Internetworking

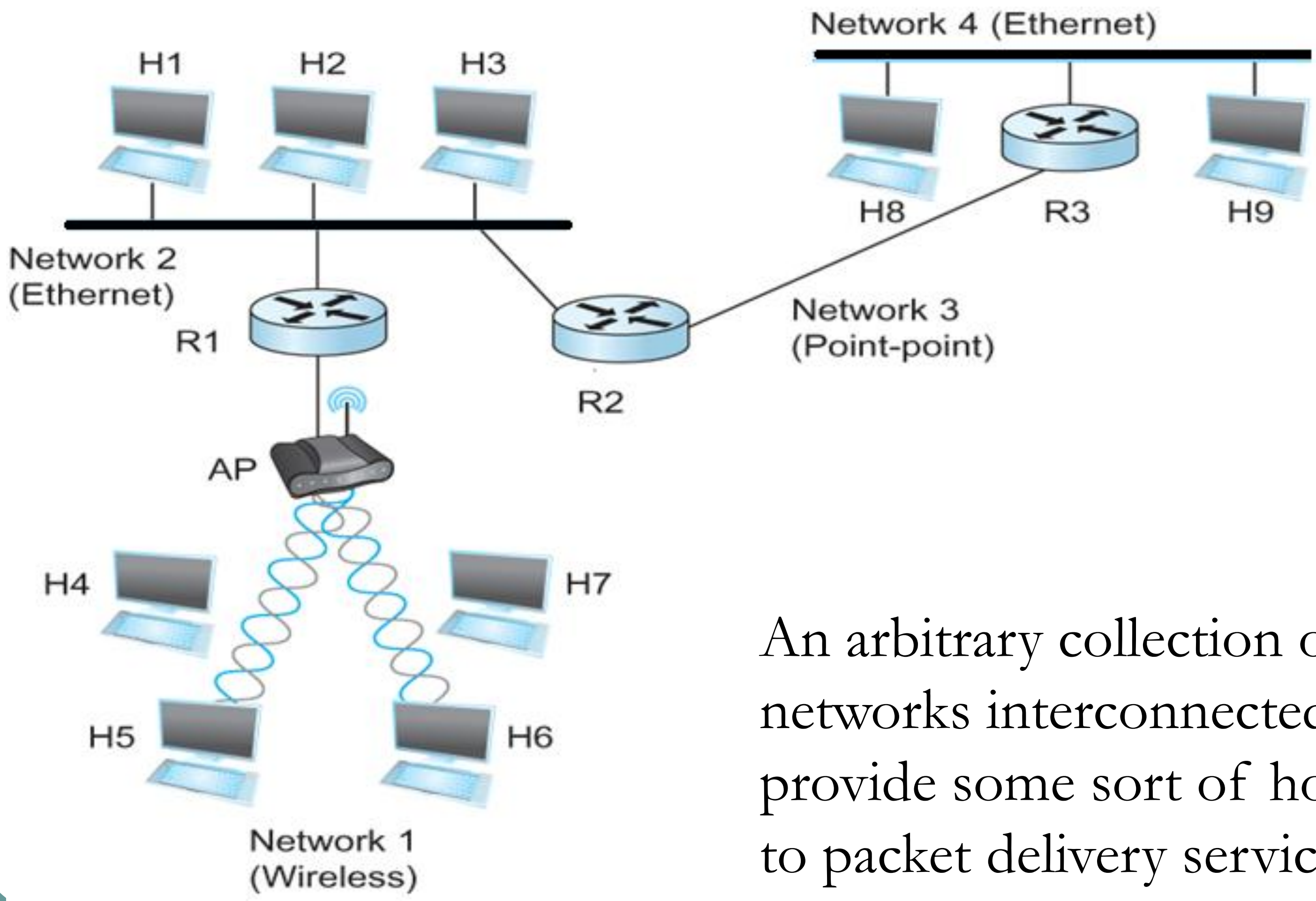
Internetwork

- Connects different types of networks with efficient routing.
- “internet” with a lowercase *i*, refers an arbitrary collection of networks interconnected to provide some sort of host to host packet delivery service
- A corporation with many sites can form a internetwork by connecting the LANs at their different sites with point-to-point links leased from the phone company

Internetwork

- "Internet" with a capital *I* refers to global internetwork to which a large percentage of networks are connected.

Internetwork

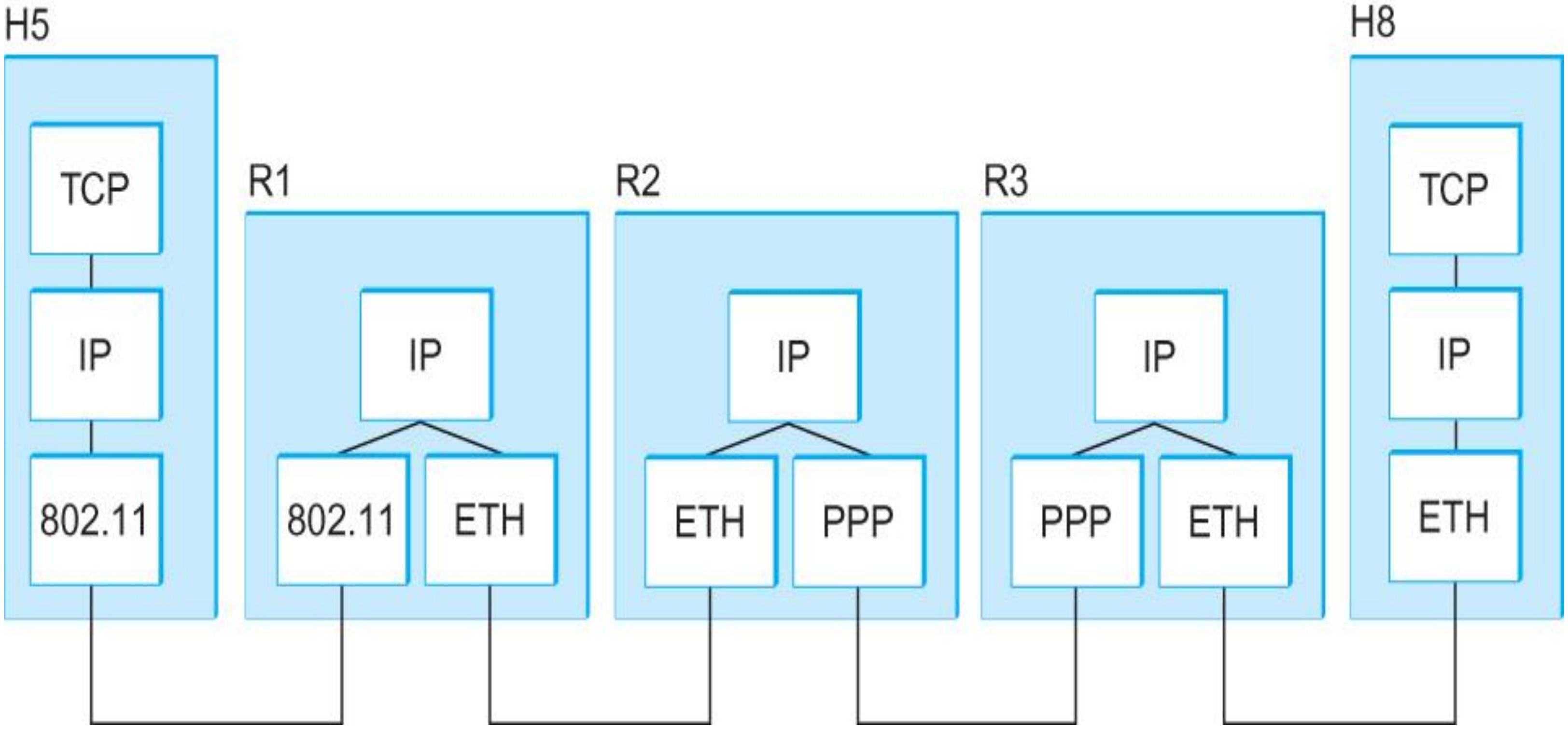


An arbitrary collection of networks interconnected to provide some sort of host-host to packet delivery service

Issues in Internetworking

- Different packet sizes
- Different protocols
- Different packet formats

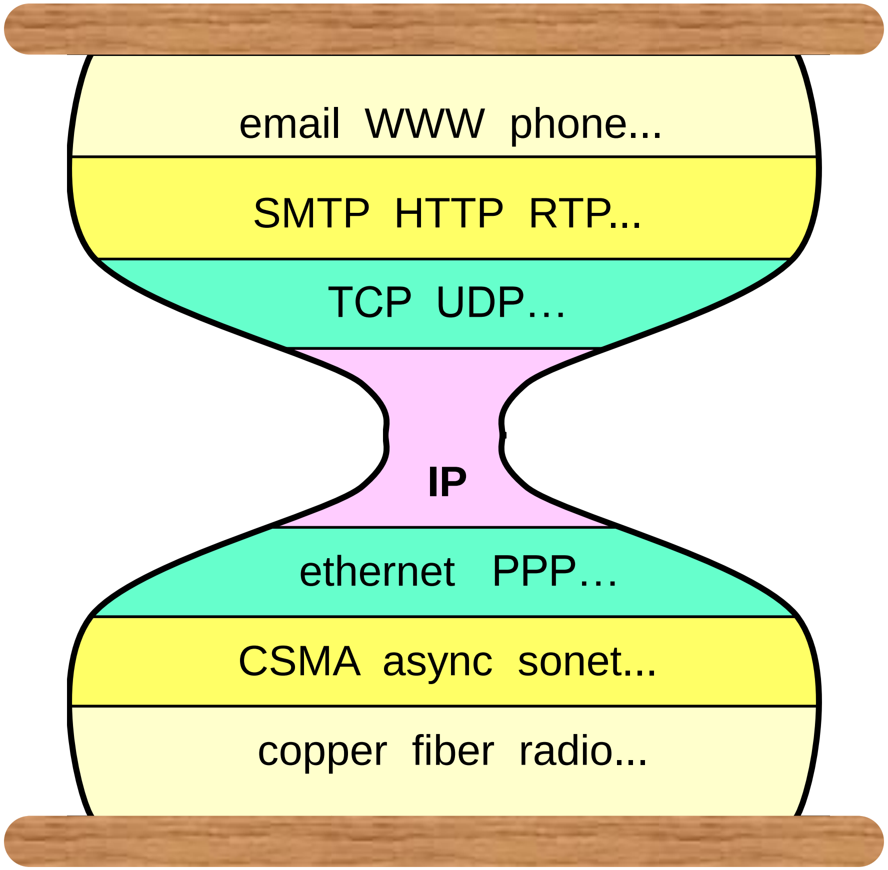
Example Internetwork



Internet Protocol

- The Internet Protocol (IP) is the key tool used today to build scalable, heterogeneous internetworks
- IP runs on all the nodes (both hosts and routers) in a collection of networks
- Functions as a single logical internetwork

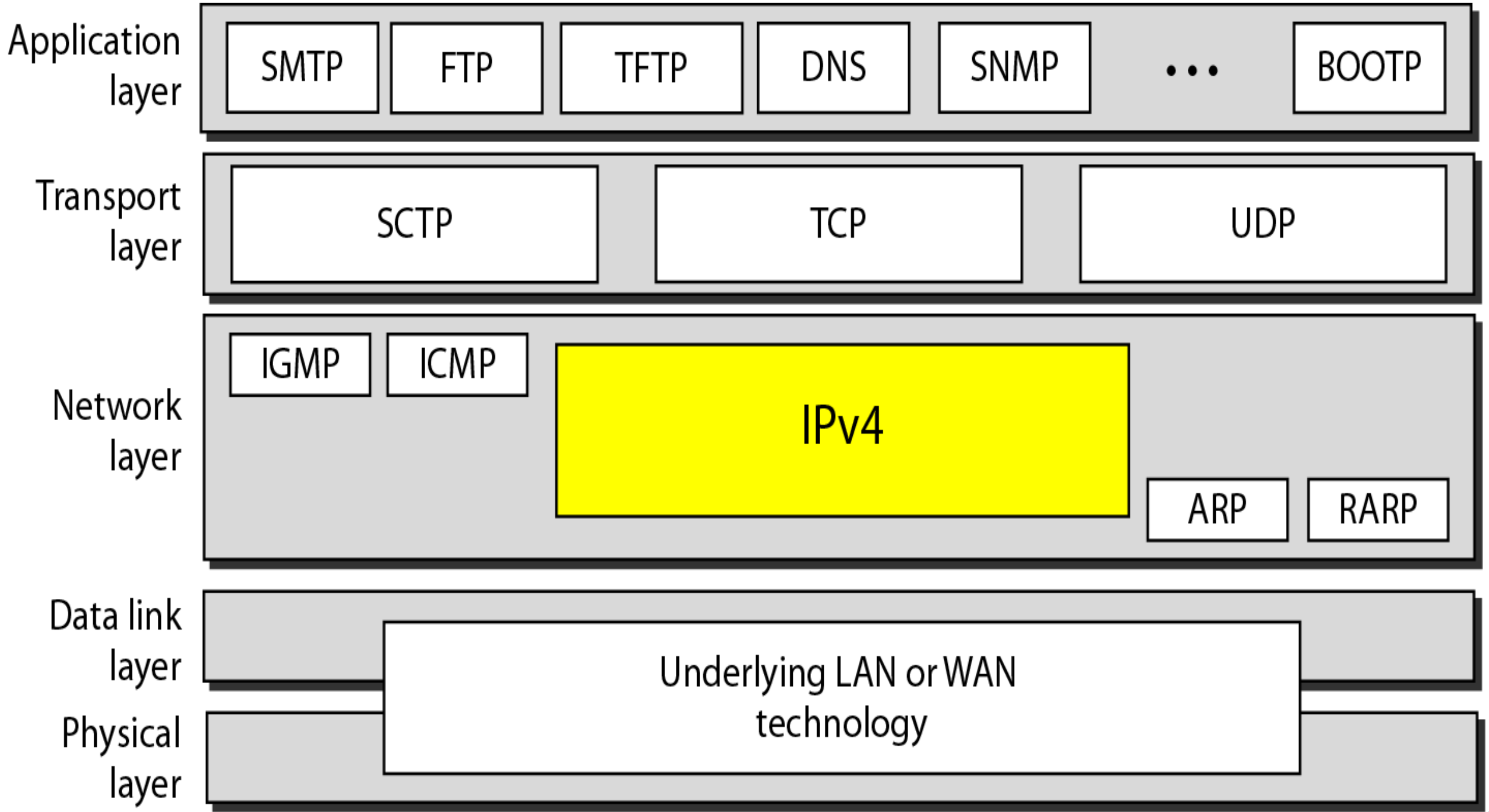
Hour Glass Model



IP Service Model

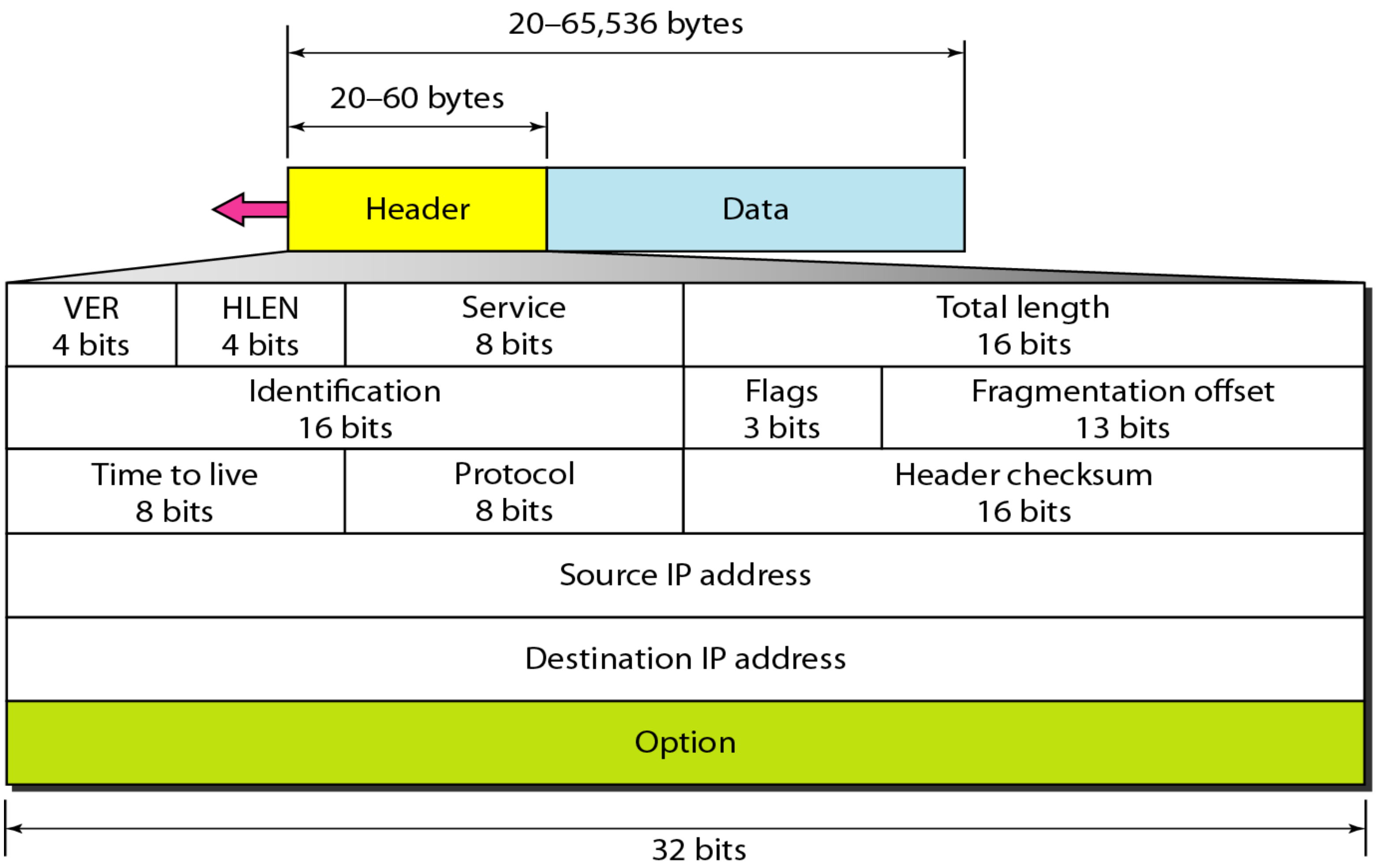
- Data Delivery Model
  - Connectionless model for data delivery
    - IP is connectionless
    - Datagram can travel from a sender to a receiver without the receiver having to send an ack.
    - Connection-oriented protocols exist at other, higher layers of that model.
  - Best-effort delivery (unreliable service)
    - Packets are lost
    - Packets are delivered out of order
    - Duplicate copies of a packet are delivered
    - Packets can be delayed for a long time
- Global Addressing Scheme
  - Provides a way to identify all hosts in the network

Protocols of Network Layer



IP Packet Format

Packet Format



Packet Format

- Version
  - Current Version 4
  - Version 6
- Header Length
  - Length → 4 bits
  - No option→ header size 20bytes [5(rows)x4(bytes)]
  - Option(Max option size) → 60 (15x4)

Packet Format

- TOS
  - Types of Service
- Total Length
  - Header + data
  - Therefore length of data =total length – header length
- Identification
  - Used in fragmentation

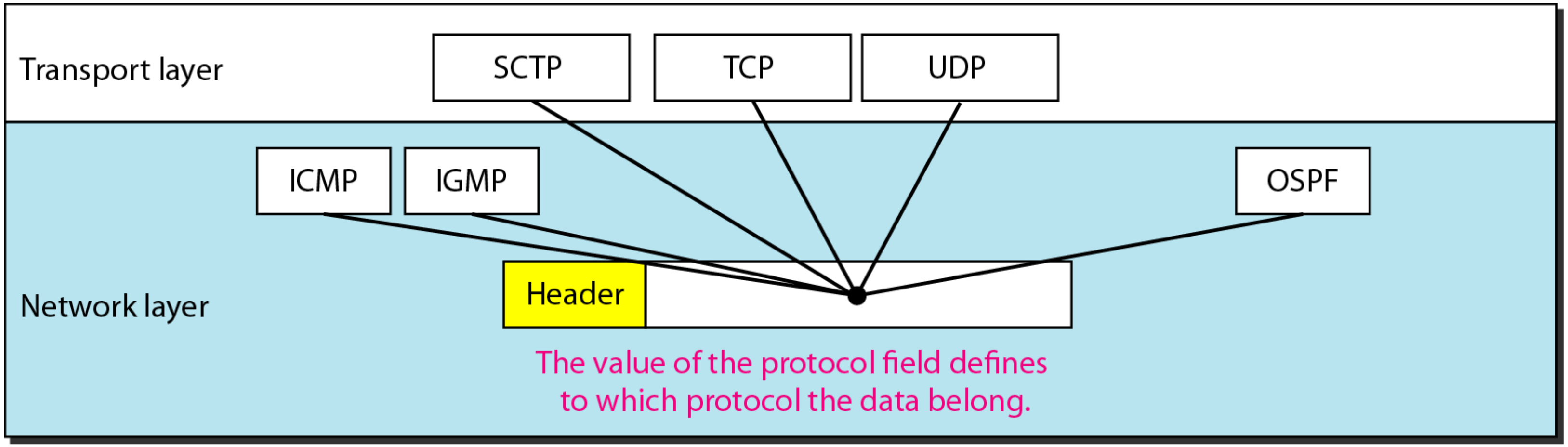
TOS Bits	*Description*
0000	Normal (default)
0001	Minimize cost
0010	Maximize reliability
0100	Maximize throughput
1000	Minimize delay

Packet Format

- Flags
  - Used in fragmentation
- Fragmentation Offset
  - Used in fragmentation
- Time to live
  - Limit packet life time
  - Support to count time in seconds
  - Maximum life time → number of hop count

Packet Format

- Protocol
  - Higher level protocol that uses the services of IPv4



Packet Format

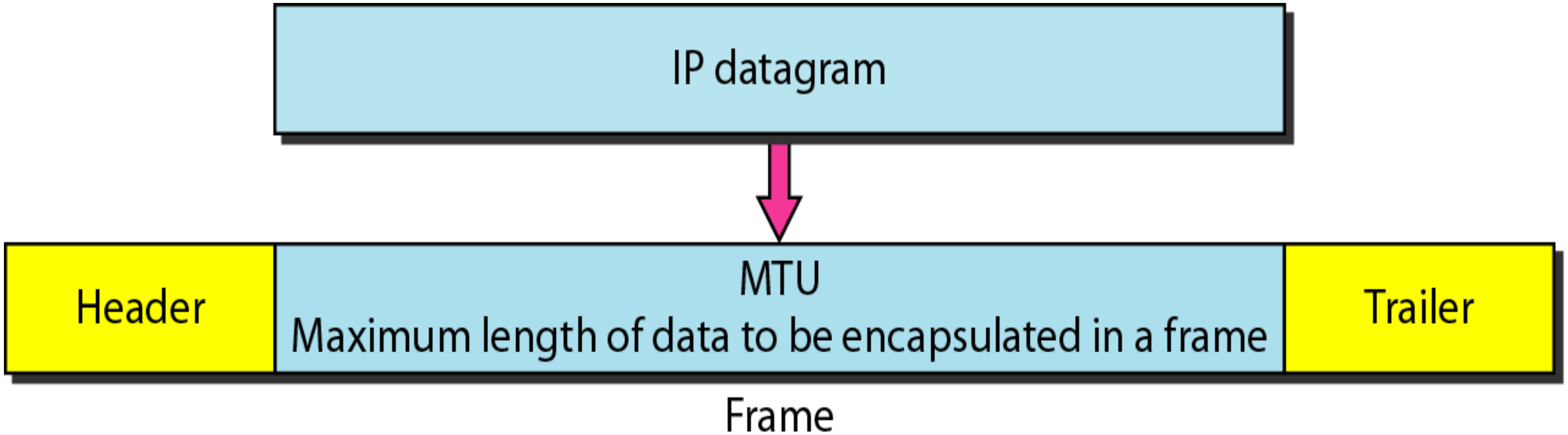
- Source Address
- Destination Address

Fragmentation

- Each router decapsulates (fragments) the IPv4 datagram from the frame it receives, process it and then encapsulates (reassembly) it in another frame.
- Received frame → Frame format and size depends on the protocol used by the physical network through which the frame has just traveled.
- Sent Frame → Frame format and size depends on the protocol used by the physical network through which the frame is going to travel.

Maximum Transfer Unit (MTU)

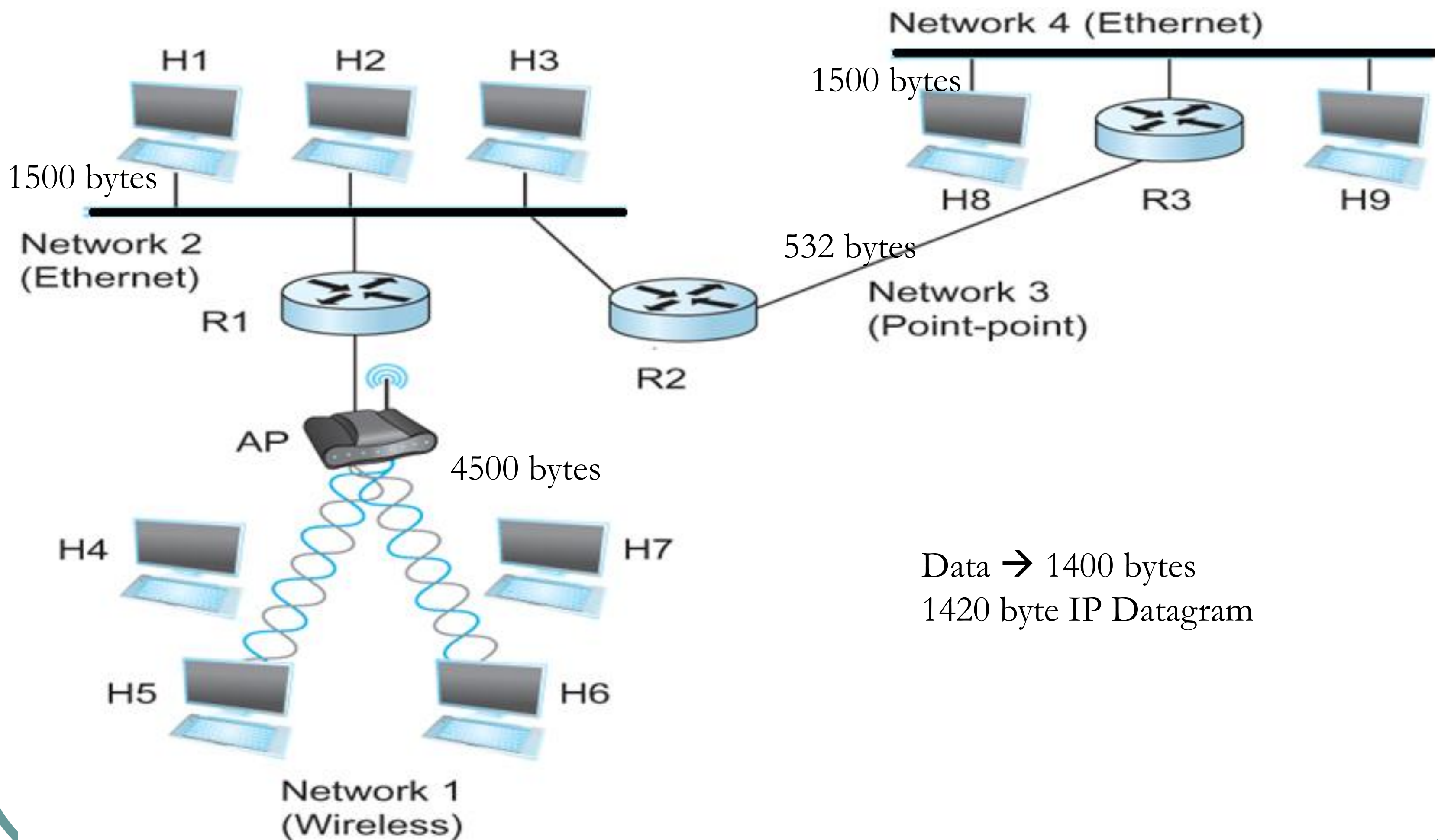
- Largest IP datagram that can carry in a frame.
- Each DLL protocol has its own frame format.
- Therefore MTU is smaller than the largest packet size on that network because the IP datagram needs to fit in the payload of the DLL frame



Maximum Transfer Unit (MTU)

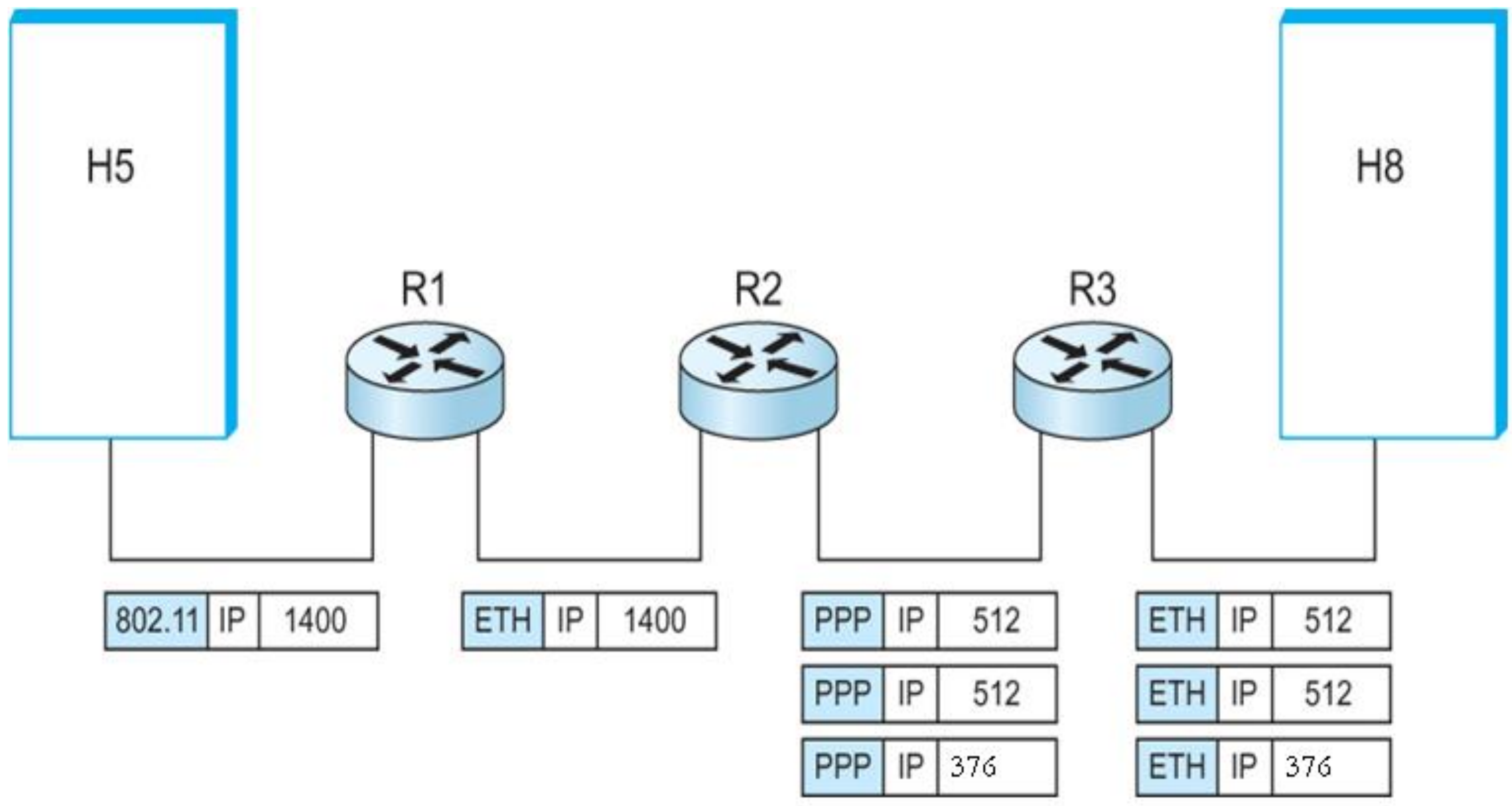
- MTU is 1500 bytes for the two Ethernets
- 4500 bytes for the Wireless Network
- 532 bytes for the point-to-point network
- Data 1400 bytes.
- A 1420-byte datagram (20-byte IP header plus 1400 bytes of data) sent from H1 makes it across the first Ethernet

Example



Data → 1400 bytes
1420 byte IP Datagram

Maximum Transfer Unit (MTU)

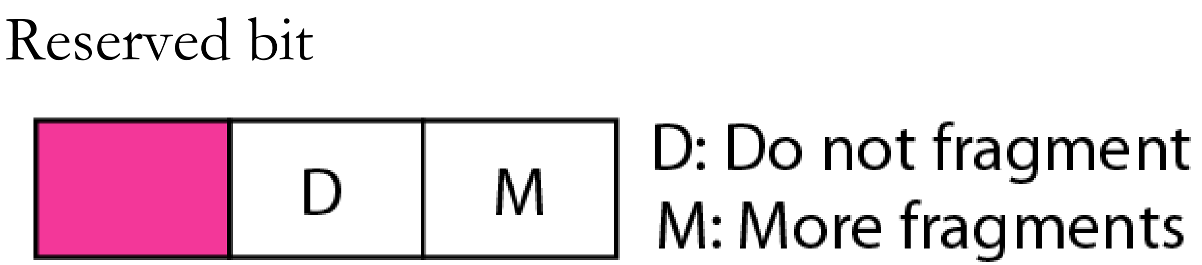


Fields Related to fragmentation

- Identification
  - 16 bit field.
  - Identifies a datagram originating from the source host.
  - Combination of Identification number and the IP address uniquely identifies a datagram
  - Counter value initially a positive number
  - Copies the counter value to the identification field of the datagram.
  - Counter is incremented to 1.
  - Fragmentation happens copies the identification value to all the fragments

Fields Related to fragmentation

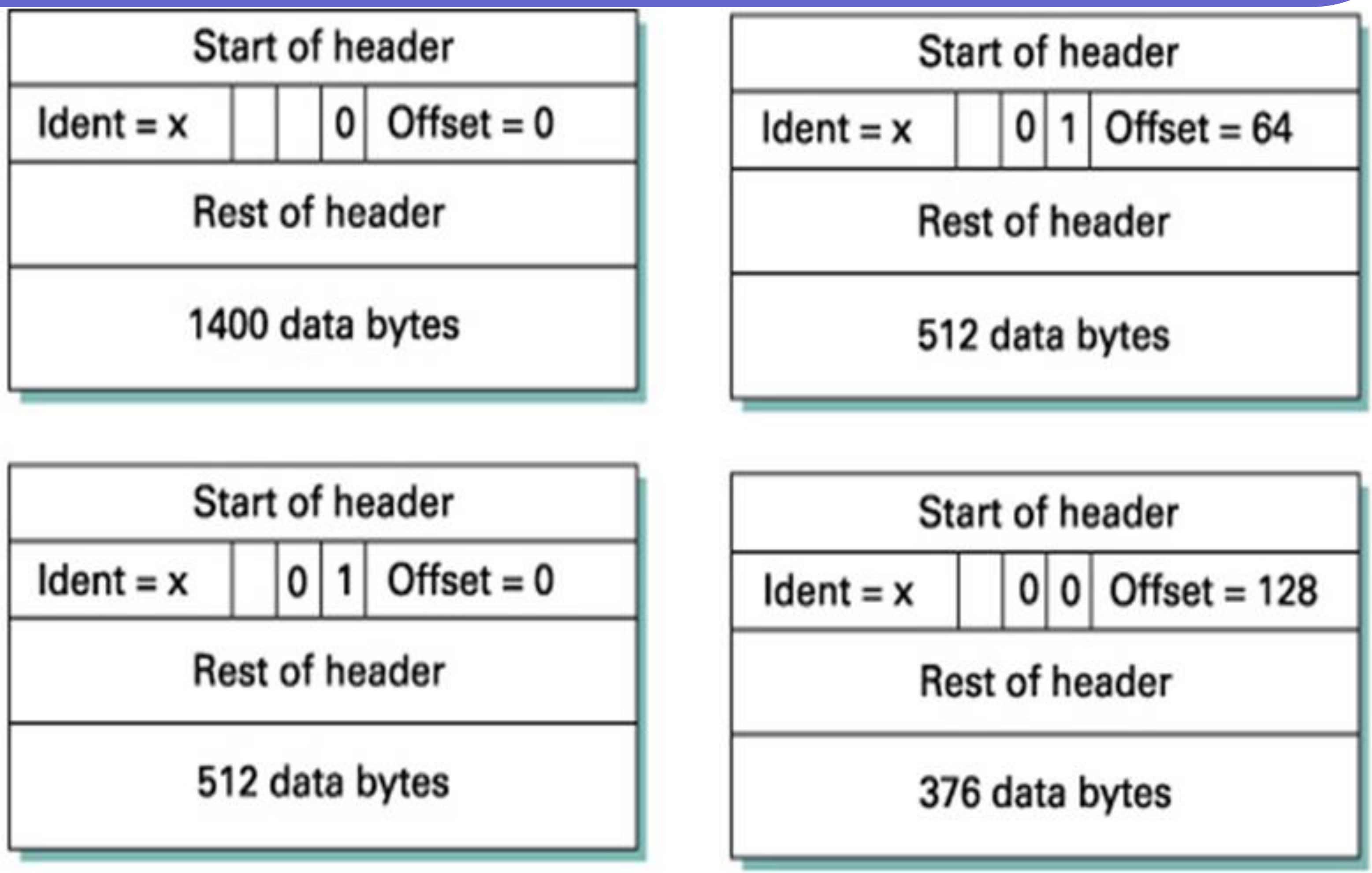
- Flags
  - D → set to 1 do not fragment
  - M → set to 1 more fragments following
    - 0 no more fragments



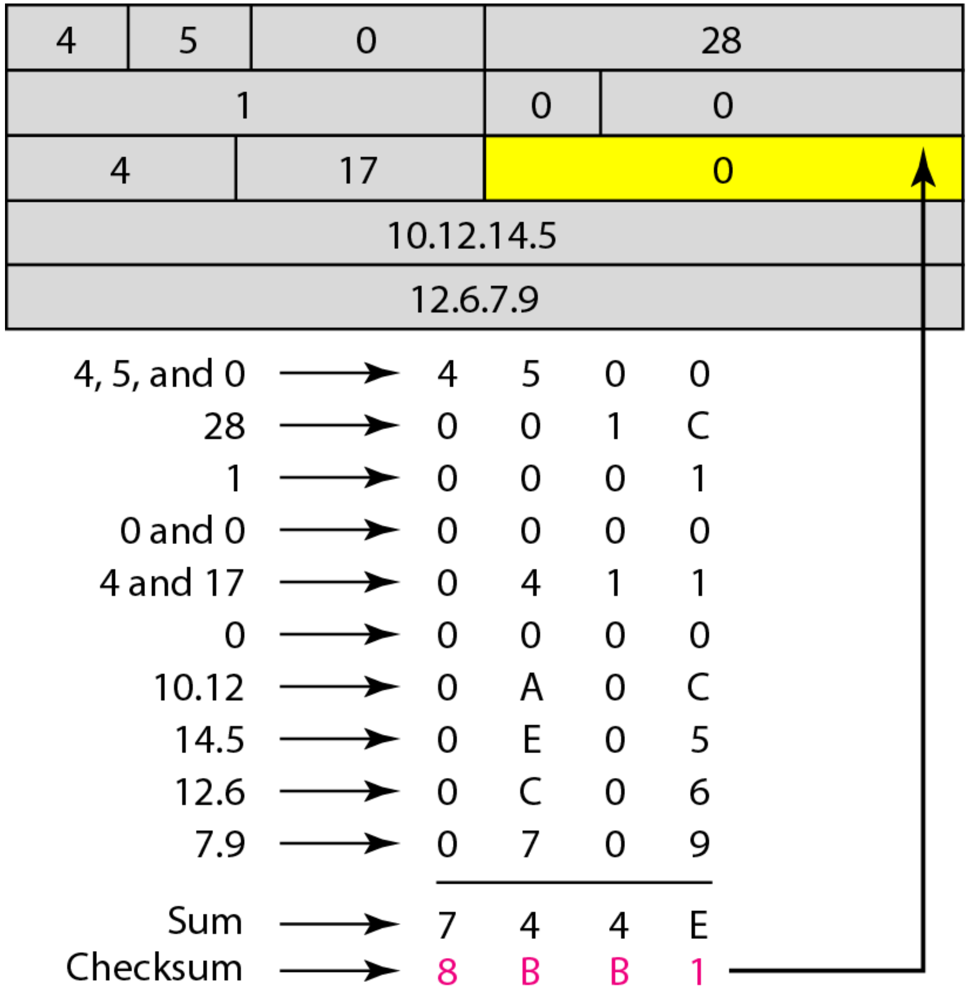
Fields Related to fragmentation

- Offset
  - 13 bit field
  - Shows the relative position of the fragmentation
  - Offset of data in original datagram measured in units of 8 bytes boundaries or 8 byte chunks. (64 bits)
  - 13 bits - all ones - value - 8191
  - ie cannot represent sequence of bytes greater than 8191
  - 8191 * 8 is 65,528, just about the maximum size allowed for an IP datagram. (65536 → IP)

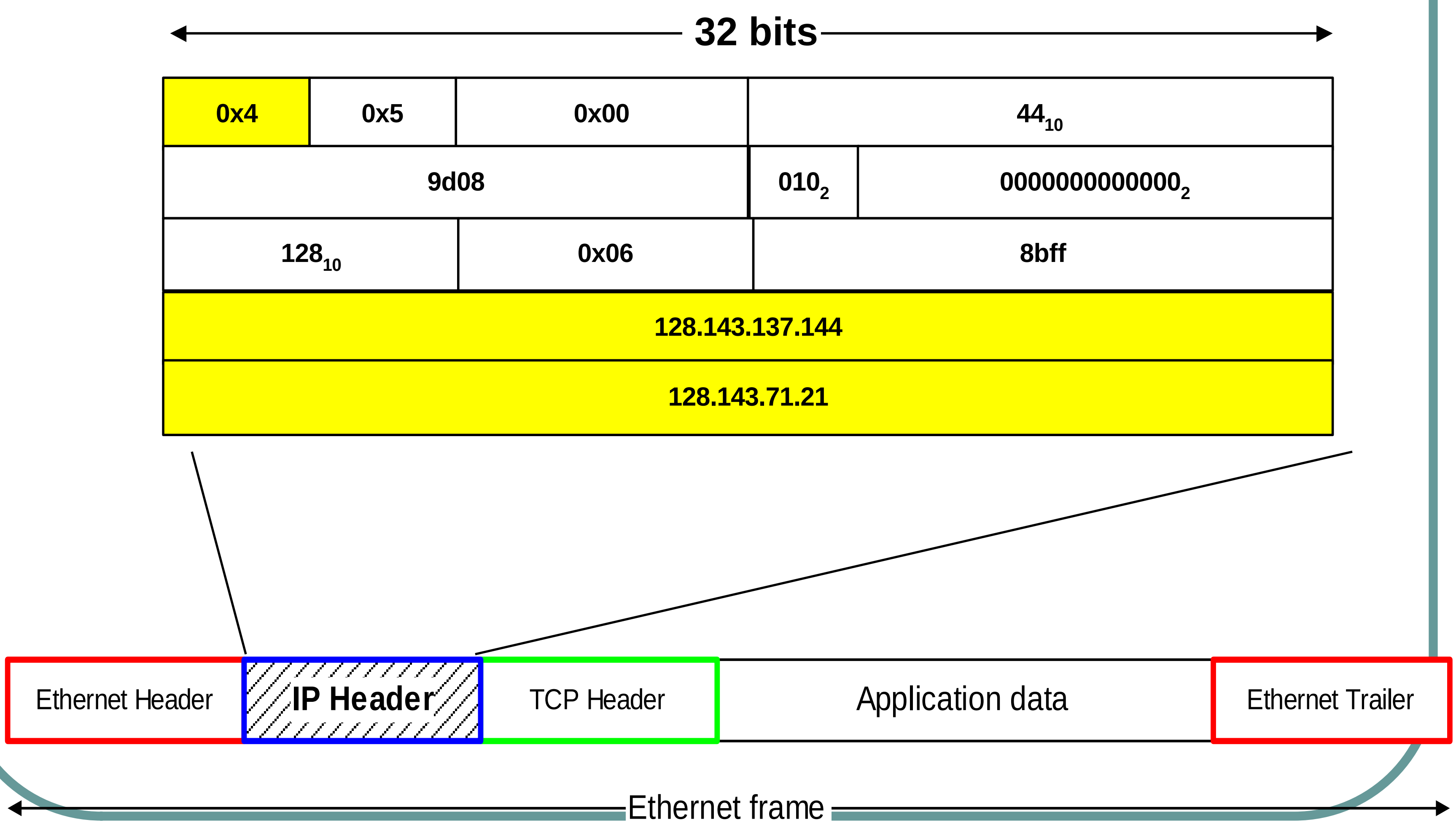
Fields Related to fragmentation



Checksum


4
5
0
28
1
0
0
4
17
0
10.12.14.5
12.6.7.9
4, 5, and 0 → 4 5 0 0
28 → 0 0 1 C
1 → 0 0 0 1
0 and 0 → 0 0 0 0
4 and 17 → 0 4 1 1
0 → 0 0 0 0
10.12 → 0 A 0 C
14.5 → 0 E 0 5
12.6 → 0 C 0 6
7.9 → 0 7 0 9
Sum → 7 4 4 E
Checksum → 8 B B 1

IP Addresses

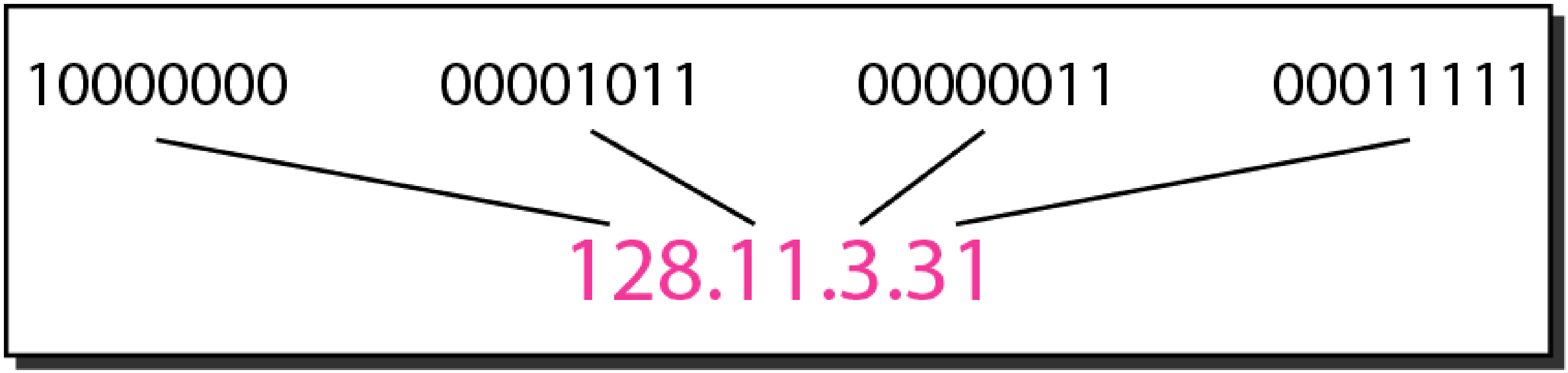


IP Addressing

IPv4 Addresses

- An IPv4 address is a 32-bit address that uniquely and universally defines the connection of a device (for example, a computer or a router) to the Internet.
- The address space of IPv4 is 2^{32} or 4,294,967,296.

Dotted-decimal notation



Example

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

a. 111.56.45.78

b. 221.34.7.82

Solution

Replace each decimal number with its binary equivalent

a. 01101111 00111000 00101101 01001110

b. 11011101 00100010 00000111 01010010

Example

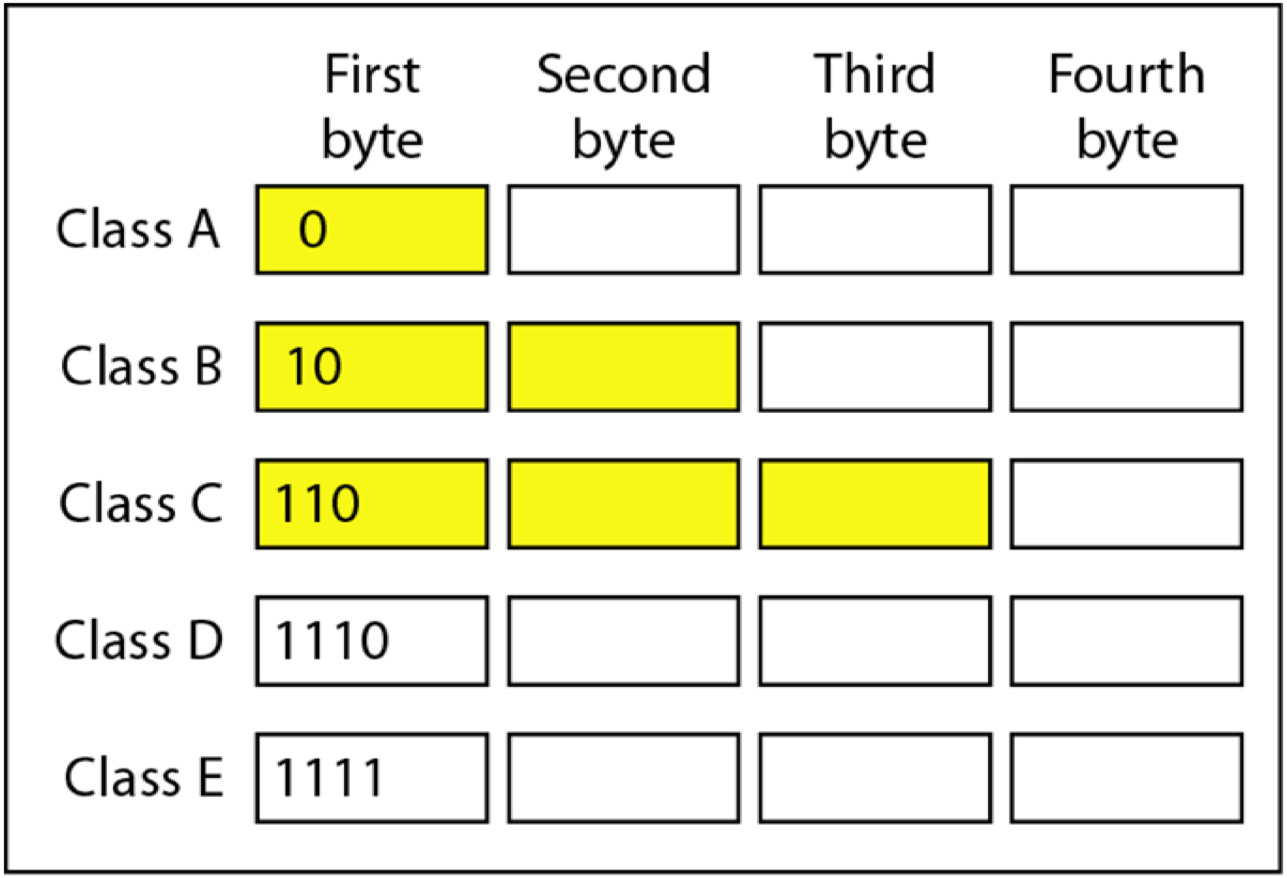
Find the error, if any, in the following IPv4 addresses.

a. 111.56.045.78
b. 221.34.7.8.20
c. 75.45.301.14
d. 11100010.23.14.67

Classful Addressing

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

Classful Addressing



a. Binary notation

Classful Addressing

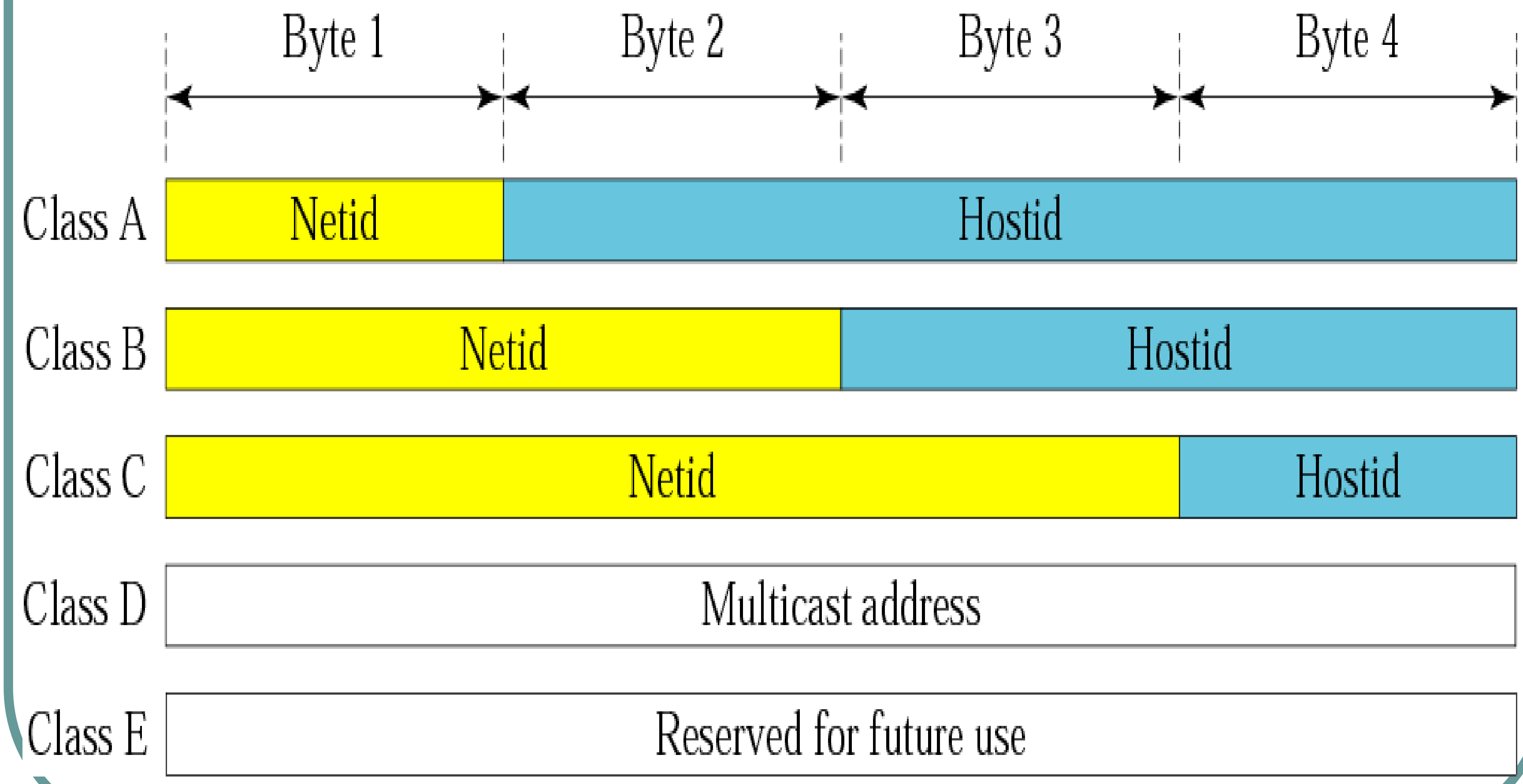
	First byte	Second byte	Third byte	Fourth byte
Class A	0–127			
Class B	128–191			
Class C	192–223			
Class D	224–239			
Class E	240–255			

b. Dotted-decimal notation

Network vs. Host

- Every IP address has 2 parts:
  - Identifying the network it resides on
  - Identifying the host address on the network
- The class of the address and the subnet mask determine which part belongs to the network address and which part belongs to the host address

Network vs. Host

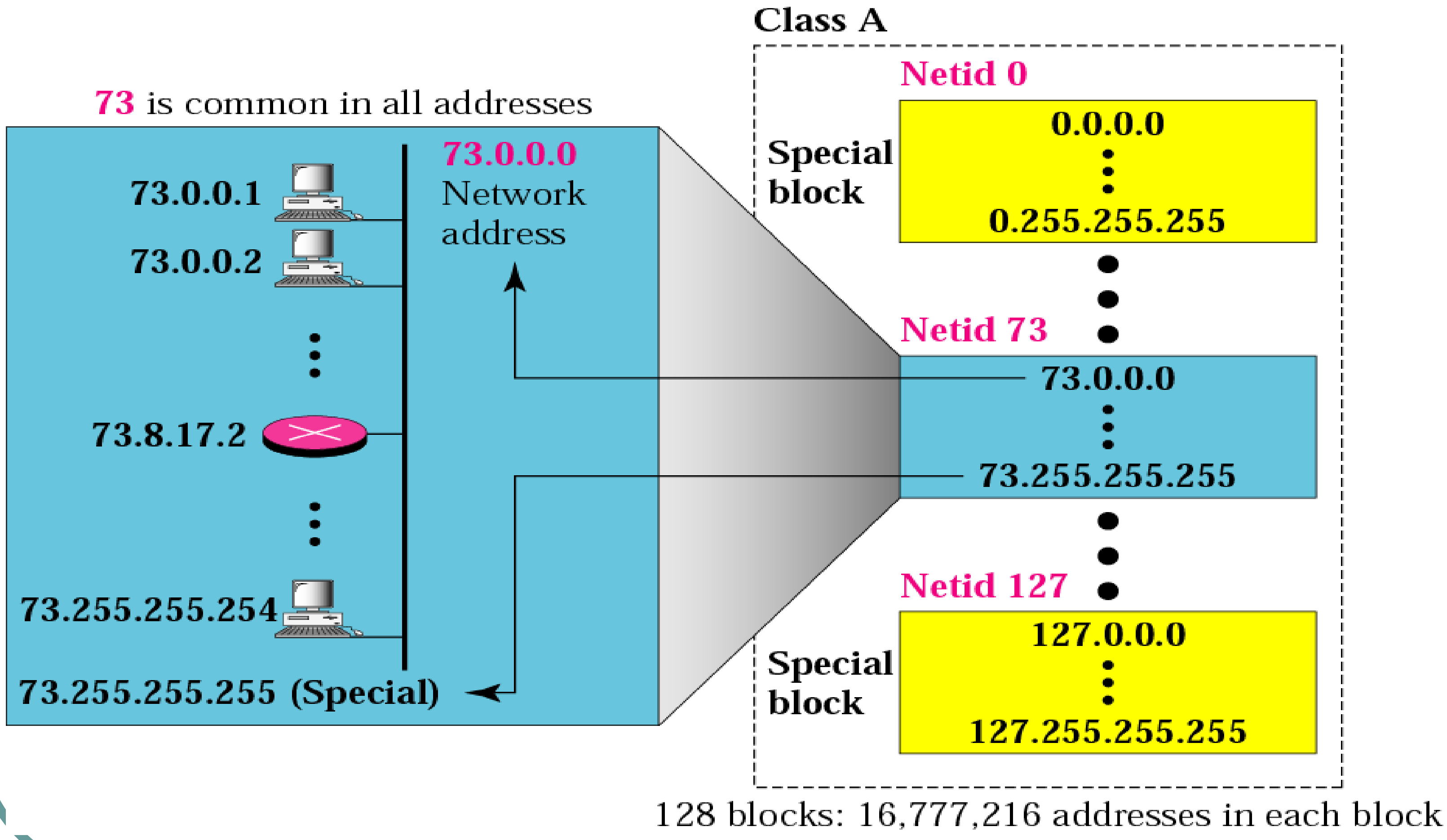


Example

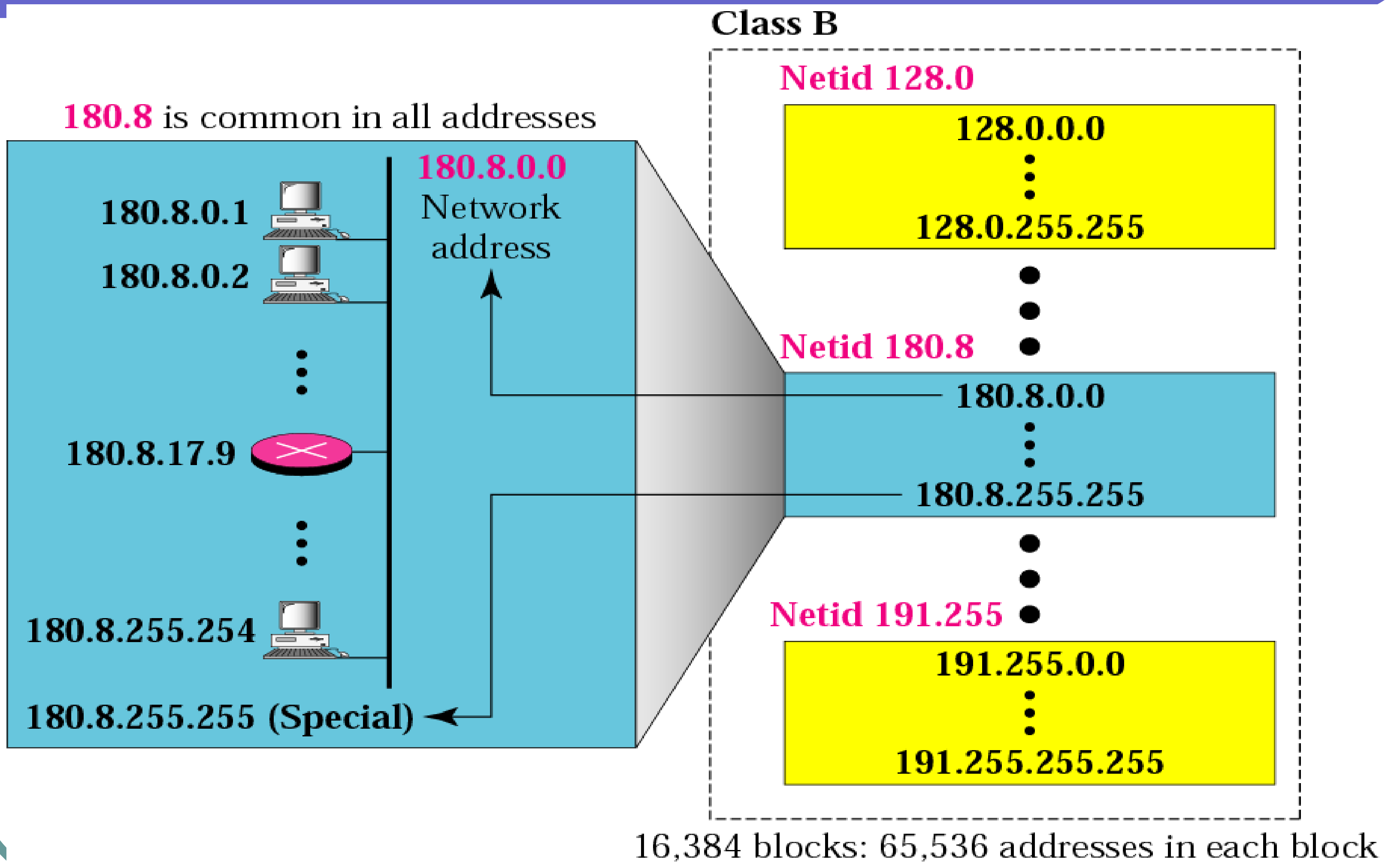
Find the class of each address.

a. 00000001 00001011 00001011 11101111
b. 11000001 10000011 00011011 11111111
c. 14.23.120.8
d. 252.5.15.111

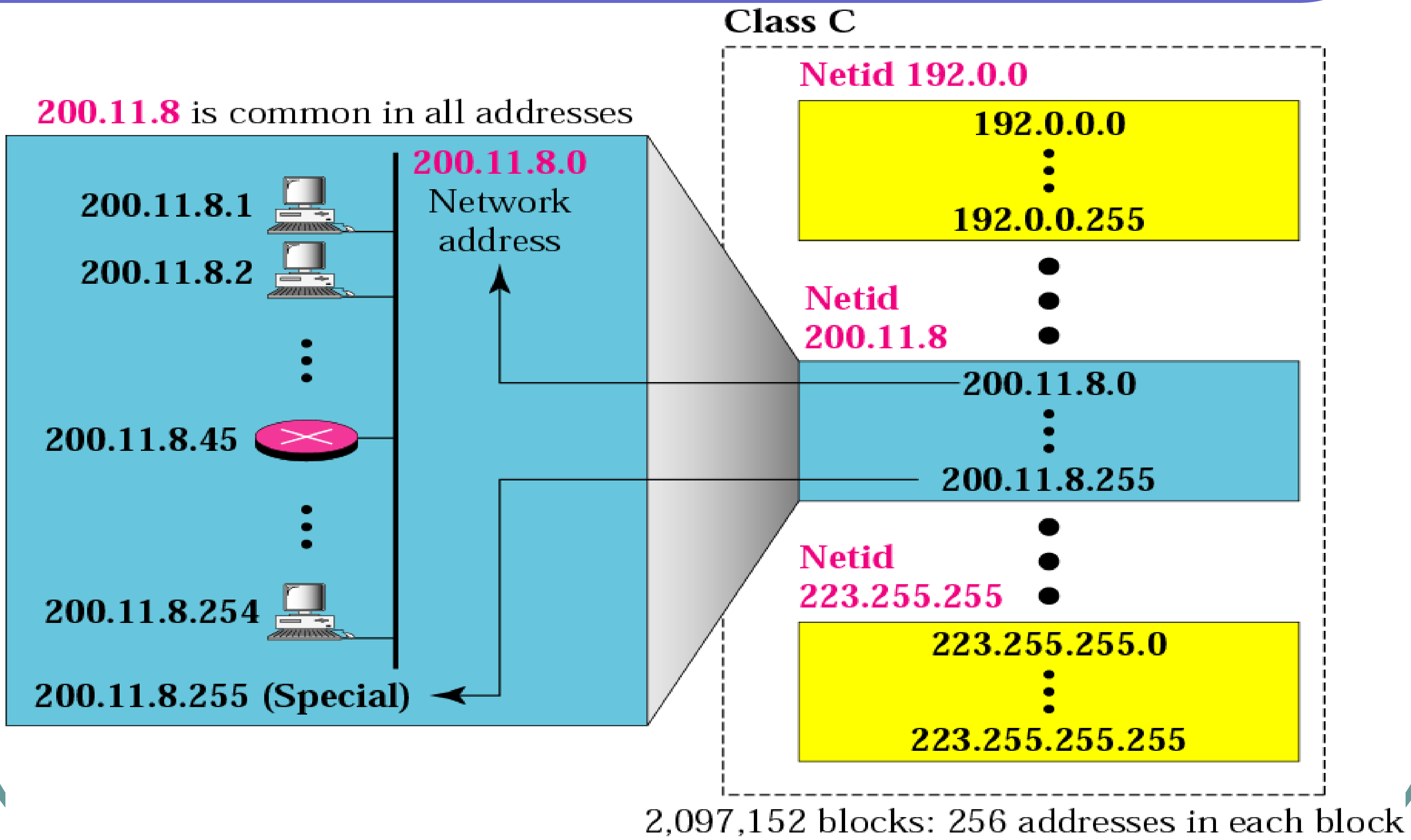
Blocks in class A



Blocks in class B



Blocks in class C



Classes and Blocks

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

Automatically assigned addresses

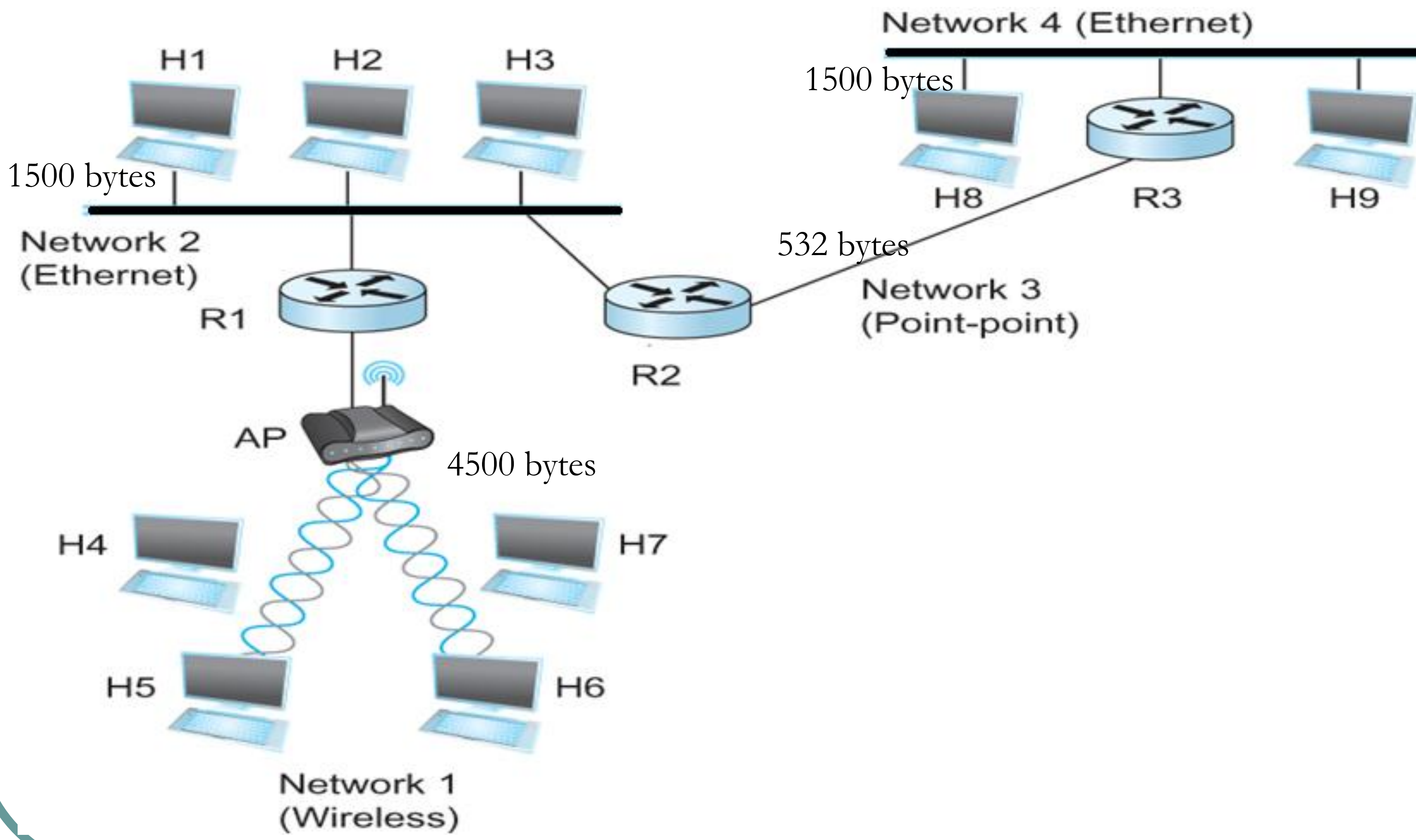
- 192.168.1.0 → 0 is the automatically assigned network address.
- 192.168.1.1 – 254 → Addresses beyond 1 are assigned to computers and devices on the network.
- 192.168.1.255 → 255 is automatically assigned on most networks as the broadcast address.

IP Datagram Forwarding

- Strategy
  - Every datagram contains destination's address
  - If directly connected to destination network, then forward to host
  - If not directly connected to destination network, then forward to some router
  - Forwarding table maps network number into next hop
  - Each host has a default router
  - Each router maintains a forwarding table
- Example (router R2)

NetworkNum	NextHop
1	R1
2	Interface 1
3	Interface 0
4	R3

Example


Network 4 (Ethernet)
H1
H2
H3
1500 bytes
1500 bytes
H8
R3
H9
Network 2
(Ethernet)
532 bytes
R1
Network 3
(Point-point)
R2
AP
4500 bytes
H4
H7
H5
H6
Network 1
(Wireless)

IP Datagram Forwarding

if (NetworkNum of destination = NetworkNum of one of my interfaces) **then**
    deliver packet to destination over that interface
else
    if (NetworkNum of destination is in my forwarding table) **then**
        deliver packet to NextHop router
    else
        deliver packet to default router

For a host with only one interface and only a default router in its forwarding table, this simplifies to

if (NetworkNum of destination = my NetworkNum)**then**
    deliver packet to destination directly
else
    deliver packet to default router