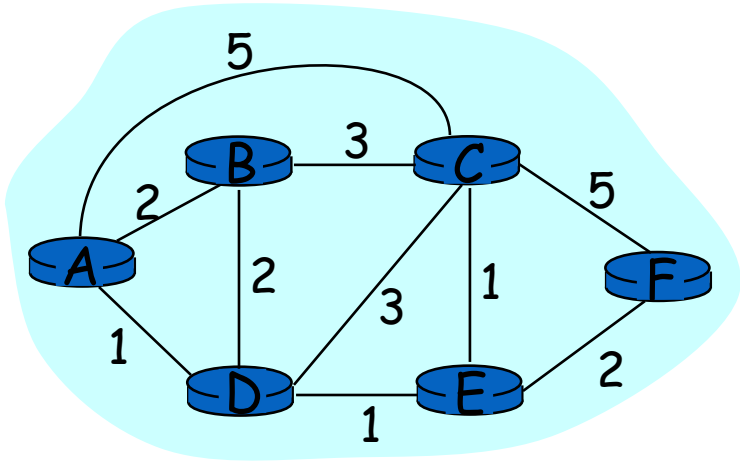
Routing

Routing protocol

Goal: determine "good" path (sequence of routers) thru network from source to dest.

Graph abstraction for routing algorithms:

- graph nodes are routers
- graph edges are physical links
  - link cost: delay, $ cost, or congestion level



- "good" path:
  - typically means minimum cost path
  - other def's possible

Routing Algorithm classification

Global or decentralized information?

Global:

- all routers have complete topology, link cost info
- “link state” algorithms

Decentralized:

- router knows physically-connected neighbors, link costs to neighbors
- iterative process of computation, exchange of info with neighbors
- “distance vector” algorithms

Static or dynamic?

Static:

- routes change slowly over time

Dynamic:

- routes change more quickly
  - periodic update
  - in response to link cost changes

Unicast Routing Protocol

- Unicast routing is a process that enable sender to send an unicast IP packets to the destination node.
- 1 router or more intermediate routers may be used, depending to the destination of the node. (Figure 1)
- Unicast routing protocol is a set of rules of forwarding unicast traffic from a source to a destination on an internetwork.

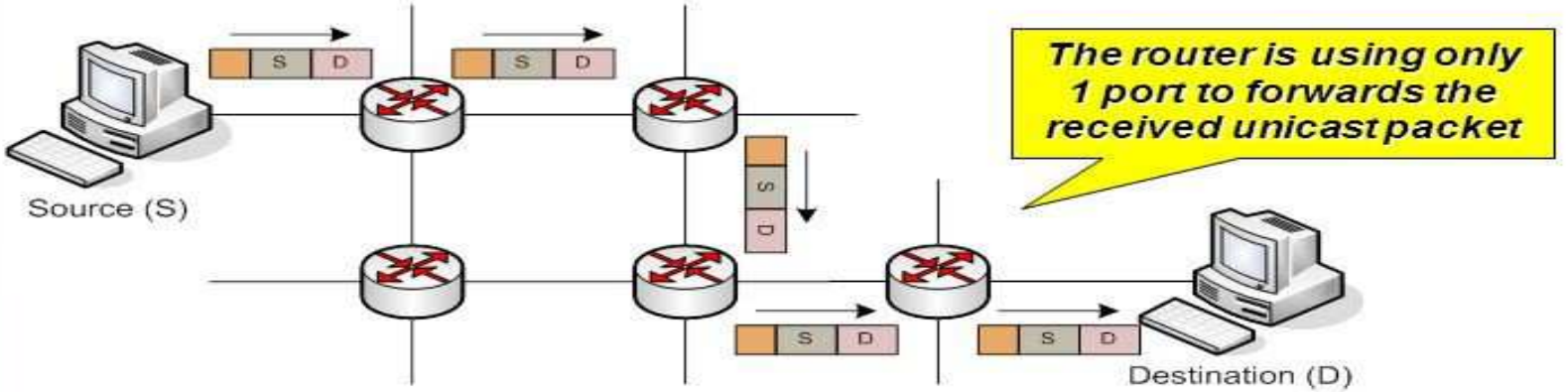
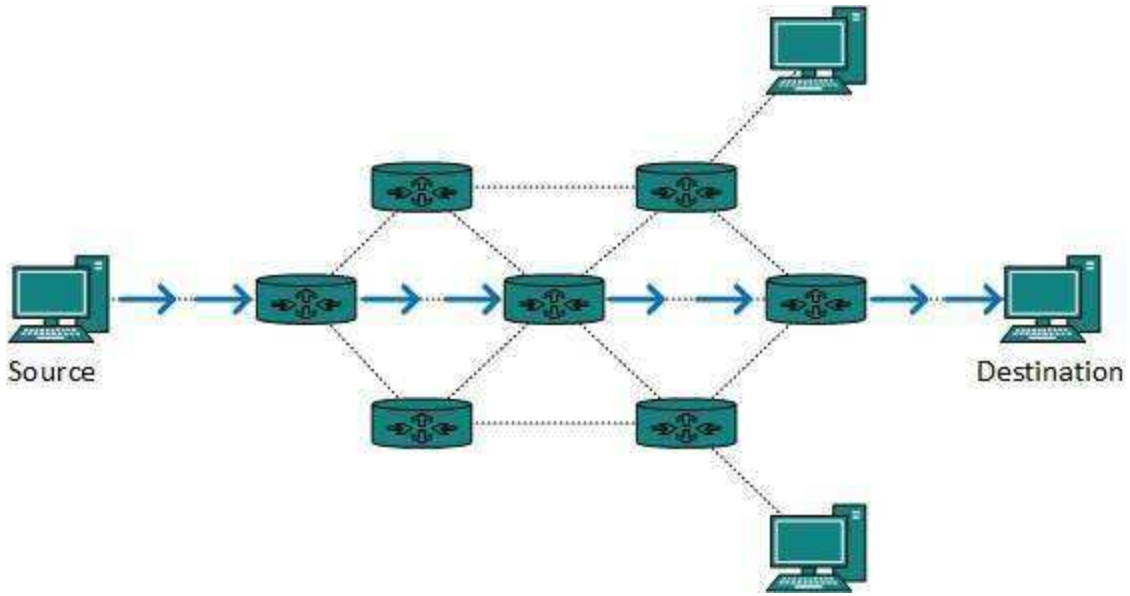


Fig. 1. Unicast Routing

- Unicast routing is the process of forwarding unicasted traffic from a source to a destination on an internetwork.
- Unicasted traffic is destined for a unique address.

Routing unicast data over the internet is called **unicast routing**. It is the simplest form of **routing** because the destination is already known. Hence the **router** just has to look up the **routing** table and **forward** the packet to next hop



Broadcast routing

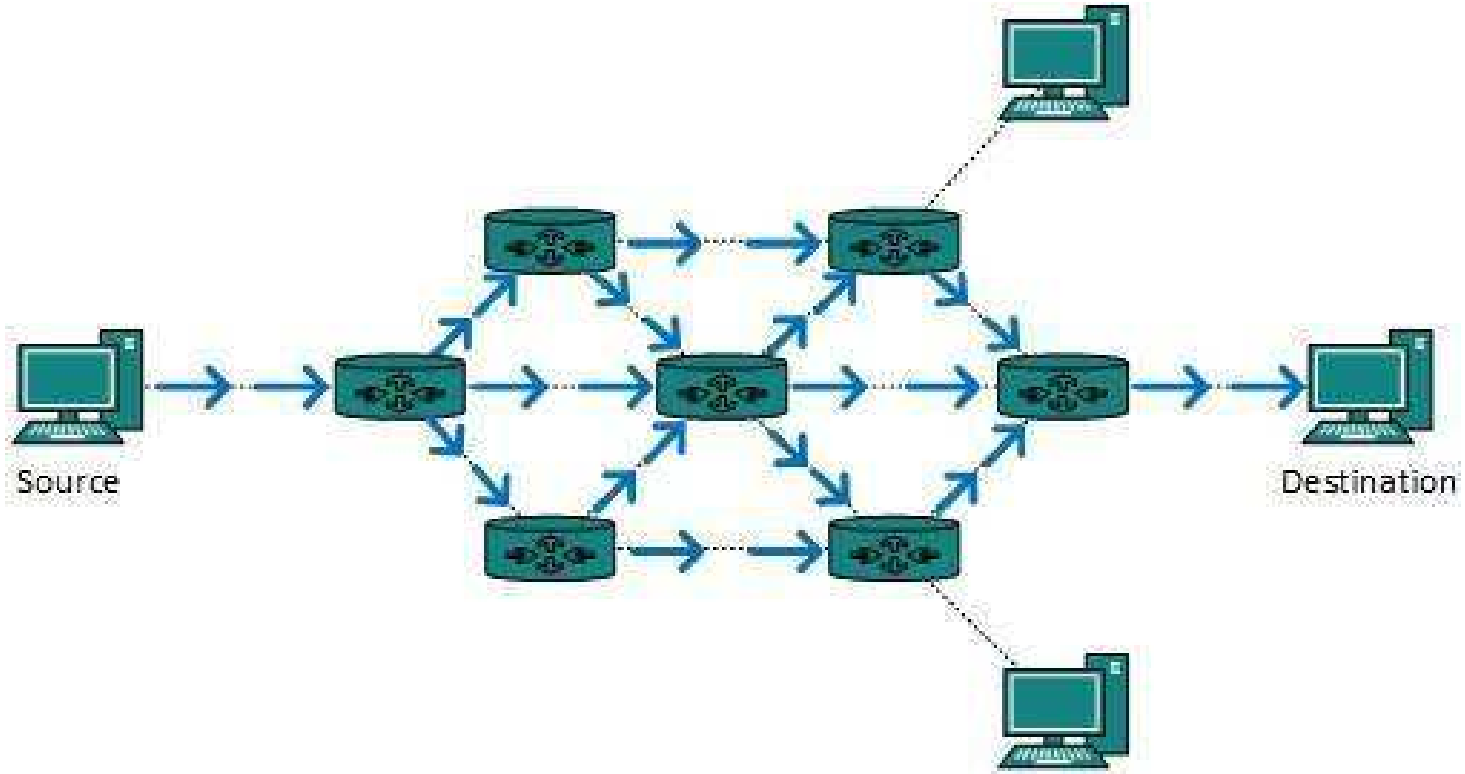
By default, the broadcast packets are not routed and forwarded by the routers on any network. Routers create broadcast domains. But it can be configured to forward broadcasts in some special cases. A broadcast message is destined to all network devices.

Broadcast routing can be done in two ways (algorithm):

- A router creates a data packet and then sends it to each host one by one. In this case, the router creates multiple copies of single data packet with different destination addresses. All packets are sent as unicast but because they are sent to all, it simulates as if router is broadcasting.

- This method consumes lots of bandwidth and router must destination address of each node.

- Secondly, when router receives a packet that is to be broadcasted, it simply floods those packets out of all interfaces. All routers are configured in the same way.

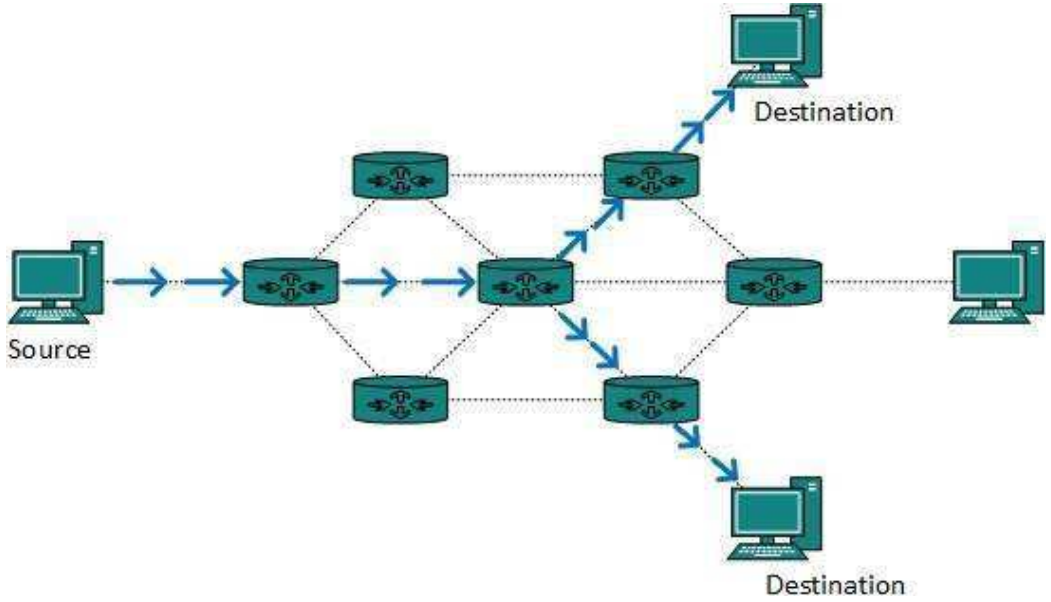


- This method is easy on router's CPU but may cause the problem of duplicate packets received from peer routers.
- Reverse path forwarding is a technique, in which router knows in advance about its predecessor from where it should receive broadcast. This technique is used to detect and discard duplicates.

Multicast Routing

Multicast routing is special case of broadcast routing with significance difference and challenges. In broadcast routing, packets are sent to all nodes even if they do not want it. But in Multicast routing, the data is sent to only nodes which wants to receive the packets.

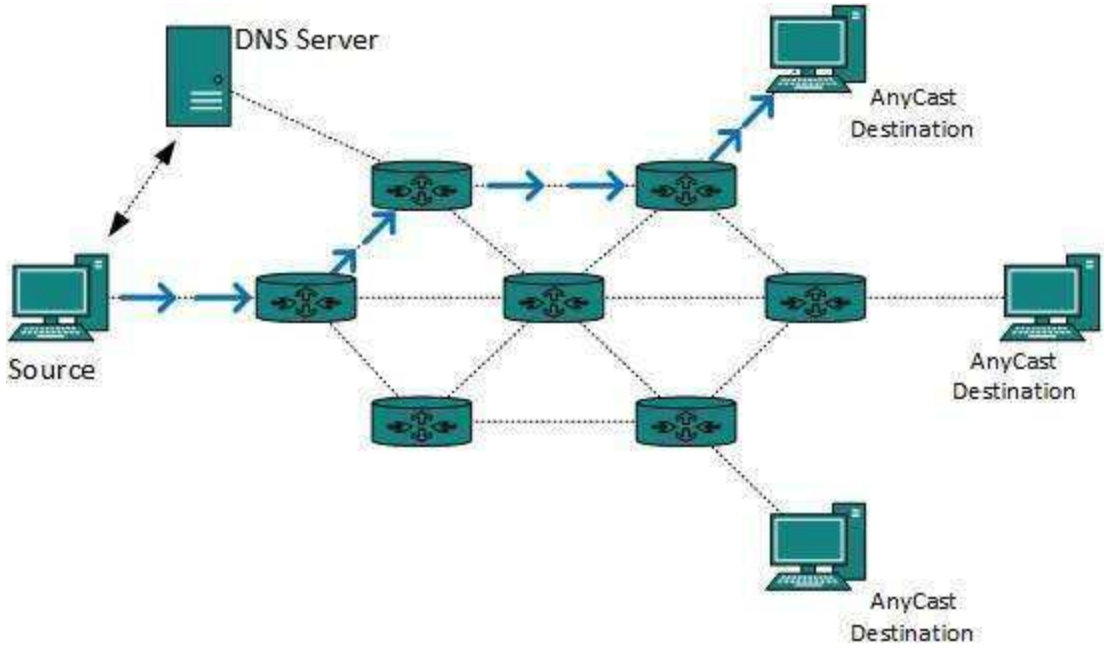
- The router must know that there are nodes, which wish to receive multicast packets (or stream) then only it should forward. Multicast routing works spanning tree protocol to avoid looping.
- Multicast routing also uses reverse path Forwarding technique, to detect and discard duplicates and loops.



Anycast Routing

Anycast packet forwarding is a mechanism where multiple hosts can have same logical address. When a packet destined to this logical address is received, it is sent to the host which is nearest in routing topology.

- Anycast routing is done with help of DNS server. Whenever an Anycast packet is received it is enquired with DNS to where to send it.
- DNS provides the IP address which is the nearest IP configured on it.



Unicast Routing Protocols

There are two kinds of routing protocols available to route unicast packets:

Distance Vector Routing Protocol

- Distance Vector is simple routing protocol which takes routing decision on the number of hops between source and destination. A route with less number of hops is considered as the best route. Every router advertises its set best routes to other routers. Ultimately, all routers build up their network topology based on the advertisements of their peer routers,

For example Routing Information Protocol (RIP).

Link State Routing Protocol

- Link State protocol is slightly complicated protocol than Distance Vector. It takes into account the states of links of all the routers in a network. This technique helps routes build a common graph of the entire network. All routers then calculate their best path for routing purposes. For example, Open Shortest Path First (OSPF) and Intermediate System to Intermediate System (ISIS).

Distance Vector Routing

- Distance Vector Technology
  - **The Meaning of Distance Vector**:
    - A router using distance vector routing protocols knows 2 things:
      - Distance to final destination
      - Vector, or direction, traffic should be directed

Key thing -- each node knows the cost of links to its neighbors.
If no link exists between two nodes, the cost of a direct link between the nodes is “infinity”.

Distance Vector Routing: overview

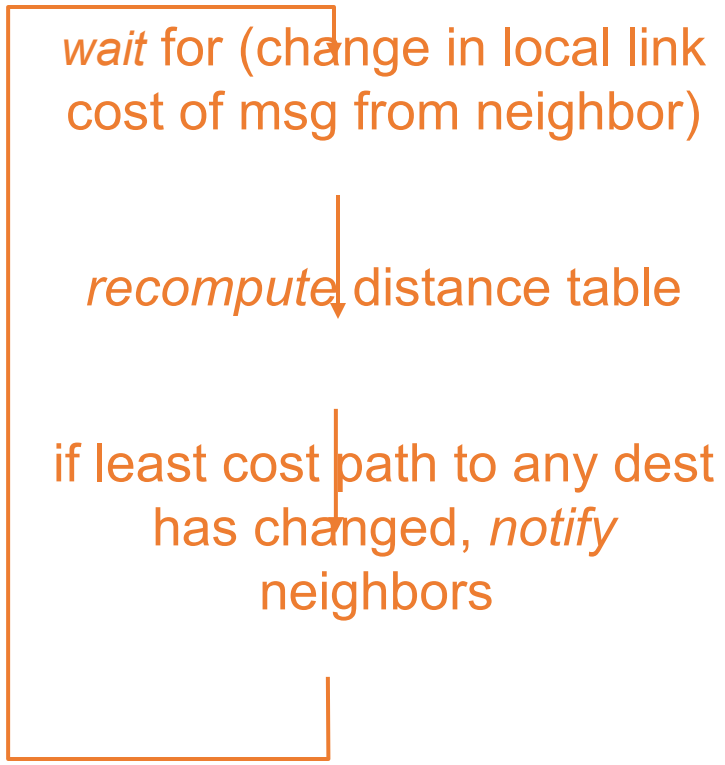
Iterative, asynchronous:
each local iteration caused by:

- local link cost change
- message from neighbor: its least cost path change from neighbor

Distributed:

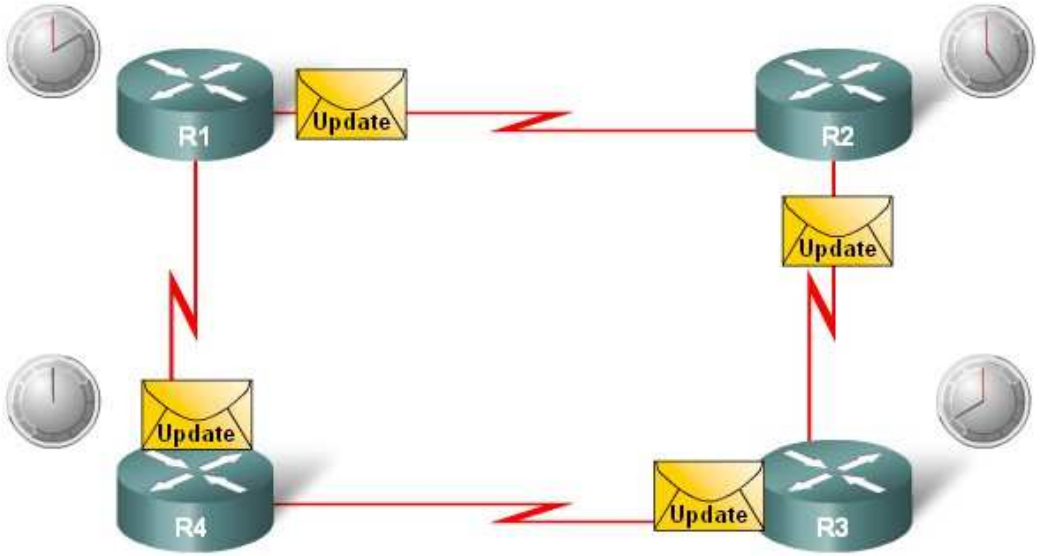
- each node notifies neighbors *only* when its least cost path to any destination changes
  - neighbors then notify their neighbors if necessary

Each node:



Characteristics of Distance Vector routing protocols:

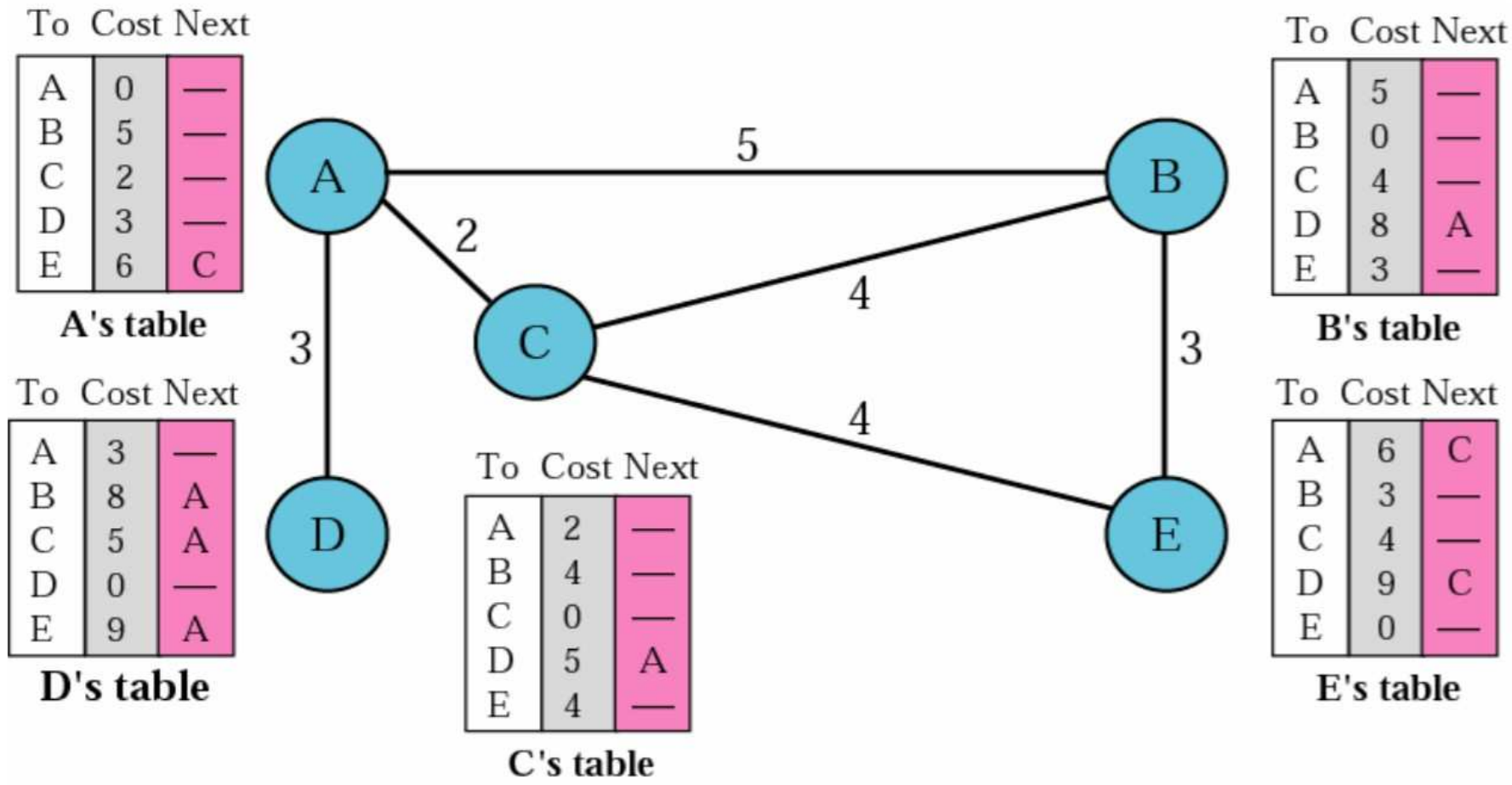
- Periodic updates
- Neighbors
- Broadcast updates
- Entire routing table is included with routing update



Characteristics of Distance Vector Routing

- **Periodic Updates:** Updates to the routing tables are sent at the end of a certain time period. A typical value is 90 seconds.
- **Triggered Updates:** If a metric changes on a link, a router immediately sends out an update without waiting for the end of the update period.
- **Full Routing Table Update**: Most distance vector routing protocol send their neighbors the entire routing table (not only entries which change).
- **Route invalidation timers:** Routing table entries are invalid if they are not refreshed. A typical value is to invalidate an entry if no update is received after 3-6 update periods.

Example

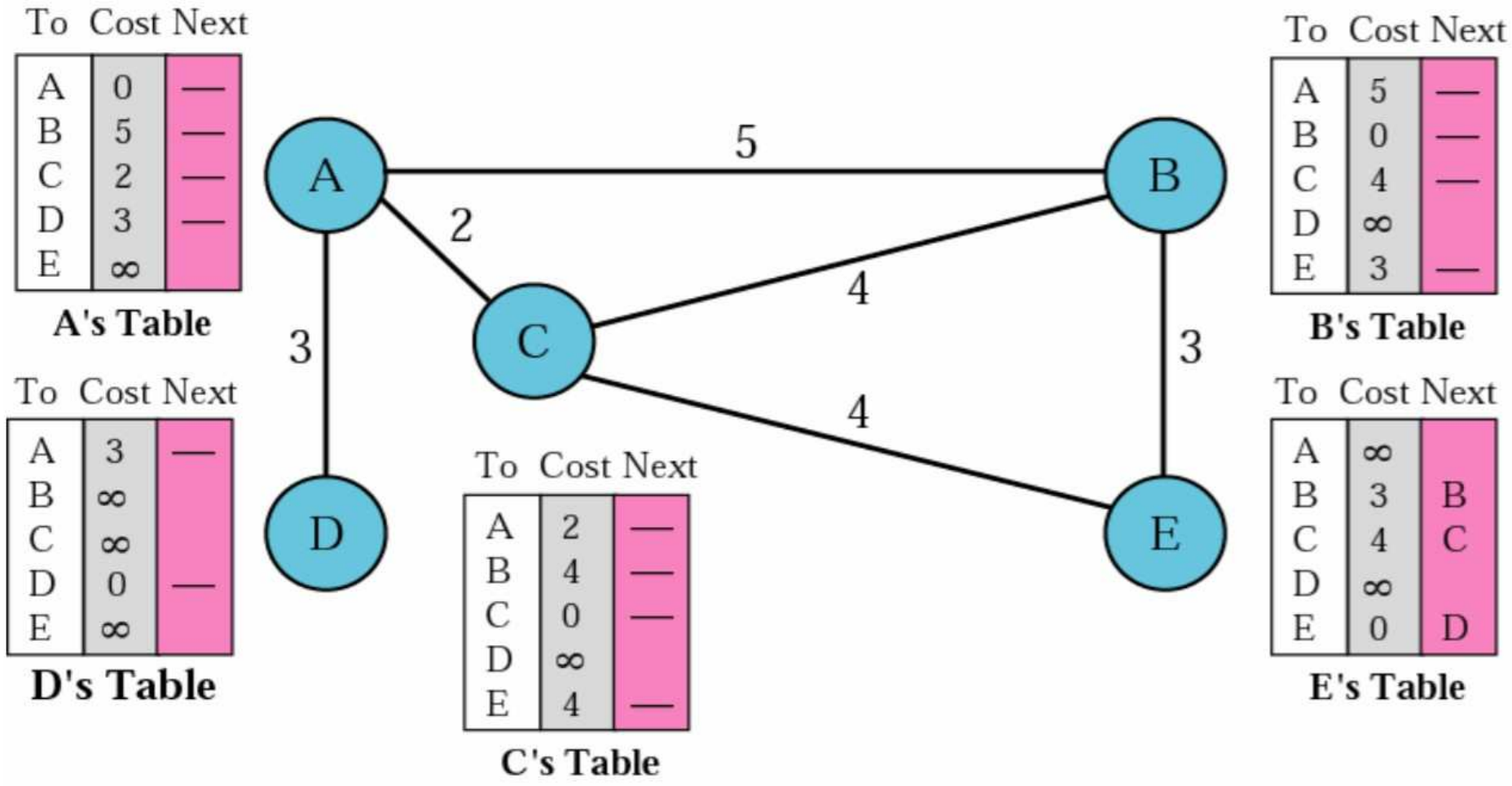

To Cost Next
A 0 —
B 5 —
C 2 —
D 3 —
E 6 C
A's table
To Cost Next
A 5 —
B 0 —
C 4 —
D 8 A
E 3 —
B's table
To Cost Next
A 3 —
B 8 A
C 5 A
D 0 —
E 9 A
D's table
To Cost Next
A 2 —
B 4 —
C 0 —
D 5 A
E 4 —
C's table
To Cost Next
A 6 C
B 3 —
C 4 —
D 9 C
E 0 —
E's table
A
B
C
D
E
5
2
4
3
3
4

Step1: Initialization

Initialization

- At the beginning n Each node can know only the distance between itself and its immediate neighbors.
- We assume each node can send a message to the immediate neighbors and find the distance.

Initialization of Tables in Distance Vector Routing



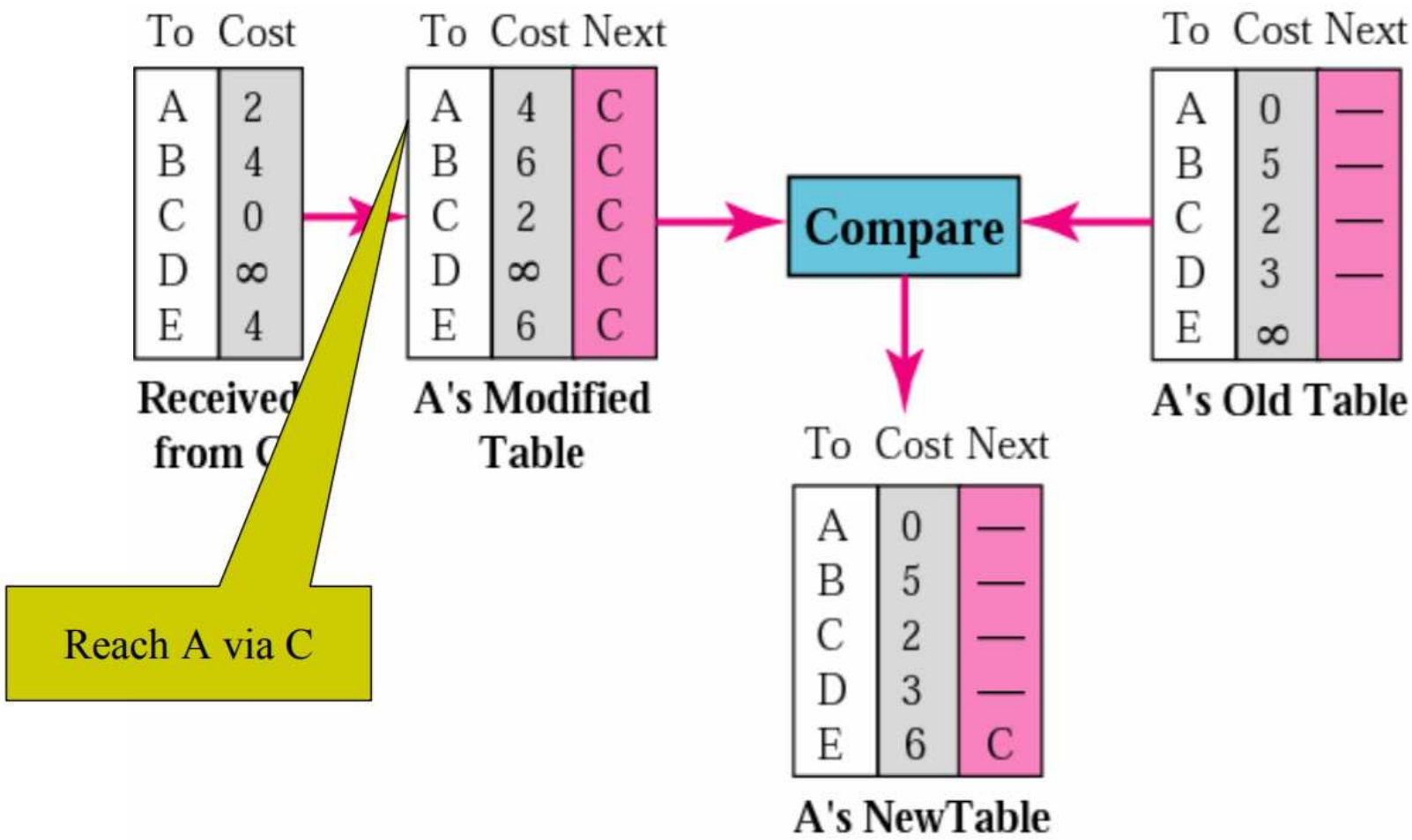
Sharing

- Idea of distance vector routing
  - Sharing of information between neighbors
  - In distance vector routing, each node shares its routing table with its immediate neighbors periodically and when there is a change
- How much of the table must be shared ?
- Send the entire table but contains only the first two columns
  - The third column must be changed

Updating

- Receipt: a two-column table from a neighbor
- Add the cost between itself and the sending node to each value in the second column
- Repeat the following steps for each advertised destination
  - If (destination not in the routing table)
    - Add the advertised information to the table
  - Else
    - If (next-hop field is the same)
      - Replace retry in the table with the new advertised one
    - Else
      - If (advertised hop count smaller than one in the table)

Updating in Distance Vector Routing



When to Share

- The table is sent both ***periodically*** and when there is a ***change*** in the table
- ***Periodic update***
  - A node sends its routing table in a periodic update
  - Normally every 30 seconds
- ***Triggered update***
  - A node receives a table from a neighbor resulting in changes in its own table
  - A node detects some failure in the neighboring links which results in a distance change to infinity

Two-Node Loop Instability

- A problem with distance vector routing is ***instability***
  - A network using this protocol can become ***unstable***
- See the following table
  - 1. both node A and B know how to reach node X
  - 2. the link between A and X fails
    - Node A change its table
  - 3a. If node A can send its routing table to B immediately
    - Everything is fine
  - 3b. However, if node B sends its routing table to A first
    - Node A assumes that B has found a way to reach X
  - 4. A sends its new update to B and B also update its routing table
  - 5. B sends its new update to A and so on…***until the cost reach infinity***
  - 6. ***Then both A and B knows that the link is broken***

Unicast Routing Protocol: RIP

- RIP is a simple vector routing protocol, the routers exchange network reachability information with their nearest neighbours.
- Routing Information Protocol (RIP) is a dynamic protocol used to find the best route or path from end-to-end (source to destination) over a network by using a routing metric/hop count algorithm.
- This algorithm is used to determine the shortest path from the source to destination, which allows the data to be delivered at high speed in the shortest time

RIP - History

- Late 1960s : Distance Vector protocols were used in the ARPANET
- Mid-1970s: XNS (Xerox Network system) routing protocol is the precursor of RIP in IP (and Novell's IPX RIP and Apple's routing protocol)
- 1982 Release of **routed** for BSD Unix
- 1988 RIPv1 (RFC 1058)
  - classful routing
- 1993 RIPv2 (RFC 1388)
  - adds subnet masks with each route entry
  - allows classless routing

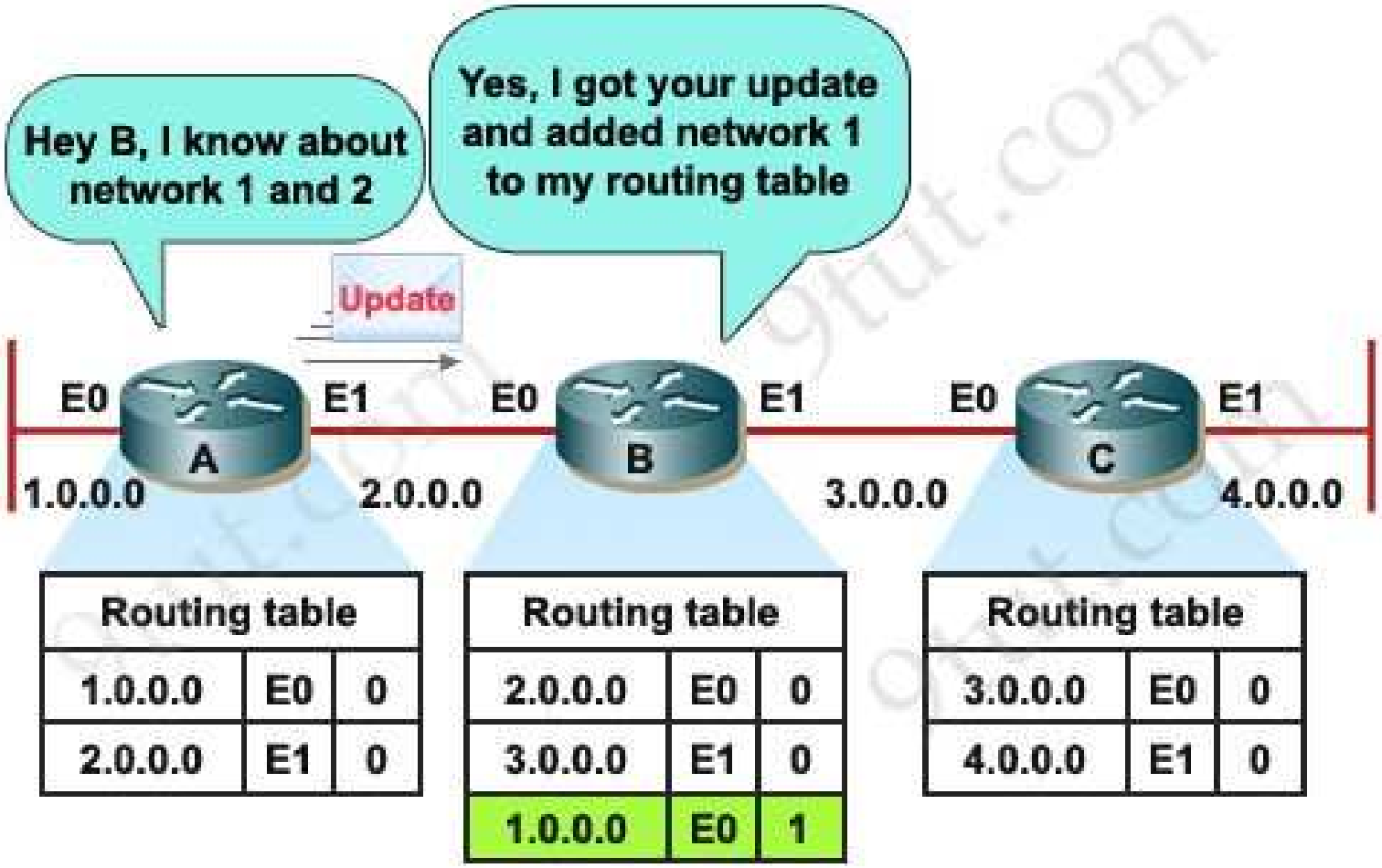
RIP - History

- Late 1960s : Distance Vector protocols were used in the ARPANET
- Mid-1970s: XNS (Xerox Network system) routing protocol is the precursor of RIP in IP (and Novell's IPX RIP and Apple's routing protocol)
- 1982 Release of **routed** for BSD Unix
- 1988 RIPv1 (RFC 1058)
  - classful routing
- 1993 RIPv2 (RFC 1388)
  - adds subnet masks with each route entry
  - allows classless routing

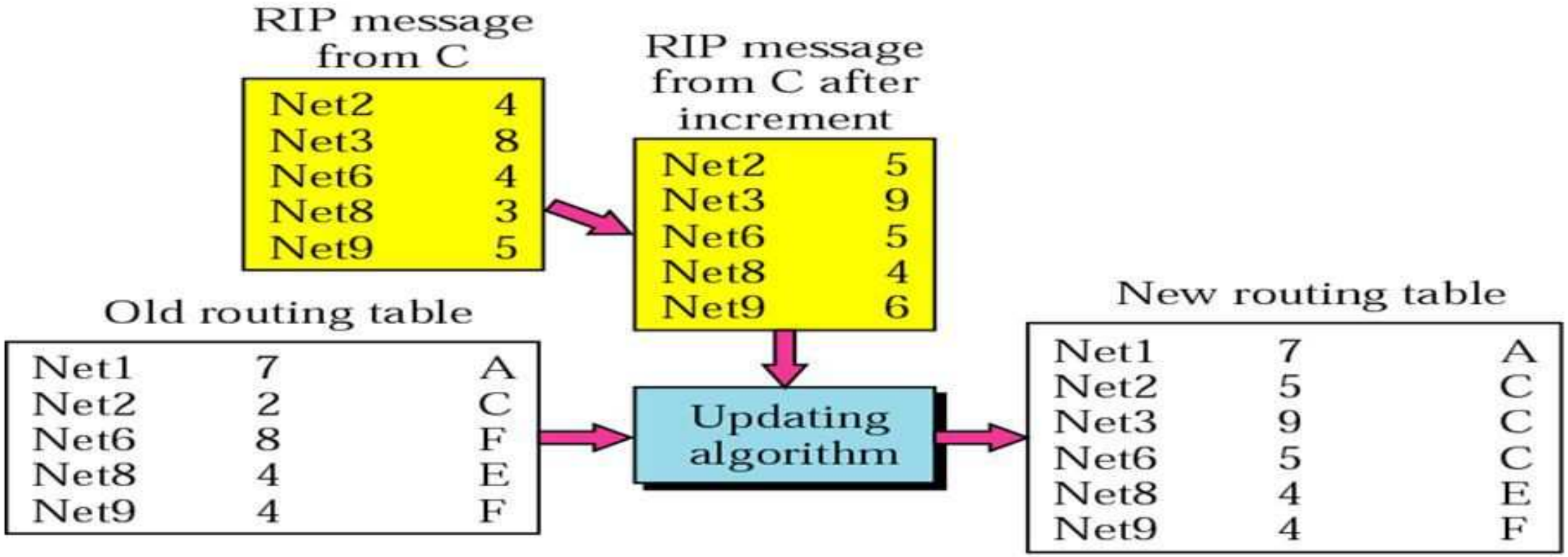
Working of RIP

- Each router initializes its routing table with a list of locally connected networks.
- Periodically, each router advertises the entire contents of its routing table over all of its RIP-enabled interfaces.
    - Whenever a RIP router receives such an advertisement, it puts all of the appropriate routes into its routing table and begins using it to forward packets. This process ensures that every network connected to every router eventually becomes known to all routers.
    - If a router does not continue to receive advertisements for a remote route, it eventually times out that route and stops forwarding packets over it. In other words, RIP is a "soft state" protocol.
- Every route has a property called a metric, which indicates the "distance" to the route's destination.
    - Every time a router receives a route advertisement, it increments the metric.
    - Routers prefer shorter routes to longer routes when deciding which of two versions of a route to program in the routing table.
    - The maximum metric permitted by RIP is 16, which means that a route is unreachable. This means that the protocol cannot scale to networks where there may be more than 15 hops to a given destination.

RIP Example


Hey B, I know about network 1 and 2
Yes, I got your update and added network 1 to my routing table
Update
E0
E1
E0
E1
E0
E1
A
B
C
1.0.0.0
2.0.0.0
3.0.0.0
4.0.0.0
Routing table
1.0.0.0 E0 0
2.0.0.0 E1 0
Routing table
2.0.0.0 E0 0
3.0.0.0 E1 0
1.0.0.0 E0 1
Routing table
3.0.0.0 E0 0
4.0.0.0 E1 0

Example of updating a routing table



Net1: No news, do not change
Net2: Same next hop, replace
Net3: A new router, add
Net6: Different next hop, new hop count smaller, replace
Net8: Different next hop, new hop count the same, do not change
Net9: Different next hop, new hop count larger, do not change

RIP - Routing Information Protocol

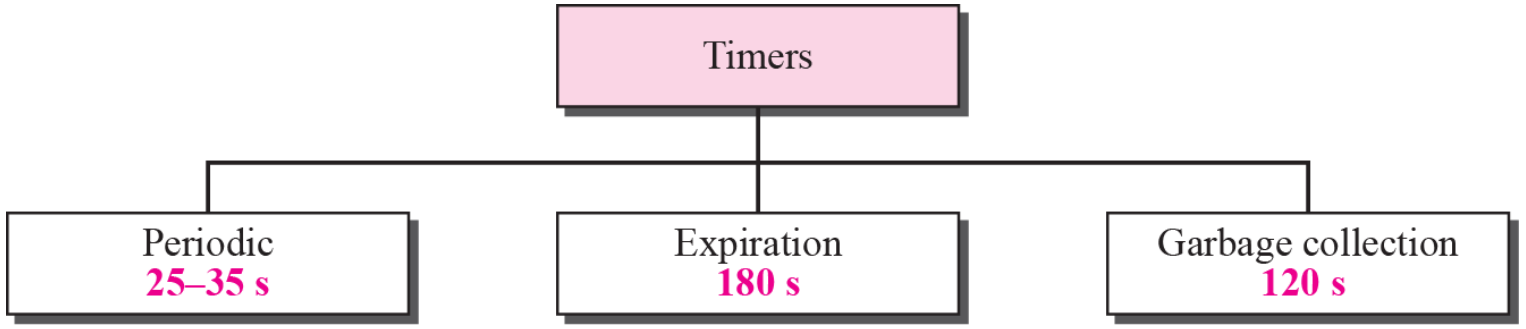
- A simple intradomain protocol
- Straightforward implementation of Distance Vector Routing
- Each router advertises its distance vector every 30 seconds (or whenever its routing table changes) to all of its neighbors
- RIP always uses 1 as link metric
- Maximum hop count is 15, with “16” equal to “ ”
- Routes are timeout (set to 16) after 3 minutes if they are not updated

RIP Messages

- This is the operation of RIP in **`routed`**. Dedicated port for RIP is UDP port 520.

- Two types of messages:
  - **Request messages**
    - used to ask neighboring nodes for an update
  - **Response messages**
    - contains an update

RIP timers

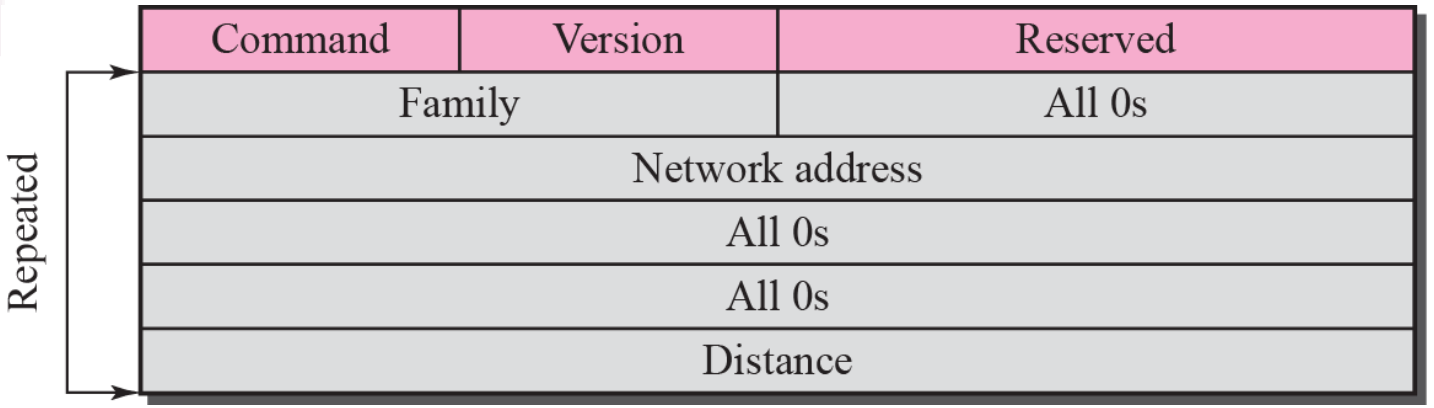


- **Periodic Timer – each router has timer set to 25-35 secs and when the timer counts down, an update message is sent**
- **Expiration Timer – governs the validity of the next-hop – when router receives next-hop update, timer is set to 180 sec. If there is a problem and the router doesn't receive it's 30 sec update, the route info expires (invalid) after the 180 sec count down – then the hop count is set to 16 (infinity)**
- **Garbage Collection Timer – once the route expires, this timer is set to 120 sec and counts downs – allows neighbors time to become aware of invalidity – after count down, info is purged**

RIP timers: REVISION

- Timers in RIP help regulate performance. They include:
- **Update timer** -- Frequency of routing updates. Every 30 seconds IP RIP sends a complete copy of its routing table, subject to split horizon. (Internetwork packet exchange RIP does this every 60 seconds.)
- **Invalid timer** -- Absence of refreshed content in a routing update. RIP waits 180 seconds to mark a route as invalid and immediately puts it into hold-down.
- **Hold-down timers and triggered updates** -- Assist with stability of routes in the Cisco environment. Hold-downs ensure that regular update messages do not inappropriately cause a routing loop. The router doesn't act on non-superior new

Figure 11.11 *RIP message format*



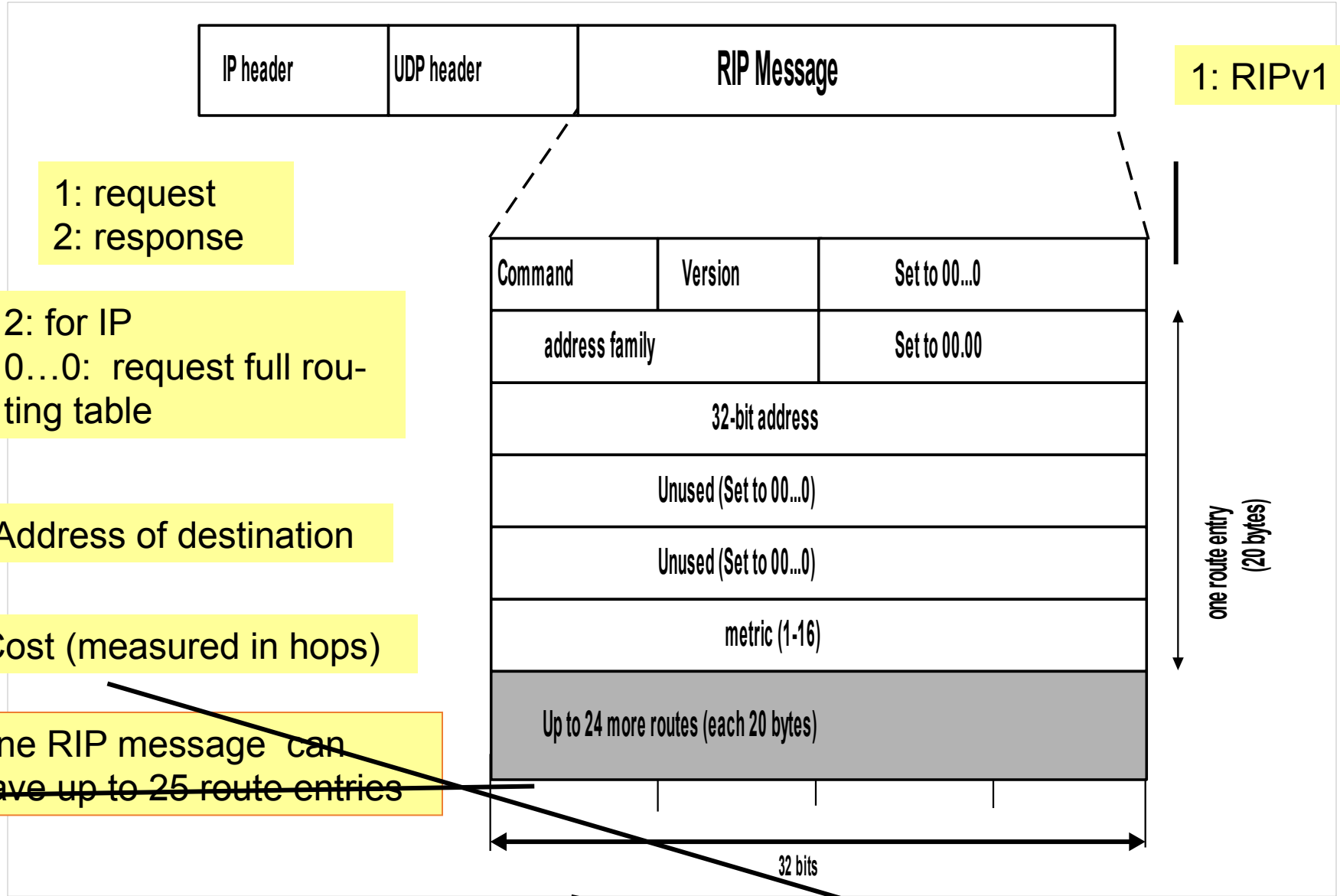
- Command – 8-bit field specifying the type of message: response (2) or request (1)
- Version – 8-bit field specifying RIP version
- Family – 16-bit specifying protocol family (TCP/IP=2)
- Network Address – address of the destination network
- Distance – 32-bit field defining the hop count from advertising router to destination network

NOTE: Request can be issued by a newly added router or by a router seeking certain info

NOTE: 2 response types: Solicited – response to request, Unsolicited – periodic updates

Gray fields repeated for each destination network

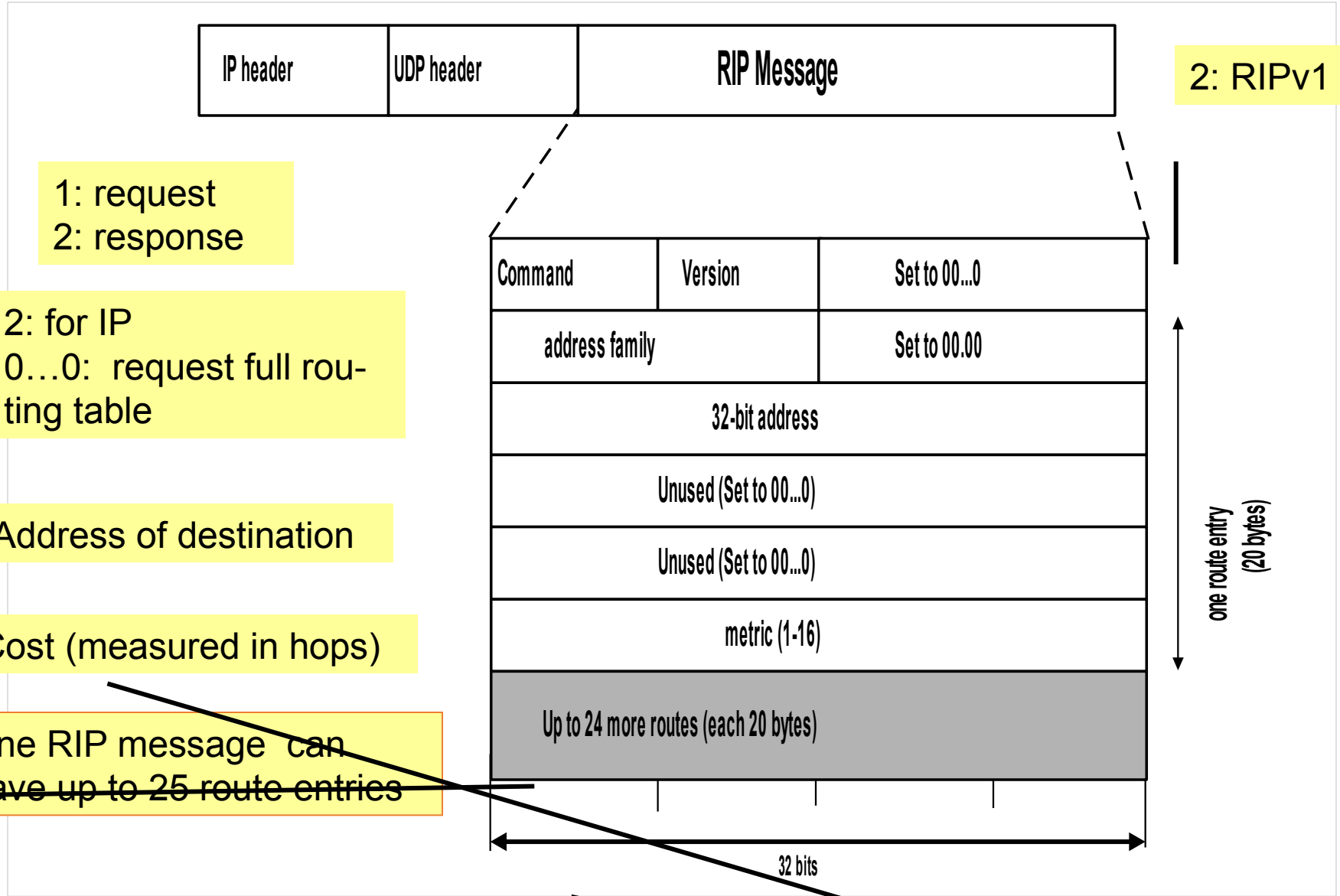
RIPv1 Packet Format



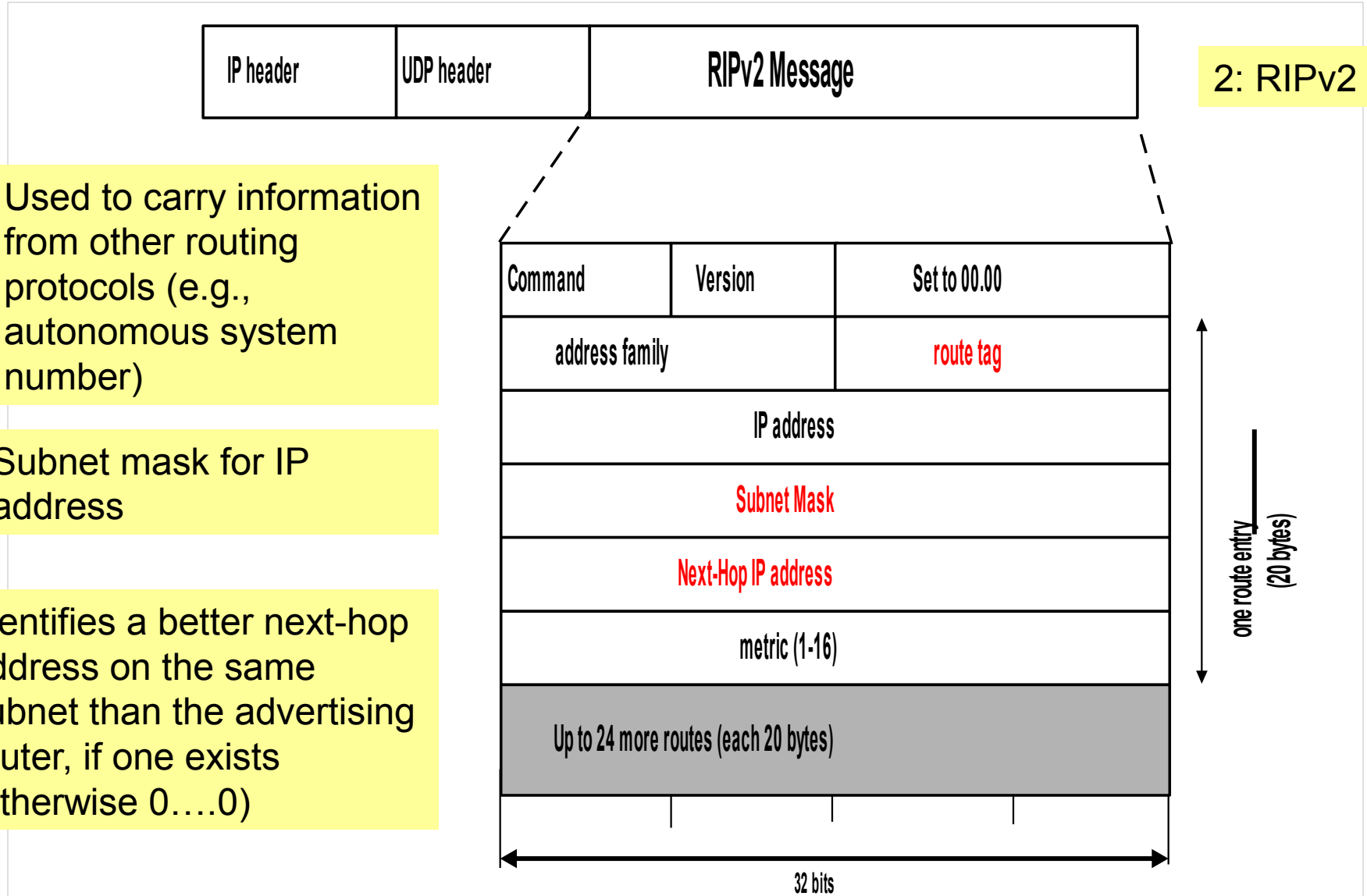
RIPv2

- RIPv2 is an extends RIPv1:
  - Subnet masks are carried in the route information
  - Authentication of routing messages
  - Route information carries next-hop address
  - Exploites IP multicasting
- Extensions of RIPv2 are carried in unused fields of RIPv1 messages

RIPv2 Packet Format



RIPv2 Packet Format



Used to carry information from other routing protocols (e.g., autonomous system number)

Subnet mask for IP address

Identifies a better next-hop address on the same subnet than the advertising router, if one exists (otherwise 0….0)

Routing with RIP

- **Initialization:** Send a **request packet** (command = 1, address family=0..0) on all interfaces:
    - RIPv1 uses broadcast if possible,
    - RIPv2 uses multicast address 224.0.0.9, if possible

  requesting routing tables from neighboring routers
- **Request received**: Routers that receive above request send their entire routing table
- **Response received**: Update the routing table

- Typically, there is a routing daemon (routed) that is an **application layer process** that provides access to routing tables.

Routing with Rip Cont.

- **Regular routing updates**: Every 30 seconds, send all or part of the routing tables to every neighbor in an response message
- **Triggered Updates:** Whenever the metric for a route change, send entire routing table.
- If a router does not hear from its neighbor once every 180 seconds, the neighbor is deemed unreachable.

Security

- Issue: Sending bogus routing updates to a router
- RIPv1: No protection
- RIPv2: Simple authentication scheme

RIP Problems

- RIP takes a long time to stabilize
  - Even for a small network, it takes several minutes until the routing tables have settled after a change
- RIP has all the problems of distance vector algorithms, e.g., count-to-Infinity
    - RIP uses split horizon to avoid count-to-infinity
- The maximum path in RIP is 15 hops

Shortest Path

- Routing decision in networks, are mostly taken on the basis of cost between source and destination. Hop count plays major role here. Shortest path is a technique which uses various algorithms to decide a path with minimum number of hops.
- Common shortest path algorithms are:
- Dijkstra's algorithm
- Bellman Ford algorithm
- Floyd Warshall algorithm

Approaches to Shortest Path Routing

- There are two basic routing algorithms found on the Internet.

1. Distance Vector Routing

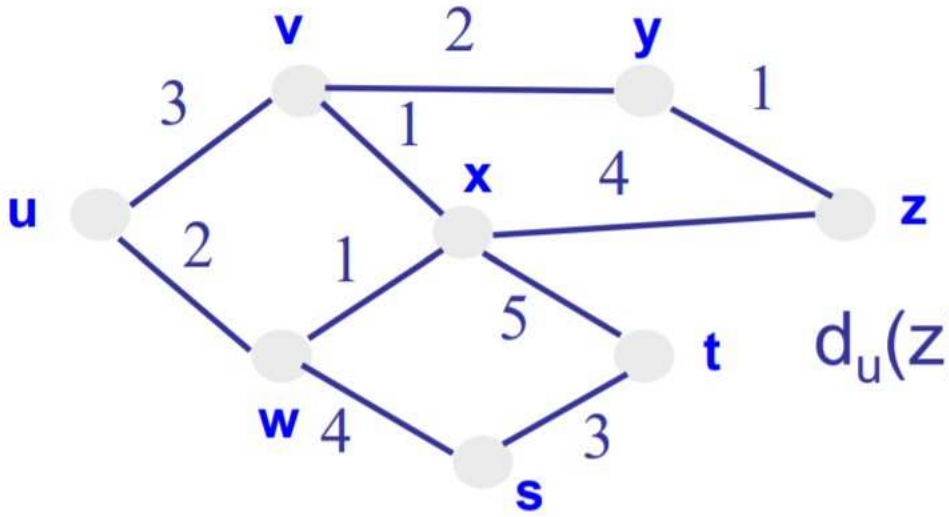
- Each node knows the distance (=cost) to its directly connected neighbors
- A node sends periodically a list of routing updates to its neighbors.
- If all nodes update their distances, the routing tables eventually converge
- New nodes advertise themselves to their neighbors

2. Link State Routing

- Each node knows the distance to its neighbors
- The distance information (=link state) is broadcast to all nodes in the network

Bellman Ford Algorithm

- Define distances at each node *X*
  - $d_x(y)$ = cost of least-cost *path* from *X* to *Y*
- Update distances based on neighbors
  - $d_x(y) = \min \{c(x,v) + d_v(y)\}$ over all neighbors *V*

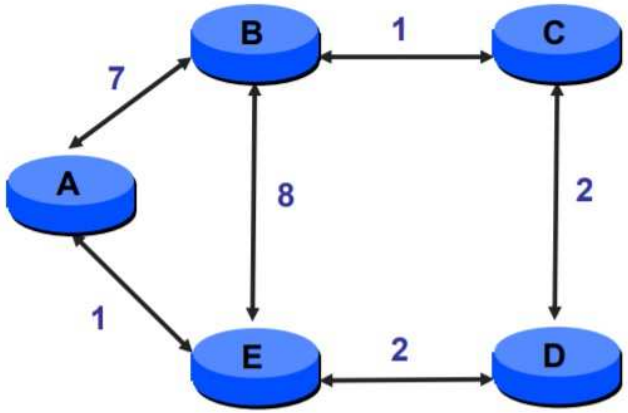


$$d_u(z) = \min\{c(u,v) + d_v(z), \; c(u,w) + d_w(z)\}$$

Step-by-Step

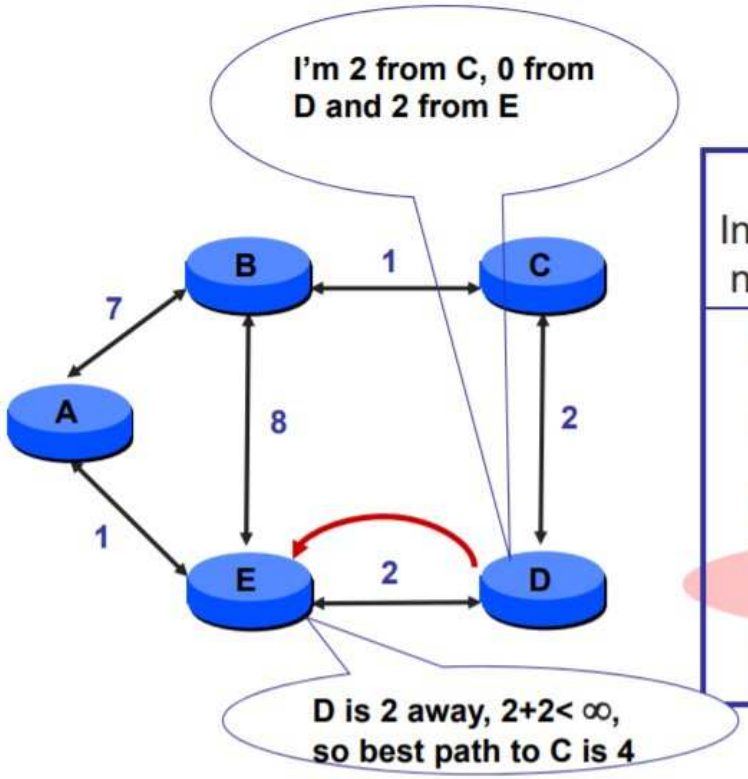
- $c(x,v)$ = cost for direct link from x to v
  - Node x maintains costs of direct links $c(x,v)$
- $D_x(y)$ = estimate of least cost from x to y
  - Node x maintains distance vector $\mathbf{D}_x = [D_x(y): y \in N]$
- Node x maintains its neighbors' distance vectors
  - For each neighbor v, x maintains $\mathbf{D}_v = [D_v(y): y \in N]$
- Each node v periodically sends D_v to its neighbors
  - And neighbors update their own distance vectors
  - $D_x(y) \leftarrow \min_v\{c(x,v) + D_v(y)\}$ for each node $y \in N$

Example



Info at node	Distance to Node				
	A	**B**	**C**	**D**	**E**
A	0	7	∞	∞	1
B	7	0	1	∞	8
C	∞	1	0	2	∞
D	∞	∞	2	0	2
E	1	8	∞	2	0

D sends vector to *E*



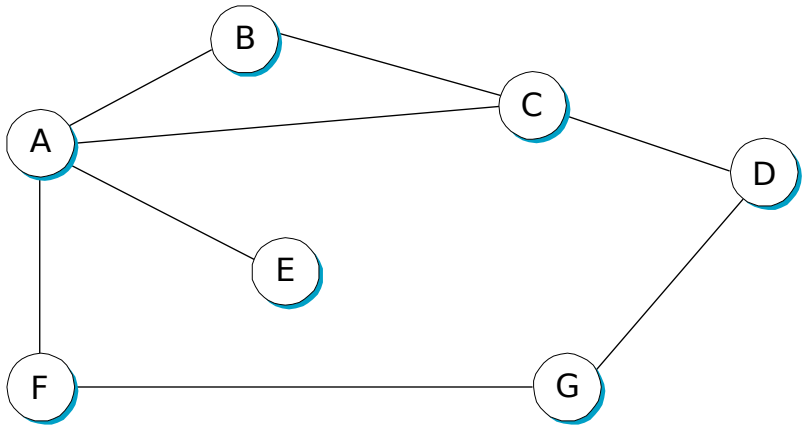
Info at node	Distance to Node				
	A	**B**	**C**	**D**	**E**
A	0	7	∞	∞	1
B	7	0	1	∞	8
C	∞	1	0	2	∞
D	∞	∞	2	0	2
E	1	8	**4**	2	0

Distance Vector Algorithm:

At all nodes, X:

1 Initialization:
2 for all adjacent nodes v:
3 $D^X(*,v) = \text{infty}$ /* the * operator means "for all rows" */
4 $D^X(v,v) = c(X,v)$
5 for all destinations, y
6 send $\min_w D^X(y,w)$ to each neighbor /* w over all X's neighbors */

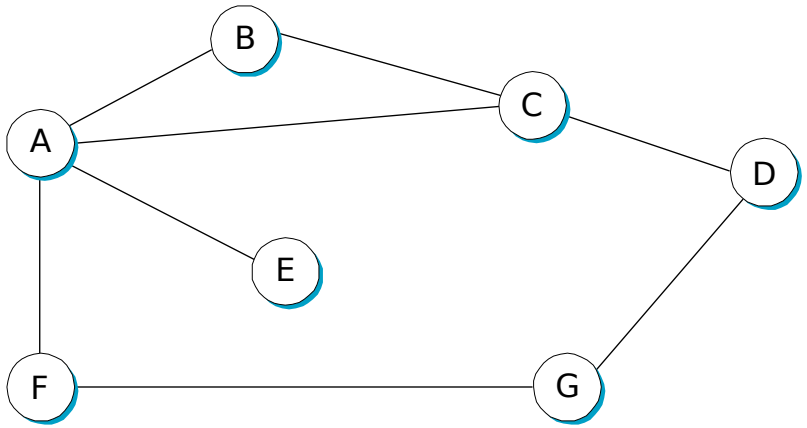
An Example



- Internal Information at each node ----->

	A	B	C	D	E	F	G
A	0	1	1	∞	1	1	∞
B	1	0	1	∞	∞	∞	∞
C	1	1	0	1	∞	∞	∞
D	∞	∞	1	0	∞	∞	1
E	1	∞	∞	∞	0	∞	∞
F	1	∞	∞	∞	∞	0	1
G	∞	∞	∞	1	∞	1	0

Routing Tables

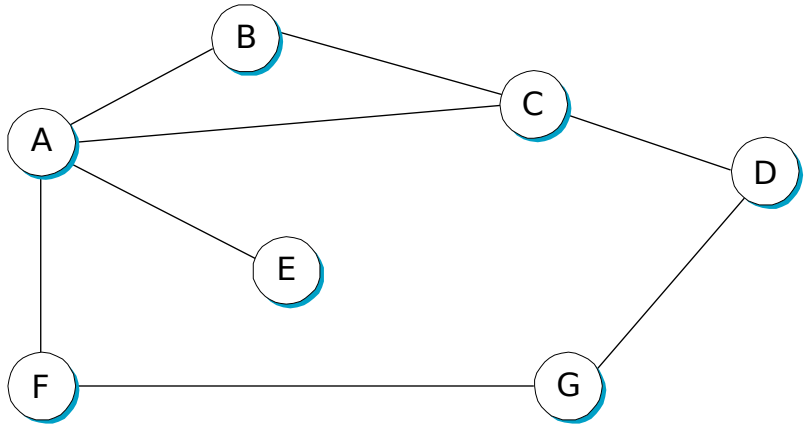


- With this information, routing table at A is -->

	Cost	Next Hop
B	1	B
C	1	C
D	∞	-
E	1	E
F	1	F
G	∞	-

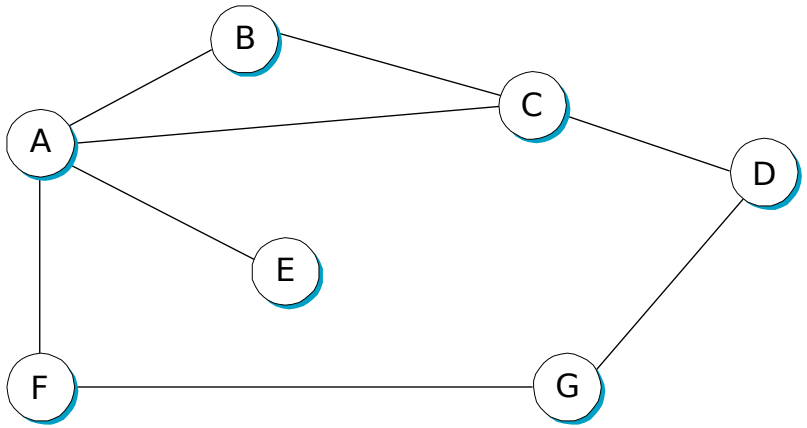
Evolution of the table.

- Each node sends a message to neighbors with a list of distances.
- F --> A with G is at a distance 1
- C --> A with D at distance 1.



	Cost	Next Hop
B	1	B
C	1	C
D	2	C
E	1	E
F	1	F
G	2	F

Final Distance Matrix



	A	B	C	D	E	F	G
A	0	1	1	2	1	1	2
B	1	0	1	2	2	2	3
C	1	1	0	1	2	2	2
D	2	2	1	0	3	2	1
E	1	2	2	3	0	2	3
F	1	2	2	2	2	0	1
G	2	3	2	1	3	1	0

Advantages & Disadvantages of Distance Vector Routing Protocols

Advantages:	Disadvantages:
Simple implementation and maintenance. The level of knowledge required to deploy and later maintain a network with distance vector protocol is not high.	**Slow convergence.** The use of periodic updates can cause slower convergence. Even if some advanced techniques are used, like triggered updates which are discussed later, the overall convergence is still slower compared to link state routing protocols.
Low resource requirements. Distance vector protocols typically do not need large amounts of memory to store the information. Nor do they require a powerful CPU. Depending of the network size and the IP addressing implemented they also typically do not require a high level of link bandwidth to send routing updates. However, this can become an issue if you deploy a distance vector protocol in a large network.	**Limited scalability.** Slow convergence may limit the size of the network because larger networks require more time to propagate routing information.
	Routing loops. Routing loops can occur when inconsistent routing tables are not updated due to slow convergence in a changing network.

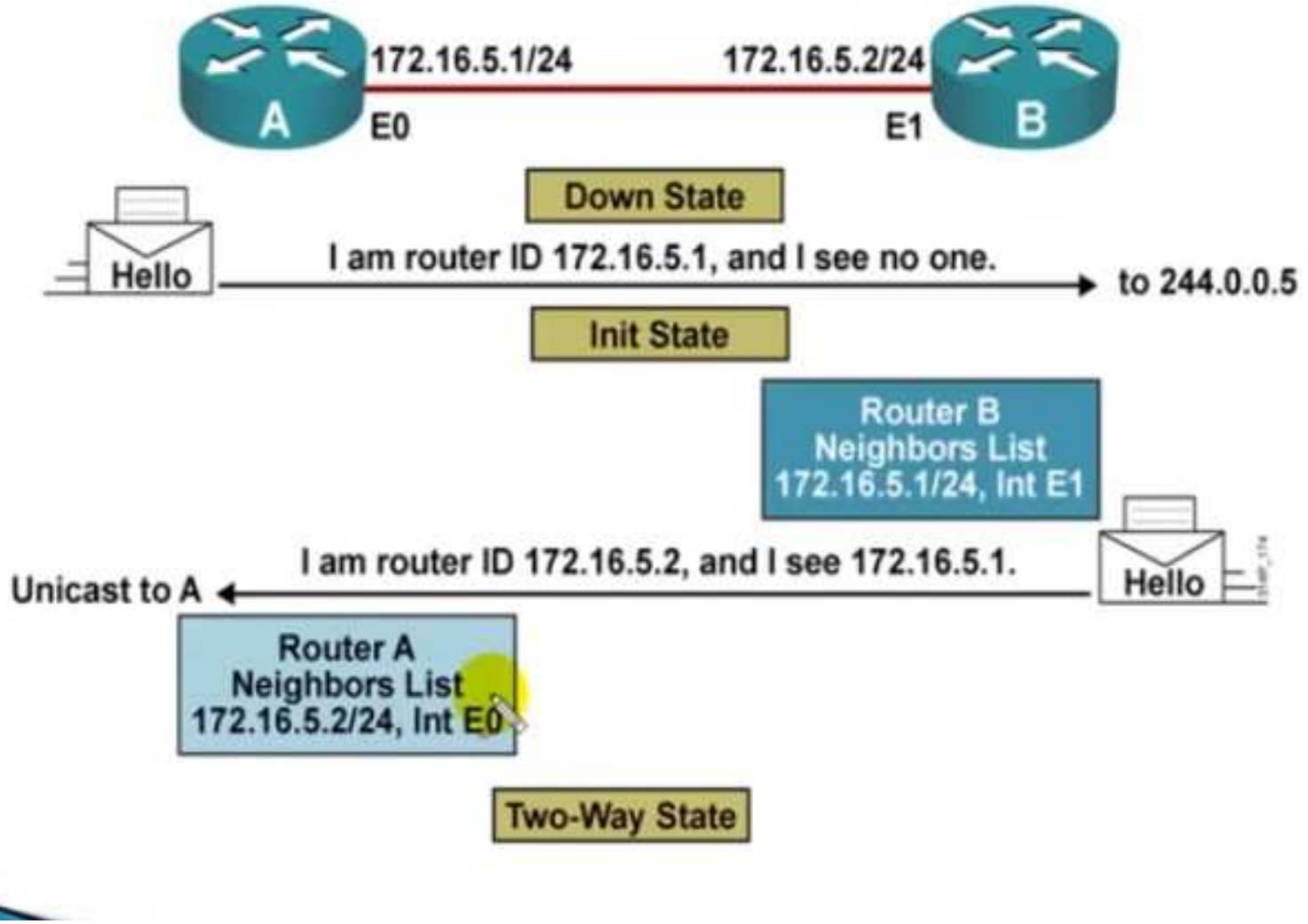
Open Shortest Path First

- Open Shortest Path First is a robust link-state interior gateway protocol (IGP). People use OSPF when they discover that RIP just isn’t going to work for their larger network, or when they need very fast convergence.
- This installment of Networking 101 will provide a conceptual overview of OSPF, and the second part of our OSPF coverage will delve a bit deeper into the protocol itself, as well as OSPF area configurations.

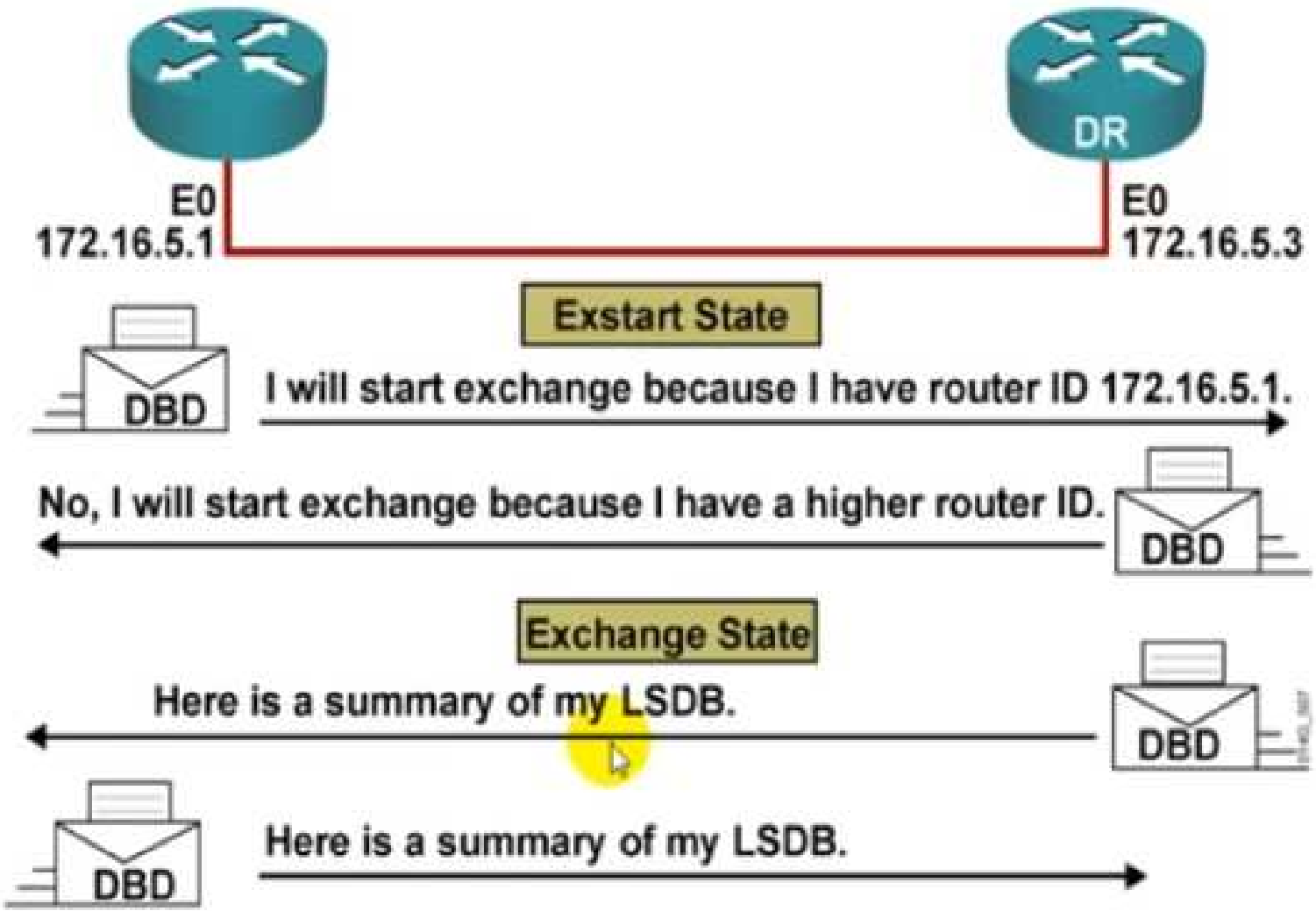
Terminology

- Router ID: In OSPF this is a unique 32-bit number assigned to each router. This is chosen as the highest IP address on a router, and can be set large by configuring an address on a loopback interface of the chosen router.
- Neighbor Routers: two routers with a common link that can talk to each other.
- Adjacency: a two-way relationship between two neighbor routers. Neighbors don’t always form adjacencies.
- LSA: Link State Advertisements are flooded; they describe routes within a given link.
- Hello Protocol: this is how routers on a network determine their

Establishing Bi Directional Communication

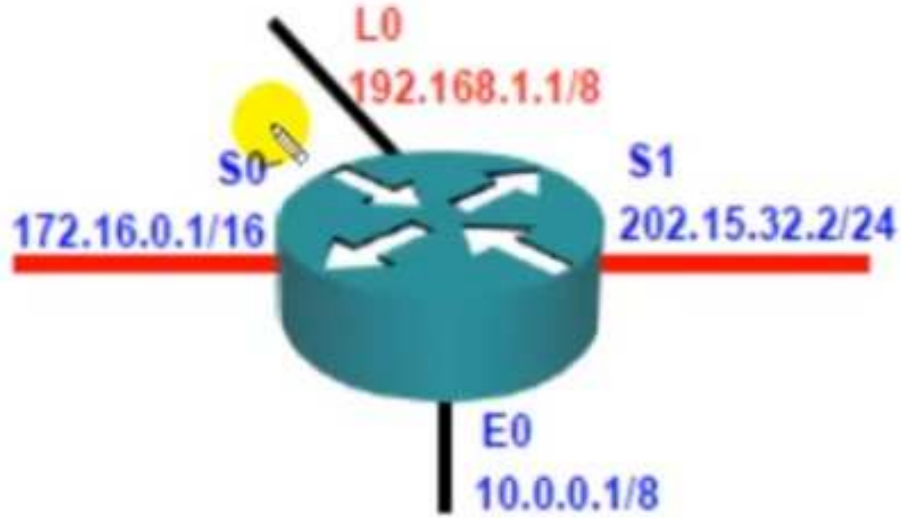


Discovering the Network Routes

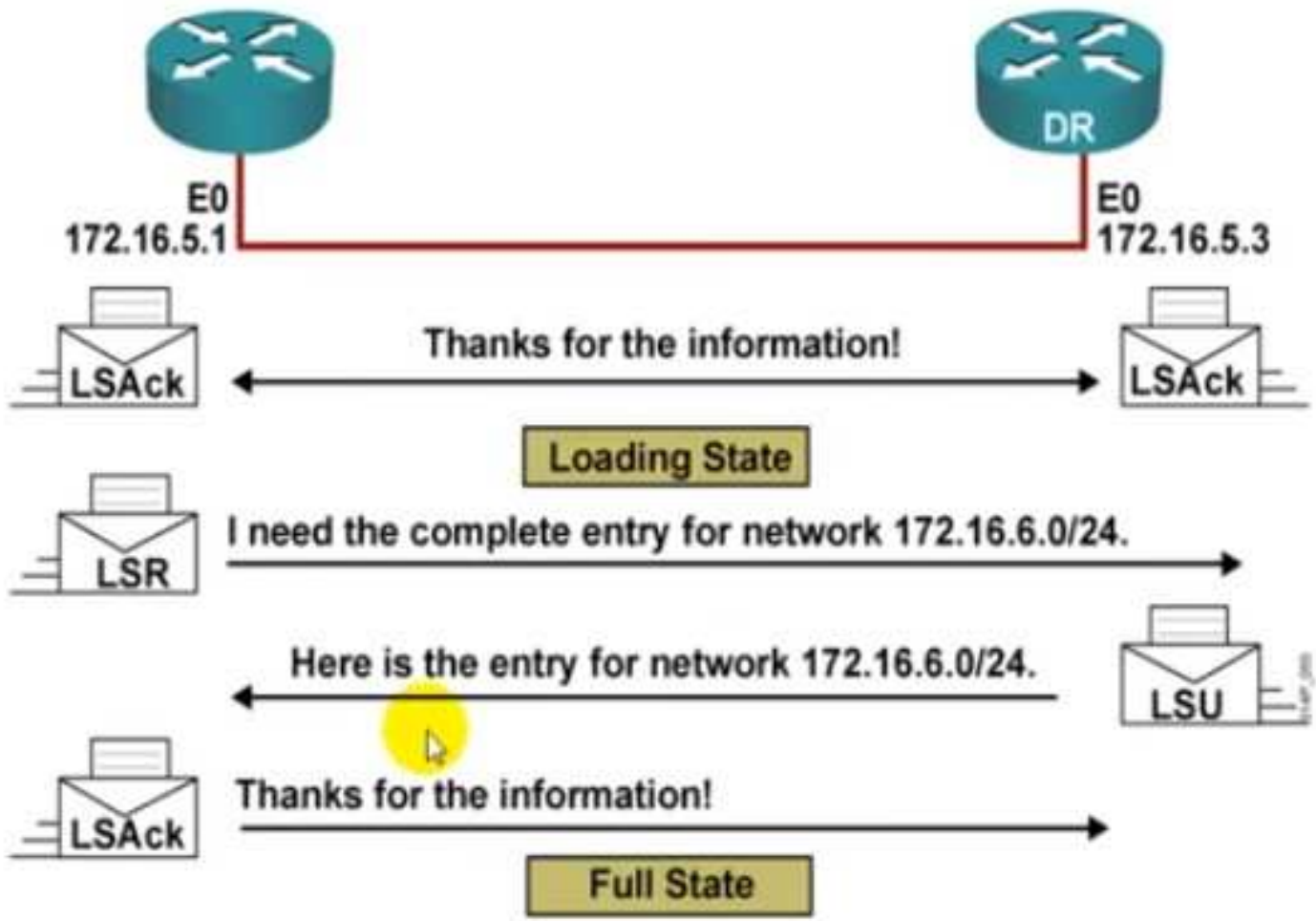


Router ID

- The highest IP address of the active physical interface of the router is Router ID.
- If logical interface is configured, the highest IP address of the logical interface is Router ID



Adding the Link-State Entries



OSPF Tables

Neighbor Table

- Also known as the adjacency database
- Contains list of directly connected routers (neighbors)
- **# Show ip ospf neighbor**

Database Table

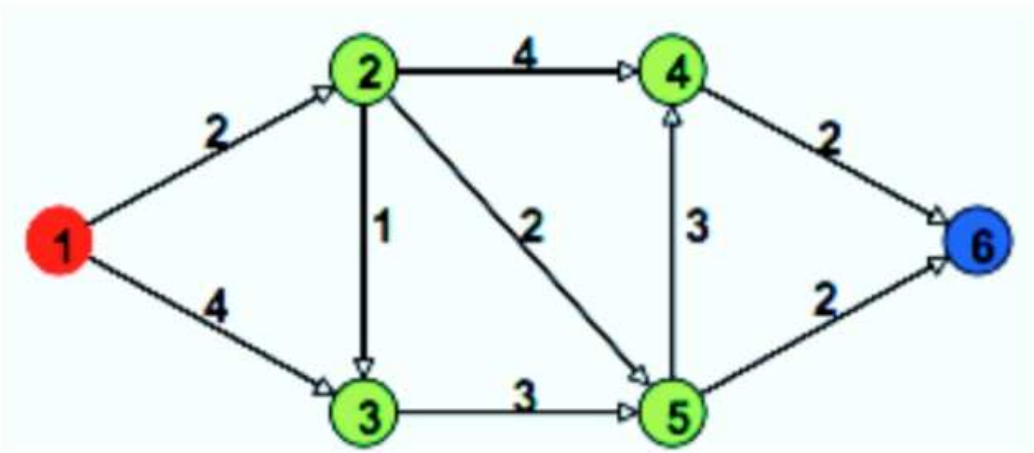
- Typically referred to as LSDB (link state database)
- Contains information about all the possible routes to the networks with in the area
- **# show ip ospf database**

Routing Table

- Contains list of best paths to each destination
- **# show ip route**

Dijkstra's Algorithm

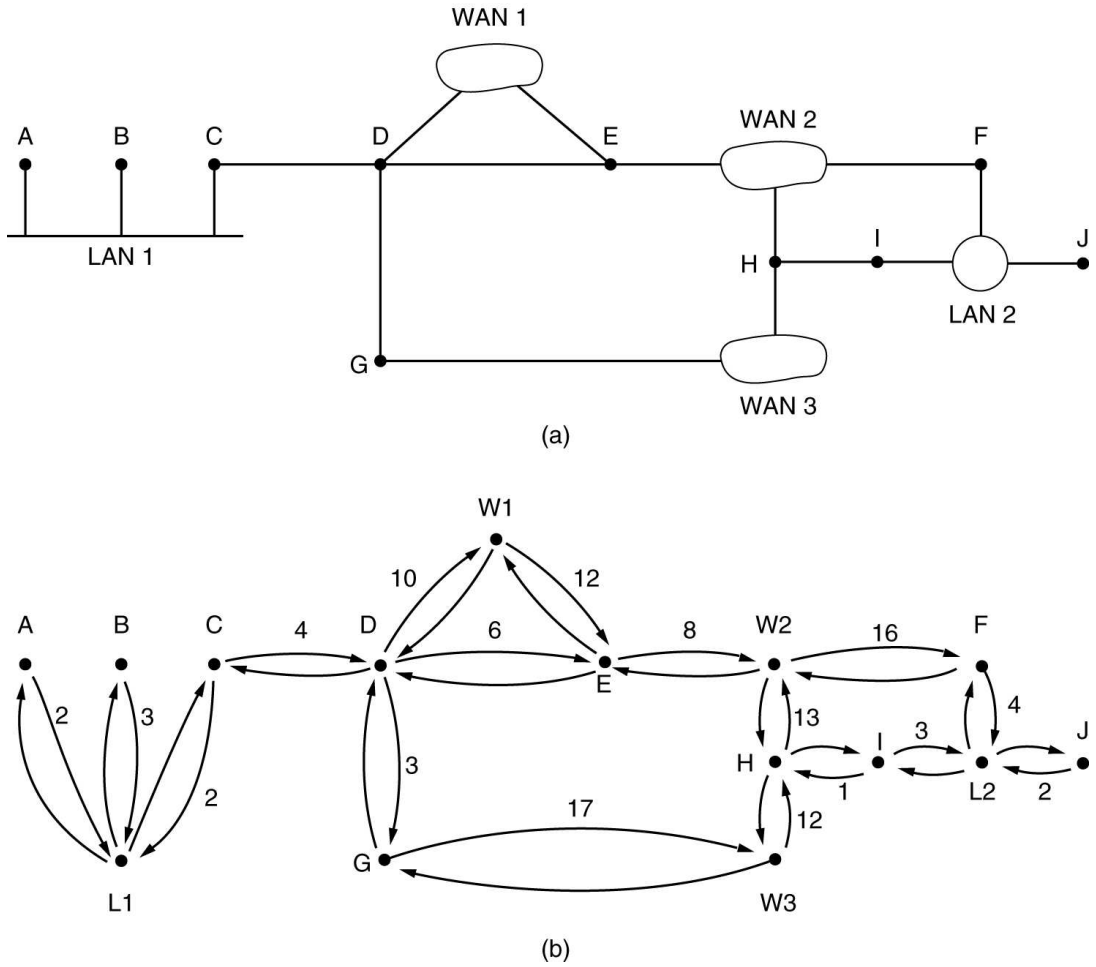
- Solution to the single-source shortest path problem in graph theory
  - Both directed and undirected graphs
  - All edges must have nonnegative weights
  - Graph must be connected



Pseudocode

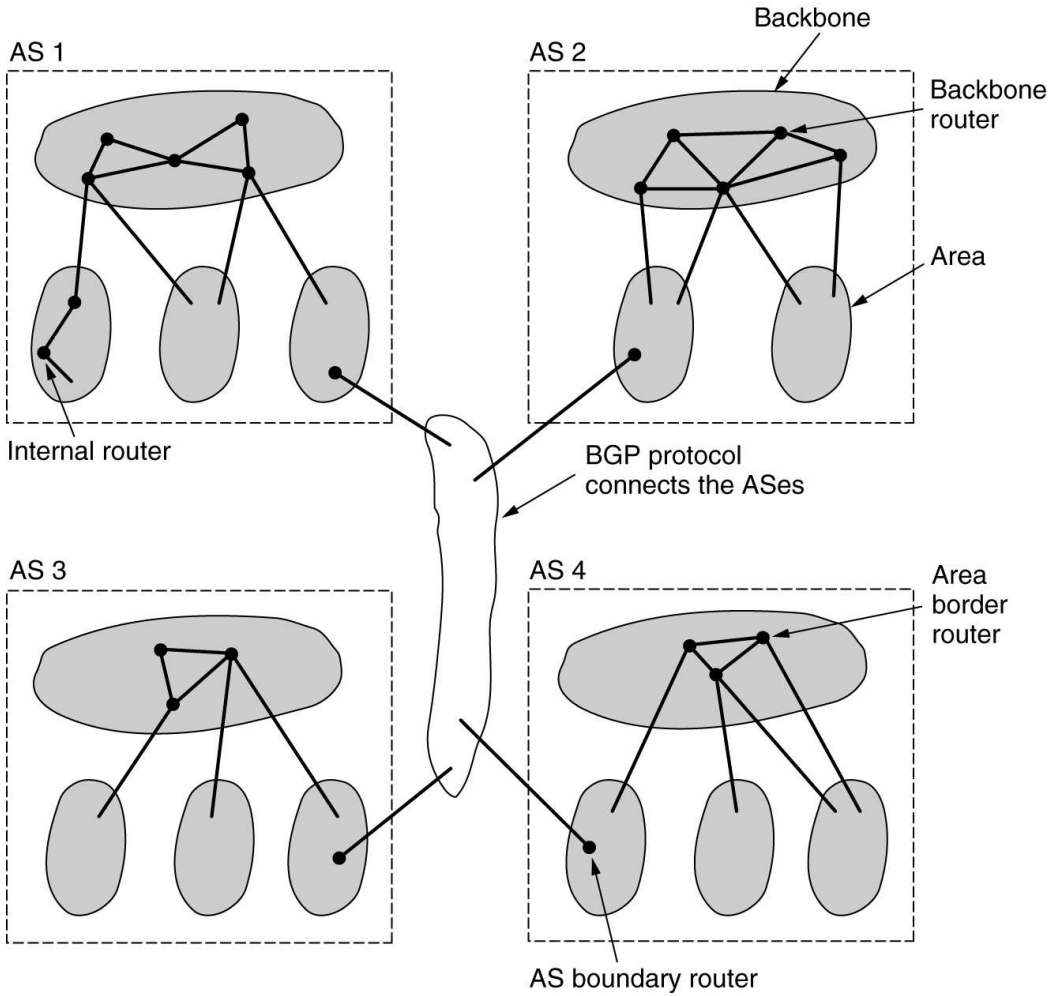
```
dist[s] ←o                          (distance to source vertex is zero)
for  all v ∈ V–{s}
      do  dist[v] ←∞                (set all other distances to infinity)
S←∅                                 (S, the set of visited vertices is initially empty)
Q←V                                 (Q, the queue initially contains all vertices)
while Q ≠∅                          (while the queue is not empty)
do   u ← mindistance(Q,dist)        (select the element of Q with the min. distance)
     S←S∪{u}                        (add u to list of visited vertices)
      for all v ∈ neighbors[u]
           do  if   dist[v] > dist[u] + w(u, v)        (if new shortest path found)
                    then     d[v] ←d[u] + w(u, v)     (set new value of shortest path)
                                                      (if desired, add traceback code)

return dist
```

OSPF – The Interior Gateway Routing Protocol



(a) An autonomous system. (b) A graph representation of (a).

OSPF (2)



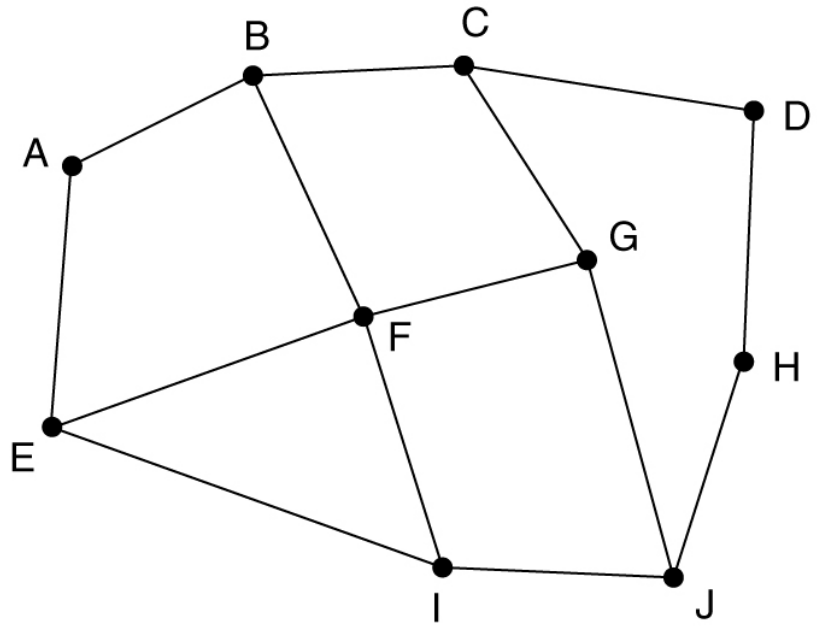
The relation between ASes, backbones, and areas in OSPF.

OSPF (3)

The five types of OSPF messeges.

Message type	Description
Hello	Used to discover who the neighbors are
Link state update	Provides the sender's costs to its neighbors
Link state ack	Acknowledges link state update
Database description	Announces which updates the sender has
Link state request	Requests information from the partner

BGP – The Exterior Gateway Routing Protocol



(a)

Information F receives from its neighbors about D

From B: "I use BCD"
From G: "I use GCD"
From I: "I use IFGCD"
From E: "I use EFGCD"

(b)

(a) A set of BGP routers. (b) Information sent to F.

Routing Information Protocol (**RIP**) is a dynamic **routing protocol** which uses hop count as a **routing** metric to find the best path between the source and the destination network.

It is a distance vector **routing protocol** works on the application layer of OSI model. **RIP** uses port number 520.

Routing Information Protocol (**RIP**) is a **protocol** that routers **can use to** exchange network topology information.

There are two versions of RIP (the managed switch supports both):

RIPv1 defined in RFC 1058.

- Routes are specified by IP destination network and hop count.
- The routing table is broadcast to all stations on the attached network.

RIPv2 defined in RFC 1723.

- Route specification also includes subnet mask and gateway.
- The routing table is sent to a multicast address, reducing network traffic.
- Authentication is used for security.

RIP uses a distance vector algorithm to decide which path to put a packet on to get to its destination.

Each **RIP router** maintains a routing table, which is a **list of all the destinations** the router knows how to reach.

Each router broadcasts its entire routing table to its closest neighbors every 30 seconds.

In this process, *neighbors* are the other routers to which a router is connected directly -- that is, the other routers on the same network segments as the selected router.

The neighbors, in turn, pass the information on to their nearest neighbors, and so on, until all RIP hosts within the network have the same knowledge of routing paths. This shared knowledge is known as ***convergence.***

C1: If a router receives an update on a route, and the new path is shorter, it will update its table entry with the length and next-hop address of the shorter path.

C2: If the new path is longer, it will wait through a "hold-down" period to see if later updates reflect the higher value as well. *It will only update the table entry if the new, longer path has been determined to be stable.*

C3: If a router crashes or a network connection is severed the network discovers this because that router stops sending updates to its neighbors, or stops sending and receiving updates along the severed connection.

C4: If a given route in the routing table isn't updated across six successive update cycles (that is, for 180 seconds) a RIP router will drop that route and let the rest of the network know about the problem through its own periodic updates.

A router running RIP sends the contents of its routing table to each of its adjacent routers every 30 seconds.
When a route is removed from the routing table, it is flagged as unusable by the receiving routers after 180 seconds, and removed from their tables after an additional 120 seconds.

Versions

There are three versions of the Routing Information Protocol: RIPv1, RIPv2 and RIPng.

RIPv1-- standardized in 1988 -- is also called Classful Routing Protocol because it does not send subnet mask information in its routing updates. On the

other hand, RIPv2 -- standardized in 1998 -- is called Classless Routing Protocol because it does send subnet mask information in its routing updates. RIPng is an extension of RIPv2 that was made to support IPv6.

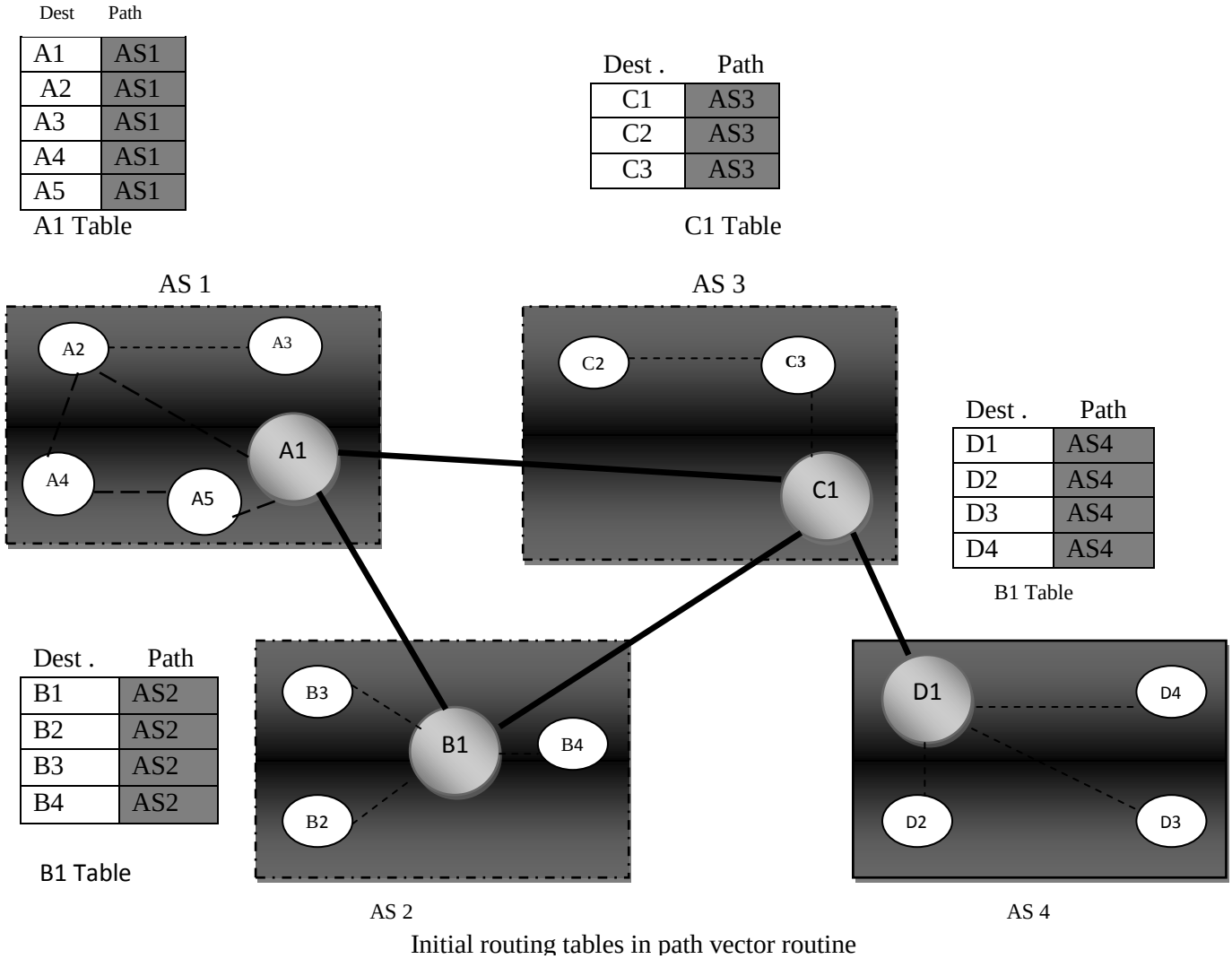
RIP allows only 15 hops in a path. If a packet can't reach a destination in 15 hops, the destination is considered unreachable.

Paths can be assigned a higher cost (as if they involved extra hops) if the enterprise wants to limit or discourage their use.

For example, a satellite backup link might be assigned a cost of 10 to force traffic to follow other routes when available.

Path Vector Routing

Path Vector Routing is a routing algorithm in unicast routing protocol of network layer, and it is useful for interdomain routing. The principle of path vector routing is similar to that of distance vector routing. It assumes that there is one node in each autonomous system that acts on behalf of the entire autonomous system is called Speaker node .The speaker node in an AS creates a routing cable and advertises to the speaker node in the neighbouring ASs .A speaker node advertises the path, not the metrics of the nodes, in its autonomous system or other autonomous systems



Initial routing tables in path vector routine

It is the initial table for each speaker node in a system made four ASs. Here Node A1 is the speaker node for AS1, B1 for AS2, C1 for AS3 and D1 for AS4, Node A1 creates an initial table that shows A1 to A5 and these are located in AS1, it can be reached through it

A speaker in an autonomous system shares its table with immediate neighbours ,here Node A1 share its table with nodes B1 and C1 , Node C1 share its table with nodes A1,B1 and D1 , Node B1 share its table with nodes A1 and C1 , Node D1 share its table with node C1

If router A1 receives a packet for nodes A3 , it knows that the path is in AS1,but if it receives a packet for D1,it knows that the packet should go from AS1,to AS2 and then to AS3 ,then the routing table shows that path completely on the other hand if the node D1 in AS4 receives a packet for node A2,it knows it should go through AS4,AS3,and AS1,

FUNCTIONS

- PREVENTION OF LOOP

The creation of loop can be avoided in path vector routing .A router receives a message it checks to see if its autonomous system is in the path list to the destination if it is looping is involved and the message is ignored

- POLICY ROUTING

When a router receives a messages it can check the path, if one of the autonomous system listed in the path against its policy, it can ignore its path and destination it does not update its routing table with this path or it does not send the messages to its neighbours.

- OPTIMUM PATH

A path to a destination that is the best for the organization that runs the autonomous system

BORDER GATEWAY PROTCOL (BGP)

It first introduced in 1989, Is an interdomain routing protocol using path vector routing. It has three versions

1. STUB AS
2. MULTHIHOMED AS
3. TRANSIT AS