



# Unit-4: Network Layer



**Prof. Maulik D Trivedi**

Computer Engineering Department

Darshan Institute of Engineering & Technology,

Rajkot

✉ [maulik.trivedi@darshan.ac.in](mailto:maulik.trivedi@darshan.ac.in)

☎ +91-9998265805





## Outline

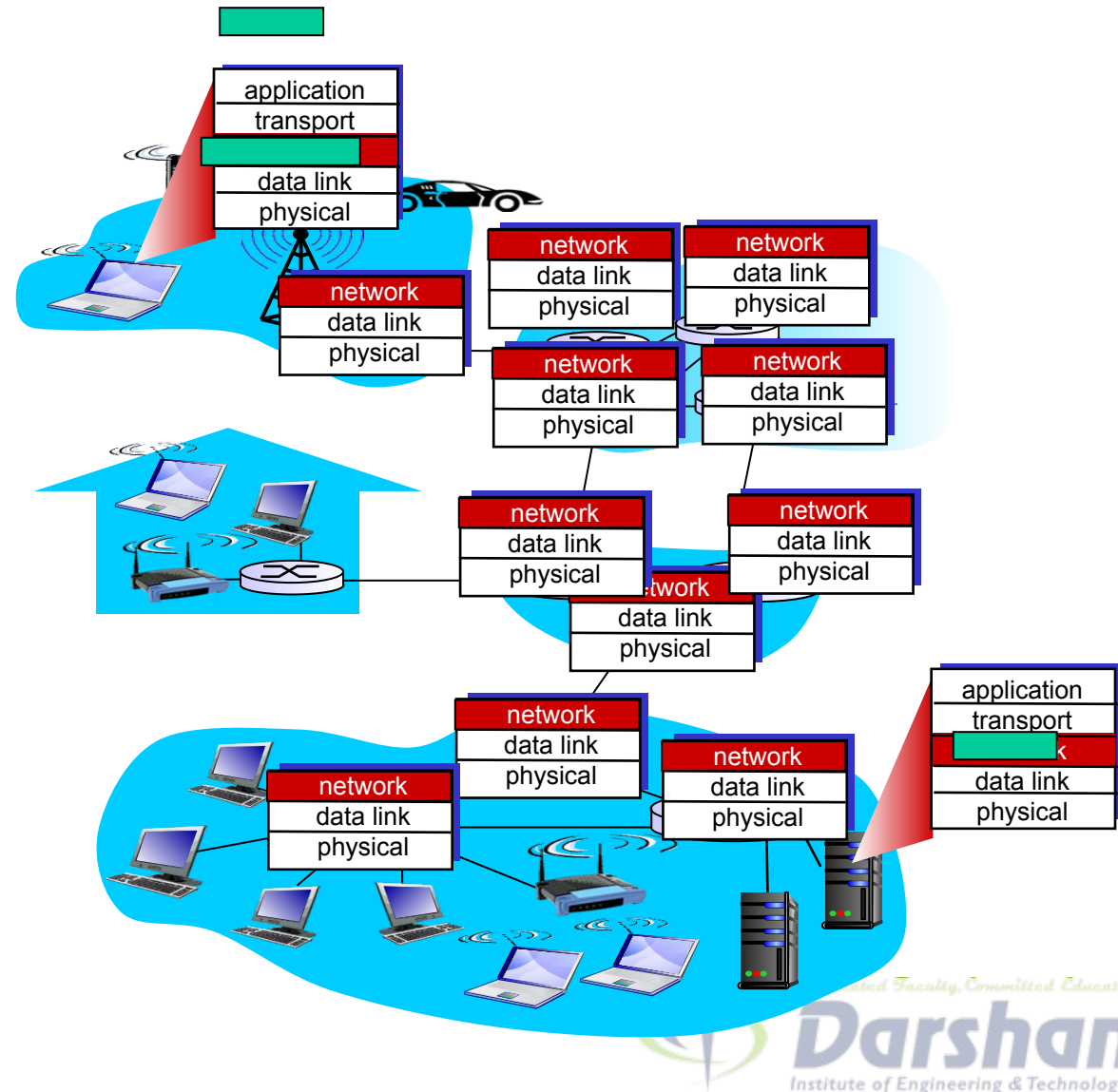
- Introduction
- Virtual and Datagram networks
- Study of Router
- IP protocol and addressing in the Internet
- Routing algorithms
- Broadcast and Multicast Routing



# Introduction of Network Layer

# Introduction: Network Layer

- ❑ To deliver segment from sending to receiving host/ router.
- ❑ On sending side, it encapsulates segments into datagrams.
- ❑ On receiving side, it delivers segments to transport layer.
- ❑ Network layer protocols in every host and router.
- ❑ Router examines header fields in all IP datagrams passing through it.



# Key Function of Network Layer

- Role of the network layer is simple - to move packets from a sending host to a receiving host.
- Two important network layer functions can be identified:

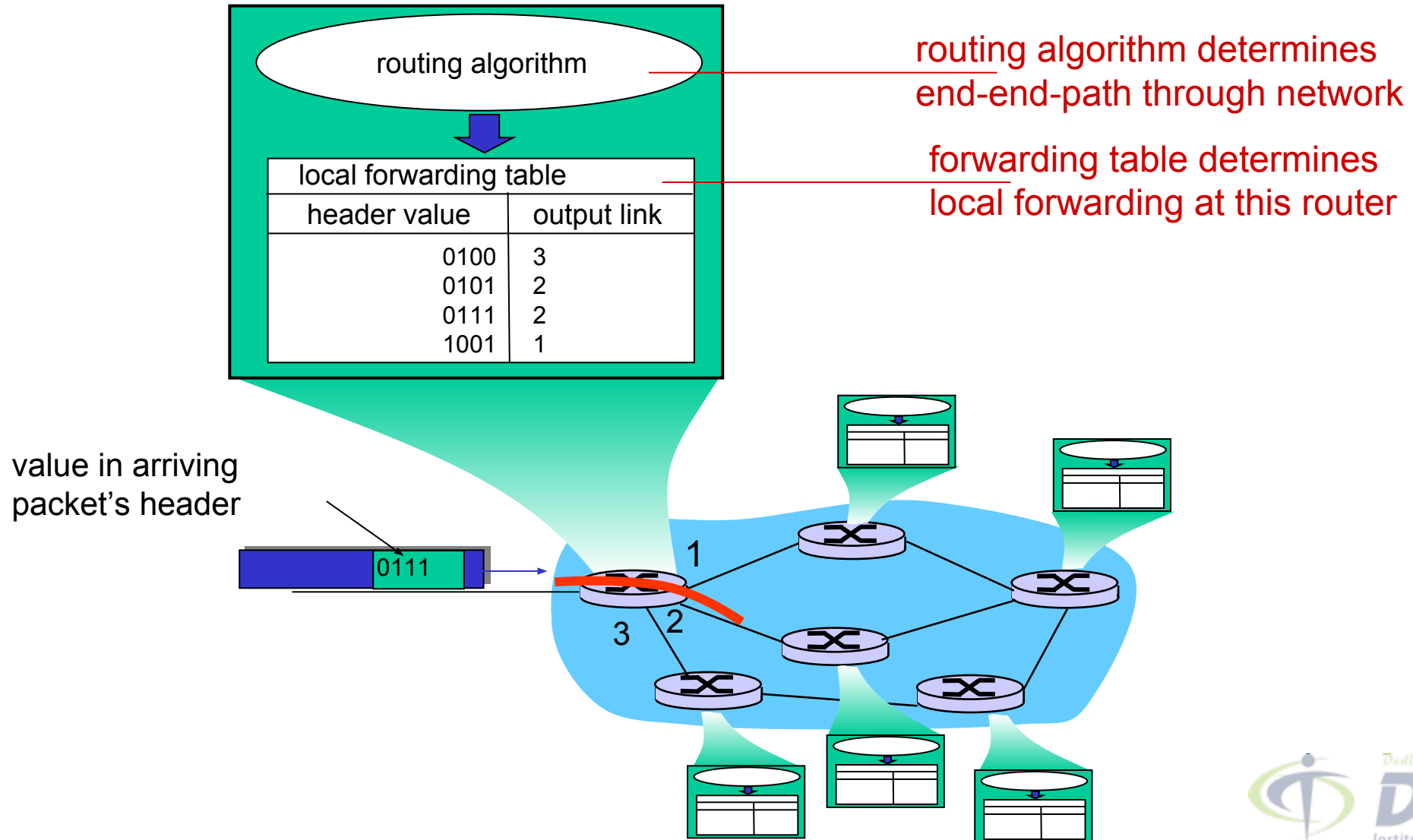
## 1. Forwarding

- When a packet arrives at a router's input link, the router must move the packet to the appropriate output link.

## 2. Routing

- It's a process of selecting best paths in a network.
- The network layer must determine the route or path taken by packets as they flow from a sender to a receiver.
- The algorithms that calculate these paths are referred to as routing algorithms.

# Routing and Forwarding



# Network Service Model

- Services provided by network layer for individual datagrams.
- **Guaranteed delivery**
  - This service guarantees that the packet will eventually arrive at its destination.
- **Guaranteed delivery with bounded delay**
  - This service not only guarantees delivery of the packet, but delivery within a specified host-to-host delay bound.

# Network Service Model – Cont...

- Services provided by network layer for a flow of datagrams.
- **In-order packet delivery**
  - This service guarantees that packets arrive at the destination in the order that they were sent.
- **Guaranteed minimal bandwidth**
  - This network-layer service emulates the behaviour of a transmission link of a specified bit rate (for example, 1 Mbps) between sending and receiving hosts.
  - As long as the sending host transmits bits at a rate below the specified bit rate, then no packet is lost.



# Network Service Model – Cont...

## ❑ Guaranteed maximum jitter

- ❑ This service guarantees that the amount of time between the transmission of two successive packets at the sender is equal to the amount of time between their receipt at the receiver.

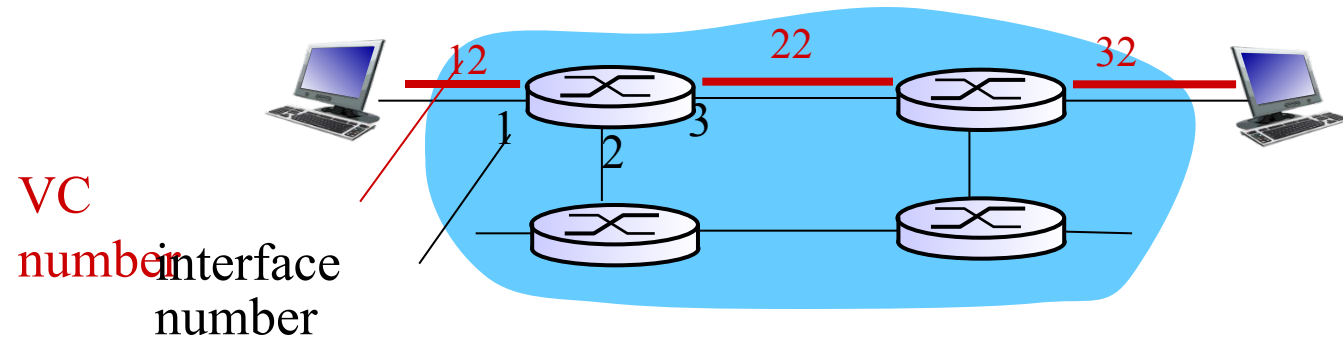
## ❑ Security services

- ❑ Using a secret session key known only by a source and destination host, the network layer in the source host could encrypt the payloads of all datagrams being sent to the destination host.
- ❑ The network layer in the destination host would then be responsible for decrypting the payloads.

# Virtual Circuit Switching

- A VC consists of
  1. A path between the source and destination hosts
  2. VC numbers, one number for each link along the path
  3. Entries in the forwarding table in each router along the path
- A packet belonging to a virtual circuit will carry a VC number in its header.
- VC number can be changed on each link
  - New VC number comes from forwarding table

# VC Forwarding Table



*Forwarding table in router:*

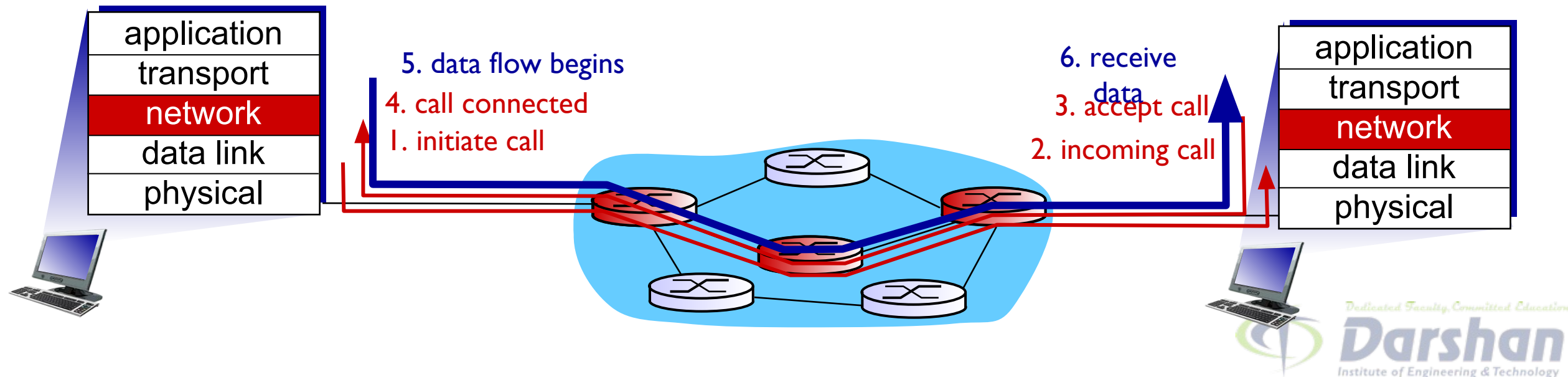
Incoming interface #	Incoming VC #	Outgoing interface	Outgoing VC
1	12	3	
2	22	1	
3	63	2	
17	7		

*VC routers maintain connection state information*

# Virtual Circuit Setup

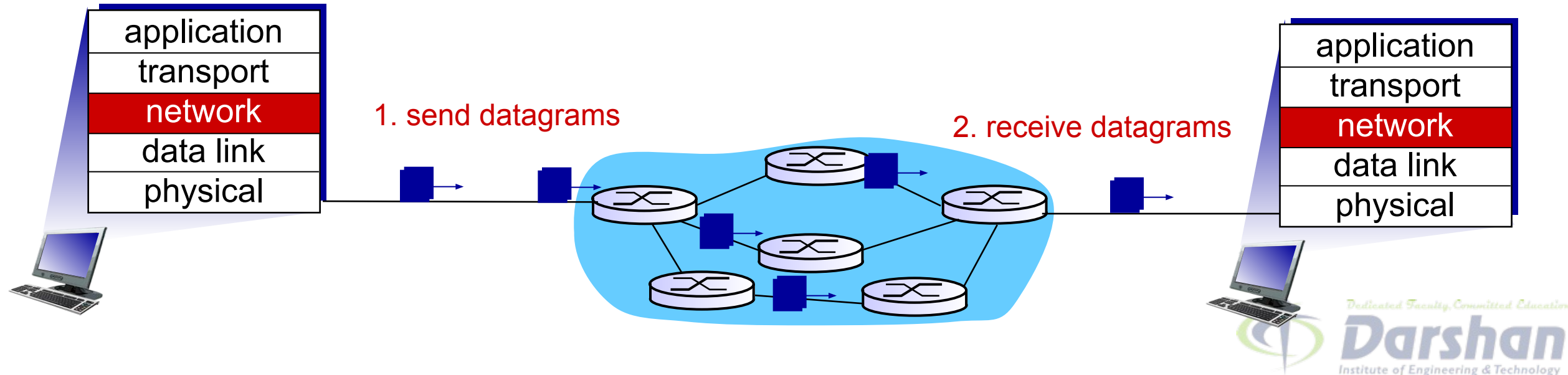
There are three identifiable phases in a virtual circuit:

1. VC setup
2. Data transfer
3. VC teardown



# Datagram Network

- ❑ In connectionless service, packets are injected into the subnet individually and routed independently of each other.
- ❑ No advance setup is needed. The packets are frequently called datagrams and the subnet is called a **datagram subnet**.
- ❑ Only directly-connected lines can be used.



# Datagram Network vs. Virtual Circuit Network

	Datagram	Virtual Circuit
<b>Connection Setup</b>	None	Required
<b>Addressing</b>	Packet contains full source and destination address	Each virtual circuit number entered to table on setup, used for routing.
<b>State Information</b>	None other than router table containing destination network	Route established at setup, all packets follow same route.
<b>Effect of Router Failure</b>	Only on packets lost during crash	All virtual circuits passing through failed router terminated.
<b>Congestion Control</b>	Difficult since all packets routed independently router resource requirements can vary.	Simple by pre-allocating enough buffers to each virtual circuit at setup, since maximum number of circuits fixed.



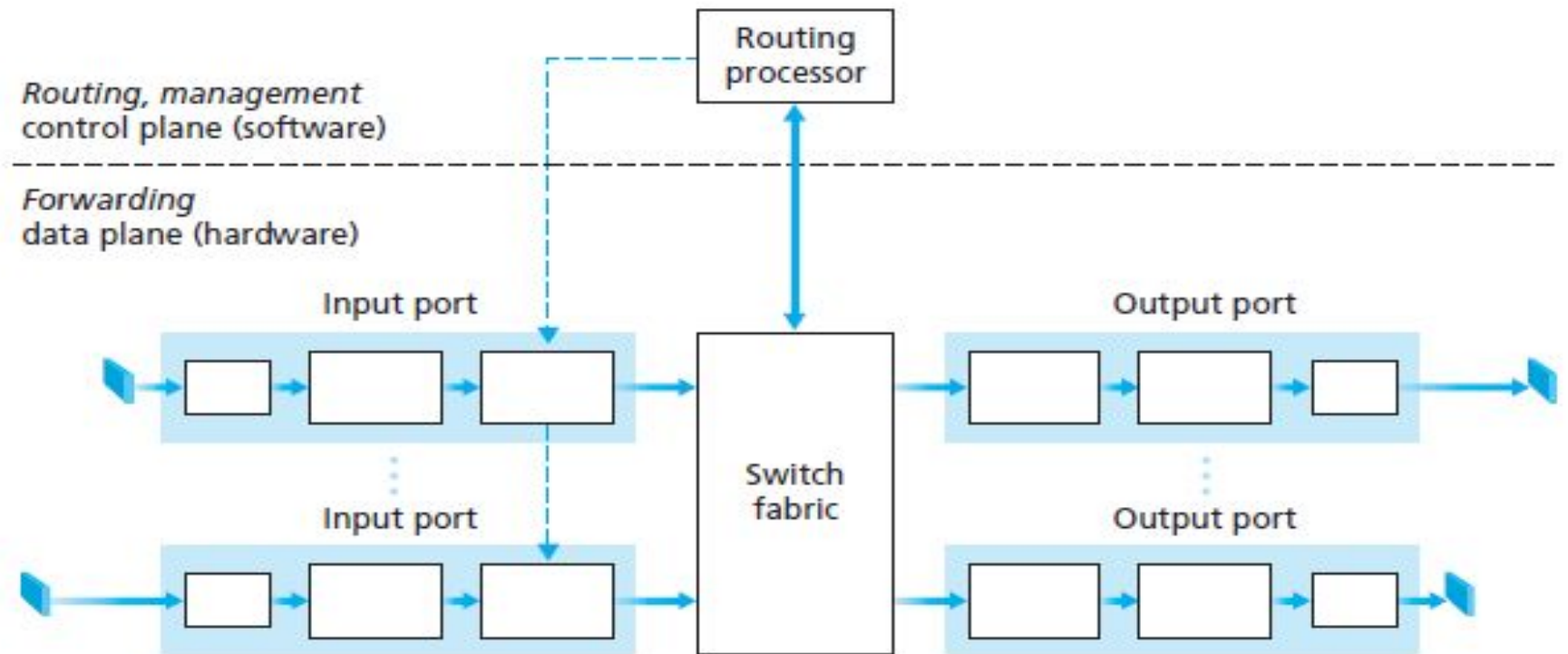
# Router Architecture



# Router Architecture

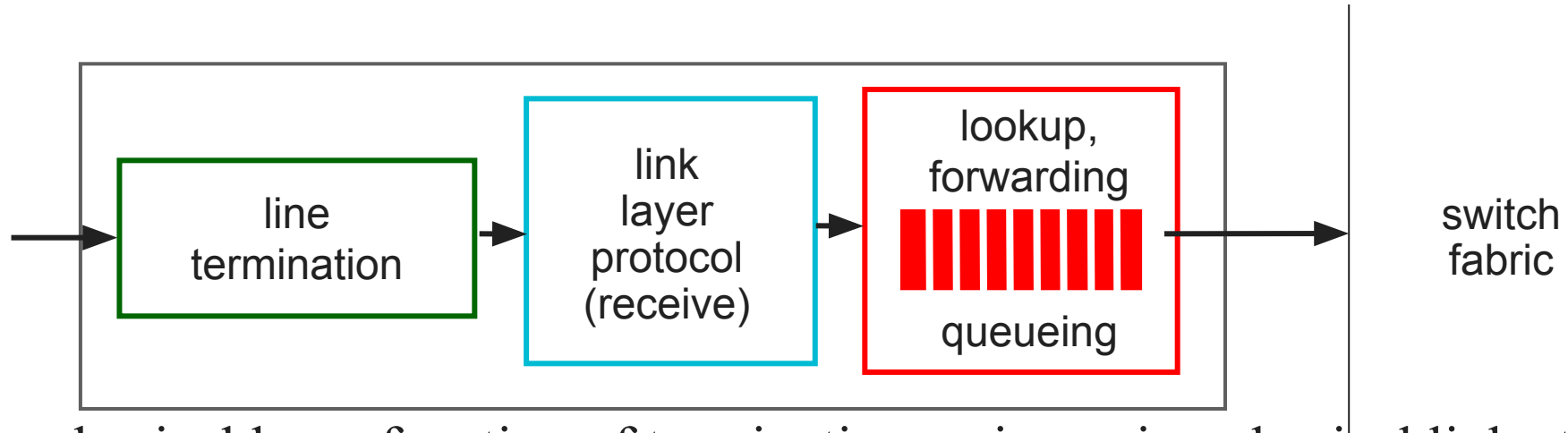
□ Routers have four components:

1. Input ports
2. Switching fabric
3. Output ports
4. Routing processor





# Input Port Functions



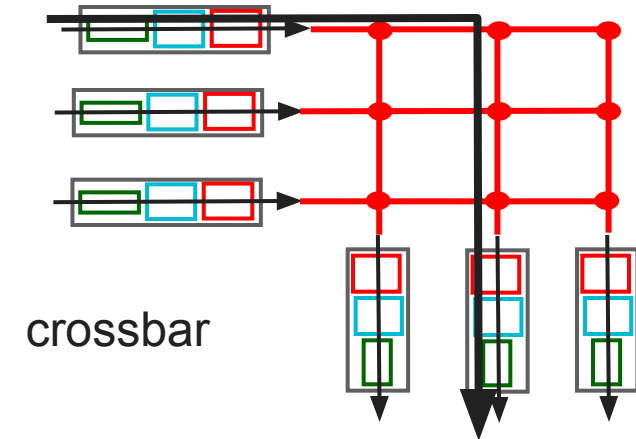
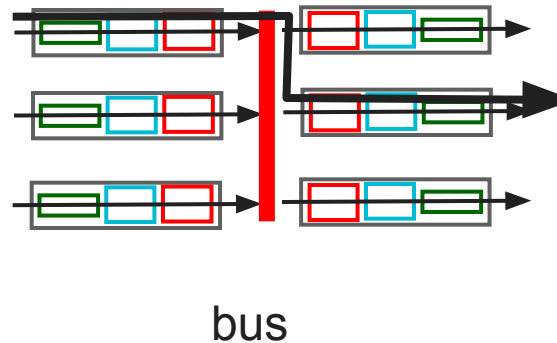
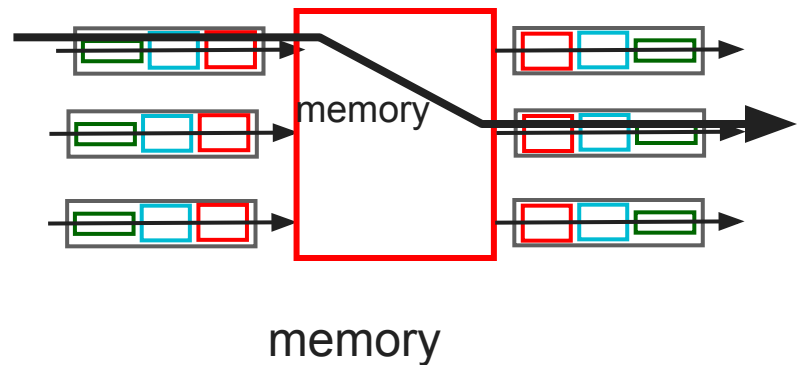
- It performs the physical layer function of terminating an incoming physical link at a router.
- It performs link-layer functions needed to interoperate with the link layer at the other side of the incoming link; this is represented by the middle boxes in the input and output ports.
- A lookup function is performed at the input port; this will occur in the rightmost box of the input port.

# Input Port Functions – Cont...

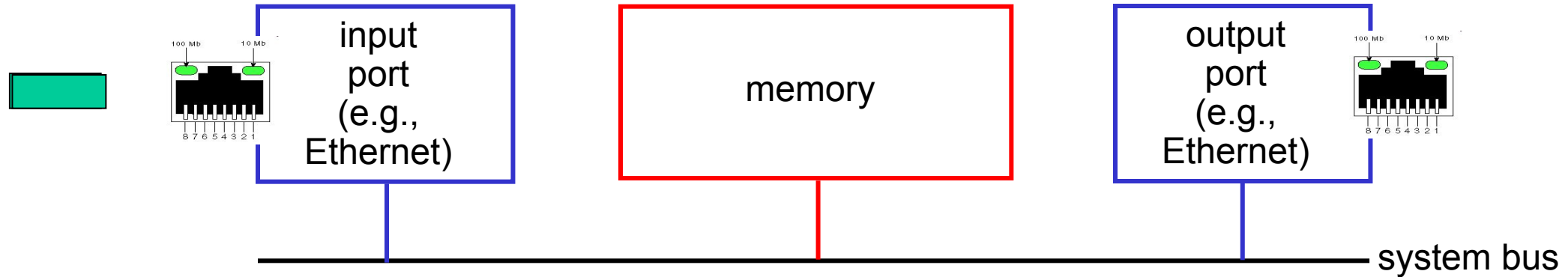
- ❑ The forwarding table is consulted to determine the router output port to which an arriving packet will be forwarded via the switching fabric.
- ❑ Control packets (for example, packets carrying routing protocol information) are forwarded from an input port to the routing processor.

# Switching Fabrics

- It connects the router's input ports to its output ports.
- It is completely contained within the router - a network inside of a network router.
- Switching rate: A rate at which packets can be transfer from inputs to outputs.
- Also measured as multiple of input/output line rate.
- Three types of switching fabrics:



# Switching via Memory



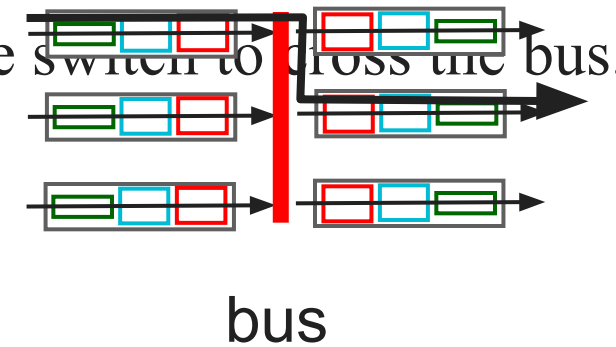
- Like traditional computers, with switching between input and output ports being done under direct control of the CPU (routing processor).
- Packet signalled a routing processor via an interrupt then copied from the input port into processor memory.

# Switching via Memory – Cont...

- ❑ Routing processor extracted the destination address from the header, from forwarding table, and copied the packet to the output port's buffers.
- ❑ If the memory bandwidth is such that  $B$  packets per second can be written into, or read from, then the overall forwarding throughput must be less than  $B/2$ .
- ❑ Two packets cannot be forwarded at the same time, even if they have different destination ports.
- ❑ Only one memory read/write over the shared system bus can be done at a time.

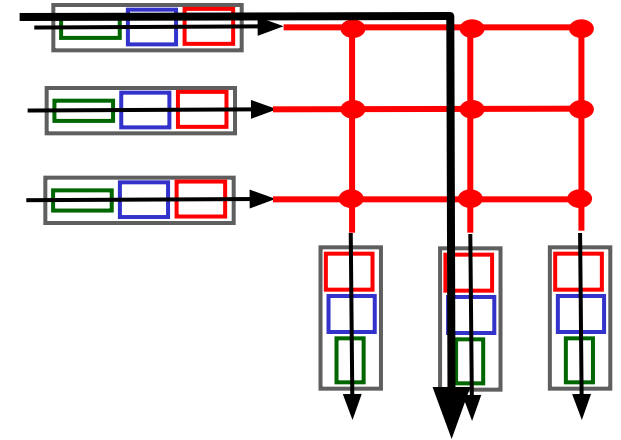
# Switching via bus

- ❑ An input port transfers a packet to the output port over a shared bus, without intervention by the routing processor.
- ❑ The packet is received by all output ports, but only the port that matches the label will keep the packet.
- ❑ The label is then removed at the output port, and only used within the switch to cross the bus.
- ❑ Switching speed of the router is limited to the bus speed.

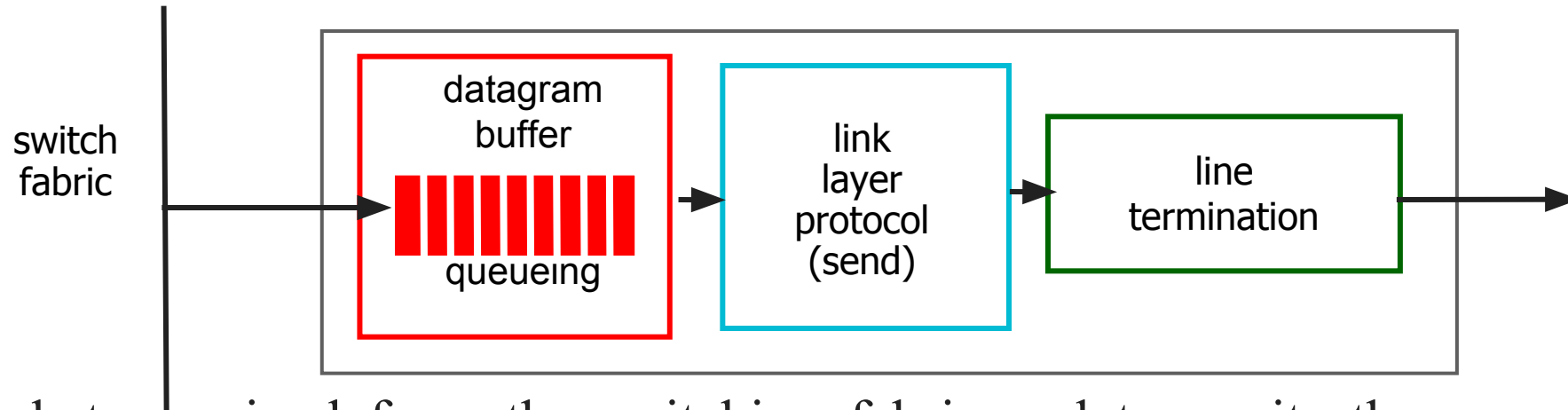


# Switching via an interconnection network

- ❑ Crossbar switch consists  $2N$  buses.
- ❑  $N$  input ports to  $N$  output ports.
- ❑ Its overcome the bandwidth limitation of shared bus; used to interconnect processors in a multiprocessor computer architecture.
- ❑ Cross-point connection can be opened or closed at any time by the switch fabric controller.
- ❑ Crossbar networks are capable of forwarding multiple packets in parallel.



# Output Port



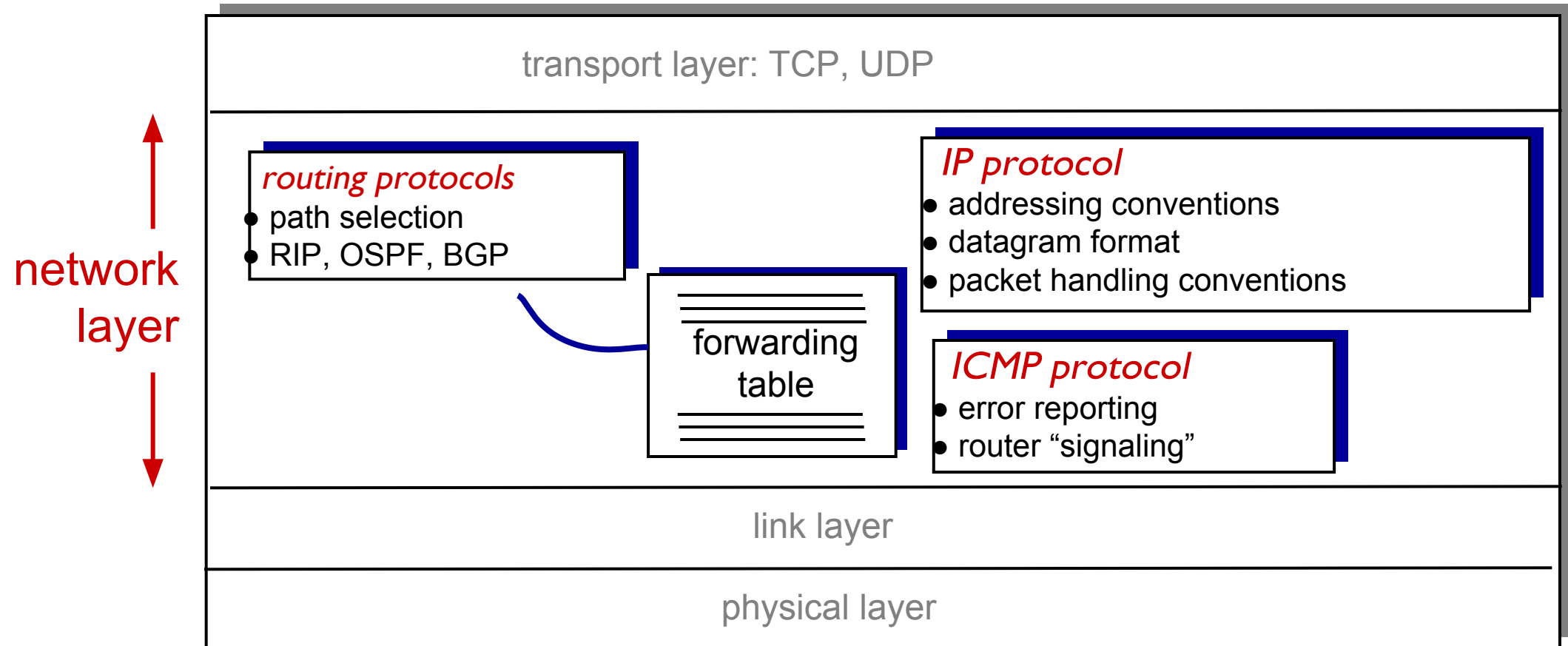
- It buffers packets received from the switching fabric and transmits these packets on the outgoing link.
- Packets can be lost due to congestions lacks of buffers.
- When a link is bidirectional, an output port will typically be paired with the input port for that link on the same line card.



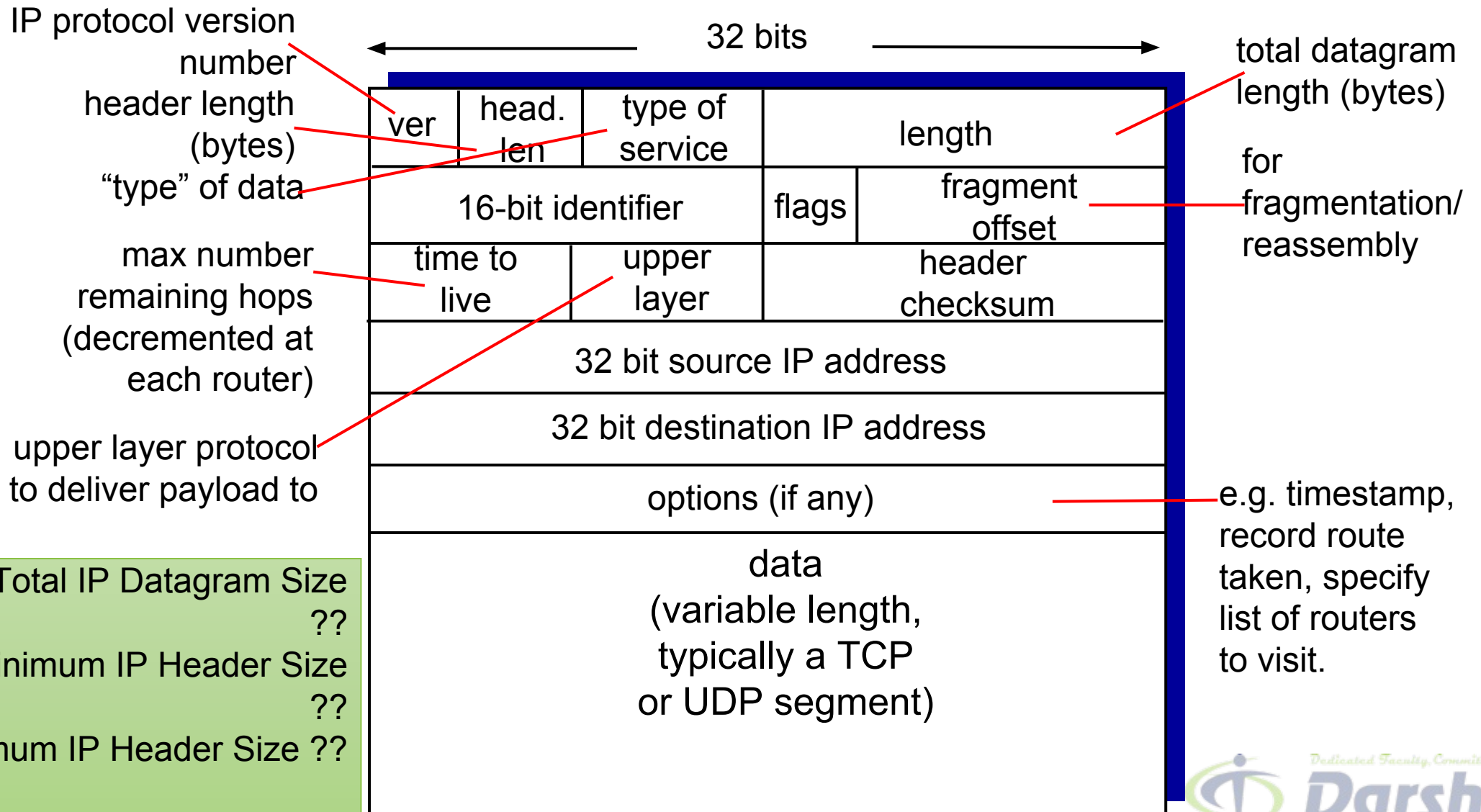
# Routing Processor

- The routing processor executes the routing protocols, maintains routing tables and attached link state information and computes the forwarding table for the router.
- It also performs the network management functions.

# Internet Network Layer



# IPv4 Datagram format



# IP Address

- ❑ IP addresses are useful in identifying a specific host in a network.
- ❑ IP addresses are 32 bit numbers which are divided into 4 octets. Each octet represents 8 bit binary number.
- ❑ Below is an example of an IP address:

10101100

**172**

00010000

**16**

11111110

**254**

00000001

**1**

**IP addresses are divided into 2 parts:**

**Network ID & Host ID**

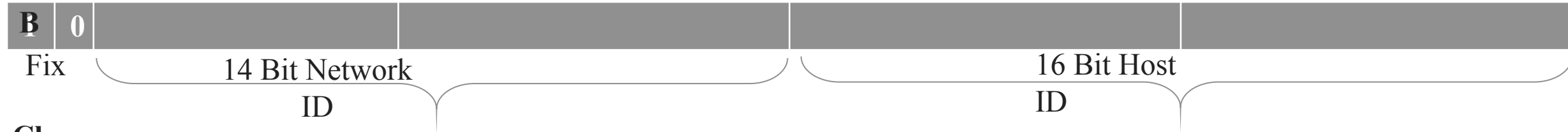
**<NID> <HID> = IP Address**

# Classification of IP Addresses (Classful Addressing)

**Class:**



**Class:**



**Class:**



**Class:**



**Class:**



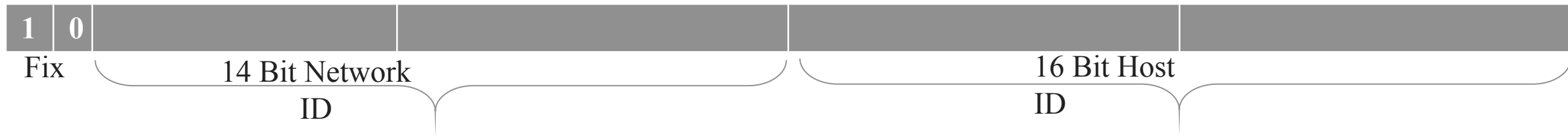
# Class A: (0.0.0.0 to 127.255.255.255)



- Only 126 addresses are used for network address.
- All 0's and 1's in Network-ID are dedicated for special IP address. So, total number of IP address in class A can be represented:

0.0.0.0	Special IP Address
00000001.0.0.1	$2^{24} - 2$ are Host IP
1.0.0.2	
1.0.0.3	
.	
.	
126.255.255.254	
127.255.255.255	Special IP Address – Loopback

# Class B: (128.0.0.0 to 191.255.255.255)



□ No special network address here. All are usable.

128.0.0.0	Special IP Address
10000001.0.0.1	$2^{16} - 2$ are Host IP
130.0.0.2	
130.0.0.3	
.	
.	
190.255.255.254	
10111111.255.255.255	Special IP Address – Loopback

# Class C: (192.0.0.0 to 223.255.255.255)



192.0.0.0	Special IP Address
11000001.0.0.1	$2^8 - 2$ are Host IP
194.0.0.2	
194.0.0.3	
.	
.	
222.255.255.254	
11011111.255.255.255	Special IP Address – Loopback



## Class D: (224.0.0.0 to 239.255.255.255)

- Very first four bits of the first octet in Class D IP addresses are set to 1110, giving a range of:

$$\begin{array}{c} \mathbf{11100000 - 11101111} \\ \mathbf{224 - 239} \end{array}$$

- Class D has IP address range from 224.0.0.0 to 239.255.255.255.
- Class D is reserved for Multicasting.
- In multicasting data is not destined for a particular host, that is why there is no need to extract host address from the IP address, and Class D does not have any subnet mask.

## Class E: (240.0.0.0 to 255.255.255.255)

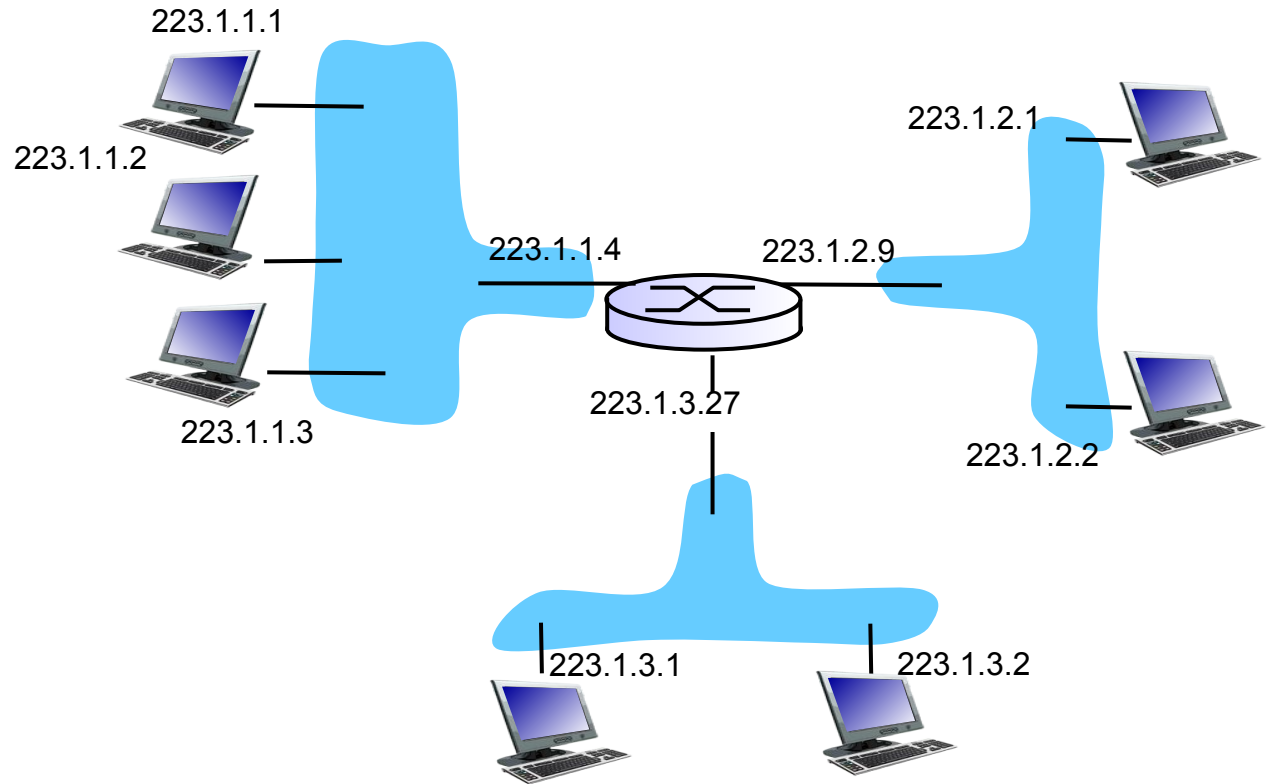
- ❑ This IP Class is reserved for experimental purposes only for R&D or Study.
- ❑ IP addresses in this class ranges from 240.0.0.0 to 255.255.255.254.
- ❑ Like Class D, this class too is not equipped with any subnet mask.

# IP Addressing Summary

Class	Leading bits	Size of <i>network number</i> bit field	Size of <i>rest</i> bit field	Number of networks	Addresses per network	Total addresses in class	Start address	End address	Default <b>subnet mask</b> in <b>dot-decimal</b> notation	<b>CIDR</b> notation
Class A	0	8	24	128 ( $2^7$ )	16,777,216 ( $2^{24}$ )	2,147,483,648 ( $2^{31}$ )	0.0.0.0	127.255.255.255	255.0.0.0	/8
Class B	10	16	16	16,384 ( $2^{14}$ )	65,536 ( $2^{16}$ )	1,073,741,824 ( $2^{30}$ )	128.0.0.0	191.255.255.255	255.255.0.0	/16
Class C	110	24	8	2,097,152 ( $2^{21}$ )	256 ( $2^8$ )	536,870,912 ( $2^{29}$ )	192.0.0.0	223.255.255.255	255.255.255.0	/24
Class D ( <b>multicast</b> )	1110	not defined	not defined	not defined	not defined	268,435,456 ( $2^{28}$ )	224.0.0.0	239.255.255.255	not defined	not defined
Class E (reserved)	1111	not defined	not defined	not defined	not defined	268,435,456 ( $2^{28}$ )	240.0.0.0	255.255.255.255	not defined	not defined

# IP Addressing - Example

- ❑ **IP address:** It is 32-bit identifier for host, router interface
- ❑ **Interface:** It is a connection between host/router and physical link.
  - ❑ A router's typically have multiple interfaces
  - ❑ A host typically has one or two interfaces
- ❑ IP addresses associated with each interface.

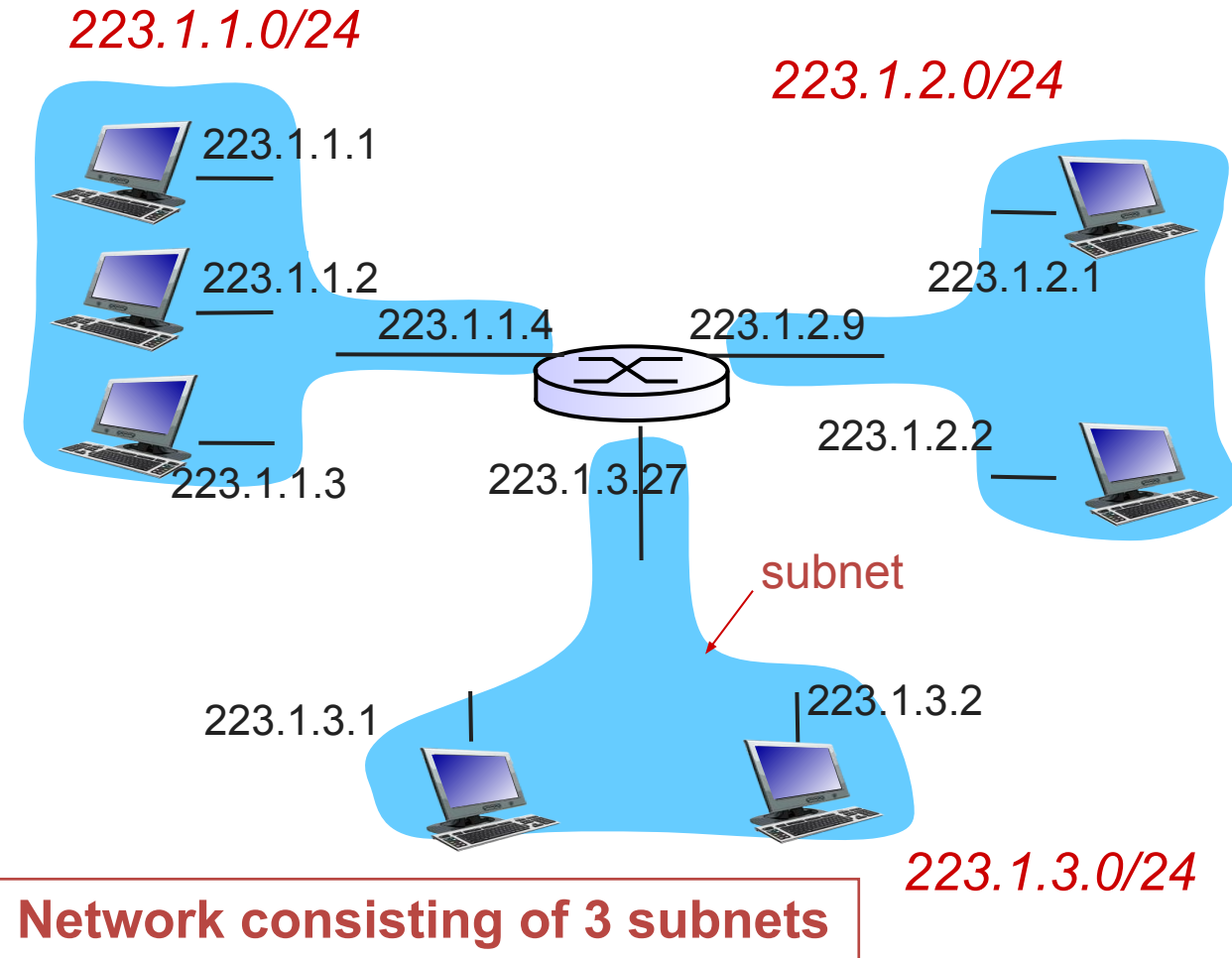


223.1.1.1 = 11011111      00000001      00000001  
00000001

223      1      1      1

# Subnet

- ❑ **Subnet:** high order bits defines subnet
- ❑ **Host:** low order bits defines host
- ❑ To determine the subnets, detach each interface from its host or router.
- ❑ Creating islands of isolated networks, with interfaces terminating the end points of the isolated networks.
- ❑ Each of these isolated networks is called a **subnet**.



# Type of addresses in IPv4 Network

❑ **Network address** - The address by which we refer to the network.

❑ E.g.: 10.0.0.0

❑ **Broadcast address** - A special address used to send data to all hosts in the network.

❑ The broadcast address uses the highest address in the network range.

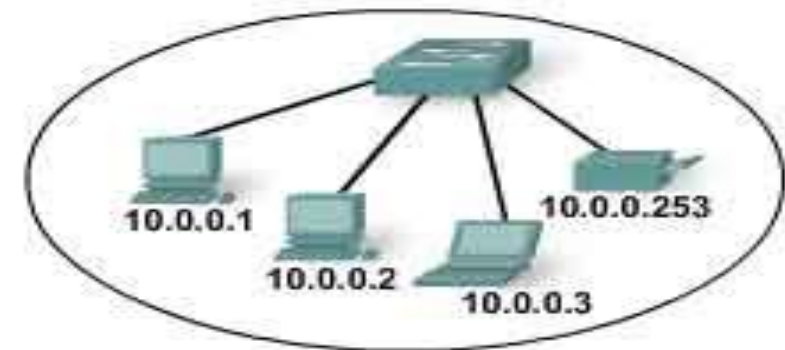
❑ E.g.: 10.0.0.255

❑ **Host addresses** - The addresses assigned to the end devices in the network.

❑ E.g.: 10.0.0.1

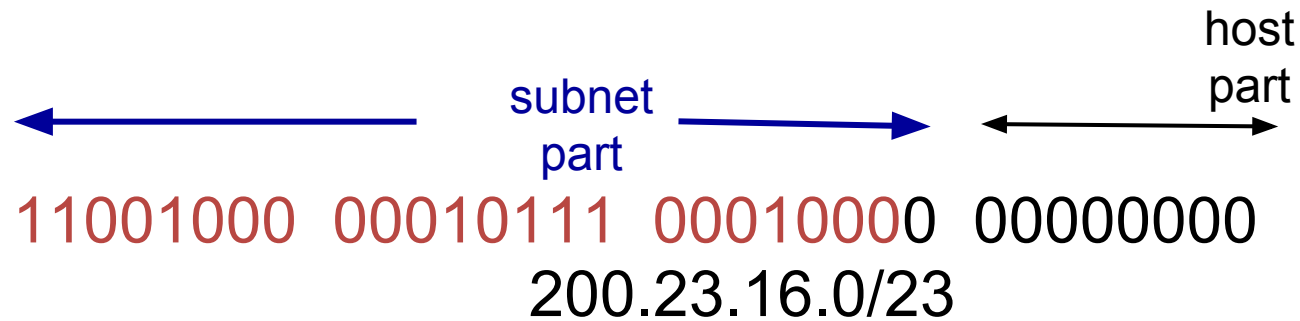
# Type of addresses – Cont...

	Address Types			
	Network			Host
Network Address	10	0	0	0
	00001010	00000000	00000000	00000000
Broadcast Address	10	0	0	255
	00001010	00000000	00000000	11111111
Host Address	10	0	0	1
	00001010	00000000	00000000	00000001



# Classless Inter-Domain Routing(CIDR)

- CIDR is a slash notation of subnet mask. CIDR tells us number of on bits in a network address.



- A single IP address can be used to designate many unique IP addresses with CIDR.
- A CIDR IP address looks like a normal IP address except that it ends with a slash followed by a number, called the **IP network prefix**.
- CIDR addresses reduce the size of routing tables and make more IP addresses available within organizations.



# Subnetting

- Subnetting take places when we extend the default subnet mask.
- We cannot perform subnetting with default subnet mask and every classes have default subnet mask.
- Now find the host bits borrowed to create subnets and convert them in decimal.
- For example find the subnet mask of address 188.25.45.48/20 ?
  1. Class B, Default Subnet mask: 255.255.0.0
  2. Borrowed 4 bit from host part so mask is now:

11111111 11111111 11110000 00000000  
255            255            240            0

# How many subnets from given subnet mask?

- To calculate the number of subnets provided by given subnet mask we use  $2^N$ , where  $N$  = number of bits borrowed from host bits to create subnets.
- For example, 192.168.1.0/27,  $N$  is 3.
- By looking at address we can determined that this address is belong to class C and default subnet mask 255.255.255.0 [/24 in CIDR].
- In given address we borrowed  $27 - 24 = 3$  host bits to create subnets.
- Now  $2^3 = 8$ , so our answer is 8.

# What are the valid subnets?

- Calculating valid subnet is two steps process.
- First calculate **total subnet** by using formula  $2^N$ .
- In second step find the **block size** and count from zero in block until subnet mask value.
- For example calculate the valid subnets for 192.168.1.0/26
  1. Borrowed host bits are 2 [26-24]
  2. Total subnets are  $2^2 = 4$
  3. Subnet mask would be 255.255.255.192
  4. Block size would be  $256 - 192 = 64$
  5. Start counting from zero at blocks of 64, so our valid subnets would be 0,64,128,192

# What are the total hosts?

- Total hosts are the hosts available per subnet
- To calculate total hosts use formula  $2^H = \text{Total hosts}$
- H is the number of host bits
- For example in address 192.168.1.0/26
- We have 32 - 26
  1. [Total bits in IP address - Bits consumed by network address] = 6
  2. Total hosts per subnet would be  $2^6 = 64$

# Network Prefixes

- ❑ For Class C, Default subnet mask of class C is 255.255.255.0
- ❑ CIDR notation of class C is /24, which means 24 bits from IP address are already consumed by network portion.
- ❑ We have 8 host bits remain.
- ❑ Subnetting moves from left to right. So Class C subnet masks can only be the following:

CIDR	Decimal	Binary
/25	128	10000000
/26	192	11000000
/27	224	11100000
/28	240	11110000
/29	248	11111000
/30	252	11111100

# Network Prefixes- Example

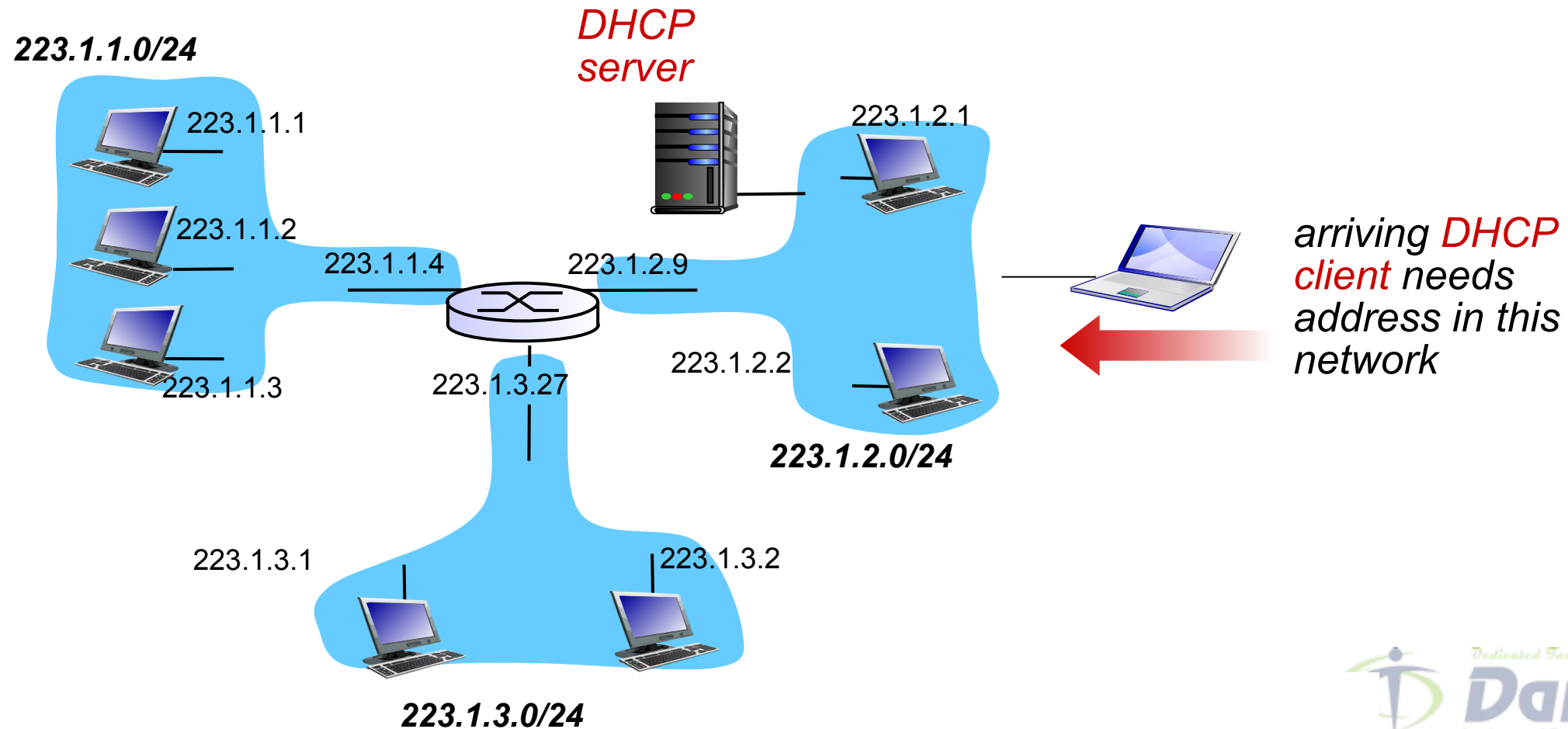
## □ /25

- CIDR /25 has subnet mask 255.255.255.128 and 128 is 10000000 in binary.
- We used one host bit in network address.
- $N = 1$  [Number of host bit]
- $H = 7$  [Remaining host bits]
- Total subnets (  $2^N$  ) :  $2^1 = 2$
- Block size (256 - subnet mask) :-  $256 - 128 = 128$
- Valid subnets ( Count blocks from 0 ) :- 0, 128
- Total hosts ( $2^H$ ) :-  $2^7 = 128$
- Valid hosts per subnet ( Total host - 2 ) :-  $128 - 2 = 126$



# Dynamic Host Configuration Protocol - DHCP

- Dynamic Host Configuration Protocol is a protocol for assigning dynamic IP addresses to devices on a network.

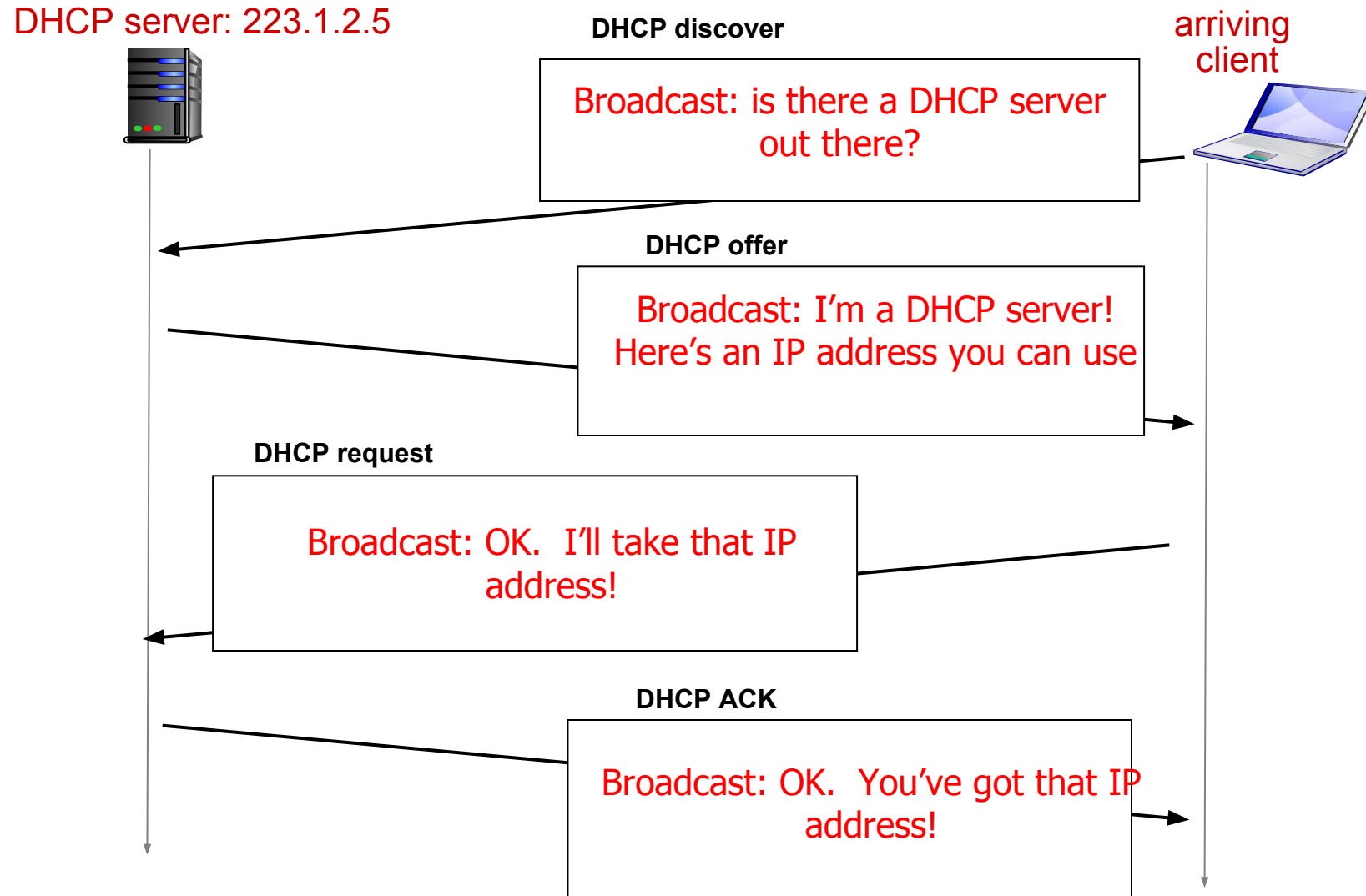


# DHCP – Cont...

- ❑ With dynamic addressing, a device can have a different IP address every time it connects to the network.
- ❑ In some systems, the device's IP address can even change while it is still connected.
- ❑ It allows reuse of addresses (only hold address while connected “on”).
- ❑ It also support mobile users who want to join network.

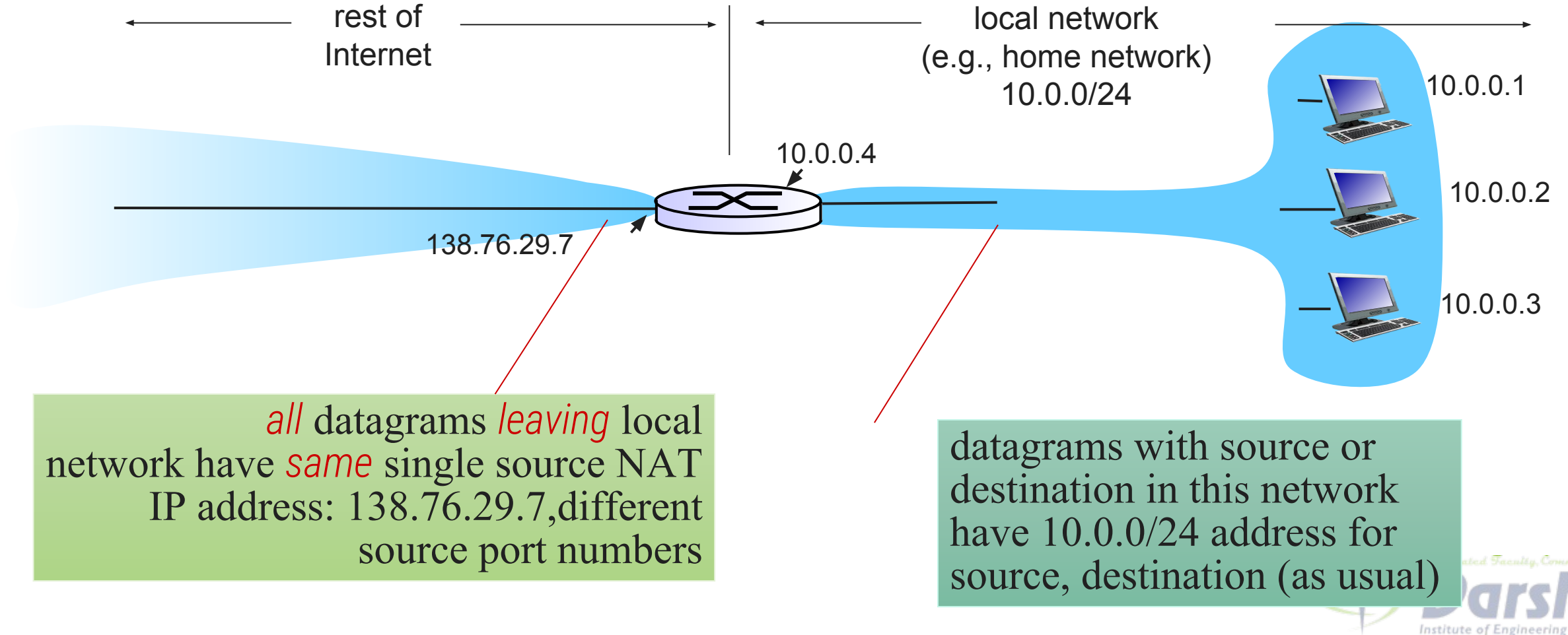


# DHCP Client Server Interaction



# Network Address Translation

□ NAT is a method that is used to translate Private IP addresses to Public IP addresses.



# NAT – Cont...

- ❑ Local network uses just one IP address as far as outside world is concerned.
- ❑ This means that only a single, unique IP address is required to represent an entire group of computers.
- ❑ The technique was originally used for ease of rerouting traffic in IP networks without readdressing every host.
- ❑ The concept of NAT as developed to solve two problems:
  - ❑ Solve shortage of IPv4 IP addresses
  - ❑ To Hide the Network Address

# NAT Terminology

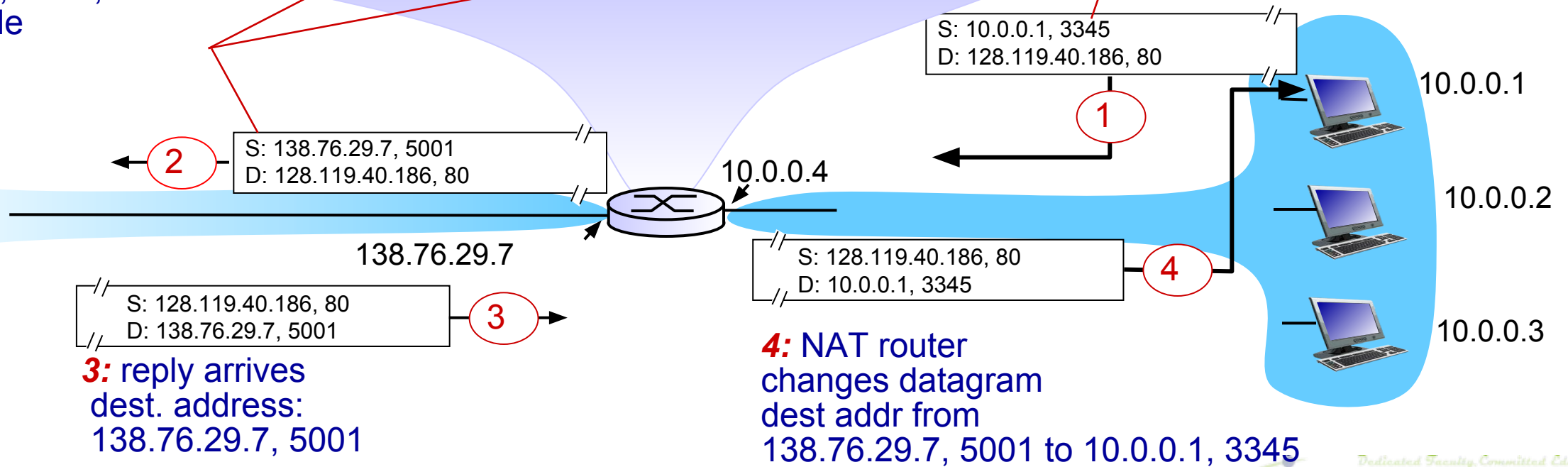
- ❑ **Inside Local Address:** The name of the inside source address before translation. This would be a Private IP.
- ❑ **Inside Global Address:** The name of the inside host after translation. This would be the Public IP.
- ❑ **Outside Local Address:** The name of the destination host before translation.
- ❑ **Outside Global Address:** The name of the destination host after translation.
- ❑ **Where:**
  - ❑ Global Addresses → Public
  - ❑ Local Addresses → Private
  - ❑ Inside Hosts → Within Local Network
  - ❑ Outside Hosts → Outside Local Network

# NAT – Cont...

**2:** NAT router changes datagram source addr from 10.0.0.1, 3345 to 138.76.29.7, 5001, updates table

NAT translation table	
WAN side addr	LAN side addr
138.76.29.7, 5001	10.0.0.1, 3345
.....	.....

**1:** host 10.0.0.1 sends datagram to 128.119.40.186, 80

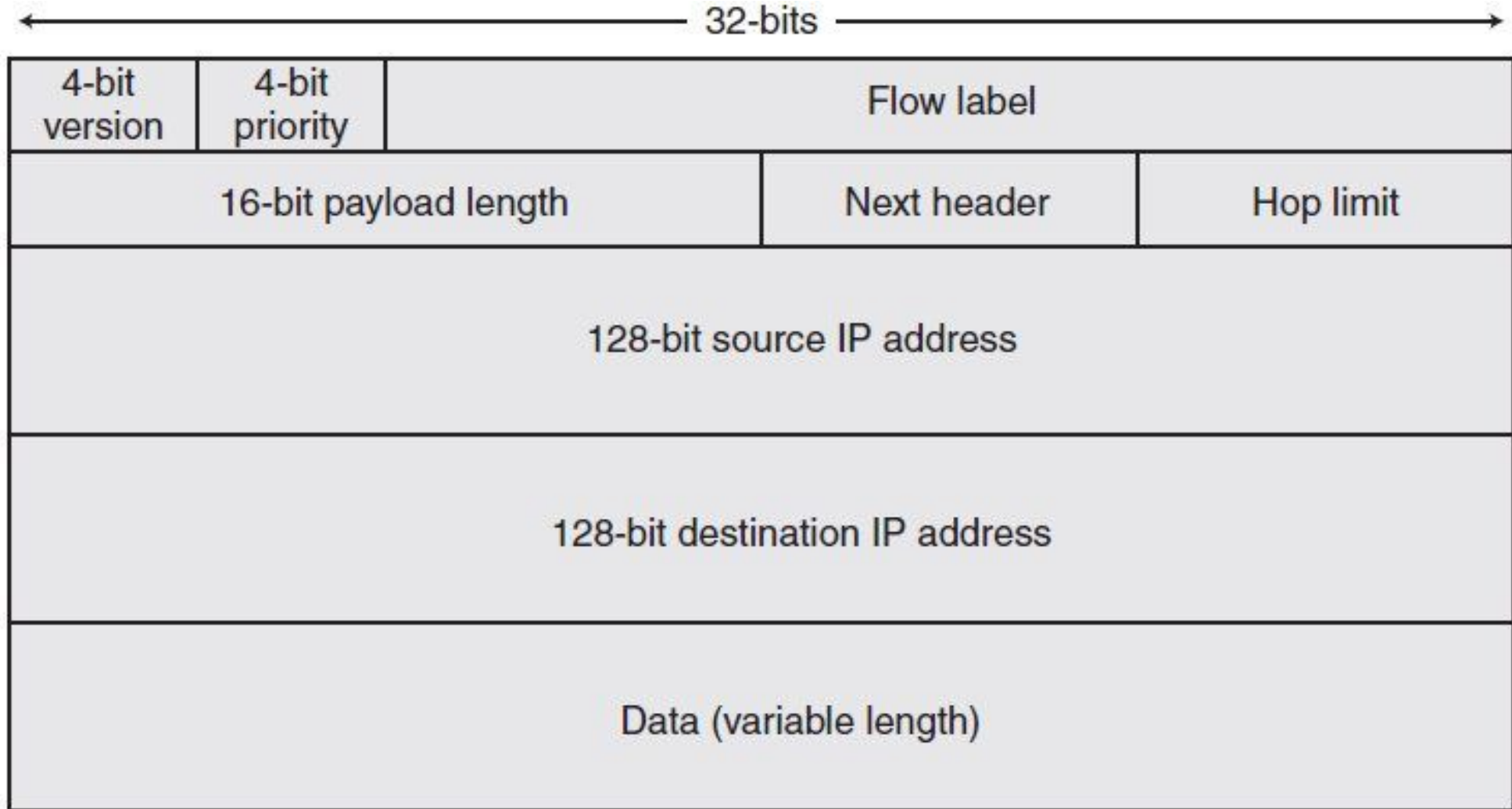


# Internet Control Message Protocol - ICMP

- When something unexpected occurs, the event is reported by the ICMP, which is also used to test the Internet.
- About a dozen types of ICMP messages are defined. The most important are listed below. Each ICMP message type is encapsulated in an IP packet.

Message Type	Description
Destination unreachable	Packet could not be delivered
Time exceeded	Time to live field hit 0
Parameter problem	Invalid header field
Source quench	Choke packet
Redirect	Teach a router about geography
Echo	Ask a machine if it is alive
Echo reply	Yes, I am alive
Timestamp request	Same as Echo request, but with timestamp
Timestamp reply	Same as Echo reply, but with timestamp

# IPv6 Datagram Format



# Difference between IPv4 & IPv6

IPv4	IPv6
✓ 32 bit length	✓ 128 bit length
✓ Fragmentation is done by sender and forwarding routers	✓ Fragmentation is done only by sender
✓ No packet flow identification	✓ Packet flow identification is available within the IPv6 header using the Flow Label field
✓ Checksum field in header	✓ No checksum field in header
✓ Options fields are available in header	✓ No option fields, but Extension headers are available
✓ Address Resolution Protocol (ARP) is available to map IPv4 addresses to MAC addresses	✓ Address Resolution Protocol (ARP) is replaced with Neighbor Discovery Protocol
✓ Broadcast messages are available	✓ Broadcast messages are not available
✓ Static IP addresses or DHCP is required to configure IP addresses	✓ Auto-configuration of addresses is available



# Link State Routing Algorithm

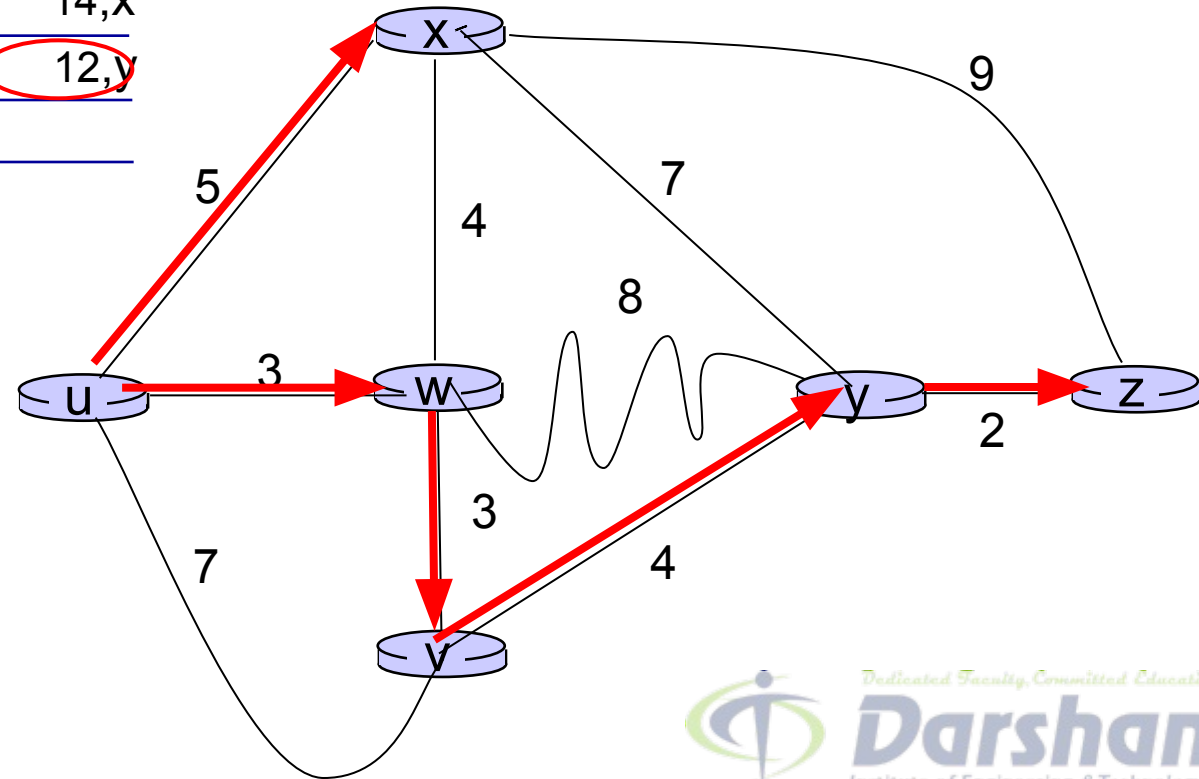
- Also known as Dijkstra's Algorithm.
- It computes the least-cost path from one node (source node) to all other nodes in the network.
- Its iterative and after the  $k^{\text{th}}$  least-cost paths are known to  $k$  destination nodes.
- **Notation:**
  - $c(x,y)$ : link cost from node  $x$  to  $y$ ;  $= \infty$  if not direct neighbours
  - $D(v)$ : current value of cost of path from source to destination  $v$
  - $p(v)$ : predecessor node along path from source to  $v$
  - $N'$ : set of nodes whose least cost path definitively known

# Dijkstra's Algorithm

```
1 Initialization:
2   $N' = \{u\}$ 
3  for all nodes  $v$ 
4    if  $v$  adjacent to  $u$ 
5      then  $D(v) = c(u,v)$ 
6    else  $D(v) = \infty$ 
7
8 Loop
9  find  $w$  not in  $N'$  such that  $D(w)$  is a minimum
10 add  $w$  to  $N'$ 
11 update  $D(v)$  for all  $v$  adjacent to  $w$  and not in  $N'$ 
12 :
13    $D(v) = \min( D(v), D(w) + c(w,v) )$ 
14   /* new cost to  $v$  is either old cost to  $v$  or known
15   shortest path cost to  $w$  plus cost from  $w$  to  $v$  */
16 until all nodes in  $N'$ 
```

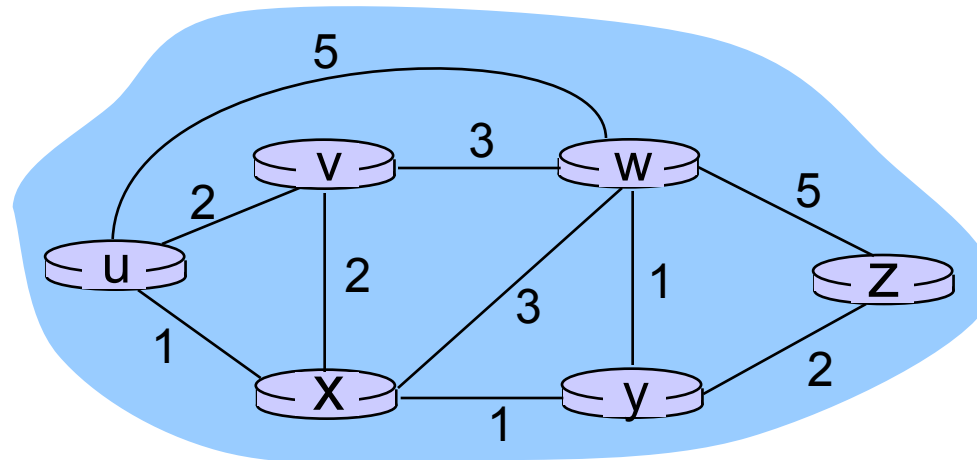
# Dijkstra's Algorithm – Example:1

Step	N'	D( <b>v</b> ) p(v)	D( <b>w</b> ) p(w)	D( <b>x</b> ) p(x)	D( <b>y</b> ) p(y)	D( <b>z</b> ) p(z)
0	u	7,u	3,u	5,u	$\infty$	$\infty$
1	uw	6,w		5,u	11,w	$\infty$
2	uwx	6,w			11,w	14,x
3	uwxv				10,v	14,x
4	uwxvy					12,y
5	uwxvyz					



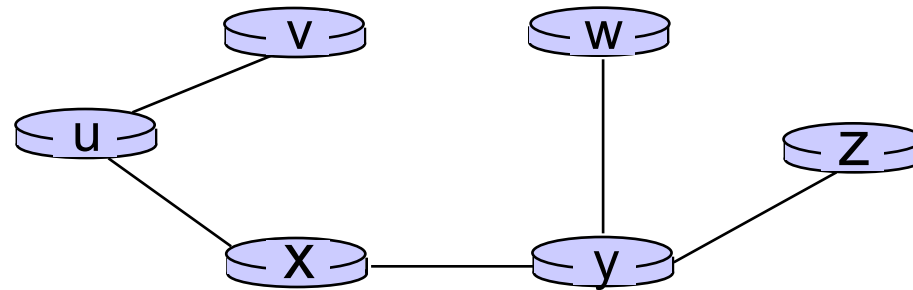
# Dijkstra's Algorithm – Example:2

Step	N'	D(v),p(v)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	u	2,u	5,u	1,u	$\infty$	$\infty$
1	ux	2,u	4,x		2,x	$\infty$
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					



# Dijkstra's Algorithm – Example:2

resulting shortest-path tree from u:



resulting forwarding table in  
u:

destination	link
v	(u,v)
x	(u,x)
y	(u,x)
w	(u,x)
z	(u,x)

# Distance Vector Algorithm

- ❑ Distance-vector (DV) algorithm is iterative, asynchronous, and distributed.
- ❑ It is distributed in that each node receives some information from one or more of its directly attached neighbours, performs a calculation, and then distributes the results of its calculation back to its neighbours.
- ❑ It is iterative. so, process continues on until no more information is exchanged between neighbours.
- ❑ The algorithm is asynchronous. It does not require all of the nodes to operate with each other.

# Distance Vector Algorithm – Cont...

- Let  $d_x(y)$  be the cost of the least-cost path from node  $x$  to node  $y$ .
- Then least costs are related by the celebrated Bellman-Ford equation:

$d_x(y)$  = cost of least-cost path from  $x$  to  $y$  then

$$d_x(y) = \min_v \{ c(x, v) + d_v(y) \}$$

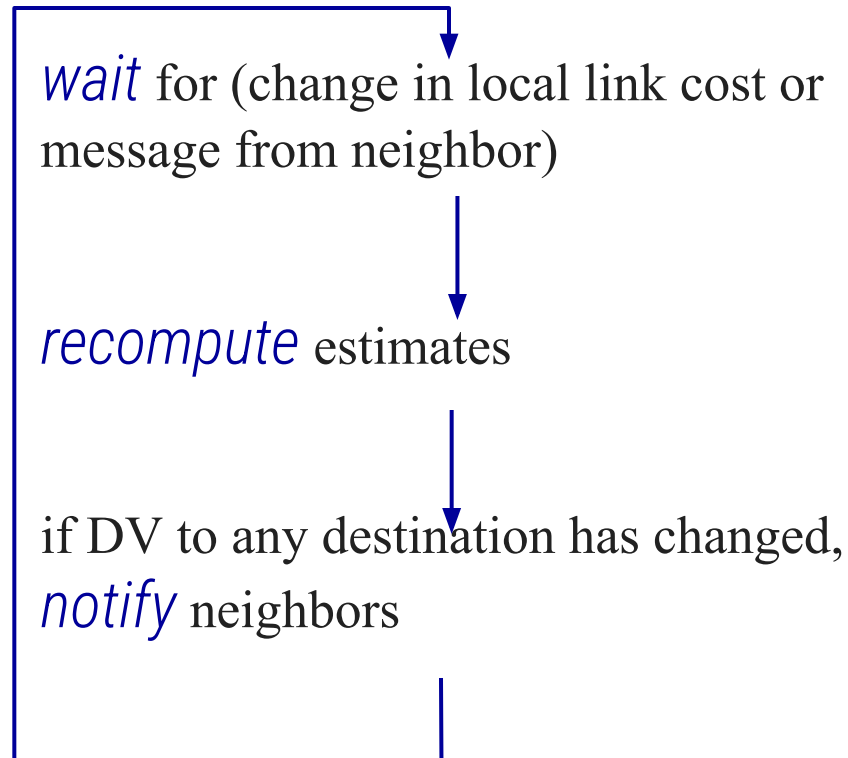
cost from neighbor  $v$  to destination  $y$

cost to neighbor  $v$

$\min$  taken over all neighbors  $v$  of  $x$

# Distance Vector Algorithm – Cont...

*each node:*





# Distance Vector Algorithm - Example

**node x table**

	cost to		
	x	y	z
from x	0	2	7
from y	$\infty$	$\infty$	$\infty$
from z	$\infty$	$\infty$	$\infty$

**node y table**

	cost to		
	x	y	z
from x	$\infty$	$\infty$	$\infty$
from y	2	0	1
from z	$\infty$	$\infty$	$\infty$

**node z table**

	cost to		
	x	y	z
from x	$\infty$	$\infty$	$\infty$
from y	$\infty$	$\infty$	$\infty$
from z	7	1	0

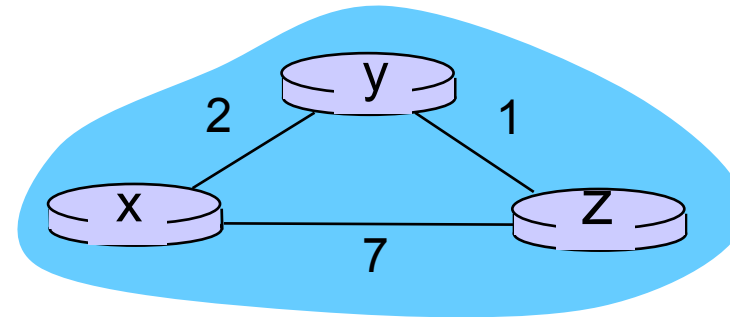
	cost to		
	x	y	z
from x	0	2	3
from y	2	0	1
from z	7	1	0

$$D_x(y) = \min\{c(x,y) + D_y(y), c(x,z) + D_z(y)\}$$

$$= \min\{2+0, 7+1\} = 2$$

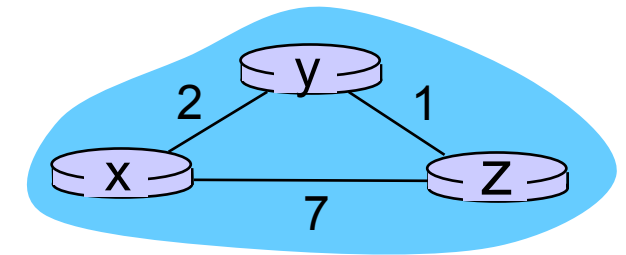
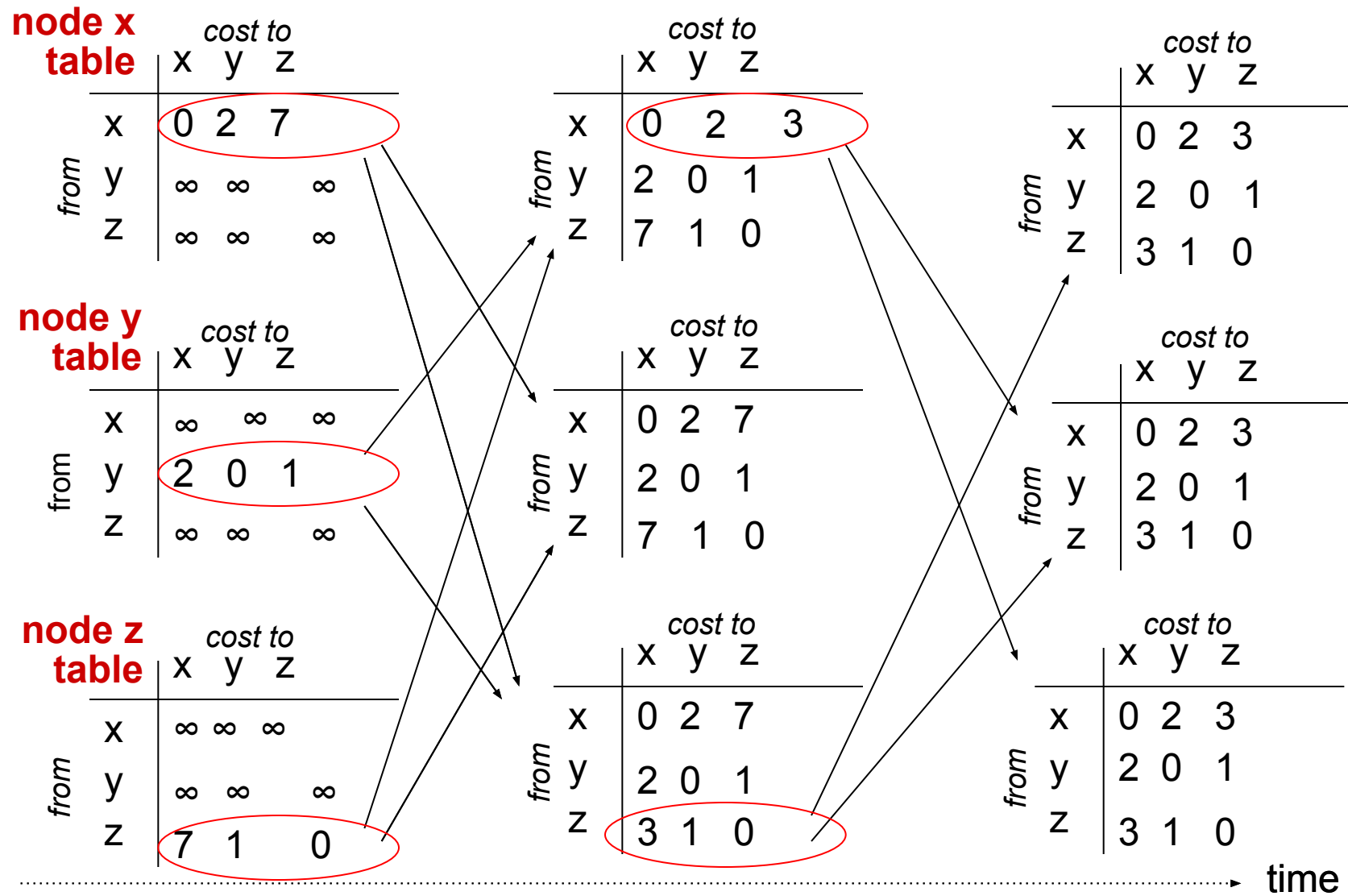
$$D_x(z) = \min\{c(x,y) + D_y(z), c(x,z) + D_z(z)\}$$

$$= \min\{2+1, 7+0\} = 3$$



time

# Distance Vector Algorithm - Example



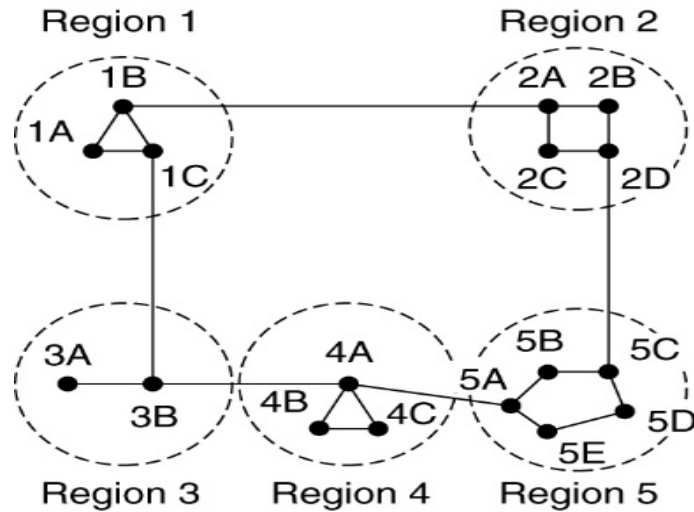
# Difference: LS and DV Routing Algorithm

Distance Vector Protocol	Link State Protocol
Entire routing table is sent as an update	Updates are incremental & entire routing table is not sent as update
Distance vector protocol send periodic update at every 30 or 90 second	Updates are triggered not periodic
Update are broadcasted	Updates are multicasted
Updates are sent to directly connected neighbour only	Update are sent to entire network & to just directly connected neighbour
Routers don't have end to end visibility of entire network.	Routers have visibility of entire network of that area only.
It is prone to routing loops	No routing loops

# Hierarchical Routing

- ❑ As networks grow in size, the router routing tables grow proportionally.
- ❑ Router memory, CPU time and more bandwidth consumed to send status reports about them.
- ❑ When hierarchical routing is used, the routers are divided into what called **regions**.
- ❑ Each router knowing all the details about how to route packets to destinations within its own region.
- ❑ But knowing nothing about the internal structure of other regions.

# Hierarchical Routing - Example



(a)

Full table for 1A

Dest.	Line	Hops
1A	—	—
1B	1B	1
1C	1C	1
2A	1B	2
2B	1B	3
2C	1B	3
2D	1B	4
3A	1C	3
3B	1C	2
4A	1C	3
4B	1C	4
4C	1C	4
5A	1C	4
5B	1C	5
5C	1B	5
5D	1C	6
5E	1C	5

(b)

Hierarchical table for 1A

Dest.	Line	Hops
1A	—	—
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

(c)

# Broadcast Routing

- ❑ Host need to send messages to many or all other hosts.
- ❑ For example
  - ❑ A service distributing weather reports
  - ❑ Stock market updates
  - ❑ Live radio programs
- ❑ In Short, Sending a packet to all destinations simultaneously is called broadcasting.
- ❑ **First broadcasting** method that simply send a distinct packet to each destination.
- ❑ So, it waste of bandwidth, but it also requires the source to have a complete list of all destinations.
- ❑ In practice this may be the only possibility, but it is the least desirable of the methods.

# Broadcast Routing – Cont...

- ❑ Flooding is **Second method**. Although flooding is for ordinary point-to-point communication, for broadcasting it might rate serious consideration, especially if none of the methods are applicable.
- ❑ The problem with flooding as a broadcast technique is the same problem it has as a point-to-point routing algorithm.
- ❑ It generates too many packets and consumes too much bandwidth.
- ❑ A **third algorithm** is Multi Destination Routing.
- ❑ If this method is used, each packet contains either a list of destinations or a bit map indicating the desired destinations.

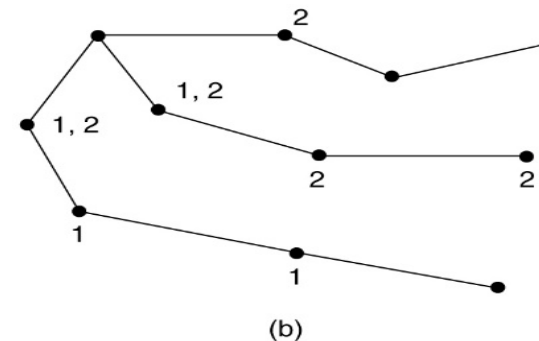
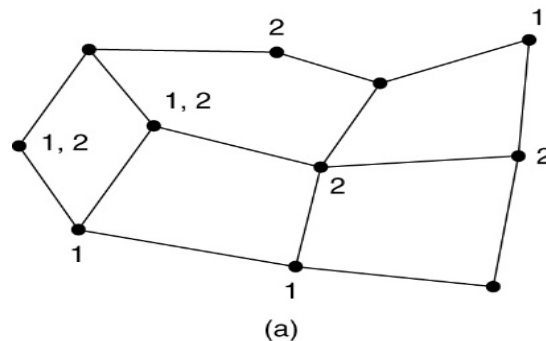
# Broadcast Routing – Cont...

- When a packet arrives at a router, the router checks all the destinations to determine the set of output lines that will be needed.
- The router generates a new copy of the packet for each output line to be used and includes in each packet only those destinations that are to use the line.
- A **fourth broadcast algorithm** makes explicit use of the sink tree for the router initiating the broadcast-or any other convenient spanning tree for that matter.
- A spanning tree is a subset of the subnet that includes all the routers but contains no loops.
- If each router knows which of its lines belong to the spanning tree, it can copy an incoming broadcast packet onto all the spanning tree lines except the one it arrived on.



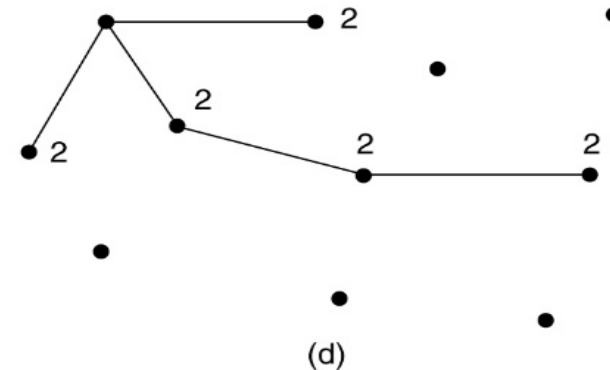
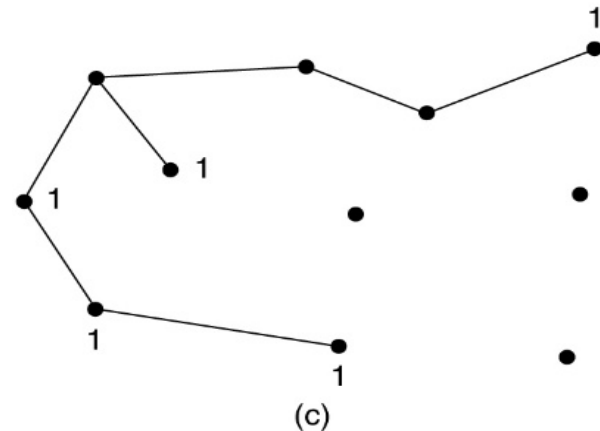
# Multicast Routing

- Sending a message to a group is called **multicasting**, and its routing algorithm is called **multicast routing**.
- Multicasting requires group management. Need to create and destroy groups, and to allow processes to join and leave groups.
- To do multicast routing, each router computes a spanning tree covering all other routers.
- For example, in Figure (a) we have two groups, 1 and 2.
- Some routers are attached to hosts that belong to one or both of these groups, as indicated in the figure.



# Multicast Routing – Cont...

- A spanning tree for the leftmost router is shown in Figure (b).
- When a process sends a multicast packet to a group, the first router examines its spanning tree and prunes it, removing all lines that do not lead to hosts that are members of the group.
- In our example, Figure (c) shows the pruned spanning tree for group 1.
- Figure(d) shows the pruned spanning tree for group 2. Multicast packets are forwarded only along the appropriate spanning tree.



# Comparison between RIP, OSPF and BGP

RIP	OSPF	BGP
RIP is intra domain routing protocol used with in the autonomous system	OSPF is also intra domain routing protocol used with in the autonomous system	It is inter domain routing protocol used between the autonomous system
RIP is used for Small networks with maximum number of hops 16	OSPF is used in large autonomous system with no limitation	The BGP protocol is used for very large-scale networks
RIP uses Distance Vector	OSPF uses Link State	BGP uses Path Vector
RIP send entire routing update to all directly connected interface	OSPF send multicast Hello packet to the neighbours, to create session	BGP send Open packet to the neighbours to create session
RIP use Bellman ford Algorithm	OSPF use Dijkstra Algorithm	BGP use Path-Vector Routing

Computer Networks (CN)

GTU #3150710



*Dedicated Faculty, Committed Education*  
**Darshan**  
Institute of Engineering & Technology

*Than  
k  
You*



**Prof. Maulik Trivedi**

Computer Engineering Department

Darshan Institute of Engineering & Technology,

Rajkot

✉ [maulik.trivedi@darshan.ac.in](mailto:maulik.trivedi@darshan.ac.in)

☎ 9998265805

