COMPUTER NETWORKS

TEAM NETWORKS

Department of Computer Science and Engineering

COMPUTER NETWORKS

Link Layer and LAN

TEAM NETWORKS

Department of Computer Science and Engineering

Unit – 4 Link Layer and LAN Roadmap

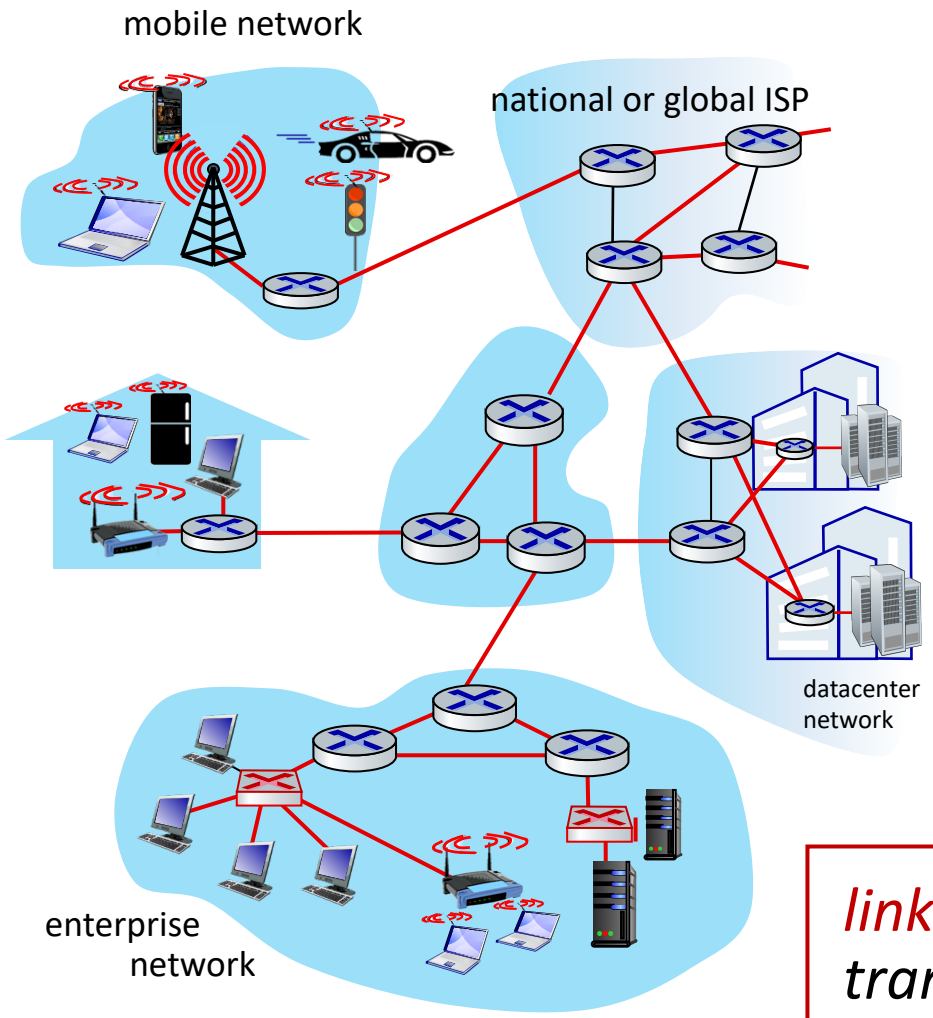
- Introduction
- Error detection, correction
- Multiple access protocols
- LANs
  - addressing, ARP
  - Ethernet
  - switches
- A day in the life of a web request
- Physical layer
- Wireless LANs: IEEE 802.11

COMPUTER NETWORKS

Unit – 4 Link Layer and LAN Roadmap

- Introduction
- Error detection, correction
- Multiple access protocols
- LANs
  - addressing, ARP
  - Ethernet
  - switches
- A day in the life of a web request
- Physical layer
- Wireless LANs: IEEE 802.11

Introduction to Link layer



Terminology:

- Hosts, routers, switches, WiFi access points: Nodes
- communication channels that connect adjacent nodes along communication path: Links
  - wired
  - wireless
- layer-2 packet: *frame*, encapsulates datagram

link layer has responsibility of transferring datagram from one node to physically adjacent node over a link

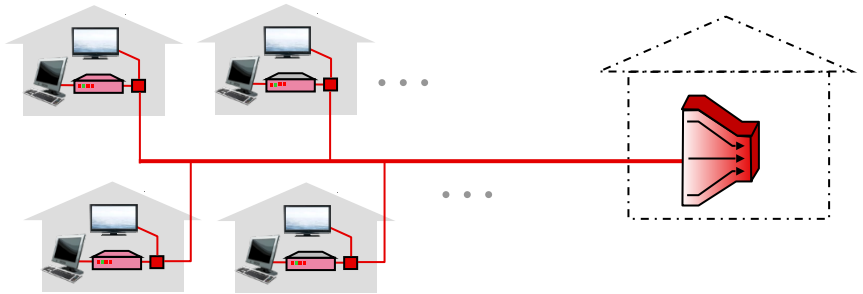
Link Layer: Context

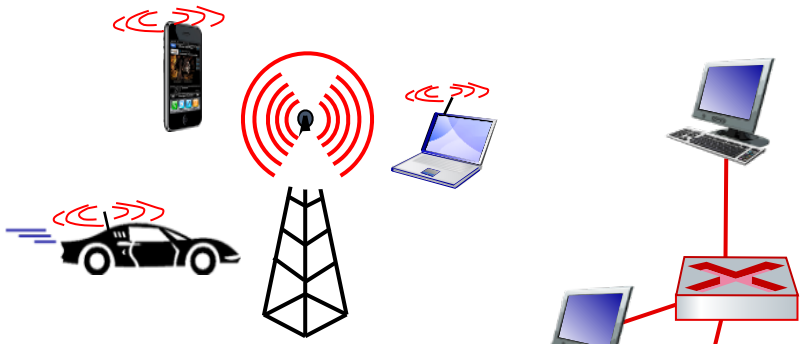
- Datagram transferred by different link protocols over different links:
  - e.g., WiFi on first link, Ethernet on next link
- Each link protocol provides different services
  - e.g., may or may not provide reliable data transfer over link

transportation analogy:

- trip from Princeton to Lausanne
  - limousine: Princeton to JFK airport
  - plane: JFK to Geneva
  - train: Geneva to Lausanne
- tourist = datagram
- transport segment = communication link
- transportation mode = link-layer protocol
- travel agent = routing algorithm

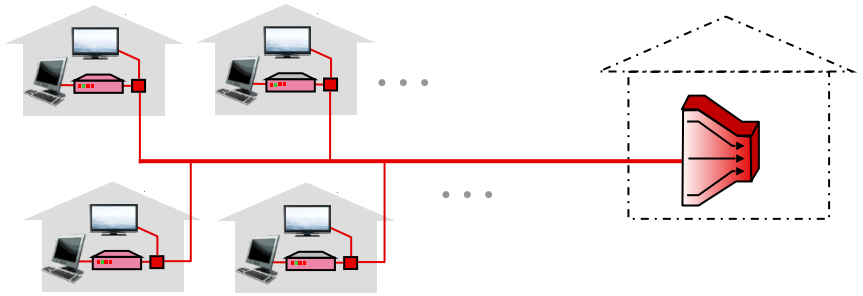
Link Layer: Services

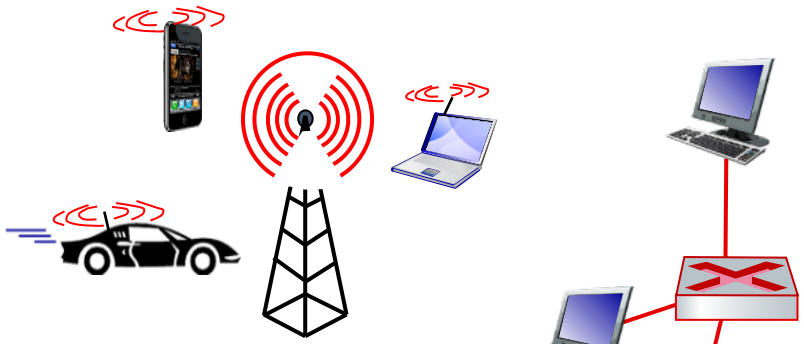




- framing, link access:
  - encapsulate datagram into frame, adding header, trailer
  - channel access if shared medium
  - "MAC" addresses in frame headers identify source, destination (different from IP address!)
- reliable delivery between adjacent nodes
  - we already know how to do this!
  - rarely used on low bit-error links (coax, fiber, etc)
  - wireless links: high error rates

Link Layer: Services



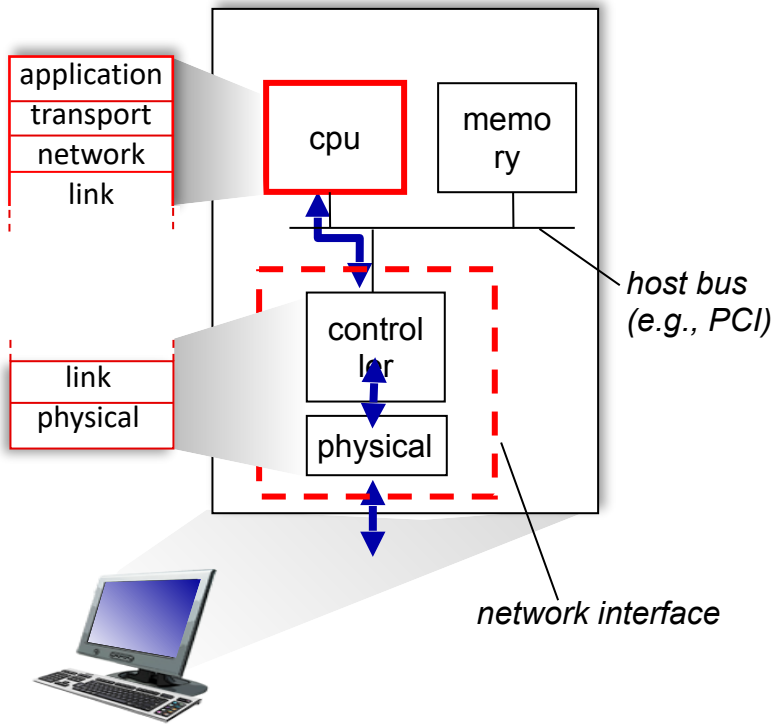


- flow control:
  - pacing between adjacent sending and receiving nodes
- error detection:
  - errors caused by signal attenuation, noise.
  - receiver detects errors, signals retransmission, or drops frame
- error correction:
  - receiver identifies *and corrects* bit error(s) without retransmission
- half-duplex and full-duplex:
  - with half duplex, nodes at both ends of link can transmit, but not at same time

COMPUTER NETWORKS

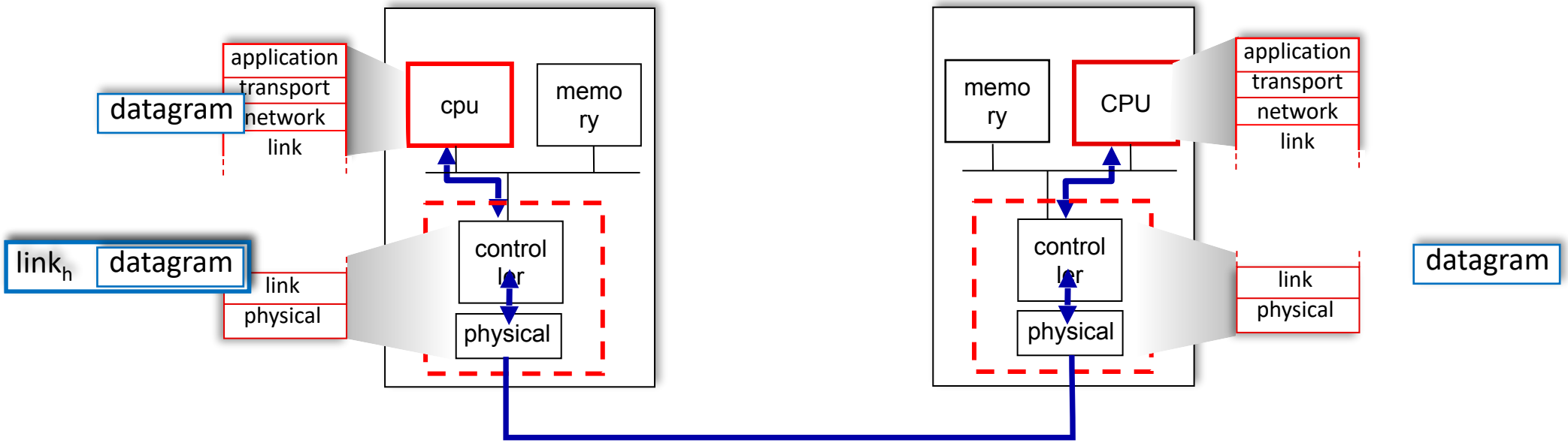
Where is the link layer implemented?

- In each-and-every host
- Link layer implemented in *network interface card* (NIC) or on a chip
  - Ethernet, WiFi card or chip
  - implements link, physical layer
- Attaches into host's system buses
- Combination of hardware, software, firmware



COMPUTER NETWORKS

Interfaces communicating



Sending side:

- encapsulates datagram in frame
- adds error checking bits, reliable data transfer, flow control, etc.

Receiving side:

- looks for errors, reliable data transfer, flow control, etc.
- extracts datagram, passes to upper layer at receiving side

COMPUTER NETWORKS

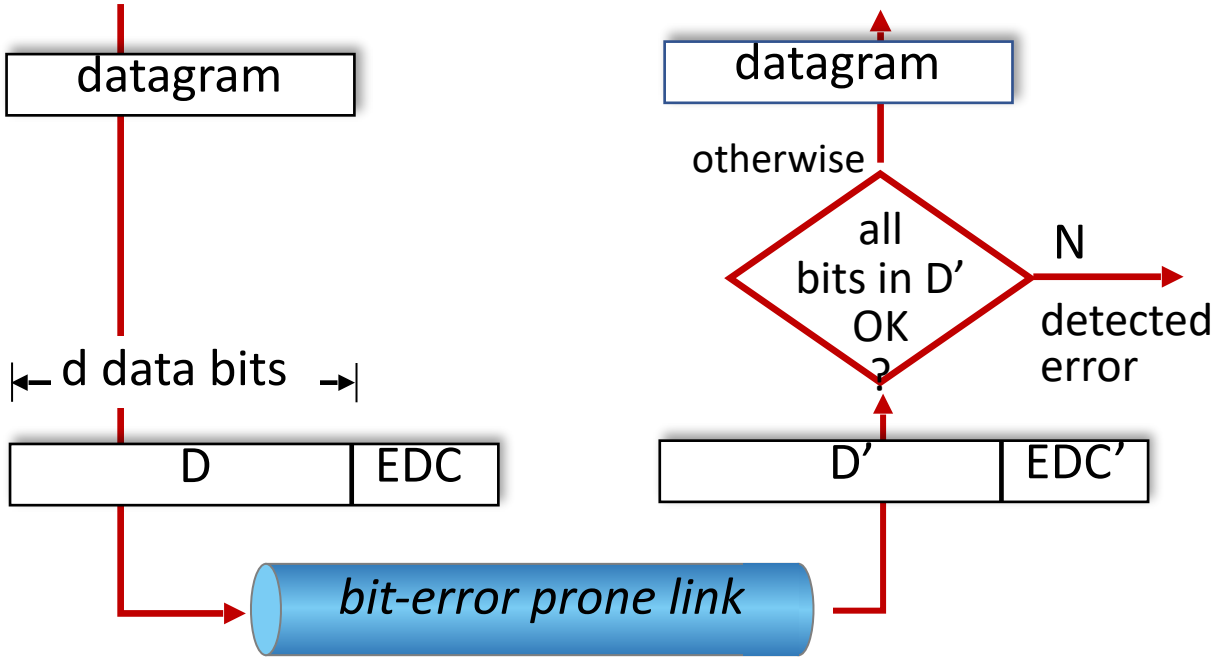
Unit – 4 Link Layer and LAN Roadmap

- Introduction
- Error detection, correction
- Multiple access protocols
- LANs
  - addressing, ARP
  - Ethernet
  - switches
- A day in the life of a web request
- Physical layer
- Wireless LANs: IEEE 802.11

Error detection

EDC: error detection and correction bits (e.g., redundancy)

D: data protected by error checking, may include header fields



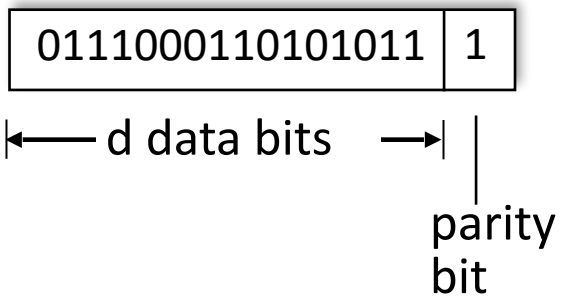
Error detection not 100% reliable!

- protocol may miss some errors, but rarely
- larger EDC field yields better detection and correction

Parity checking

single bit parity:

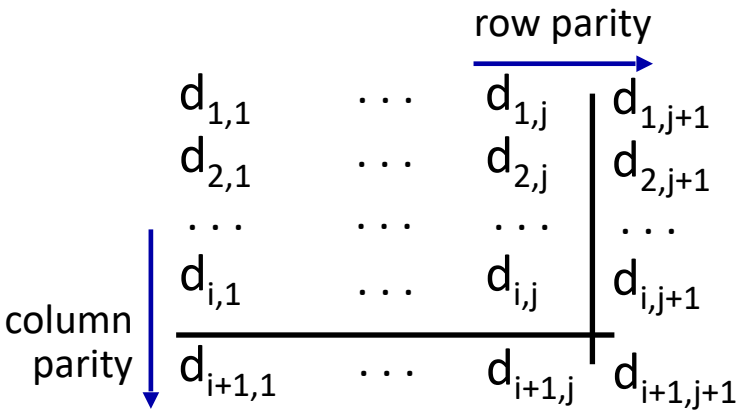
- detect single bit errors



Even parity: set parity bit so there is an even number of 1's

two-dimensional bit parity:

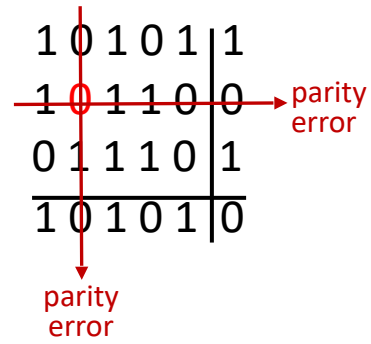
- detect *and correct* single bit errors



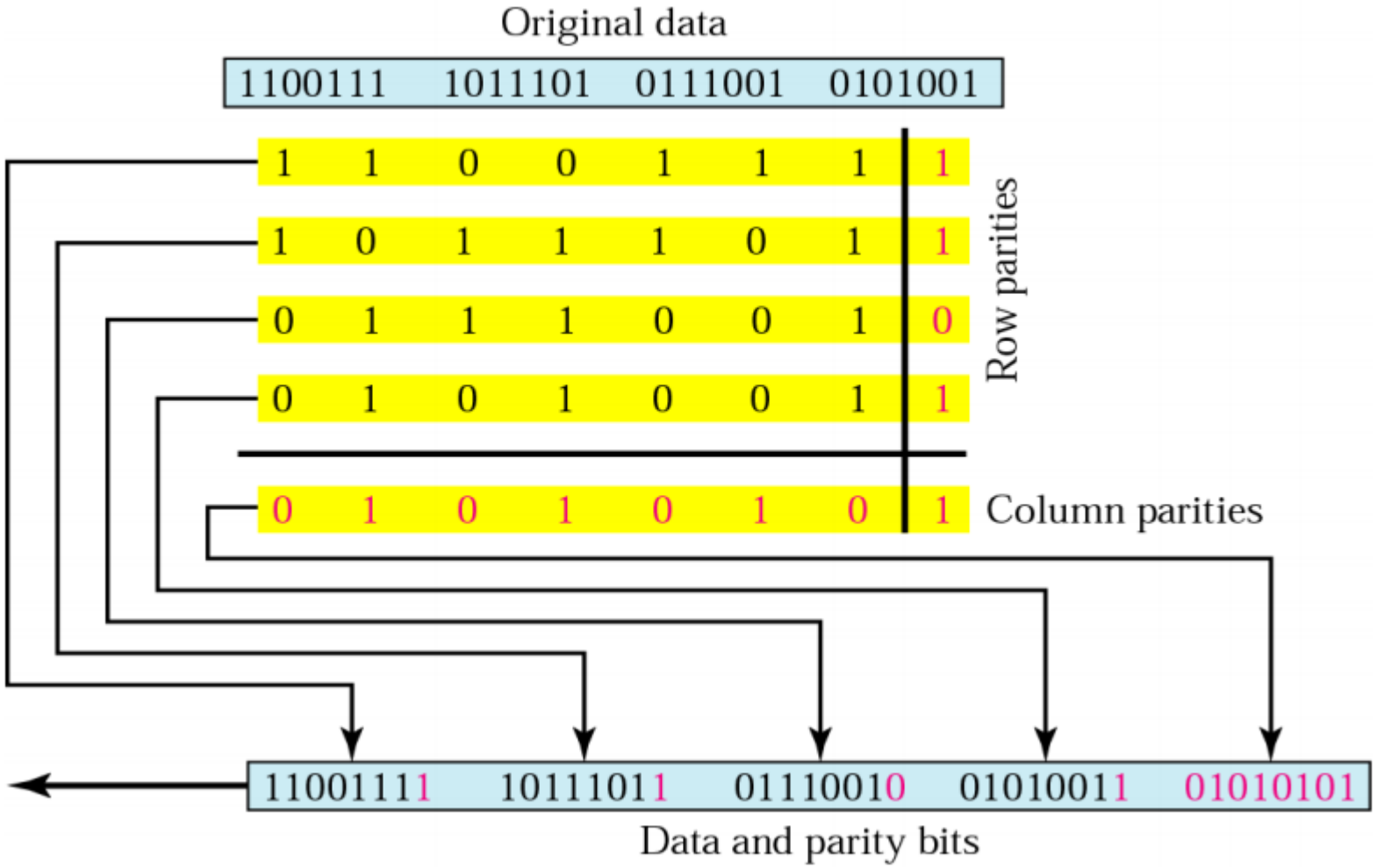
no errors:

1	0	1	0	1	1
1	1	1	1	0	0
0	1	1	1	0	1
0	0	1	0	1	0

detected and correctable single-bit error:



Two dimensional bit parity check - Example



Internet checksum (review)

Goal: detect errors (*i.e.,* flipped bits) in transmitted segment

Sender:

- treat contents of UDP segment (including UDP header fields and IP addresses) as sequence of 16-bit integers
- checksum: addition (one's complement sum) of segment content
- checksum value put into UDP checksum field

Receiver:

- compute checksum of received segment
- check if computed checksum equals checksum field value:
  - not equal - error detected
  - equal - no error detected. *But maybe errors nonetheless?* More later ….

COMPUTER NETWORKS

Two dimensional bit parity check - Example

0110011001100110
0101010101010101
0000111100001111
} Three 16 bits words

The sum of first of two 16-bit words is:

0110011001100110
0101010101010101

1011101110111011 → Sum of 1st two 16 bit words.

Adding the third word to the above sum gives:

1011101110111011 → Sum of 1st two 16 bit words.
0000111100001111 → Third 16 bits word

1100101011001010 → Sum of all three 16 bit words.

Taking 1's complement for the final sum:

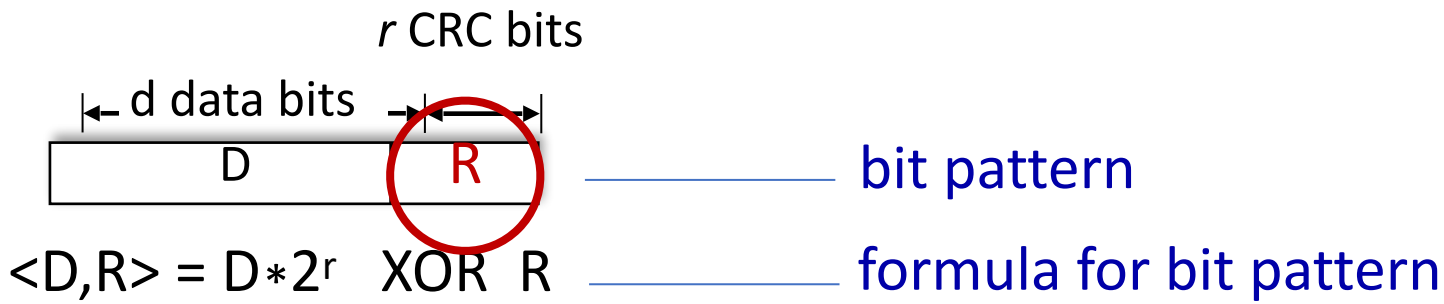
1100101011001010 → Sum of all three 16 bit words.
0011010100110101 → 1's complement for the final sum.

The 1's complement value is called as **Checksum**

COMPUTER NETWORKS

Cyclic Redundancy Check (CRC Codes)

- more powerful error-detection coding
- D: data bits (given, think of these as a binary number)
- G: bit pattern (generator), of *r+1* bits (given)



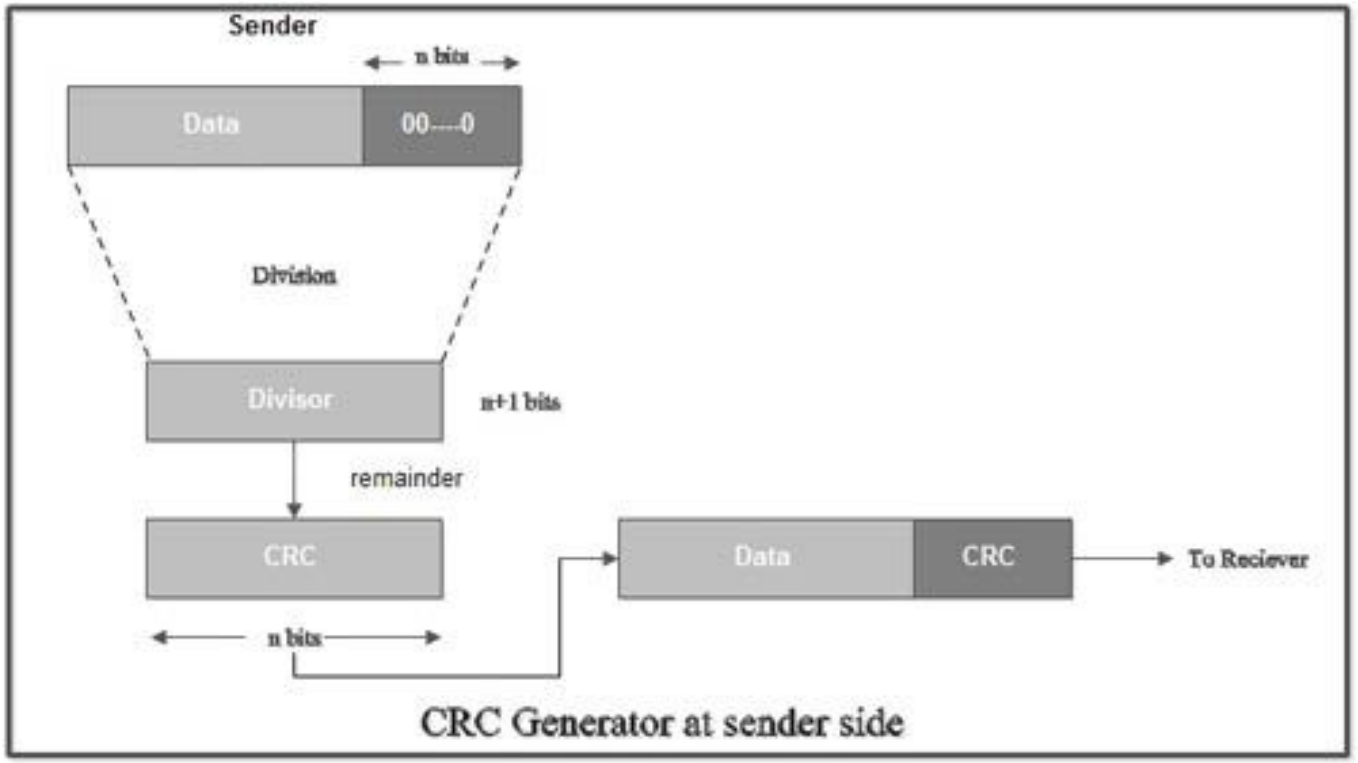
goal: choose *r* CRC bits, R, such that <D,R> exactly divisible by G (mod 2)

- receiver knows G, divides <D,R> by G. If non-zero remainder: error detected!
- can detect all burst errors less than r+1 bits
- widely used in practice (Ethernet, 802.11 WiFi)

COMPUTER NETWORKS

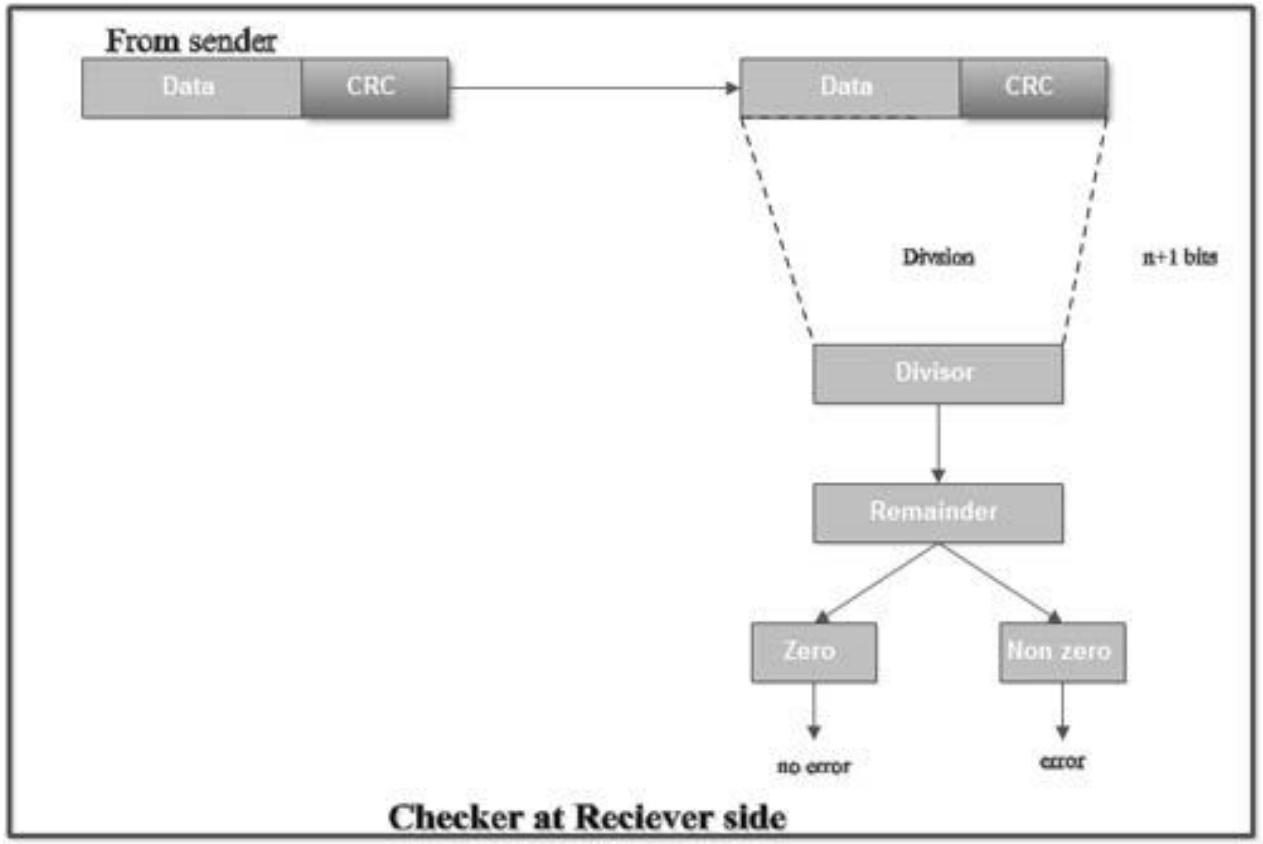
Sender Side – Calculation of CRC

- A string of n 0's is appended to the data unit to be transmitted.
- Here, n is one less than the number of bits in CRC generator.
- Binary division is performed of the resultant string with the CRC generator.
- After division, the remainder so obtained is called as CRC.
- It may be noted that CRC also consists of n bits.



Receiver Side – Check

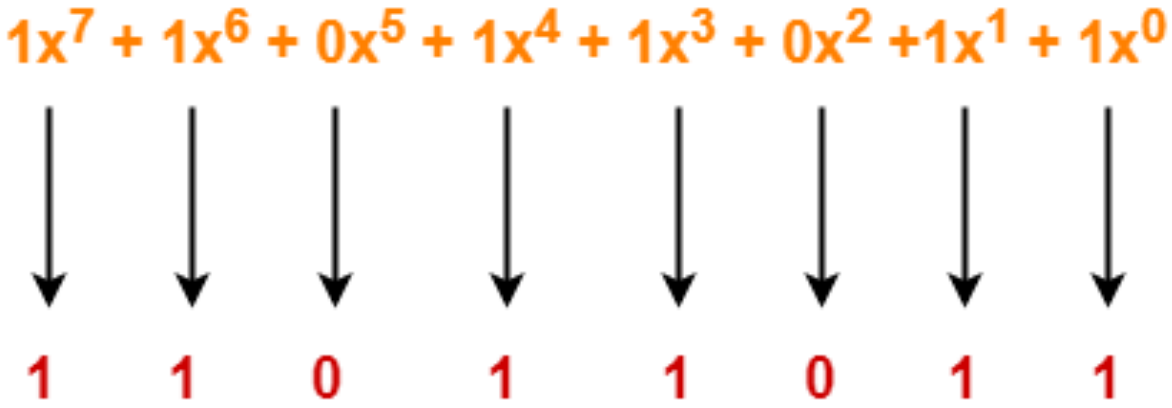
- The received code word is divided with the same CRC generator.
- On division, the remainder so obtained is checked.
- **Case-1: Remainder = 0,**
  - No error
  - Receiver accepts the data
- **Case-2: Remainder ≠ 0**
  - Receiver assumes that some error occurred
  - Receiver rejects the data and asks the sender for retransmission.



COMPUTER NETWORKS

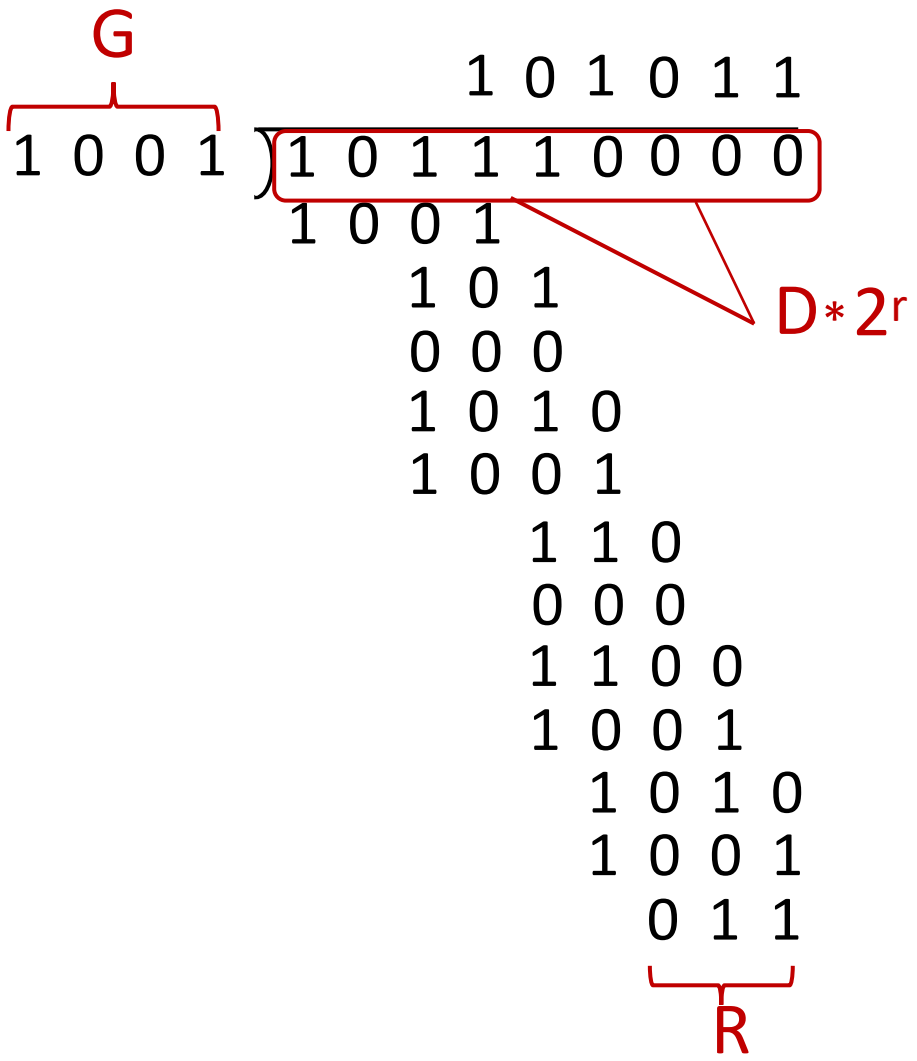
CRC Generator

- an algebraic polynomial represented as a bit pattern
- Rule - The power of each term gives the position of the bit and the coefficient gives the value of the bit.
- Consider the CRC generator is $x^7 + x^6 + x^4 + x^3 + x + 1$.



Cyclic Redundancy Check (CRC): Example

Calculation for the case of:
D = 101110, d = 6,
G = 1001, and r = 3.

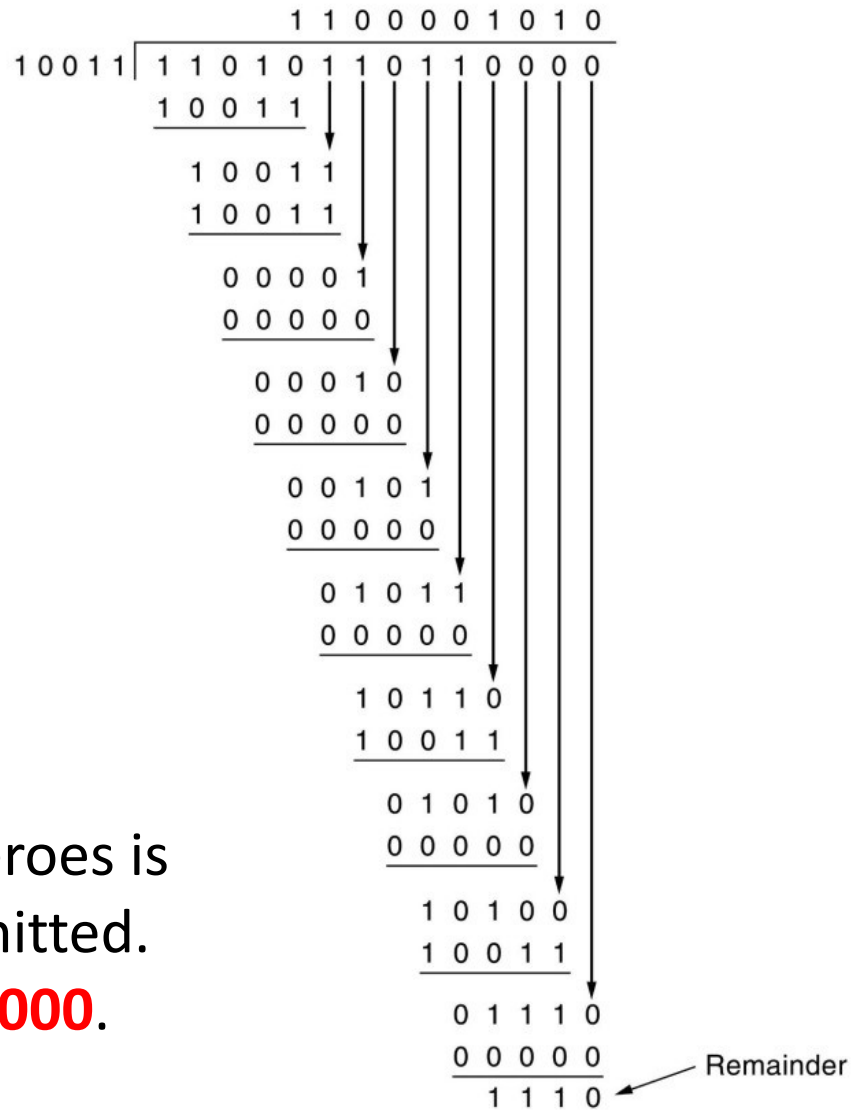


COMPUTER NETWORKS

CRC – Numerical Example

A bit stream 1101011011 is transmitted using the standard CRC method. The generator polynomial is x^4+x+1.

What is the actual bit string transmitted?



Solution

- ✔ Generator polynomial G(x) = $x^4 + x + 1$ is encoded as **10011**.
- ✔ G(x) consists of 5 bits. So, a string of 4 zeroes is appended to the bit stream to be transmitted.
- ✔ The resulting bit stream is 11010110110**0000**.

CRC – Numerical Example

- A bit stream 10011101 is transmitted using the standard CRC method. The generator polynomial is x^3+1.
  - What is the actual bit string transmitted?
  - Suppose the third bit from the left is inverted during transmission. How will receiver detect this error?

Solution

✔ Generator polynomial G(x) = x^3 + 1 is encoded as **1001**.

✔ G(x) consists of 4 bits. 4 zeroes is appended now.

✔ The resulting bit stream is 10011101**000**.

Solution (Part-1)

✔ Code word to be transmitted is obtained by replacing the last 3 zeroes of 10011101**000** with the CRC.

✔ Code word transmitted to the receiver = 10011101**100**.

```
          10001100
      ┌──────────
1001  │ 10011101000
        1001
        -----
        00001
        0000
        ----
        00011
        0000
        ----
        00110
        0000
        ----
        01101
        1001
        ----
        01000
        1001
        ----
        00010
        0000
        ----
        00100
        0000
        ----
        0100   ← CRC
```

CRC – Numerical Example

Solution (Part-2)

According to the question,

- ✔ The bit stream received by the receiver = 10111101100.
- ✔ Receiver performs the binary division with the same G(x).
- ✔ Remainder obtained on division is a non-zero value.
- ✔ Error occurred

```
          10101000
      ┌──────────
1001  │ 10111101100
      1001
      ─────
      00101
      0000
      ─────
      01011
      1001
      ─────
      00100
      0000
      ─────
      01001
      1001
      ─────
      00001
      0000
      ─────
      00010
      0000
      ─────
      00100
      0000
      ─────
      0100   ← Remainder
```

COMPUTER NETWORKS

Unit – 4 Link Layer and LAN Roadmap

- Introduction
- Error detection, correction
- Multiple access protocols
- LANs
  - addressing, ARP
  - Ethernet
  - switches
- A day in the life of a web request
- Physical layer
- Wireless LANs: IEEE 802.11

Multiple access links, Protocols

two types of “links”:

- point-to-point
  - point-to-point link between Ethernet switch, host
  - PPP, HDLC
- broadcast (shared wire or medium)
  - old-fashioned Ethernet
  - upstream HFC in cable-based access network
  - 802.11 wireless LAN, 4G/4G, satellite

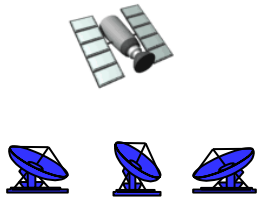
As humans, we’ve evolved an elaborate set of protocols for sharing the broadcast channel:

“Give everyone a chance to speak.”
“Don’t speak until you are spoken to.”
“Don’t monopolize the conversation.”
“Raise your hand if you have a question.”
“Don’t interrupt when someone is speaking.”
“Don’t fall asleep when someone is talking.”

shared wire (e.g., cabled Ethernet)

shared radio: 4G/5G

shared radio: WiFi



shared radio: satellite

humans at a cocktail party (shared air, acoustical)

Multiple access protocols

- Single shared broadcast channel
- Two or more simultaneous transmissions by nodes: interference
  - *collision* if node receives two or more signals at the same time

Multiple access protocol

- Distributed algorithm that determines how nodes share channel, i.e., determine when node can transmit
- Communication about channel sharing must use channel itself!
  - no out-of-band channel for coordination

An ideal Multiple access protocol

Given: multiple access channel (MAC) of rate *R* bps

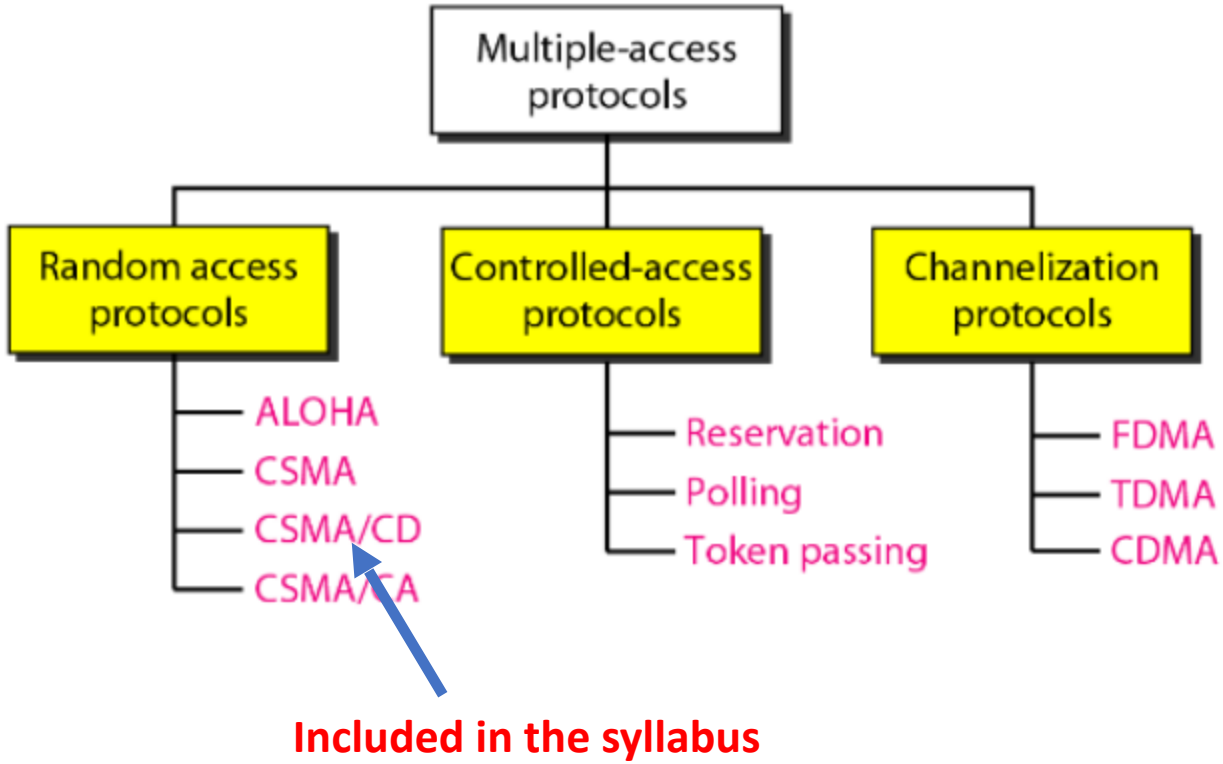
desiderata(needed):

1. when one node wants to transmit, it can send at rate *R*.
2. when M nodes want to transmit, each can send at average rate *R/M*
3. fully decentralized:
   - no special node to coordinate transmissions
   - no synchronization of clocks, slots
4. simple

MAC protocols : Taxonomy

Three broad classes:

- channel partitioning (TDMA, FDMA, CDMA)
  - divide channel into smaller “pieces” (time slots, frequency, code)
  - allocate piece to node for exclusive use
- *random access (ALOHA, slotted ALOHA, CSMA, CSMA/CD, CSMA/CA*
  - channel not divided, allow collisions
  - “recover” from collisions
- “taking turns”
  - nodes take turns, but nodes with more to send can take longer turns



Included in the syllabus

CSMA (Carrier Sense Multiple Access)

Simple CSMA: listen before transmit:

- if channel sensed idle: transmit entire frame
- if channel sensed busy: defer transmission

- Human analogy: don't interrupt others!

***Listen before speaking* – Carrier Sensing**

***If someone else begins talking at the same time, stop talking* – Collision Detection**

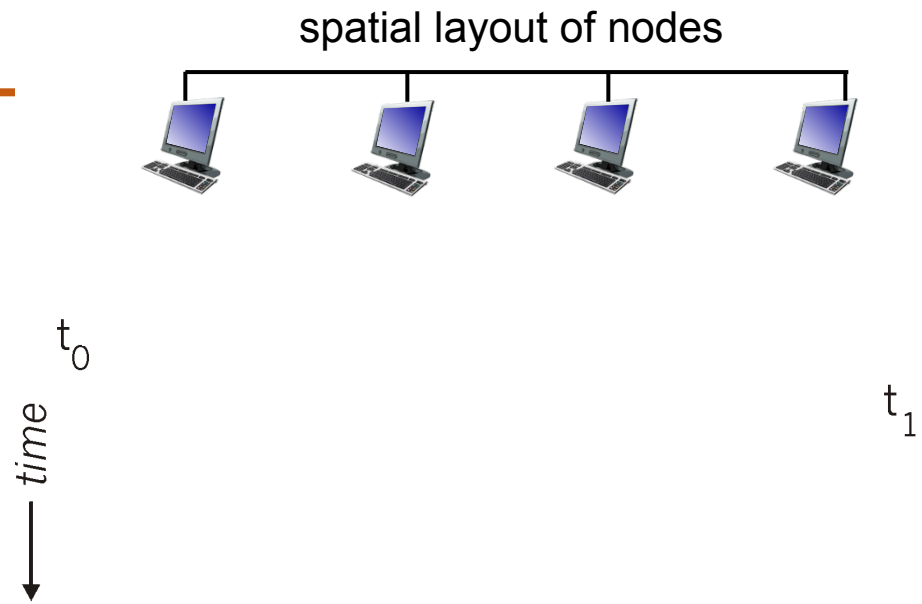
CSMA/CD: CSMA with *collision detection*

- collisions *detected* within short time
- colliding transmissions aborted, reducing channel wastage
- collision detection easy in wired, difficult with wireless

- human analogy: the polite conversationalist

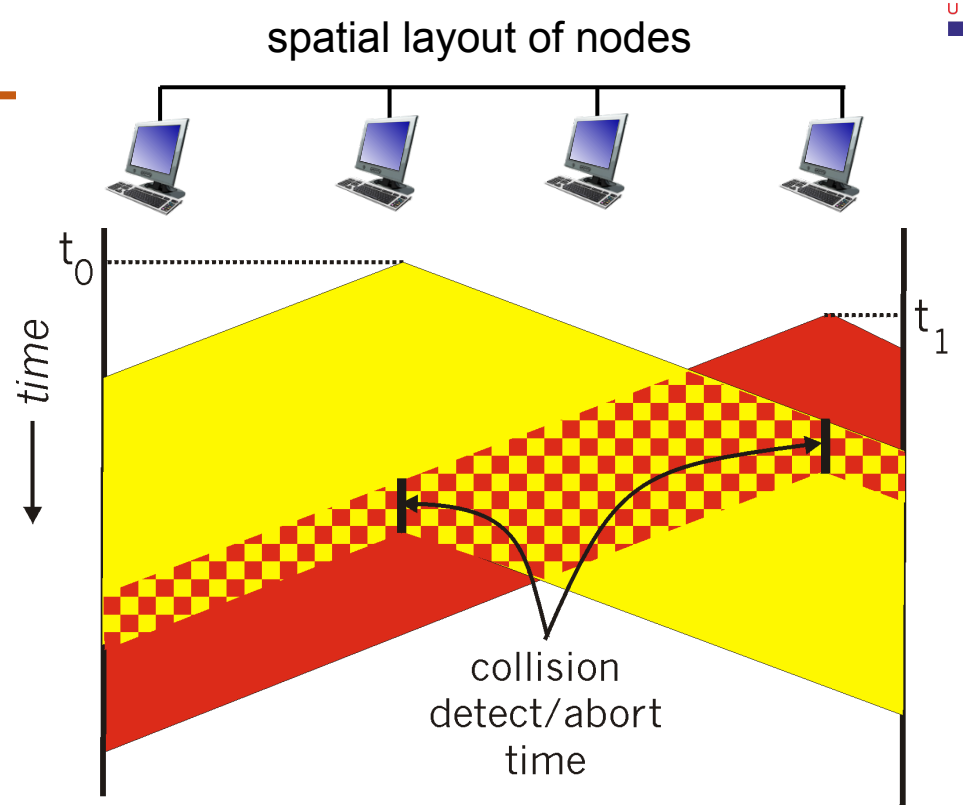
CSMA Collisions

- collisions *can* still occur with carrier sensing:
  - propagation delay means two nodes may not hear each other's just-started transmission
- collision: entire packet transmission time wasted
  - distance & propagation delay play role in in determining collision probability



CSMA/CD

- CSMA/CS reduces the amount of time wasted in collisions
  - transmission aborted on collision detection



Ethernet CSMA/CD Algorithm

1. NIC receives datagram from network layer, creates frame
2. If NIC senses channel:
   if idle: start frame transmission.
   if busy: wait until channel idle, then transmit
3. If NIC transmits entire frame without collision, NIC is done with frame!
4. If NIC detects another transmission while sending: abort, send jam signal
5. After aborting, NIC enters *binary (exponential) backoff:*
   - after *n* collisions, NIC chooses *K* at random from $\{0,1,2, \ldots, 2^m-1\}$.
     NIC waits $K \cdot 512$ bit times, returns to Step 2
   - more collisions: longer backoff interval

CSMA (Carrier Sense Multiple Access)

- T_{prop} = max prop delay between 2 nodes in LAN
- t_{trans} = time to transmit max-size frame

$$efficiency = \frac{1}{1 + 5t_{prop}/t_{trans}}$$

- efficiency goes to 1
  - as t_{prop} goes to 0
  - as t_{trans} goes to infinity
- better performance than ALOHA: and simple, cheap, decentralized!

Unit – 4 Link Layer and LAN Roadmap

- Introduction
- Error detection, correction
- Multiple access protocols
- Switched LANs
  - LL addressing, ARP
  - Ethernet
  - switches
- A day in the life of a web request
- Physical layer
- Wireless LANs: IEEE 802.11

MAC Addresses

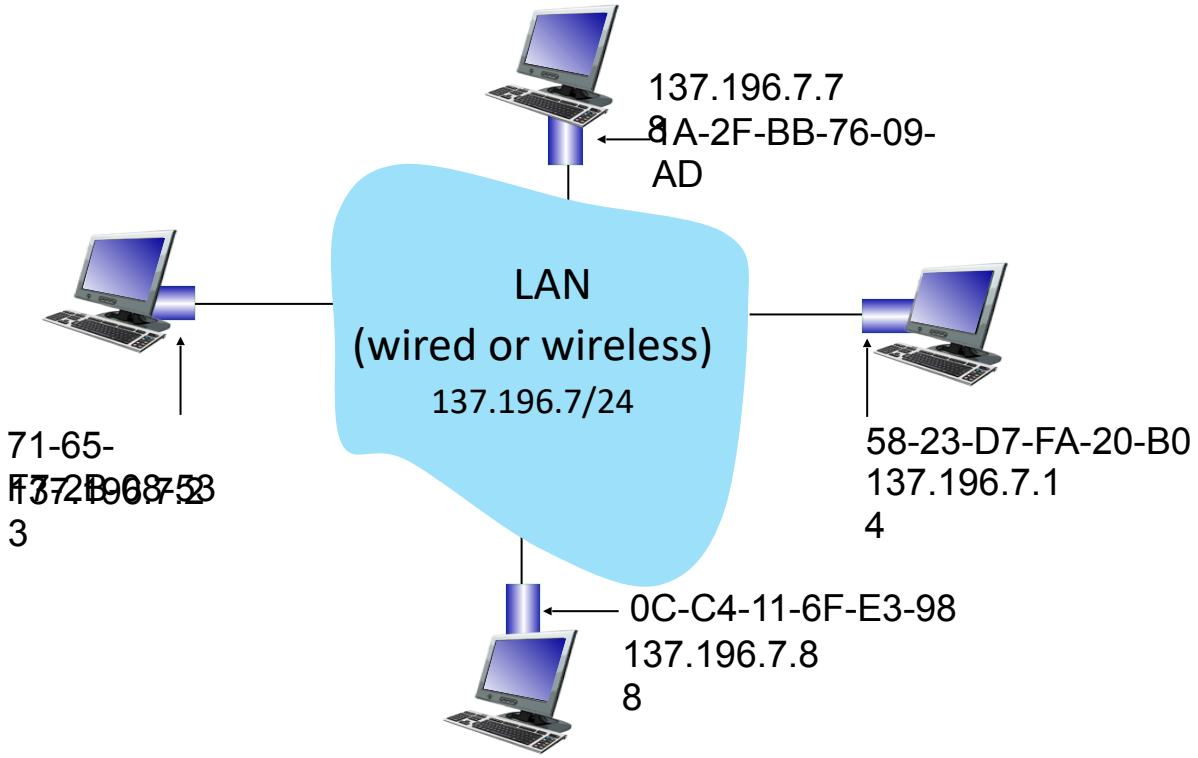
- 32-bit IP address:
  - *network-layer* address for interface
  - used for layer 3 (network layer) forwarding
  - e.g.: 128.119.40.136
- MAC (or LAN or physical or Ethernet) address:
  - function: used "locally" to get frame from one interface to another physically-connected interface (same subnet, in IP-addressing sense)
  - 48-bit MAC address (for most LANs) burned in NIC ROM, also sometimes software settable
  - e.g.: 1A-2F-BB-76-09-AD

hexadecimal (base 16) notation
(each "numeral" represents 4 bits)

MAC Addresses

Each interface on LAN

- has unique 48-bit MAC address
- has a locally unique 32-bit IP address (as we've seen)

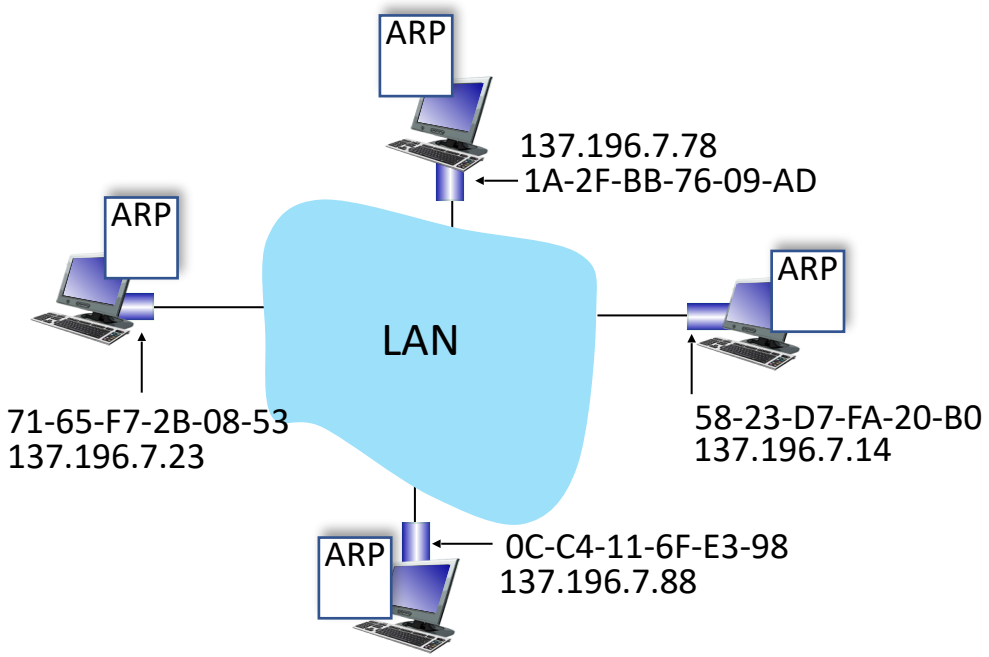


MAC Addresses

- MAC address allocation administered by IEEE
- manufacturer buys portion of MAC address space (to assure uniqueness)
- analogy:
  - MAC address: like Social Security Number
  - IP address: like postal address
- MAC flat address: portability
  - can move interface from one LAN to another
  - recall IP address *not* portable: depends on IP subnet to which node is attached

ARP: Address Resolution Protocol

Question: how to determine interface's MAC address, knowing its IP address?



ARP table: each IP node (host, router) on LAN has table

- IP/MAC address mappings for some LAN nodes:

  < IP address; MAC address; TTL>

- TTL (Time To Live): time after which address mapping will be forgotten (typically 20 min)

ARP Protocol in action

Example: A wants to send datagram to B

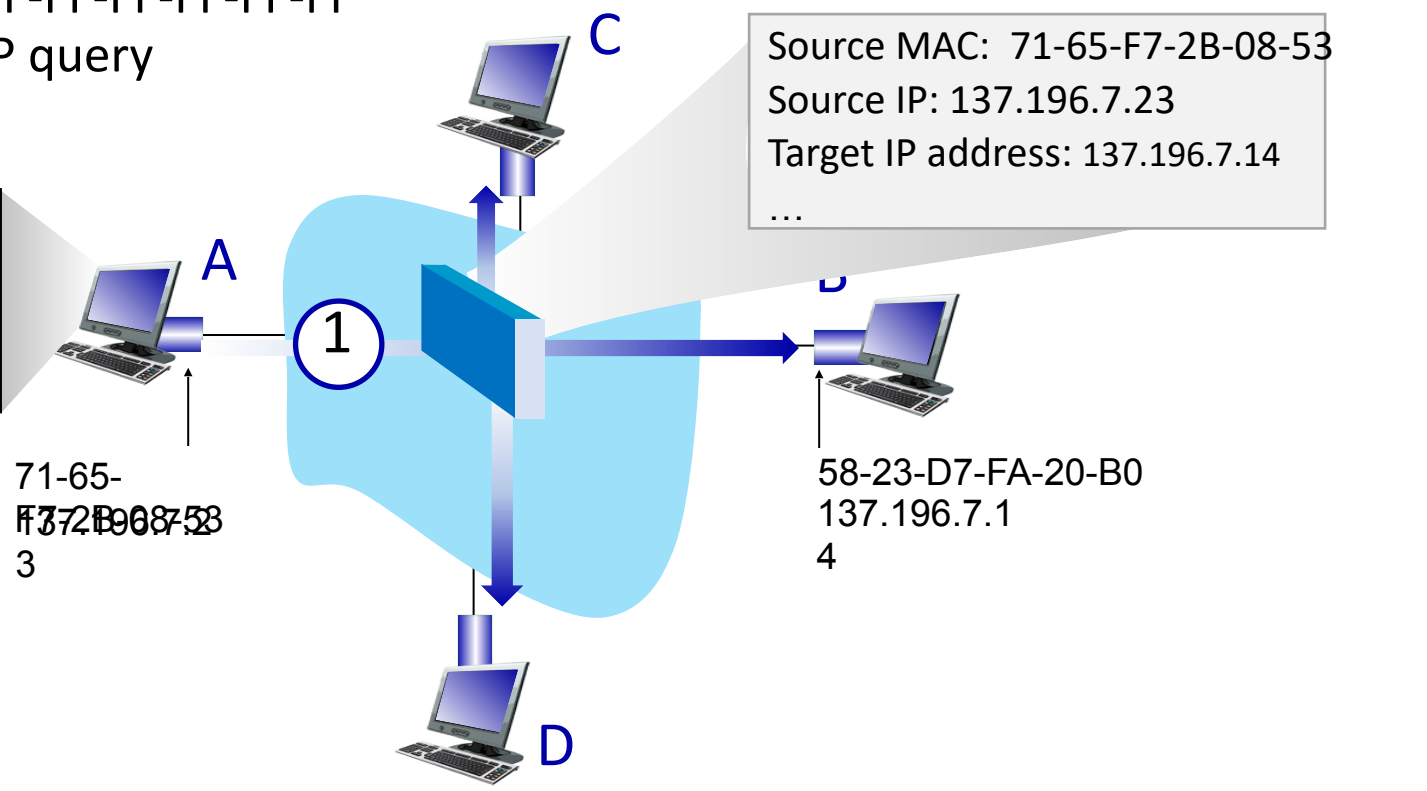
- B's MAC address not in A's ARP table, so A uses ARP to find B's MAC address

A broadcasts ARP query, containing B's IP addr

1 • destination MAC address = FF-FF-FF-FF-FF-FF
• all nodes on LAN receive ARP query

ARP table in A

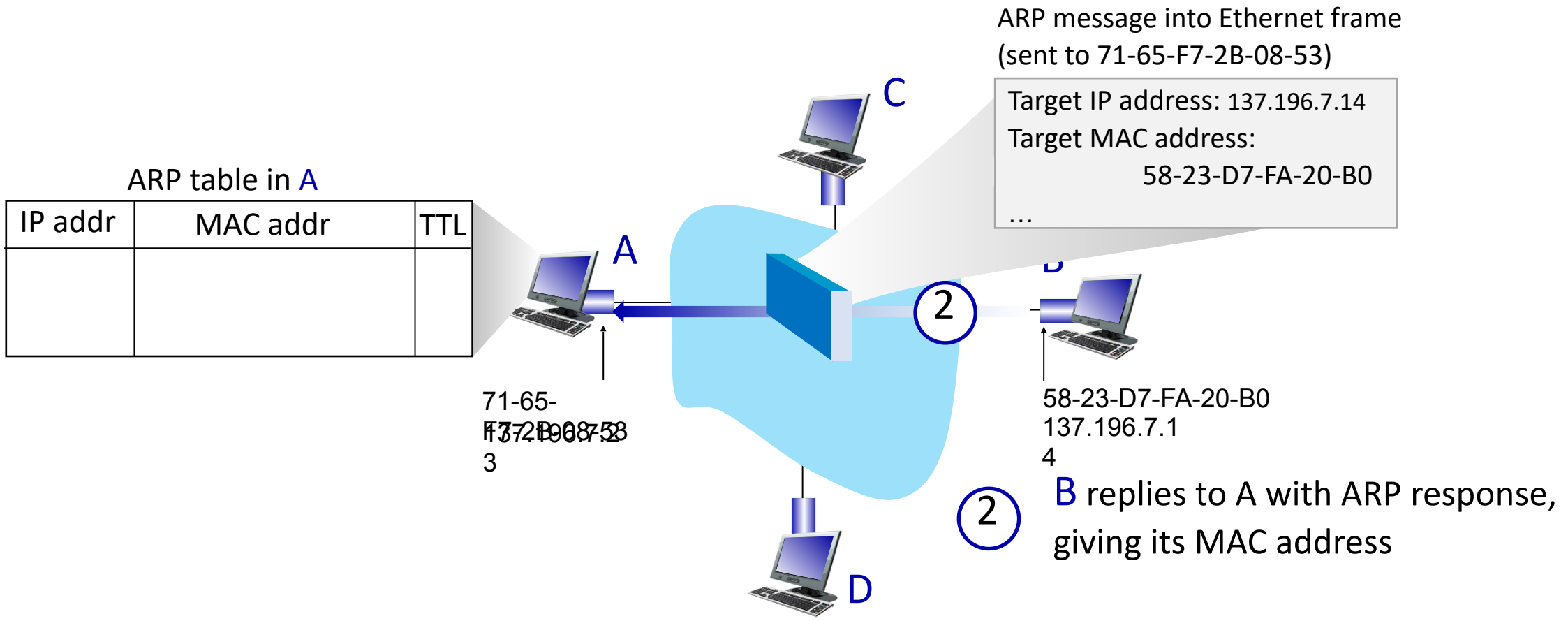
IP addr	MAC addr	TTL



ARP Protocol in action

Example: A wants to send datagram to B

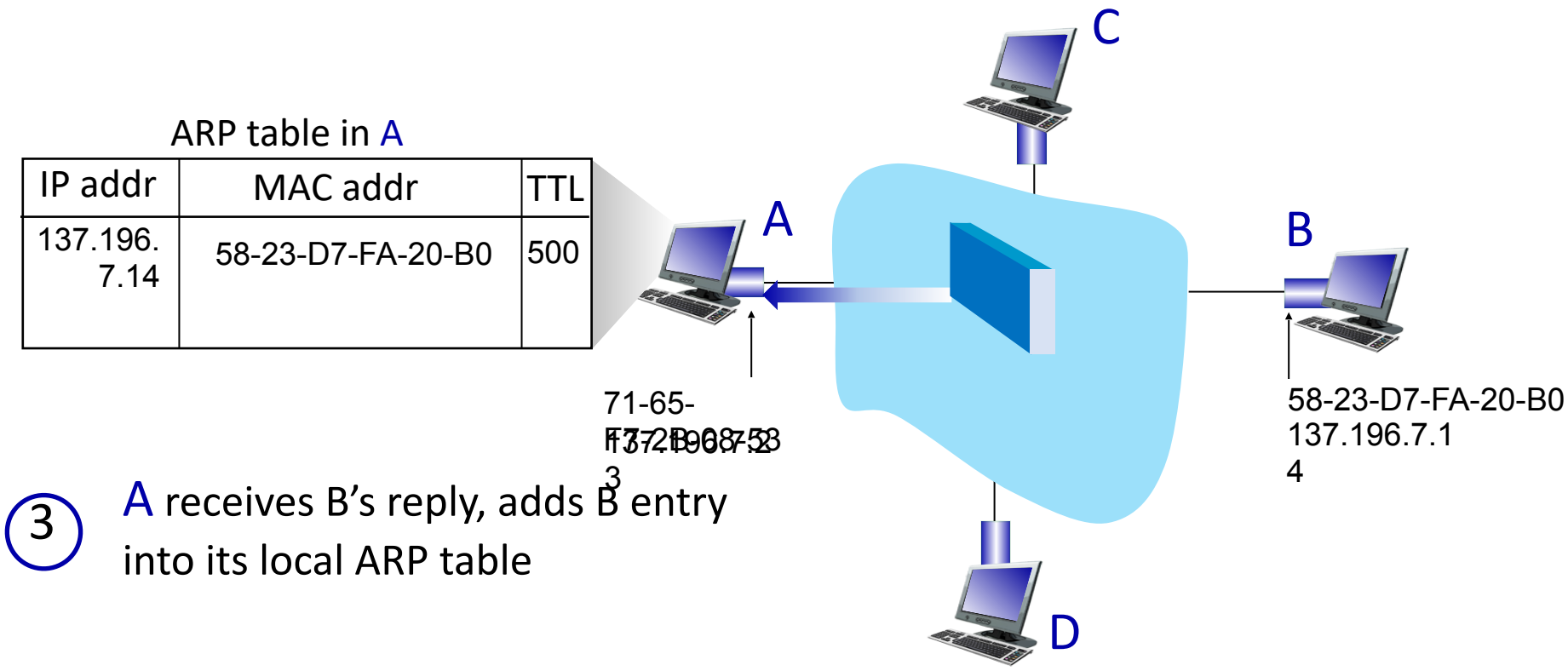
- B's MAC address not in A's ARP table, so A uses ARP to find B's MAC address



ARP Protocol in action

Example: A wants to send datagram to B

- B's MAC address not in A's ARP table, so A uses ARP to find B's MAC address

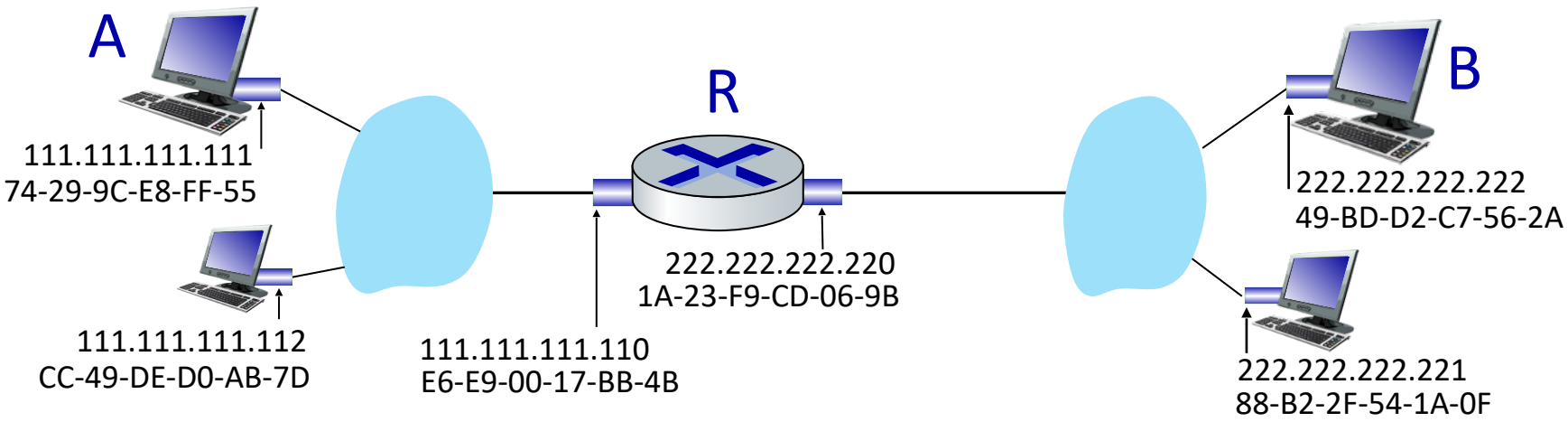


3 A receives B's reply, adds B entry into its local ARP table

Routing to another Subnet : Addressing

Walkthrough: sending a datagram from *A* to *B* via *R*

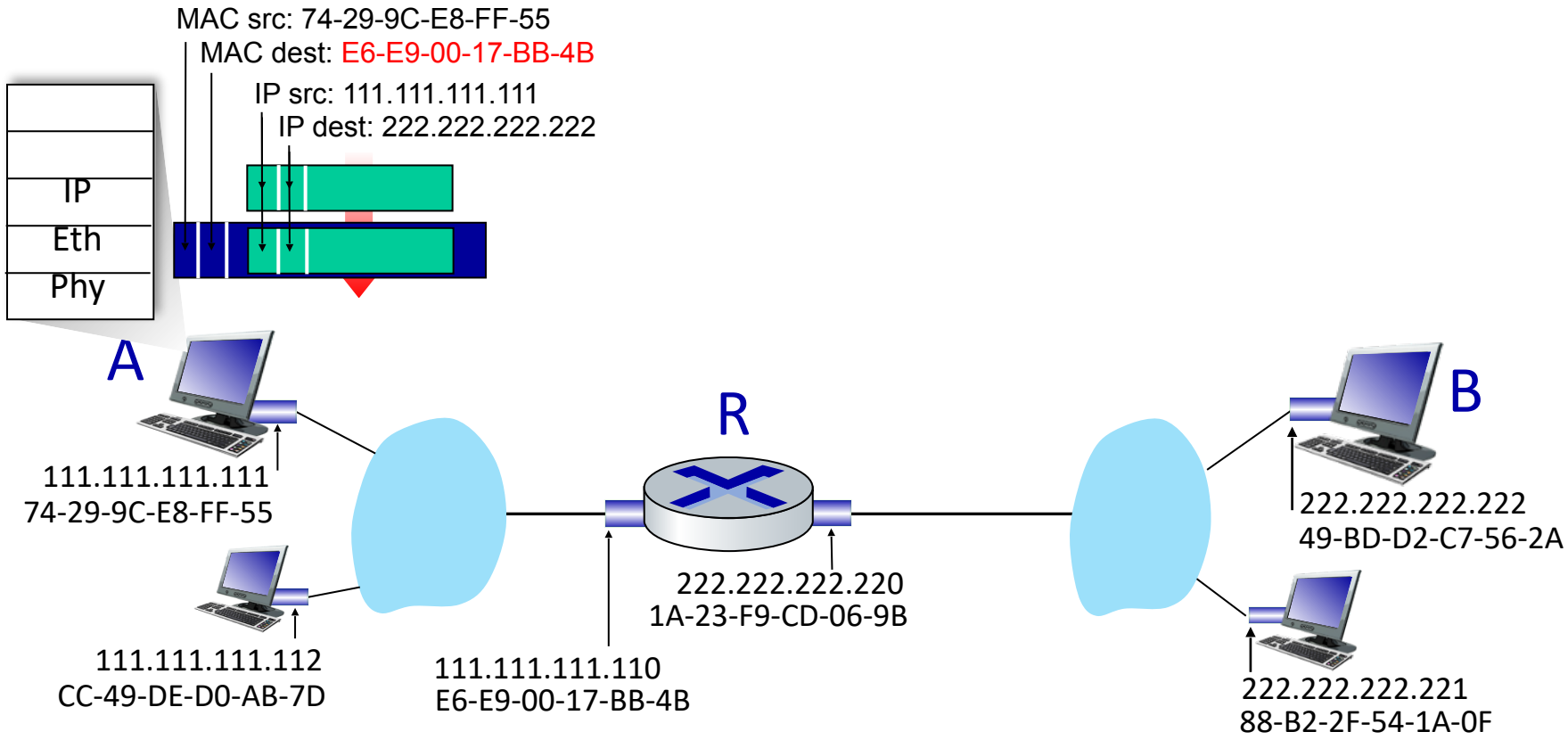
- Focus on addressing – at IP (datagram) and MAC layer (frame) levels
- Assume that:
  - A knows B's IP address
  - A knows IP address of first hop router, R (how?)
  - A knows R's MAC address (how?)



COMPUTER NETWORKS

Routing to another Subnet : Addressing

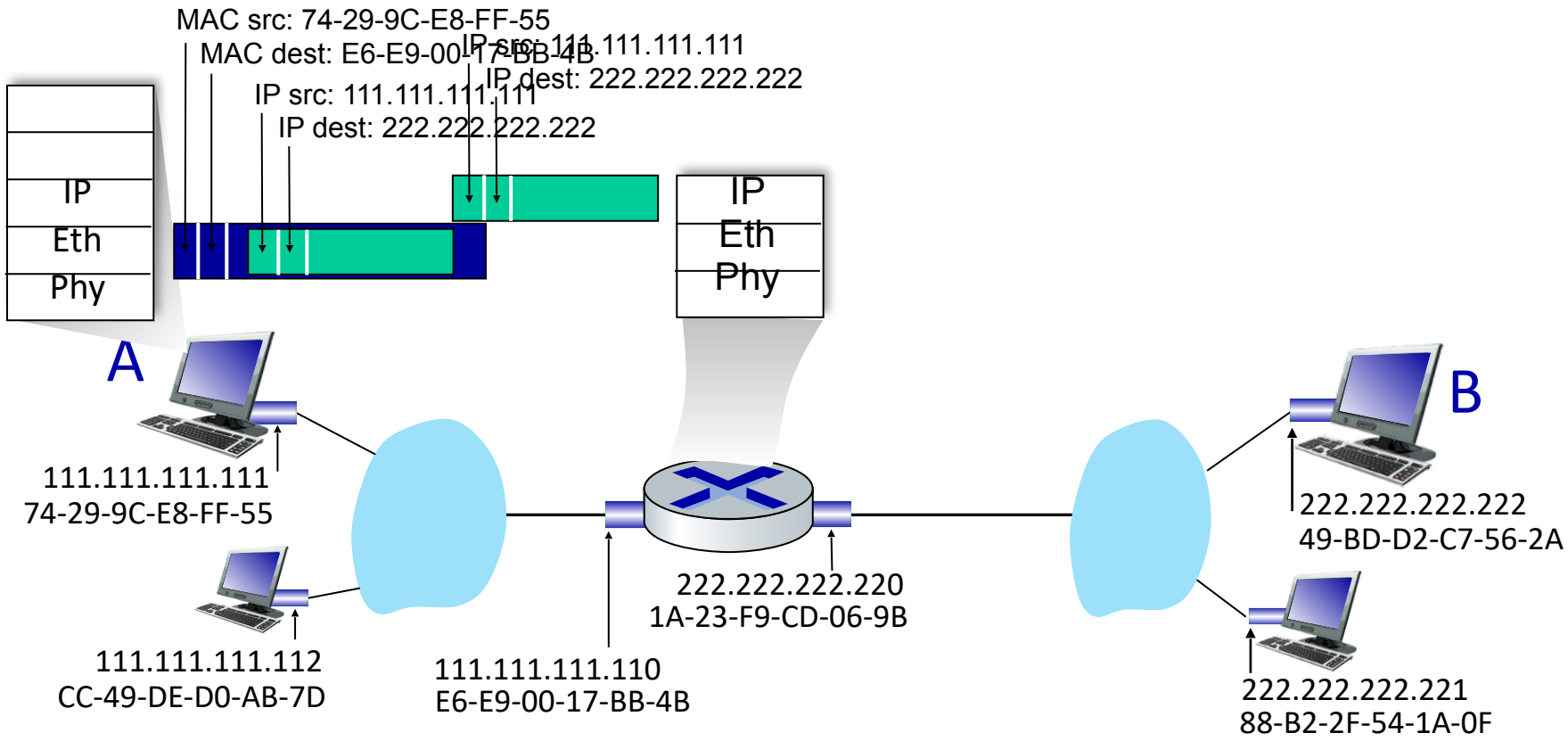
- A creates IP datagram with IP source A, destination B
- A creates link-layer frame containing A-to-B IP datagram
  - R's MAC address is frame's destination



COMPUTER NETWORKS

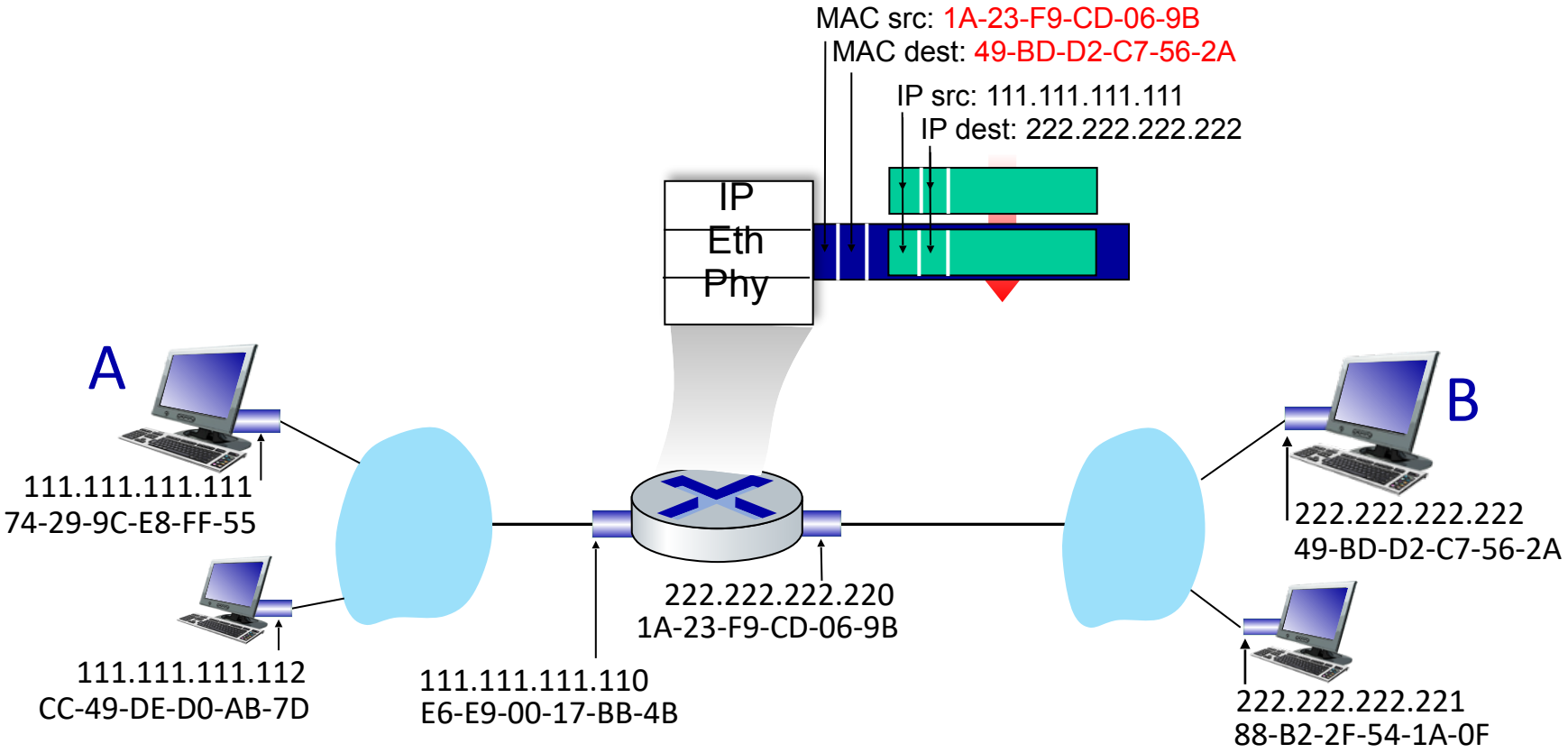
Routing to another Subnet : Addressing

- frame sent from A to R
- frame received at R, datagram removed, passed up to IP



Routing to another Subnet : Addressing

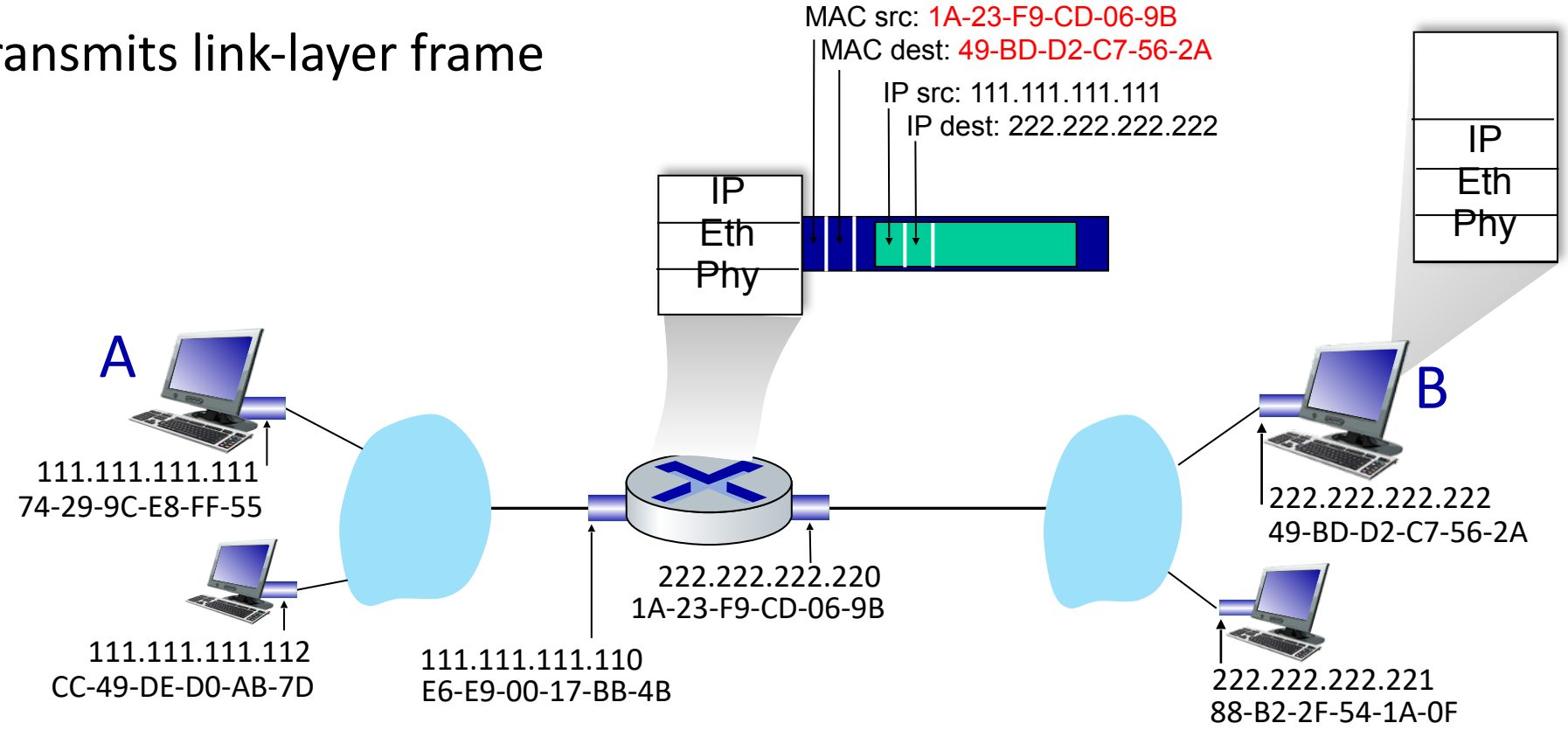
- R determines outgoing interface, passes datagram with IP source A, destination B to link layer
- R creates link-layer frame containing A-to-B IP datagram. Frame destination address: B's MAC address



COMPUTER NETWORKS

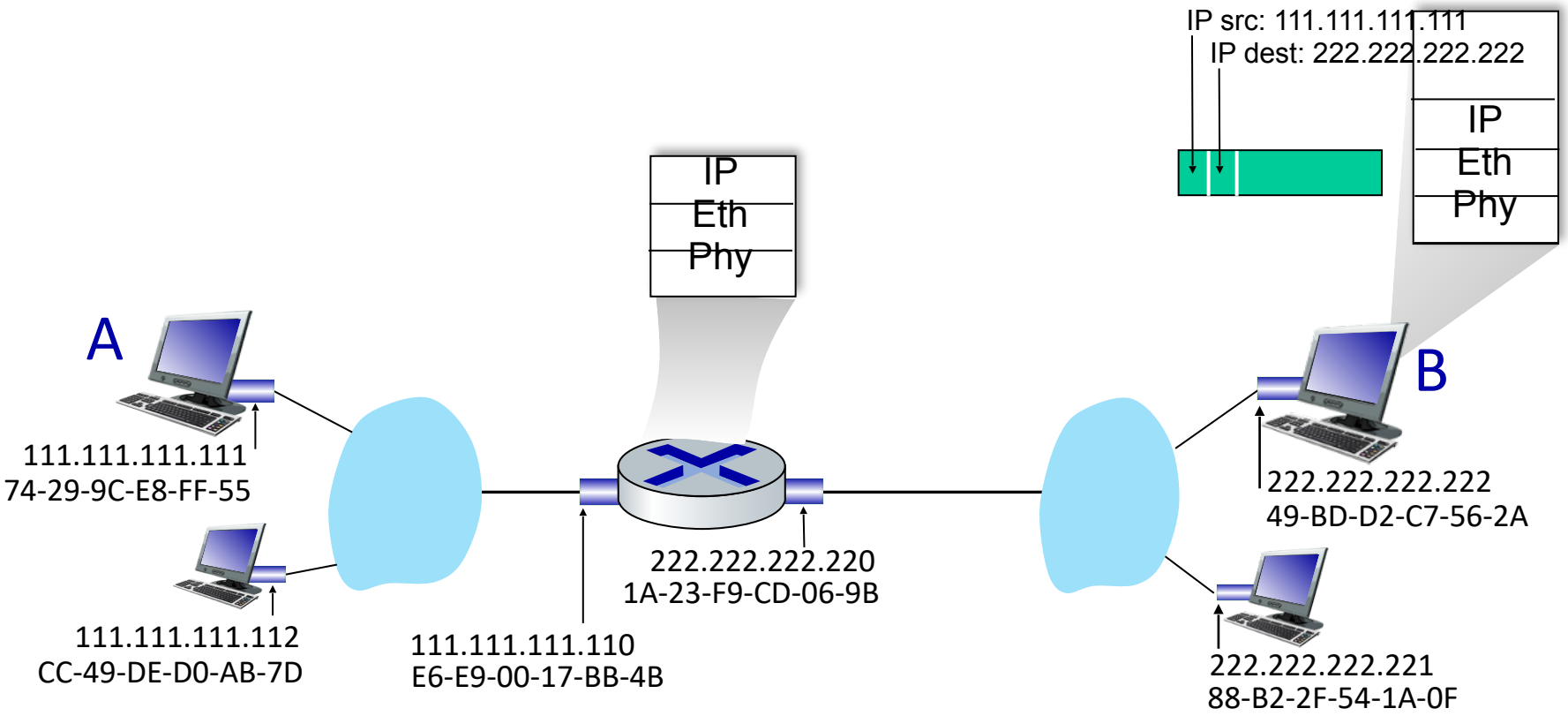
Routing to another Subnet : Addressing

- R determines outgoing interface, passes datagram with IP source A, destination B to link layer
- R creates link-layer frame containing A-to-B IP datagram. Frame destination address: B's MAC address
- Transmits link-layer frame



Routing to another Subnet : Addressing

- B receives frame, extracts IP datagram destination B
- B passes datagram up protocol stack to IP



Unit – 4 Link Layer and LAN Roadmap

- Introduction
- Error detection, correction
- Multiple access protocols
- Switched LANs
  - LL addressing, ARP
  - Ethernet
  - switches
- A day in the life of a web request
- Physical layer
- Wireless LANs: IEEE 802.11

Ethernet

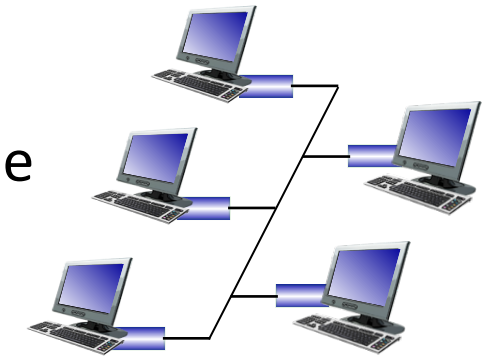
"dominant" wired LAN technology:

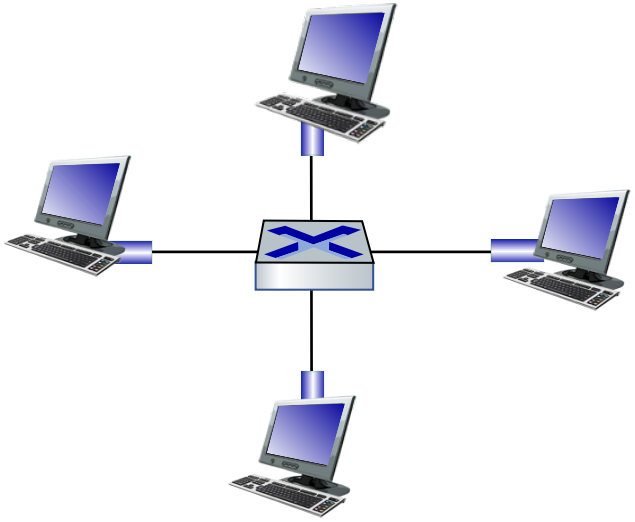
- first widely used LAN technology
- simpler, cheap
- kept up with speed race: 10 Mbps – 400 Gbps
- single chip, multiple speeds (e.g., Broadcom BCM5761)
- Ethernet' Success
  - First widely deployed high-speed LAN
  - Token ring, FDDI, and ATM were more complex and expensive than Ethernet
  - Operated at equal data rates or higher.

Ethernet : Physical Topology

- Bus: popular through mid 90s
  - all nodes in same collision domain (can collide with each other)
- Switched: prevails today
  - active link-layer 2 *switch* in center
  - each “spoke” runs a (separate) Ethernet protocol (nodes do not collide with each other)

bus: coaxial cable





switched

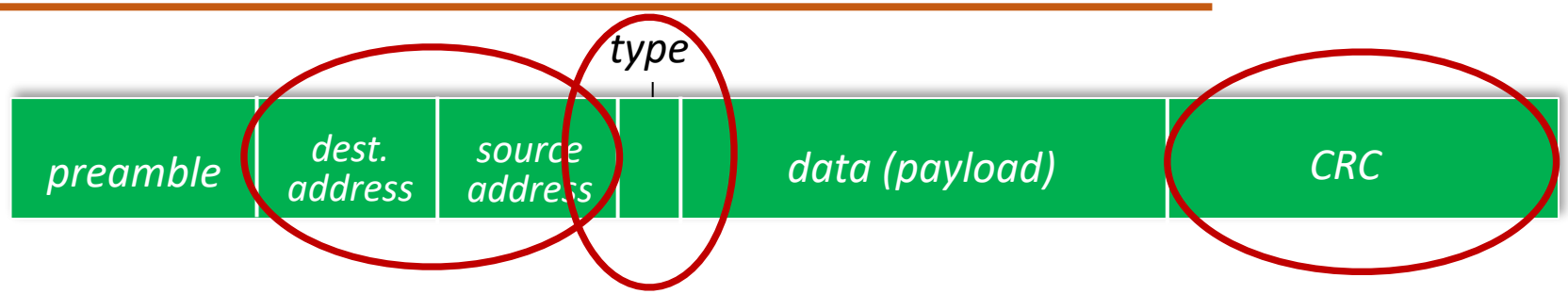
Ethernet Frame Structure

Sending interface encapsulates IP datagram (or other network layer protocol packet) in Ethernet frame

Preamble:

- Used to synchronize receiver, sender clock rates
- 7 bytes of 10101010 followed by one byte of 10101011

Ethernet Frame Structure (more)



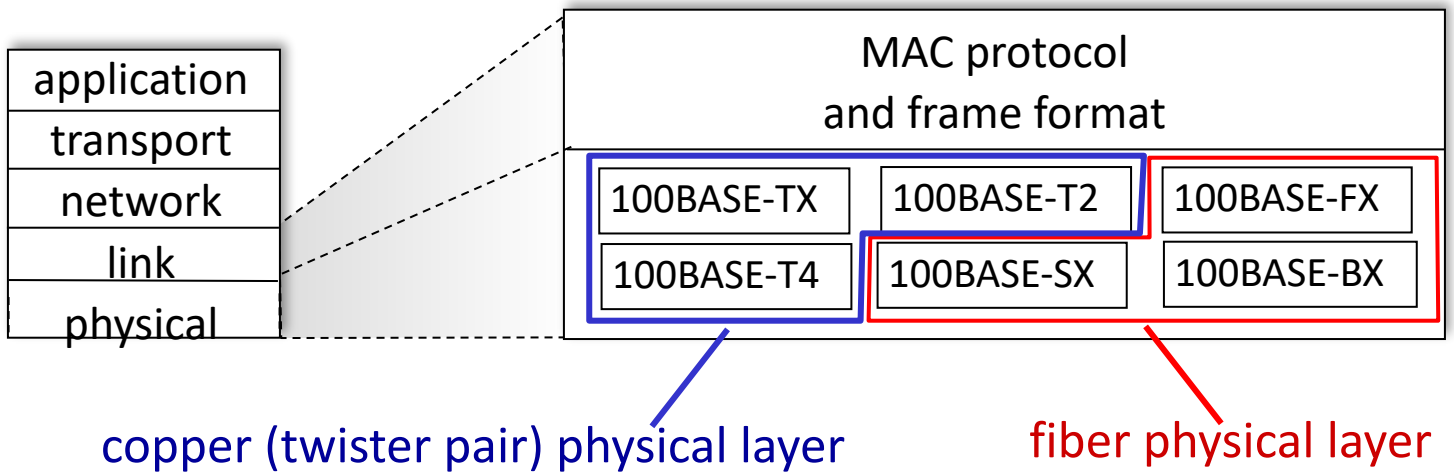
- Addresses: 6 byte source, destination MAC addresses
  - if adapter receives frame with matching destination address, or with broadcast address (e.g., ARP packet), it passes data in frame to network layer protocol
  - otherwise, adapter discards frame
- Type: indicates higher layer protocol
  - mostly IP but others possible, e.g., Novell IPX, AppleTalk
  - used to demultiplex up at receiver
- CRC: cyclic redundancy check at receiver
  - error detected: frame is dropped

Ethernet : Unreliable Connectionless

- Connectionless: no handshaking between sending and receiving NICs
- Unreliable: receiving NIC doesn't send ACKs or NAKs to sending NIC
  - data in dropped frames recovered only if initial sender uses higher layer rdt (e.g., TCP), otherwise dropped data lost
- Ethernet's MAC protocol: unslotted CSMA/CD with binary backoff

802.3 Ethernet Standards: Link and Physical Layers

- *Many* different Ethernet standards
  - Common MAC protocol and frame format
  - Different speeds: 2 Mbps, 10 Mbps, 100 Mbps, 1Gbps, 10 Gbps, 40 Gbps
  - Different physical layer media: fiber, cable



Unit – 4 Link Layer and LAN Roadmap

- Introduction
- Error detection, correction
- Multiple access protocols
- Switched LANs
  - Addressing, ARP
  - Ethernet
  - Switches
- A day in the life of a web request
- Physical layer
- Wireless LANs: IEEE 802.11

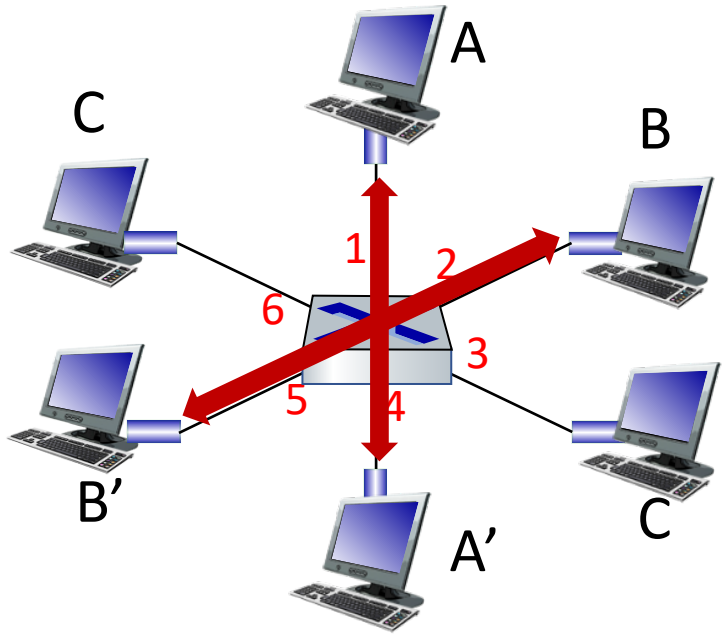
Ethernet switch

- Switch is a link-layer device: takes an *active* role
  - Store, forward Ethernet frames
  - Examine incoming frame's MAC address, *selectively* forward frame to one-or-more outgoing links when frame is to be forwarded on segment, uses CSMA/CD to access segment
- Transparent: hosts *unaware* of presence of switches
- Plug-and-play, self-learning
  - Switches do not need to be configured

Switch : Multiple Simultaneous Transmissions

- Hosts have dedicated, direct connection to switch
- Switches buffer packets
- Ethernet protocol used on *each* incoming link, so:
  - no collisions; full duplex
  - each link is its own collision domain

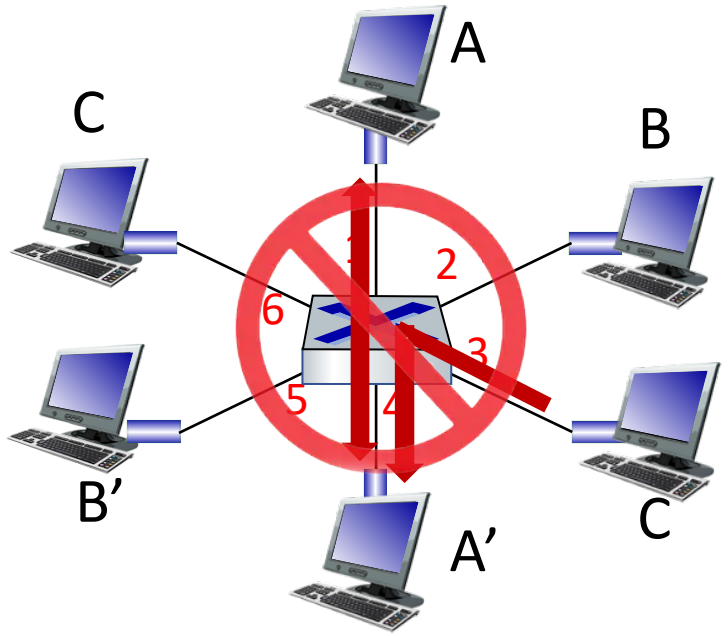
- Switching: A-to-A' and B-to-B' can transmit simultaneously, without collisions



switch with six interfaces (1,2,3,4,5,6)

Switch: Multiple Simultaneous Transmissions

- Hosts have dedicated, direct connection to switch
- Switches buffer packets
- Ethernet protocol used on *each* incoming link, so:
  - No collisions; full duplex
  - Each link is its own collision domain
- Switching: A-to-A' and B-to-B' can transmit simultaneously, without collisions
  - but A-to-A' and C to A' can *not* happen simultaneously



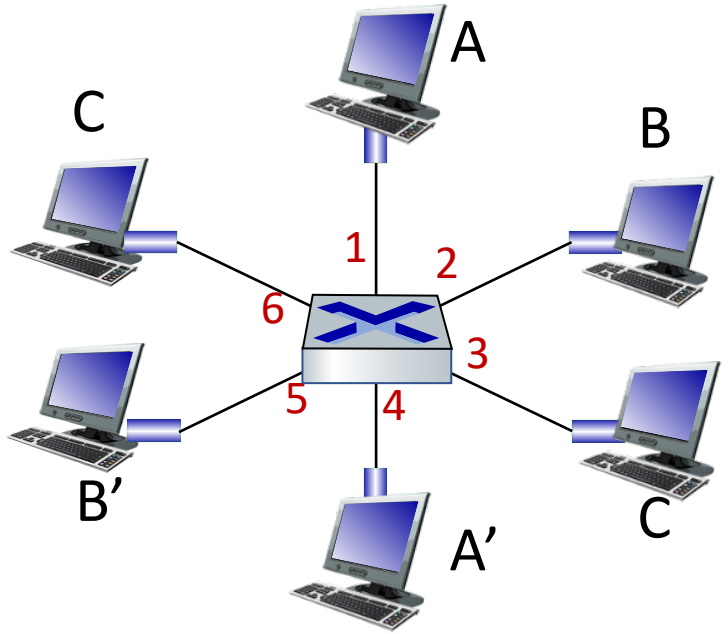
switch with six interfaces (1,2,3,4,5,6)

Switch Forwarding Table

Q: How does switch know A' reachable via interface 4, B' reachable via interface 5?

A: Each switch has a switch table, each entry:

- (MAC address of host, interface to reach host, time stamp)
- looks like a routing table!

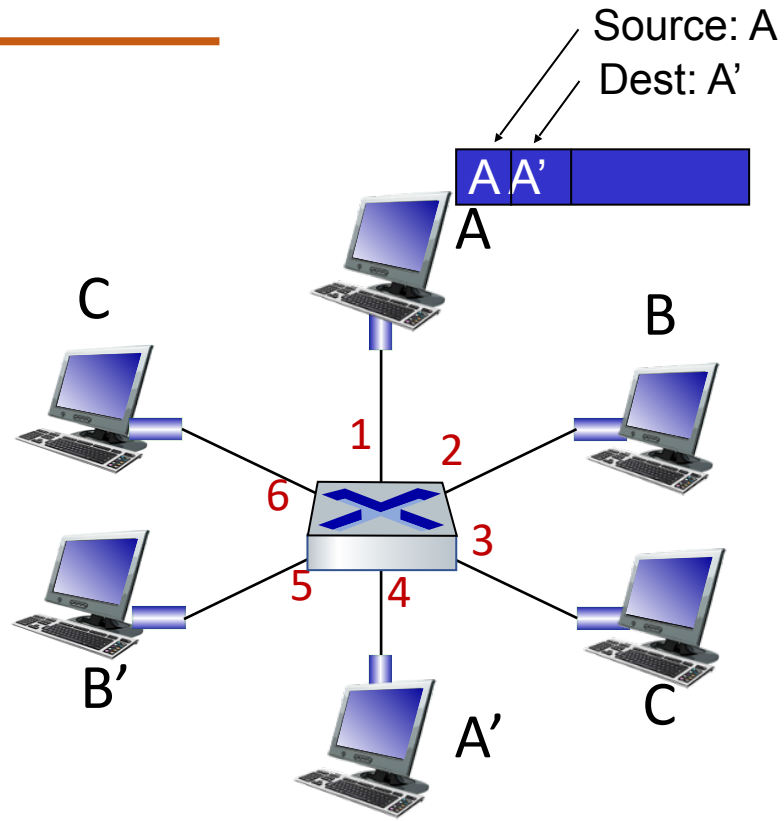


Q: How are entries created, maintained in switch table?

- something like a routing protocol?

Switch: Self-learning

- Switch *learns* which hosts can be reached through which interfaces
  - When frame received, switch "learns" location of sender: incoming LAN segment
  - Records sender/location pair in switch table



Switch table (initially empty)

MAC addr	interface	TTL
A	*1*	*60*

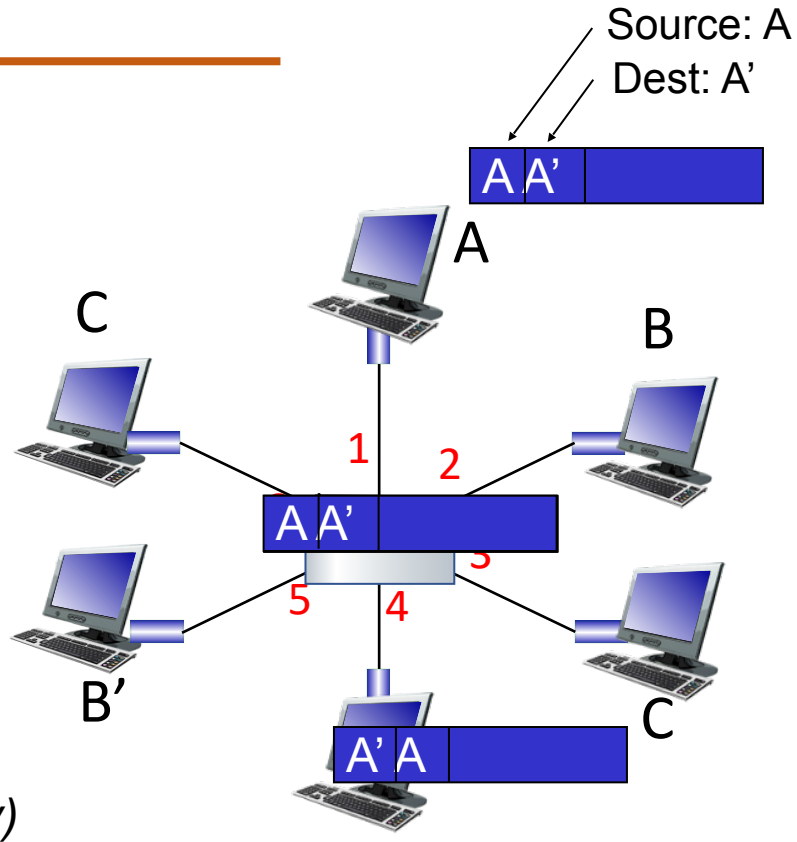
Switch: Frame Filtering / Forwarding

When frame received at switch:

1. Record incoming link, MAC address of sending host
2. Index switch table using MAC destination address
3. If entry found for destination
   then {
   If destination on segment from which frame arrived
   then drop frame
   else forward frame on interface indicated by entry
   }
   else flood /* forward on all interfaces except arriving interface */

Self-learning, Forwarding: Example

- Frame destination, A', location unknown: Flood
- Destination A location known: Selectively send on just one link



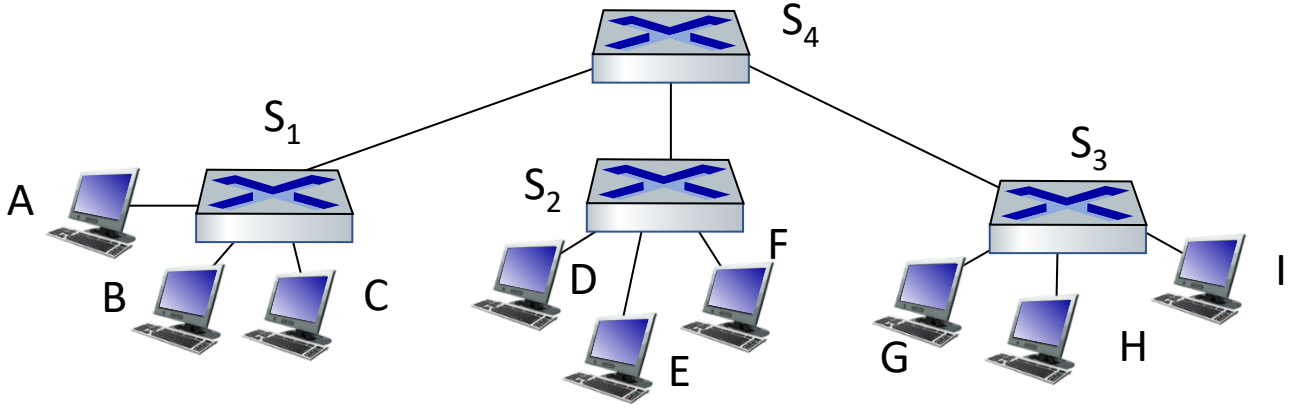
switch table (initially empty)

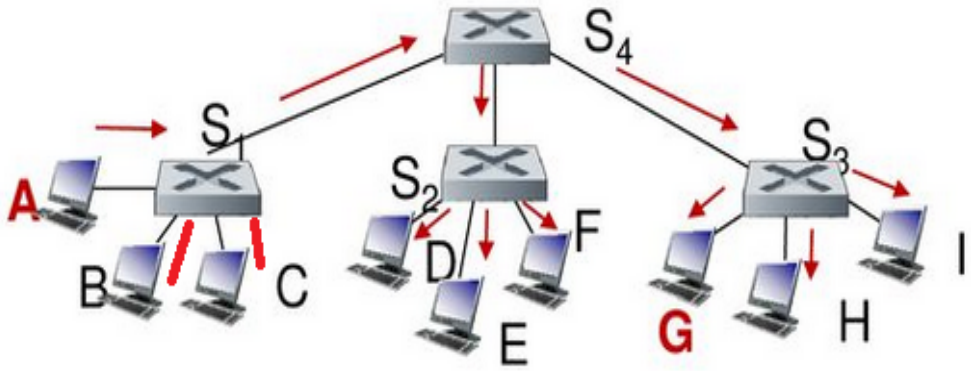
MAC addr	interface	TTL
A	*1*	*60*
A'	*4*	*60*

COMPUTER NETWORKS

Interconnecting Switches

Self-learning switches can be connected together:





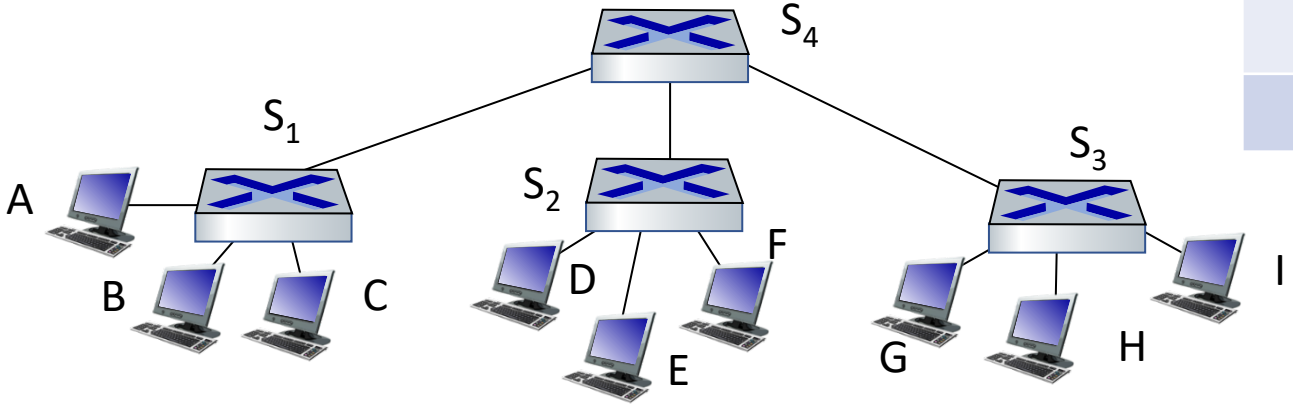
Q: sending from A to G - how does S_1 know to forward frame destined to G via S_4 and S_3?

- *A:* self learning! (works exactly the same as in single-switch case!)

COMPUTER NETWORKS

Self-learning Multi-switch Example

Suppose C sends frame to I, I responds to C



Switch 1

Address	Port
C	1
I	4

Switch 4

Address	Port
C	1
I	3

Switch 2

Address	Port
C	4

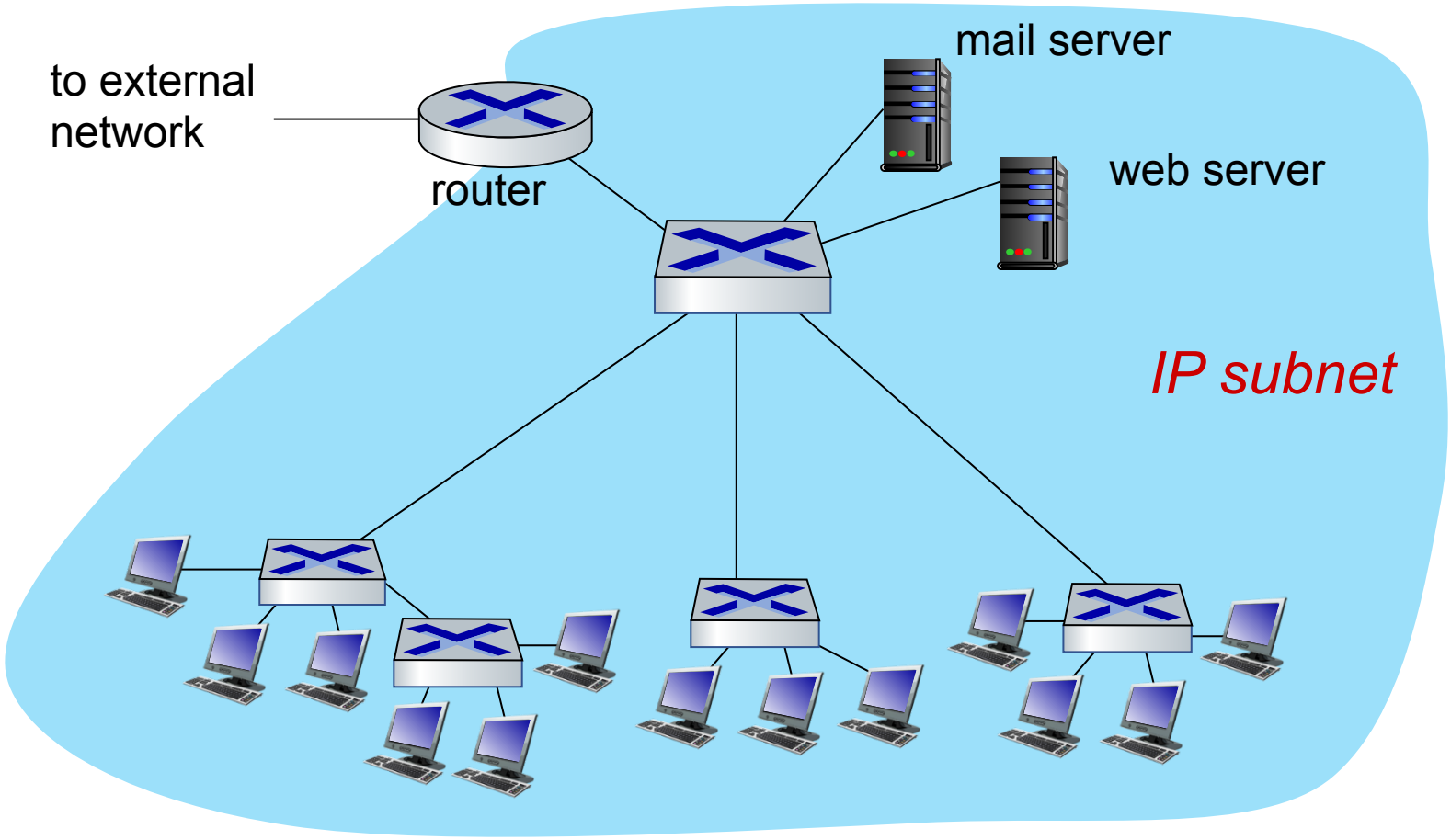
Switch 3

Address	Port
C	4
I	3

Q: show switch tables and packet forwarding in S_1, S_2, S_3, S_4

COMPUTER NETWORKS

Small Institutional Network



COMPUTER NETWORKS

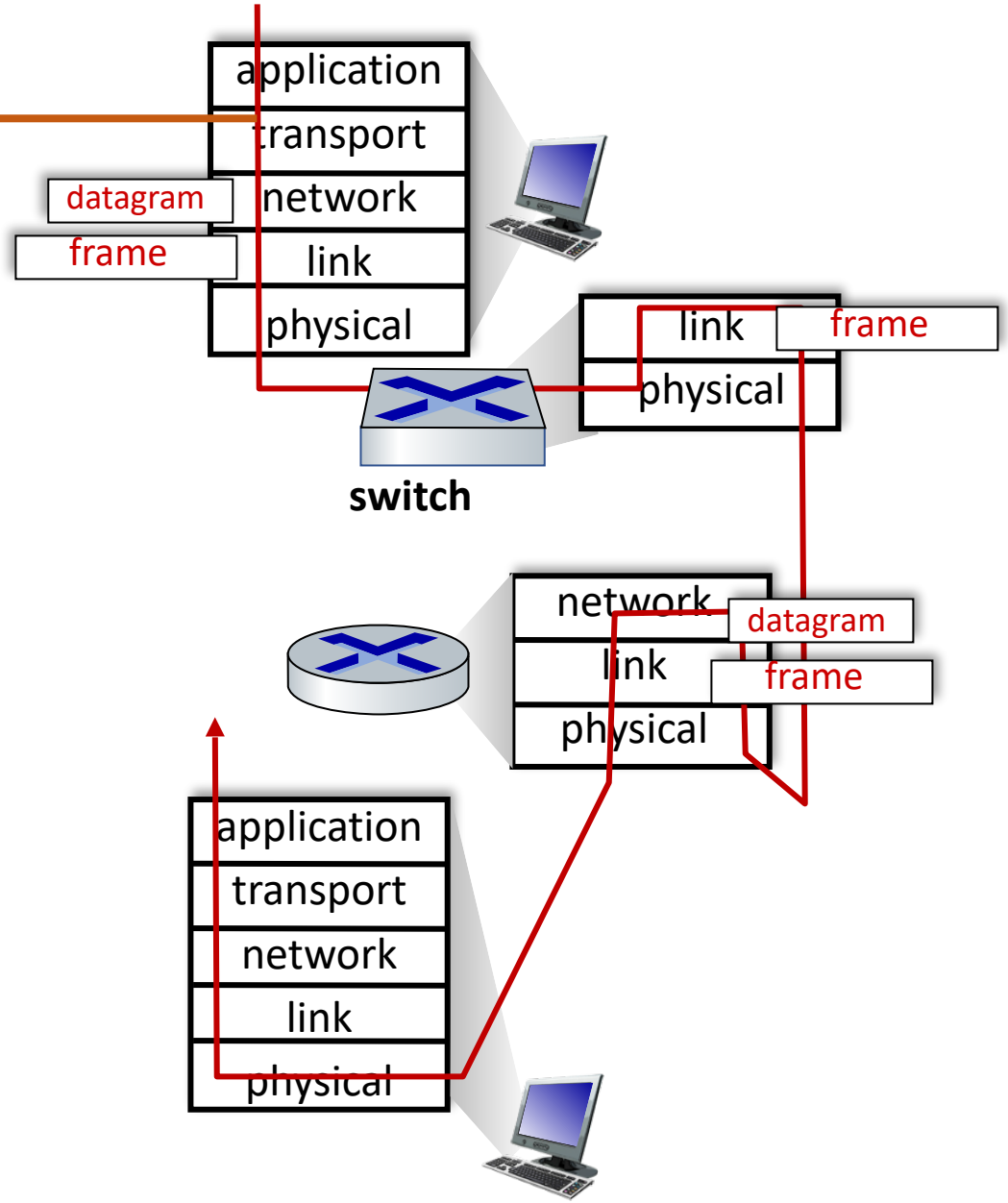
Switches Vs Routers

Both are store-and-forward:

- *Routers*: network-layer devices (examine network-layer headers)
- *Switches:* link-layer devices (examine link-layer headers)

Both have forwarding tables:

- *Routers:* compute tables using routing algorithms, IP addresses
- *Switches:* learn forwarding table using flooding, learning, MAC addresses



Unit – 4 Link Layer and LAN Roadmap

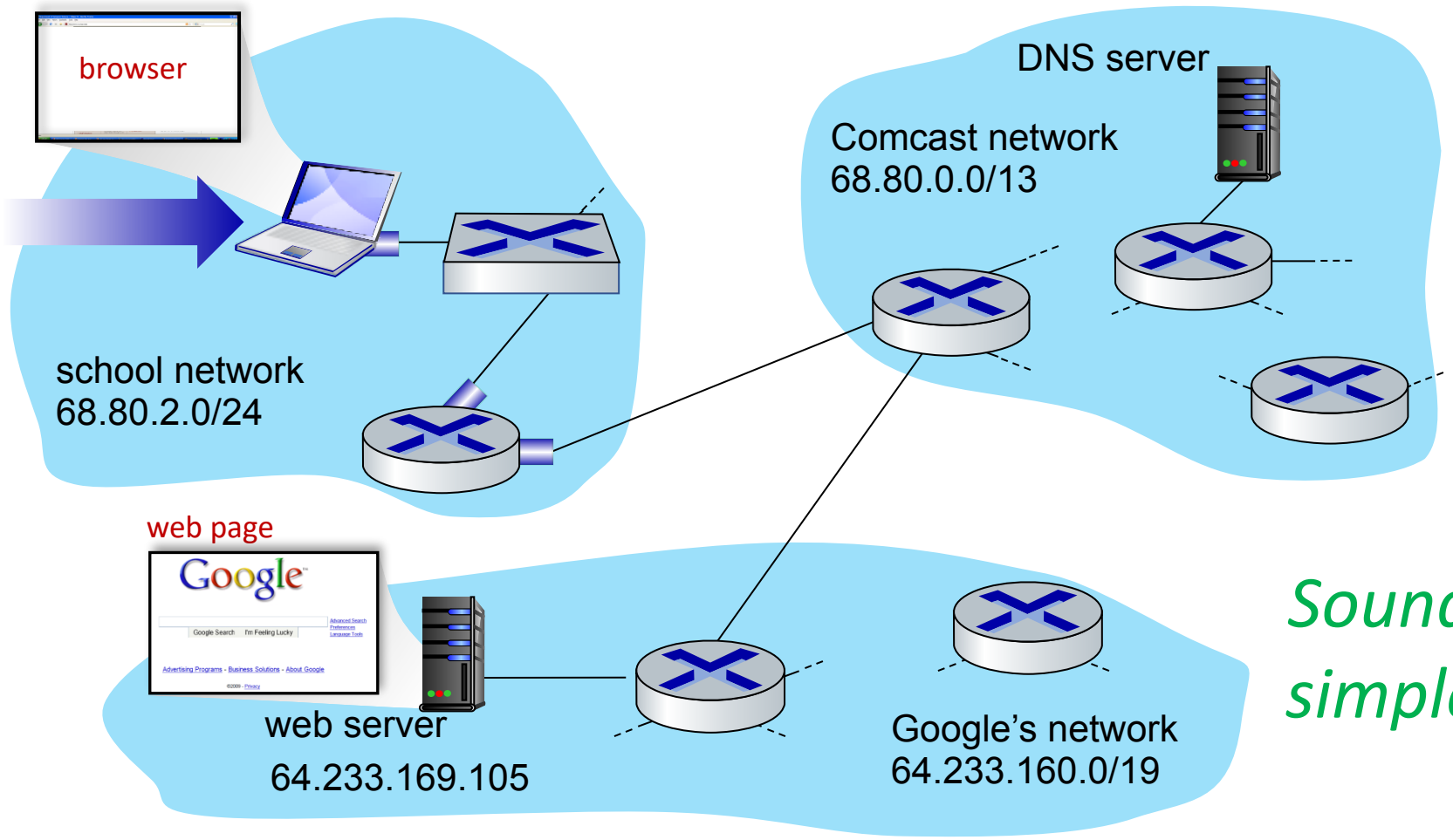
- Introduction
- Error detection, correction
- Multiple access protocols
- Switched LANs
  - addressing, ARP
  - Ethernet
  - switches
- A day in the life of a web request
- Physical layer
- Wireless LANs: IEEE 802.11

Synthesis : A day in the life of a web request

- Our journey down the protocol stack is now complete!
  - application, transport, network, link
- Putting-it-all-together: synthesis!
  - *Goal:* identify, review, understand protocols (at all layers) involved in seemingly simple scenario: requesting www page
  - *Scenario:* student attaches laptop to campus network, requests/receives www.google.com

COMPUTER NETWORKS

A day in the life of a web request

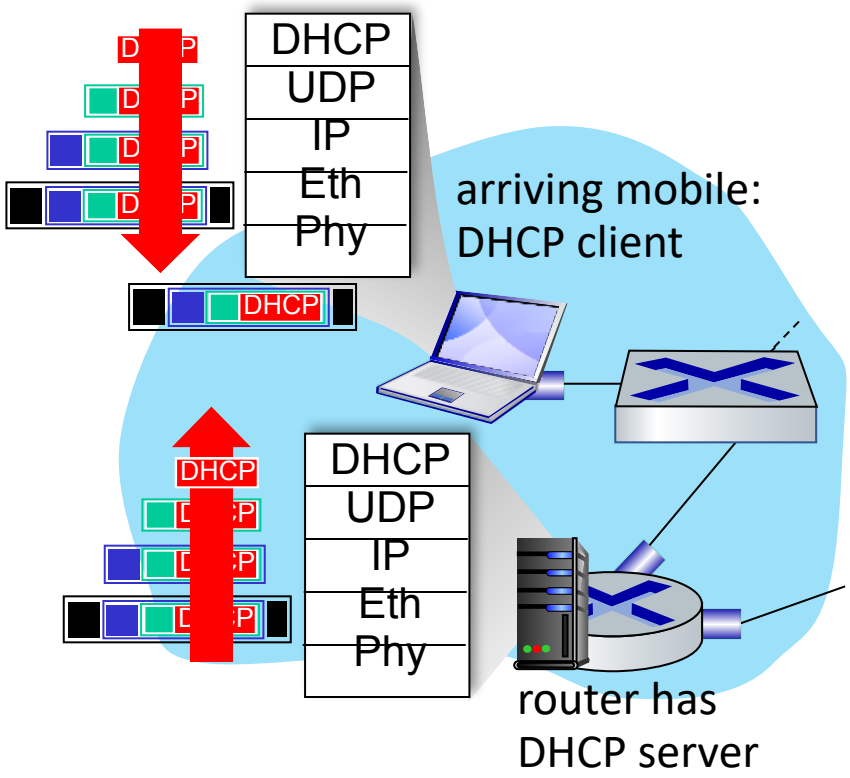


Scenario:

- Arriving mobile client attaches to network …
- Requests web page: www.google.com

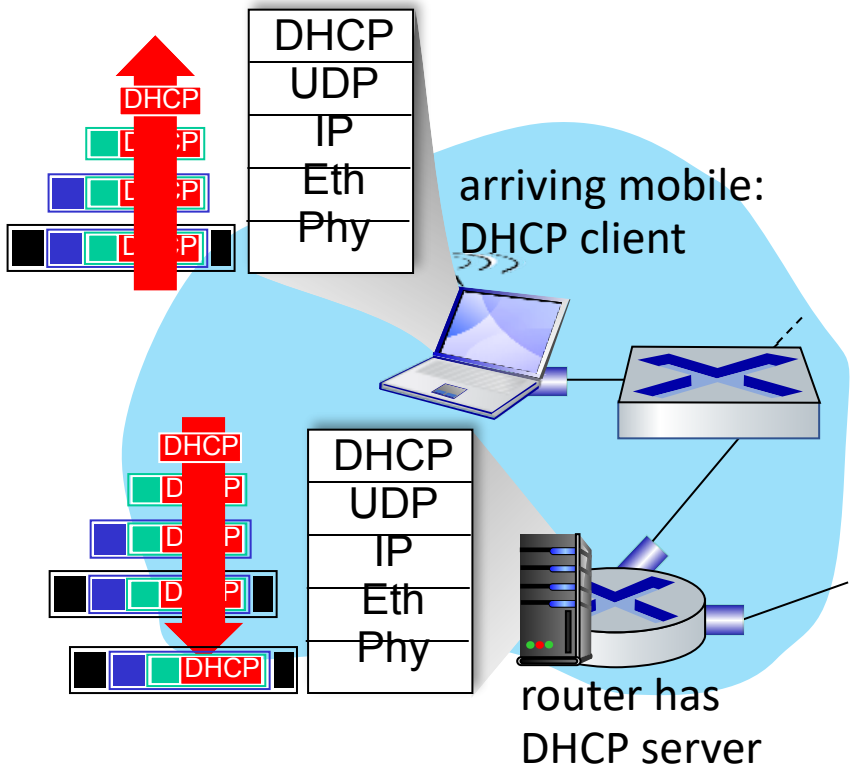
Sounds simple!

A day in the life of a web request



- Connecting laptop needs to get its own IP address, addr of first-hop router, addr of DNS server: use DHCP
- DHCP request encapsulated in UDP, encapsulated in IP, encapsulated in 802.3 Ethernet
- Ethernet frame broadcast (dest: FFFFFFFFFFFF) on LAN, received at router running DHCP server
- Ethernet demuxed to IP demuxed, UDP demuxed to DHCP

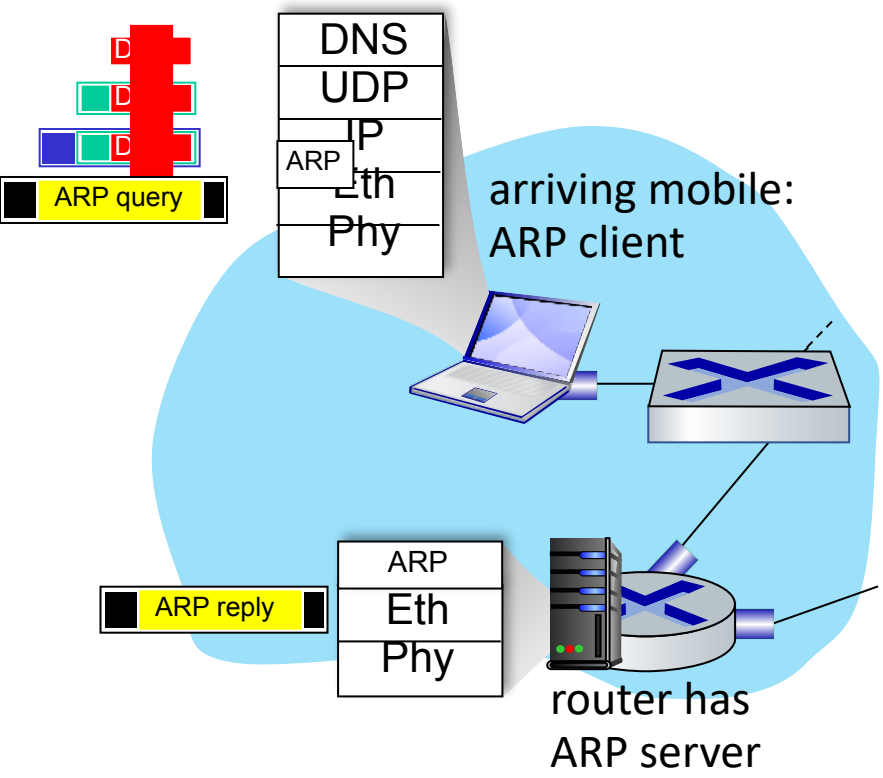
A day in the life : Connecting to the Internet



- DHCP server formulates DHCP ACK containing client's IP address, IP address of first-hop router for client, name & IP address of DNS server
- Encapsulation at DHCP server, frame forwarded (switch learning) through LAN, demultiplexing at client
- DHCP client receives DHCP ACK reply

Client now has IP address, knows name & addr of DNS server, IP address of its first-hop router

A day in the life…. ARP (Before DNS, Before HTTP)

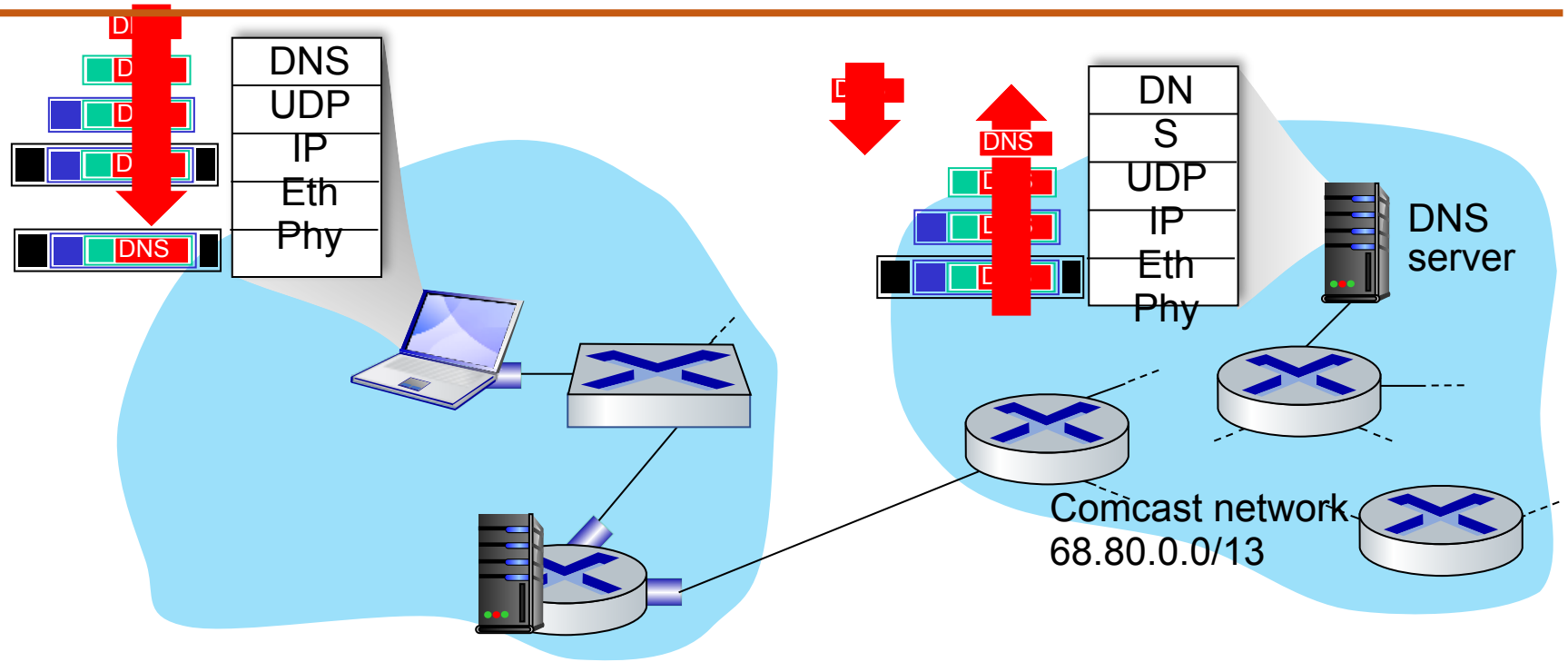


- Before sending HTTP request, need IP address of www.google.com: DNS
- DNS query created, encapsulated in UDP, encapsulated in IP, encapsulated in Eth. To send frame to router, need MAC address of router interface: ARP
- ARP query broadcast, received by router, which replies with ARP reply giving MAC address of router interface

- Client now knows MAC address of first hop router, so can now send frame containing DNS query

COMPUTER NETWORKS

A day in the life…. Using DNS

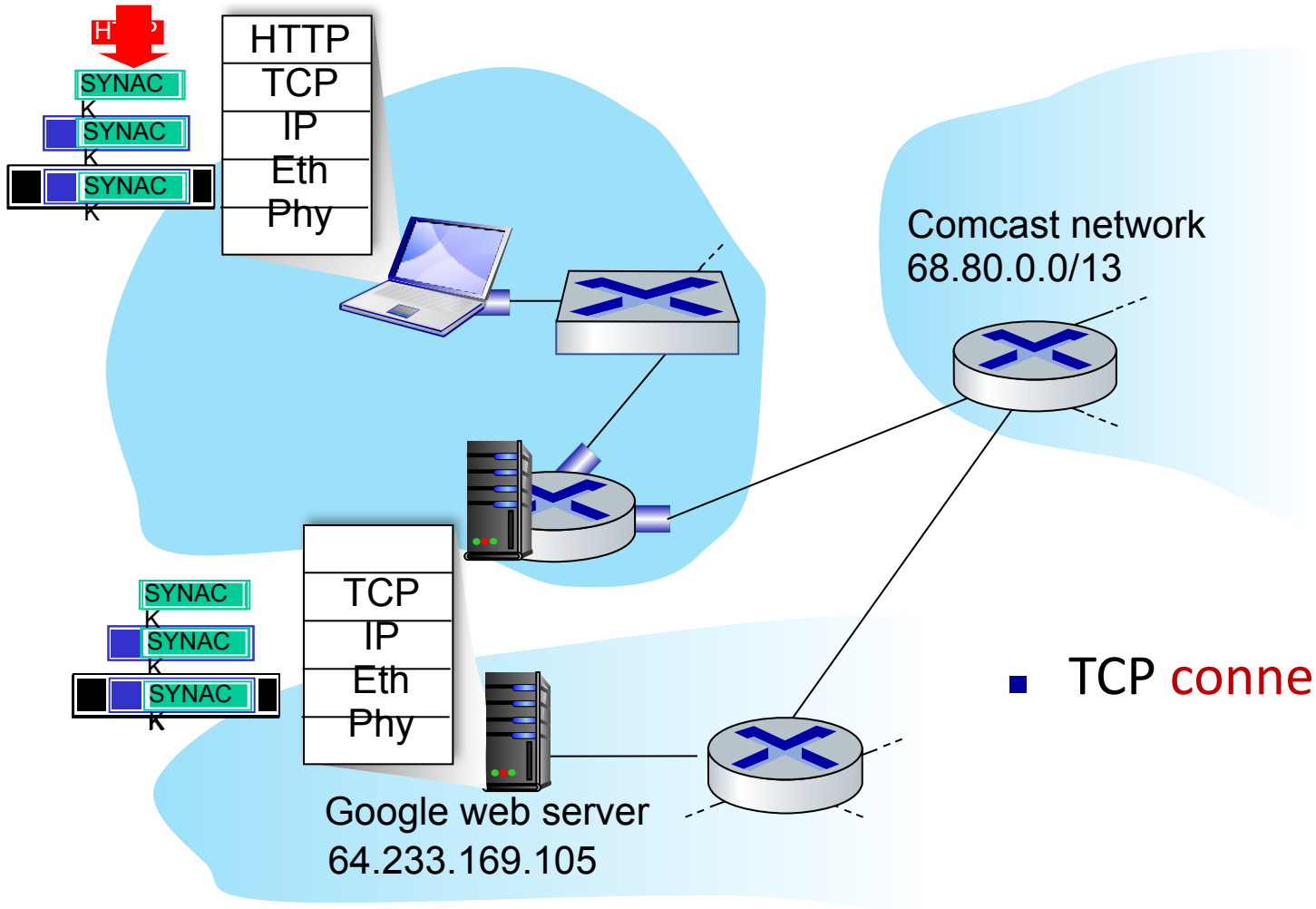


- Demuxed to DNS
- DNS replies to client with IP address of www.google.com

- IP datagram containing DNS query forwarded via LAN switch from client to 1st hop router
- IP datagram forwarded from campus network into Comcast network, routed (tables created by RIP, OSPF, IS-IS and/or BGP routing protocols) to DNS server

COMPUTER NETWORKS

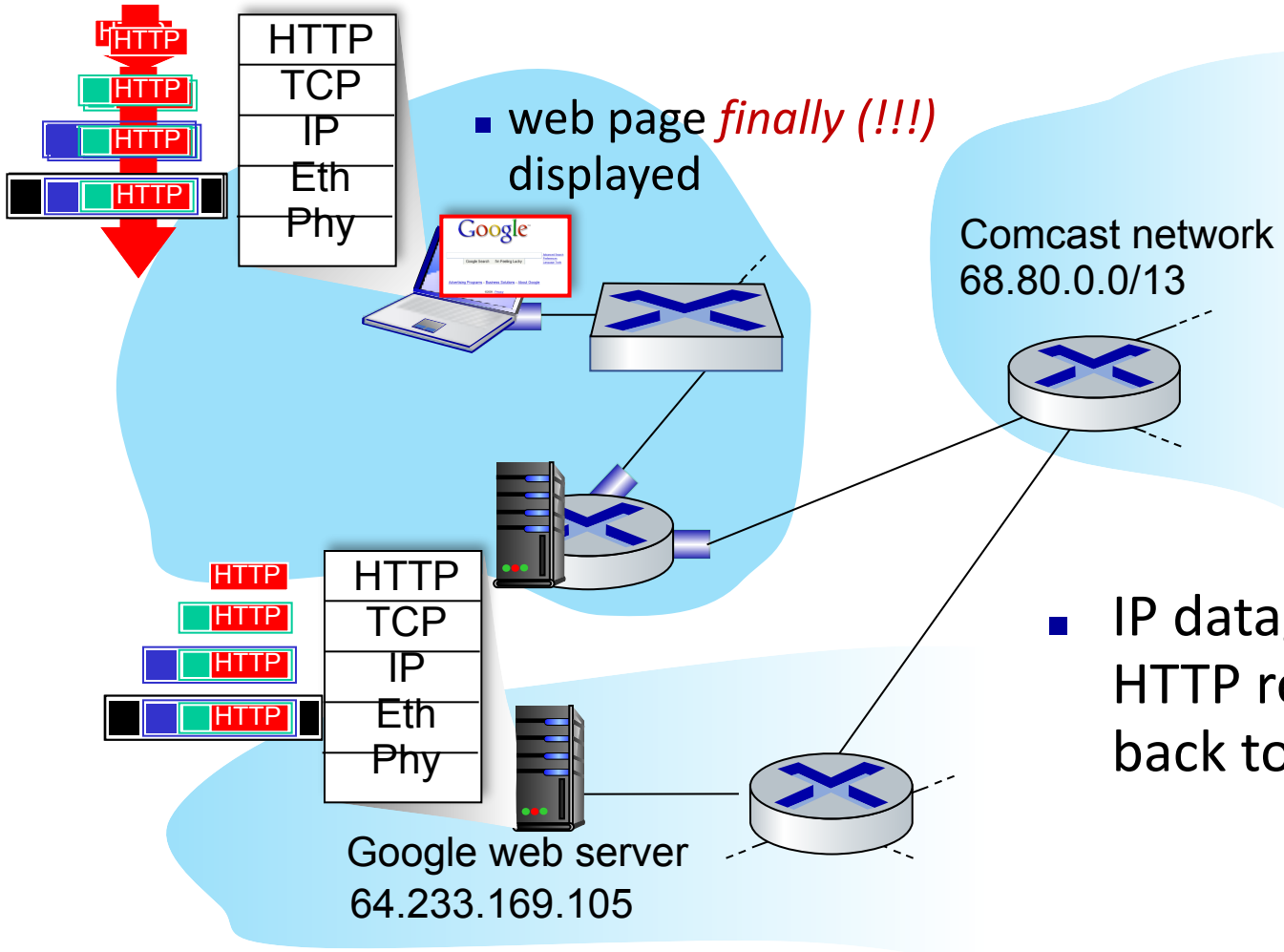
A day in the life…. TCP Connection carrying HTTP



- To send HTTP request, client first opens TCP socket to web server
- TCP SYN segment (step 1 in TCP 3-way handshake) inter-domain routed to web server
- Web server responds with TCP SYNACK (step 2 in TCP 3-way handshake)
- TCP connection established!

COMPUTER NETWORKS

A day in the life…. HTTP Request / Reply

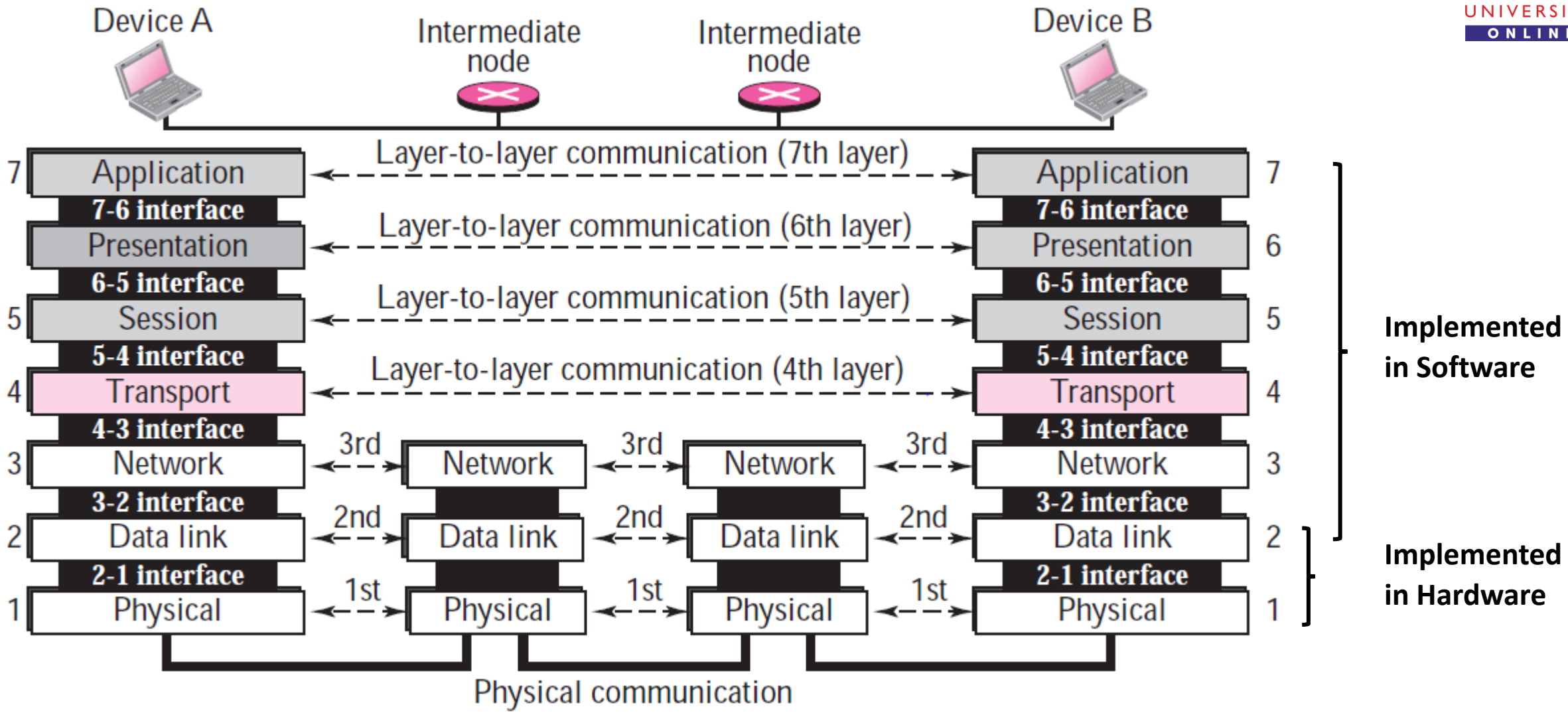


- HTTP request sent into TCP socket
- IP datagram containing HTTP request routed to www.google.com
- Web server responds with HTTP reply (containing web page)
- IP datagram containing HTTP reply routed back to client

Unit – 4 Link Layer and LAN Roadmap

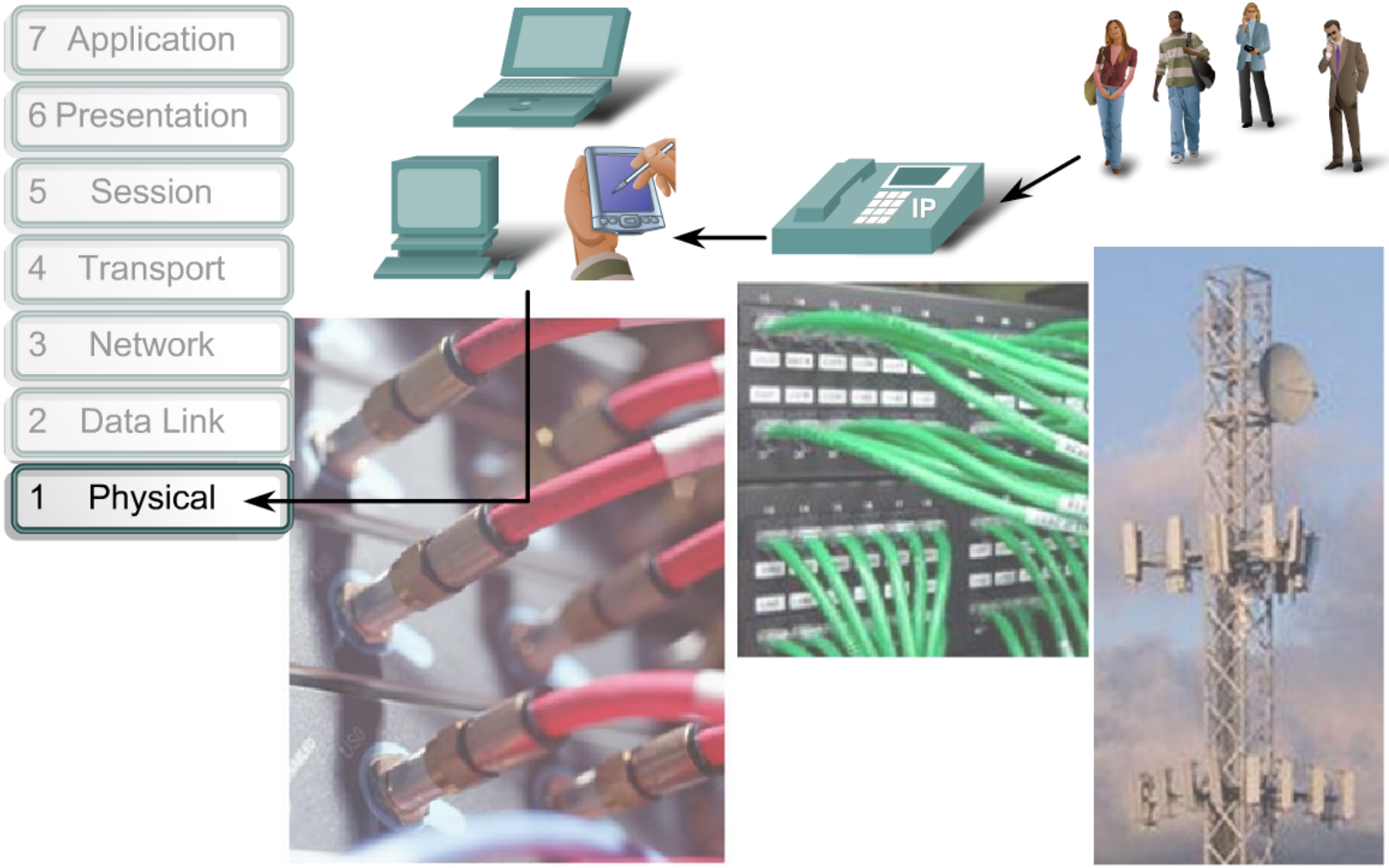
- Introduction
- Error detection, correction
- Multiple access protocols
- Switched LANs
  - addressing, ARP
  - Ethernet
  - switches
- A day in the life of a web request
- Physical layer
- Wireless LANs: IEEE 802.11

Physical Layer – OSI Reference Model



COMPUTER NETWORKS

Purpose of Physical Layer



The Physical layer interconnects our data networks.

COMPUTER NETWORKS

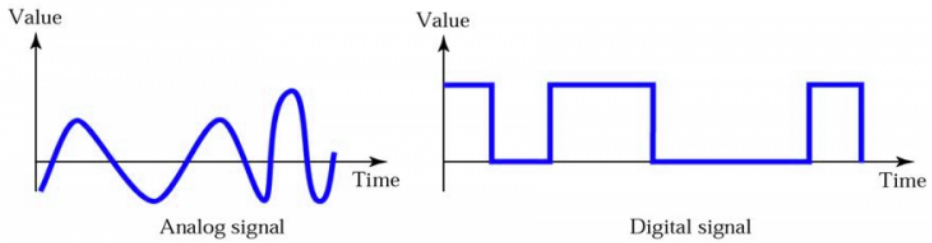
Physical Layer Services

The role of Physical Layer is to:

- encode the binary digits that represent data link layer frames into signals
- to transmit and receive these signals cross the physical media
  - copper wires, optical fiber, and wireless that connect network devices.
- Physical medium - conduct a signal in the form of voltage, light, or radio waves from one device to another.

Analog signal – continuous-valued (amplitudes, frequencies)

Digital signal – discrete-valued (0s & 1s)



Media	Signal Type
Copper cable	Patterns of electrical pulses
Fiber-optic cable	Patterns of light pulses
Wireless	Patterns of radio transmissions

Physical Layer Services

The delivery of frames across the local media requires the following physical layer elements:

- The physical media and associated connectors
- A representation of bits on the media
- Encoding of data and control information
- Transmitter and receiver circuitry on the network devices

Components to understand Physical Layer functions includes:

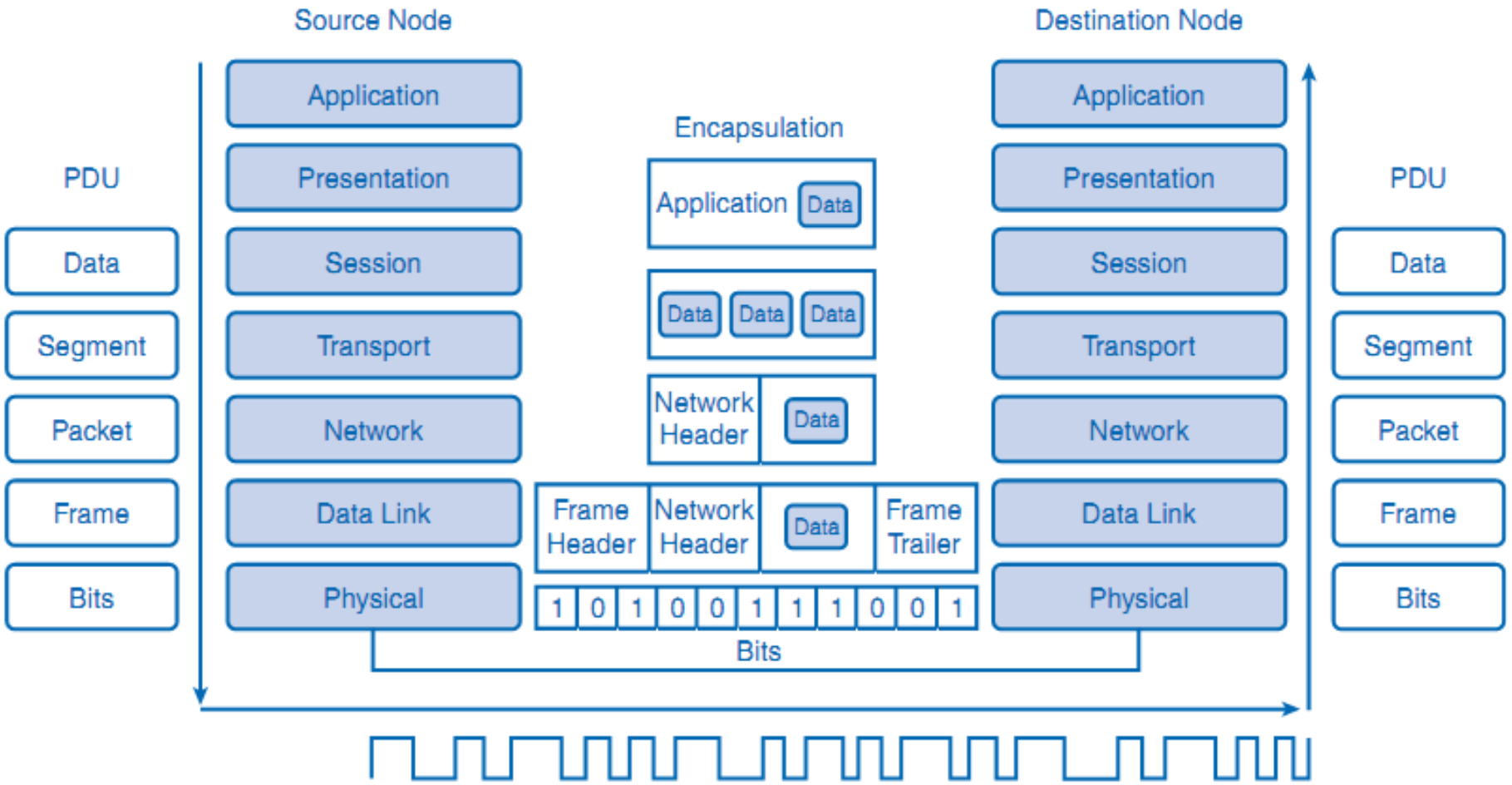
- Physical components
- Encoding
- Signaling

Physical Layer Devices

- Hub
- Repeater
- Modem
- Cables
- Network Adapters

COMPUTER NETWORKS

Physical Layer Services



Physical Layer Encoding

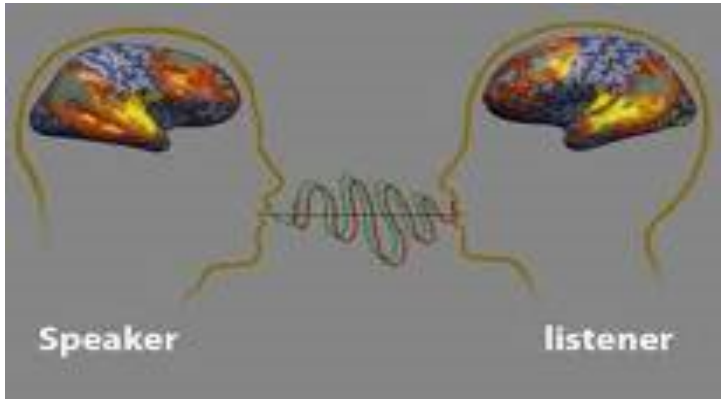
Physical Layer – Human Analogy

A person wants to communicate an idea:

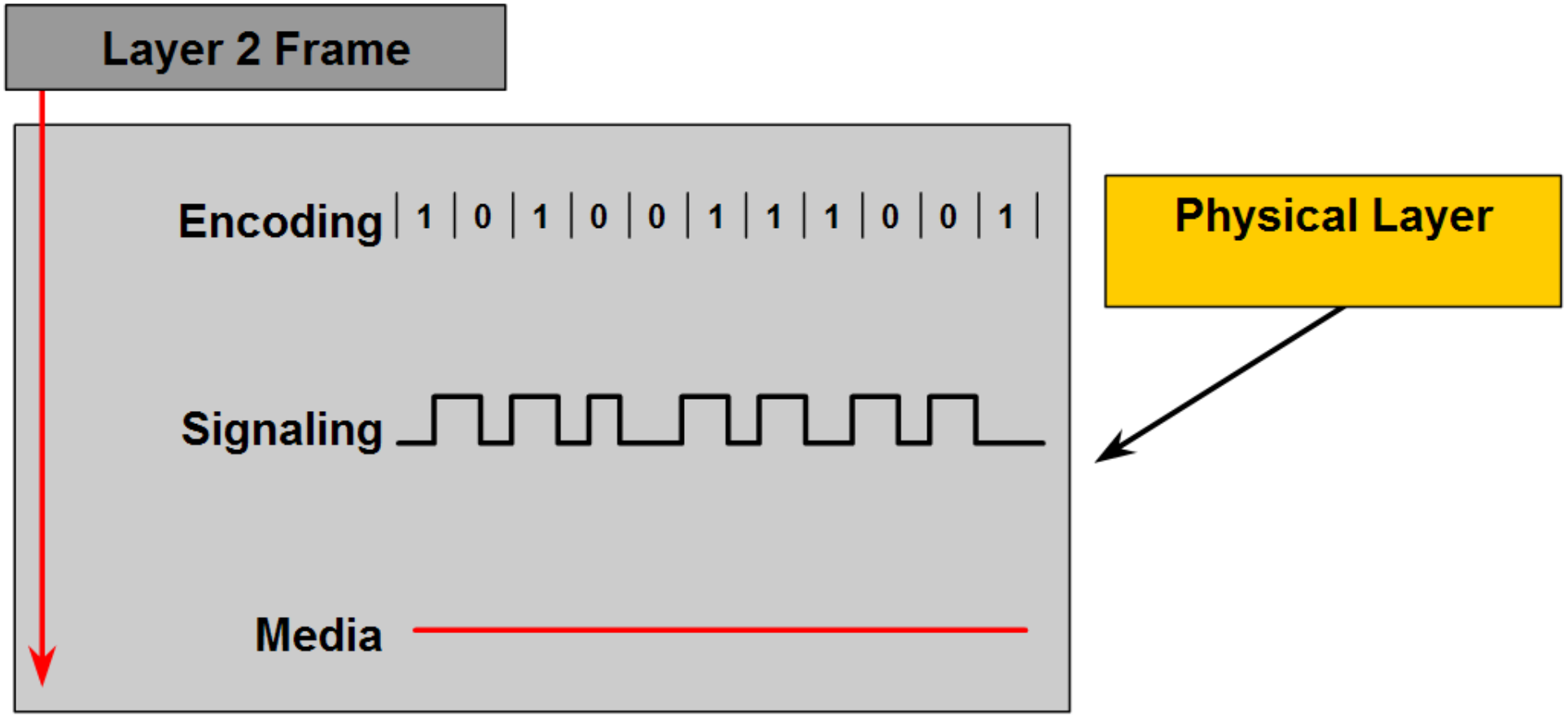
- processes an abstract thought into words (signaling) ->
- encoded into speech sounds (encoding) ->
- sent out through the medium of air (physical).

At the other end:

- the receiver interprets the signal of sound,
- recognizes patterns in the sound that denote words &
- then processes the meaning of the words into the original idea

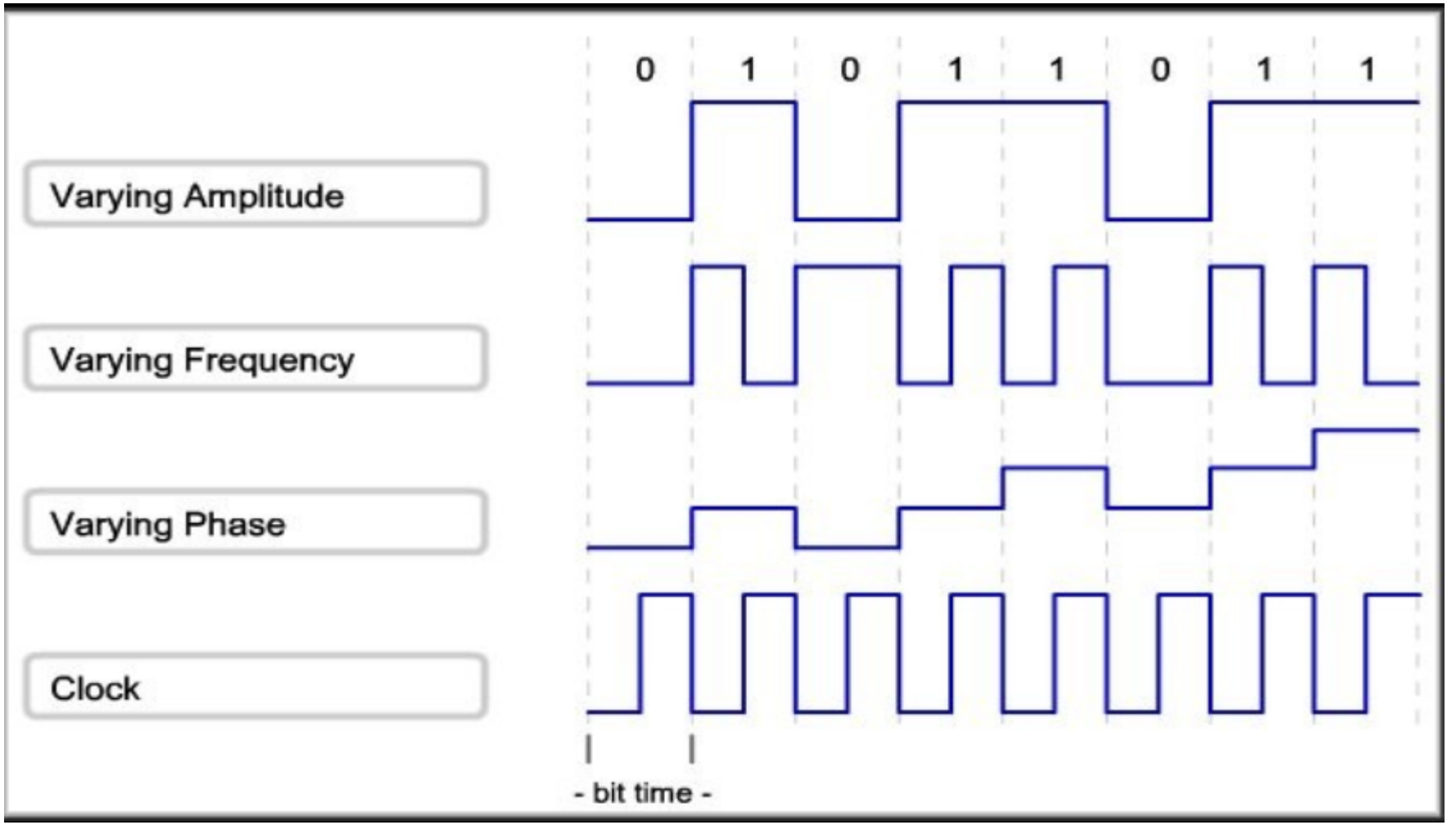


Physical Layer Principles



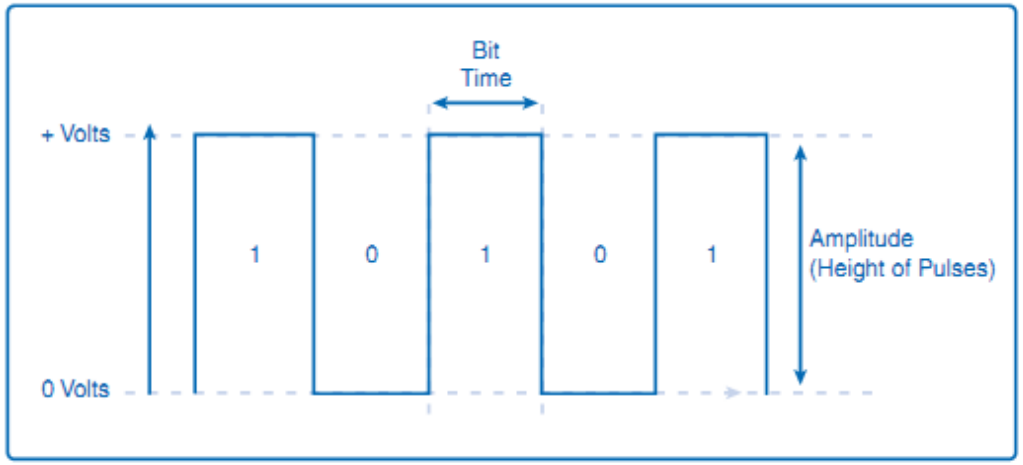
COMPUTER NETWORKS

Signalling Bits for the Media

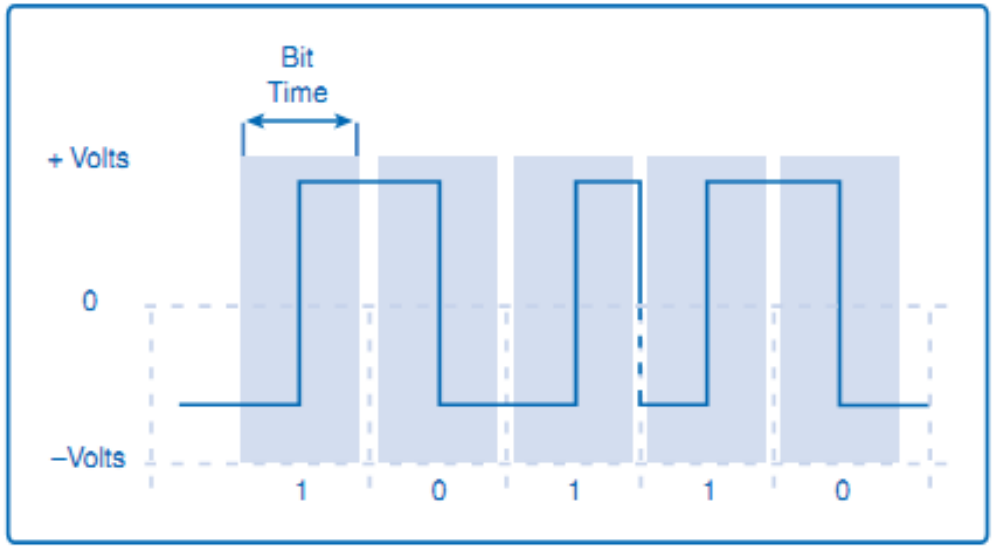


COMPUTER NETWORKS

Signalling Methods



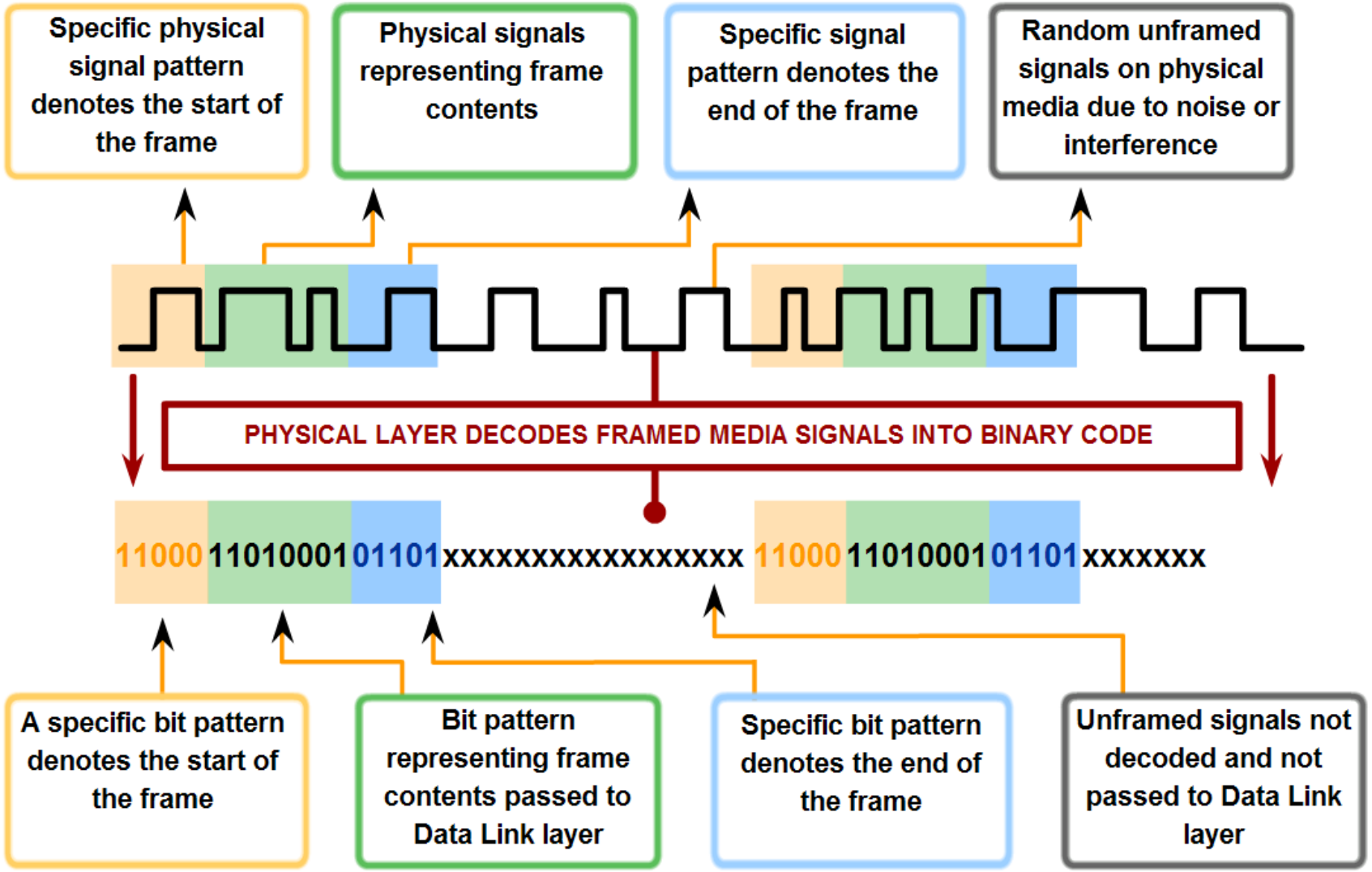
NRZ (Non Return to Zero) Encoding – A low voltage being 0 and a higher voltage representing a 1.



Manchester Encoding

- A voltage change from low to high within the bit time represents a 1.
- A voltage drop within the bit time from a high to a low voltage represents a 0.

Physical Layer – Signalling and Encoding



Physical Layer Standards

Some of the key organizations:

- The International Organization for Standardization (ISO)
- The Institute of Electrical and Electronics Engineers (IEEE)
- The American National Standards Institute (ANSI)
- The International Telecommunication Union (ITU)
- The Electronics Industry Alliance/Telecommunications Industry Association (EIA/TIA)
- National telecommunications authorities such as the Federal Communications Commission (FCC) in the United States

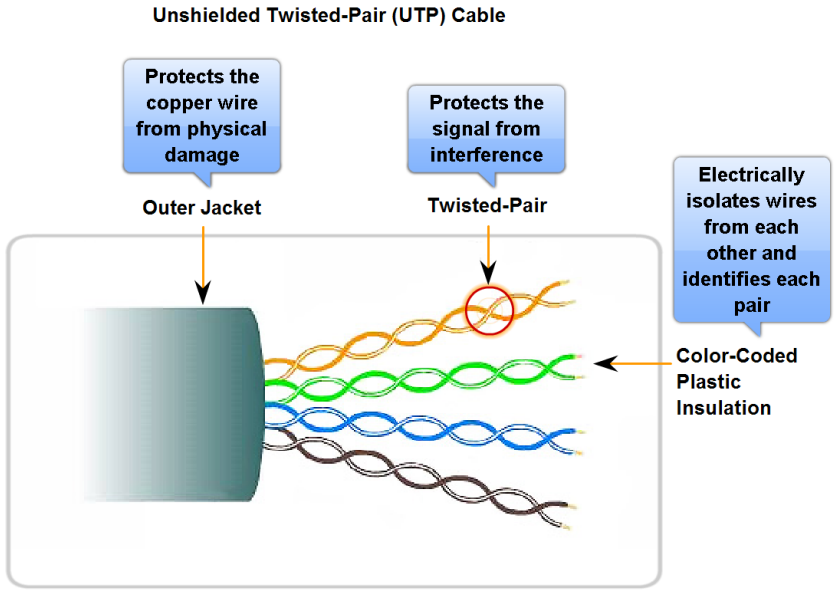
TCP/IP Standards set by:
IETF

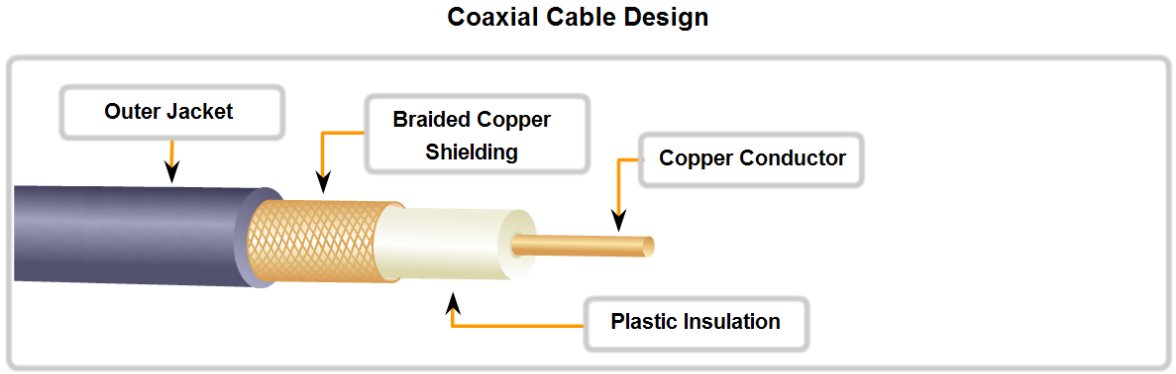
Standards set by:

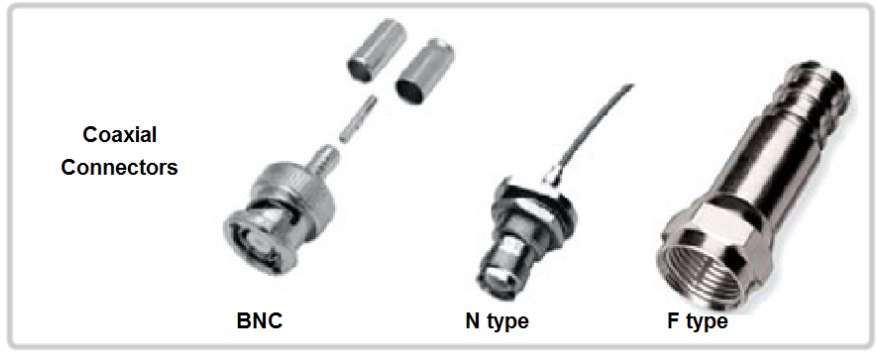
ISO	IEEE
ANSI	ITU
EIA/TIA	FCC

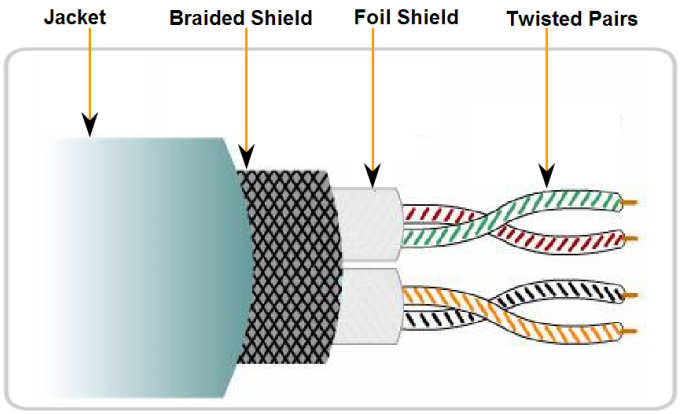
COMPUTER NETWORKS

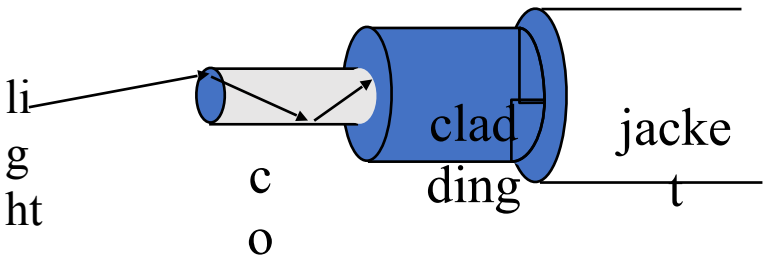
Transmission Media

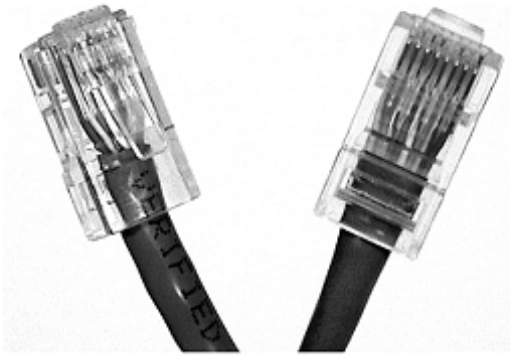












COMPUTER NETWORKS

Types of Physical Transmission Media

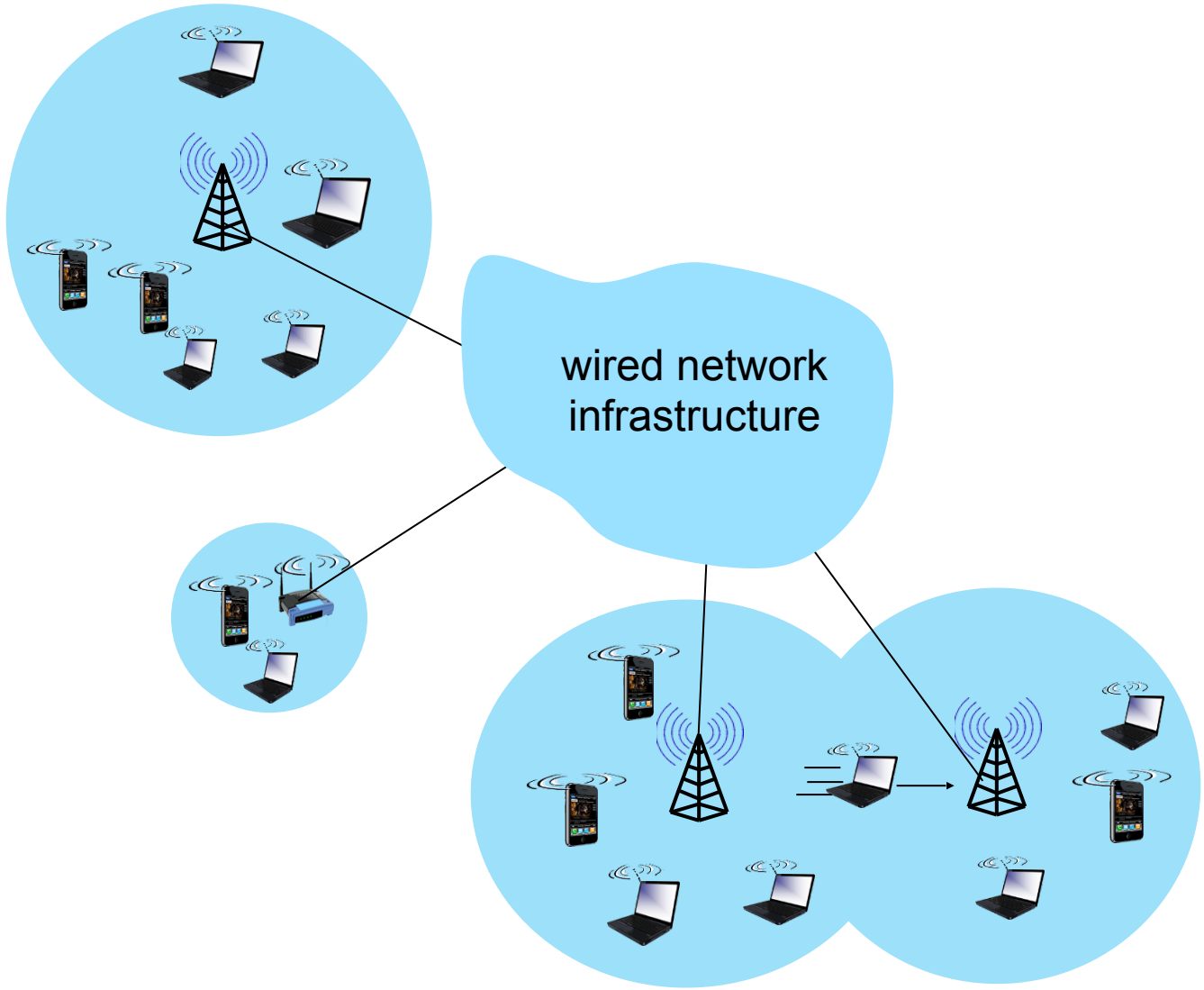
Specification	Media	Maximum Segment Length	Connector
10BASE-T	CAT 3,4 or 5 UTP (4 pair)	100m	RJ-45
100BASE-TX	CAT 5 UTP (2 pair)	100m	RJ-45
100BASE-FX	62.5/125 multimode fiber	2km	
1000BASE-CX	STP	25m	RJ-45
1000BASE-T	CAT 5 UTP (4 pair)	100m	RJ-45
1000BASE-SX	62.5/50 multimode fiber	62.5 – 275m 50 – 550m	
1000BASE-LX	62.5/50 multimode 9-micron single-mode fiber	62.5/50 – 550m 9 –10 km	
1000BASE-ZX	9-micron single-mode fiber	70km	
10GBASE-ZR	9-micron single-mode fiber	80km	

Unit – 4 Link Layer and LAN Roadmap

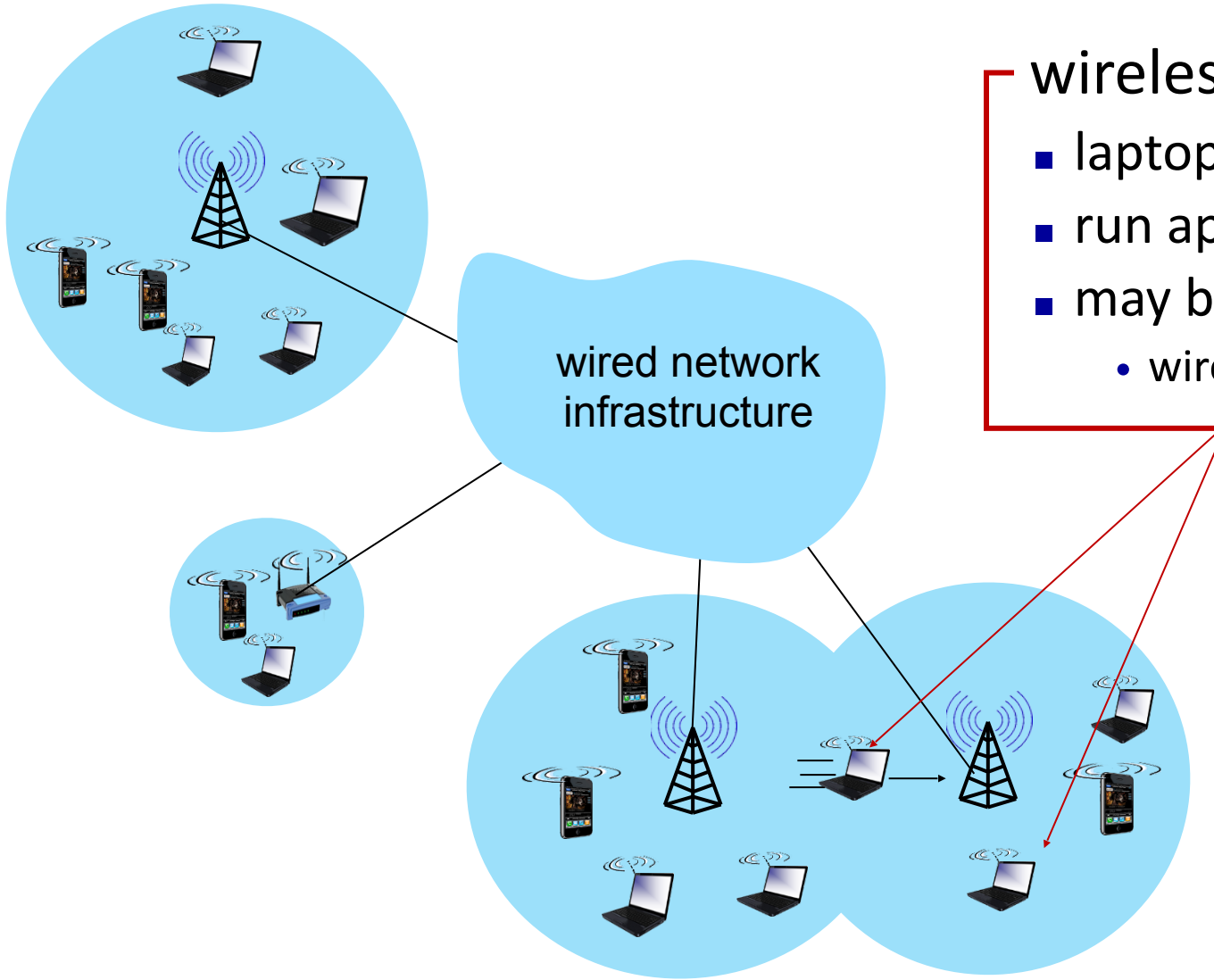
- Introduction
- Error detection, correction
- Multiple access protocols
- Switched LANs
  - addressing, ARP
  - Ethernet
  - switches
- A day in the life of a web request
- Physical layer
- Wireless LANs: IEEE 802.11

COMPUTER NETWORKS

Elements of a Wireless Network



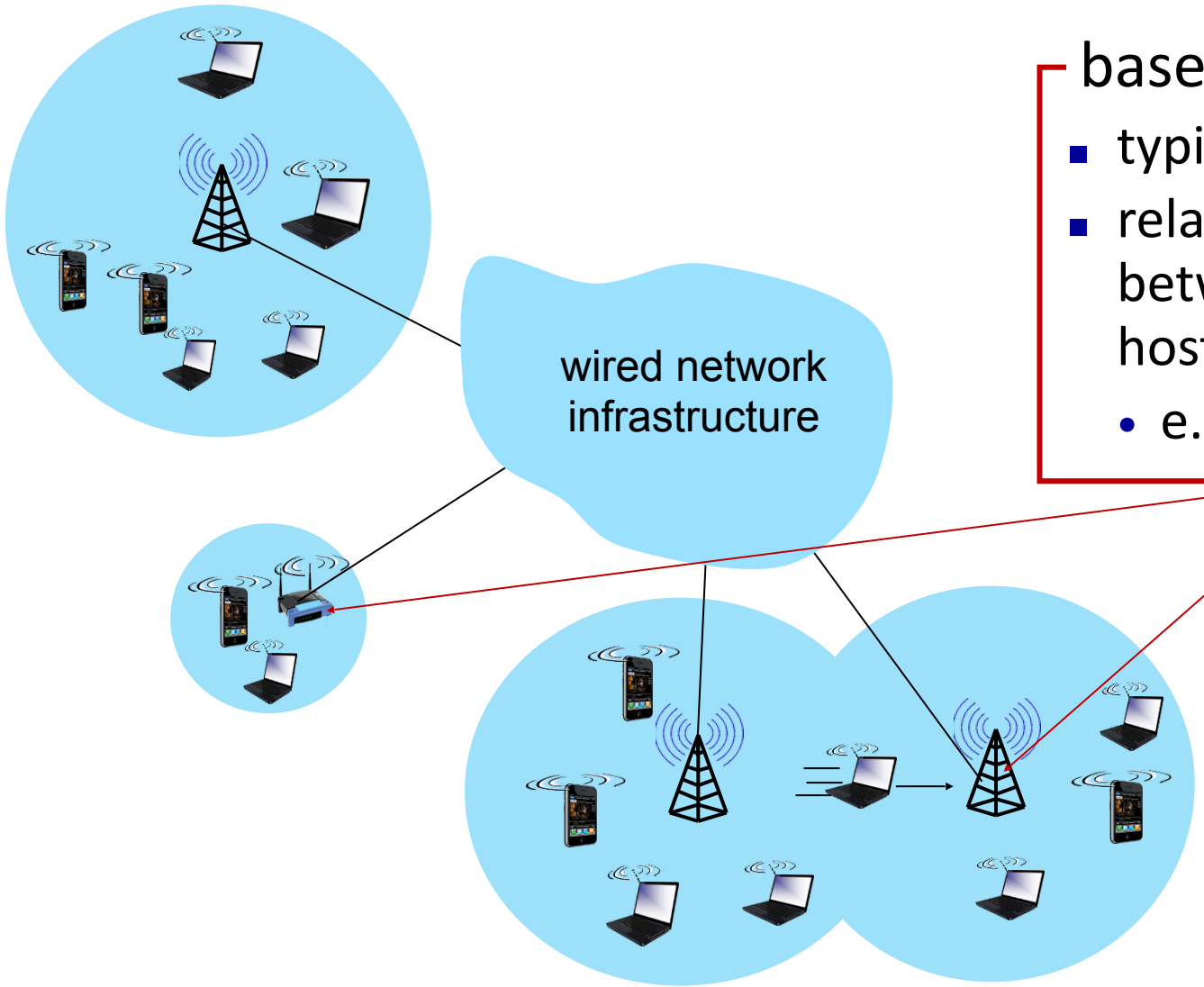
Elements of a Wireless Network



wireless hosts

- laptop, smartphone, IoT
- run applications
- may be stationary (non-mobile) or mobile
  - wireless does *not* always mean mobility!

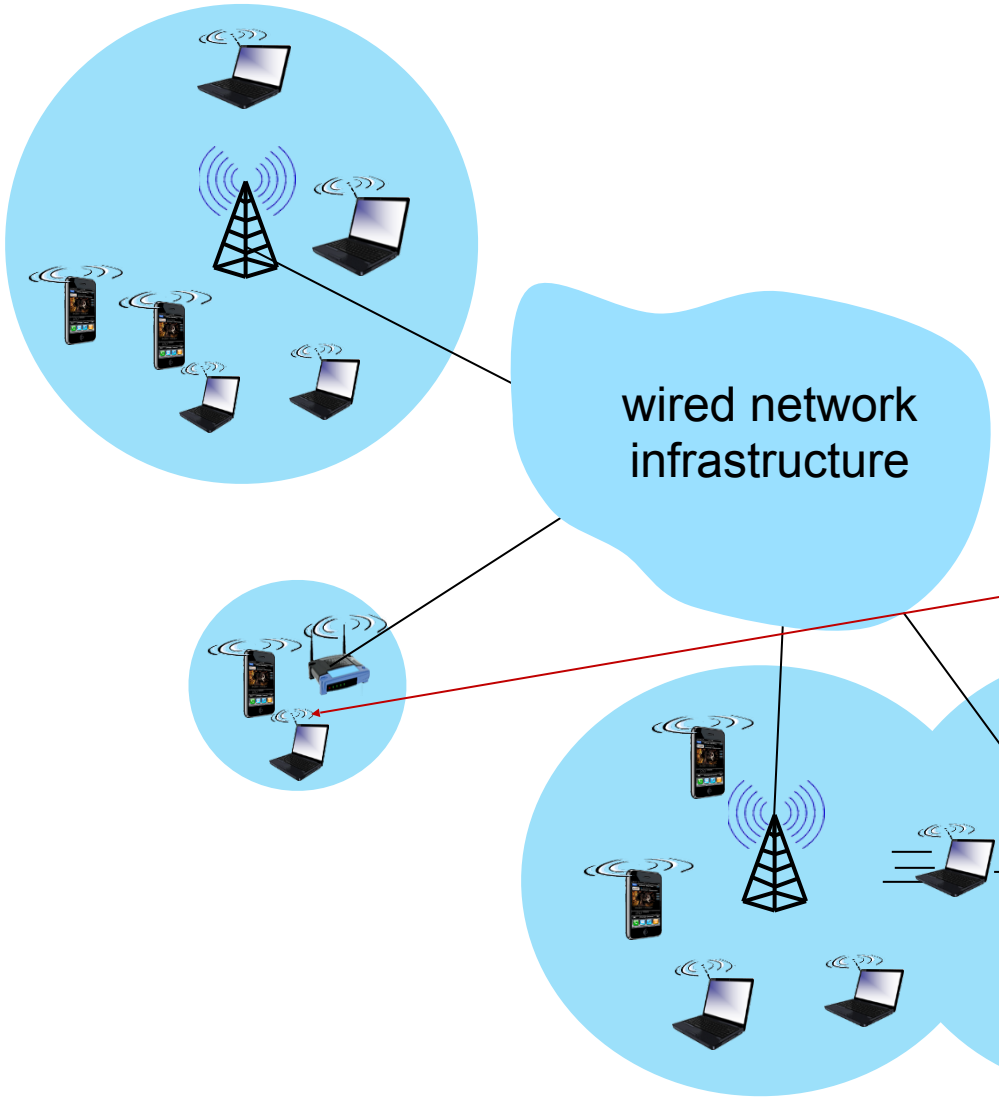
Elements of a Wireless Network



base station

- typically connected to wired network
- relay - responsible for sending packets between wired network and wireless host(s) in its “area”
  - e.g., **cell towers**, **802.11 access points**

Elements of a Wireless Network

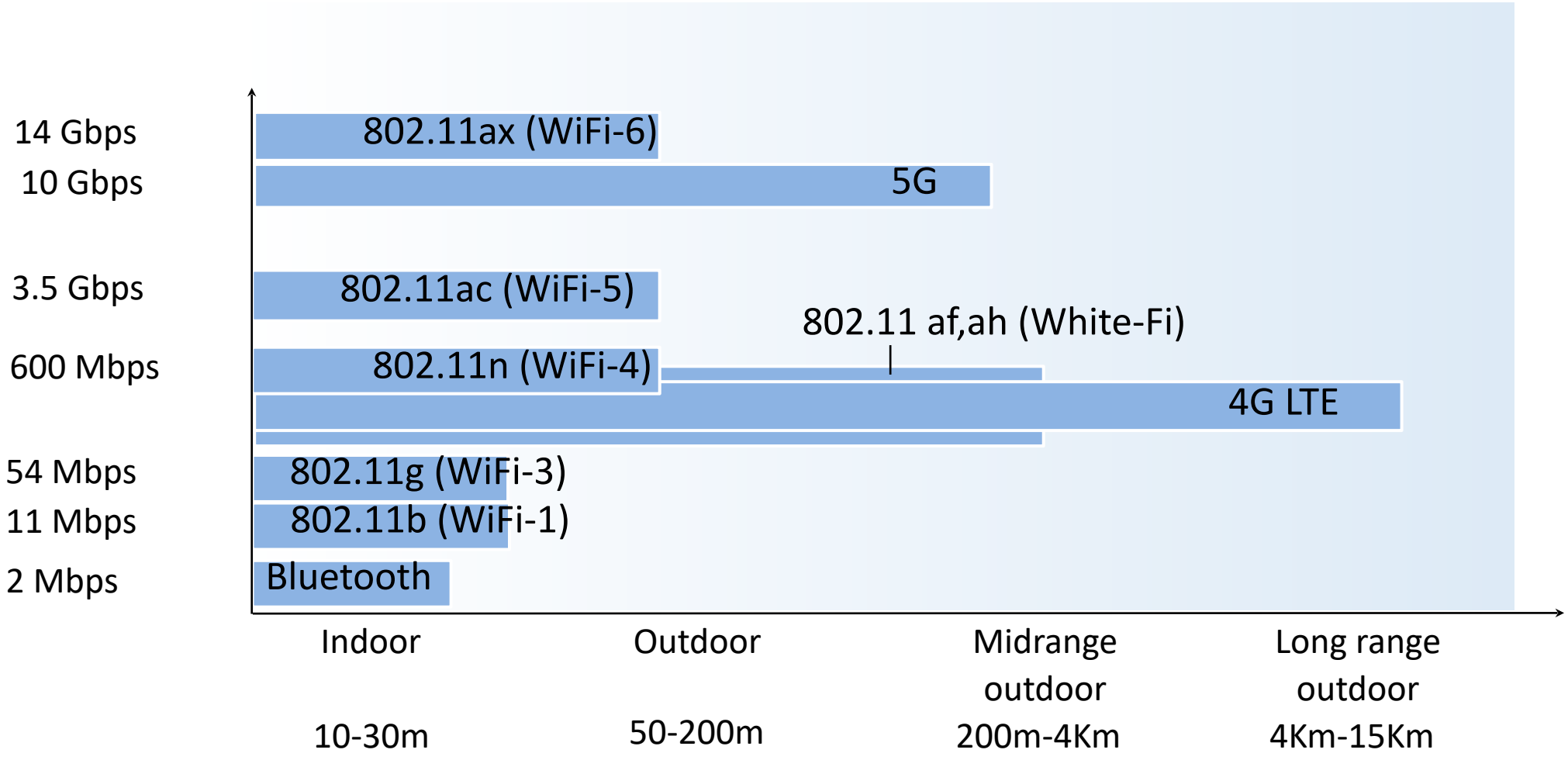


wireless link

- typically used to connect mobile(s) to base station, also used as backbone link
- multiple access protocol coordinates link access
- various transmission rates and distances, frequency bands

COMPUTER NETWORKS

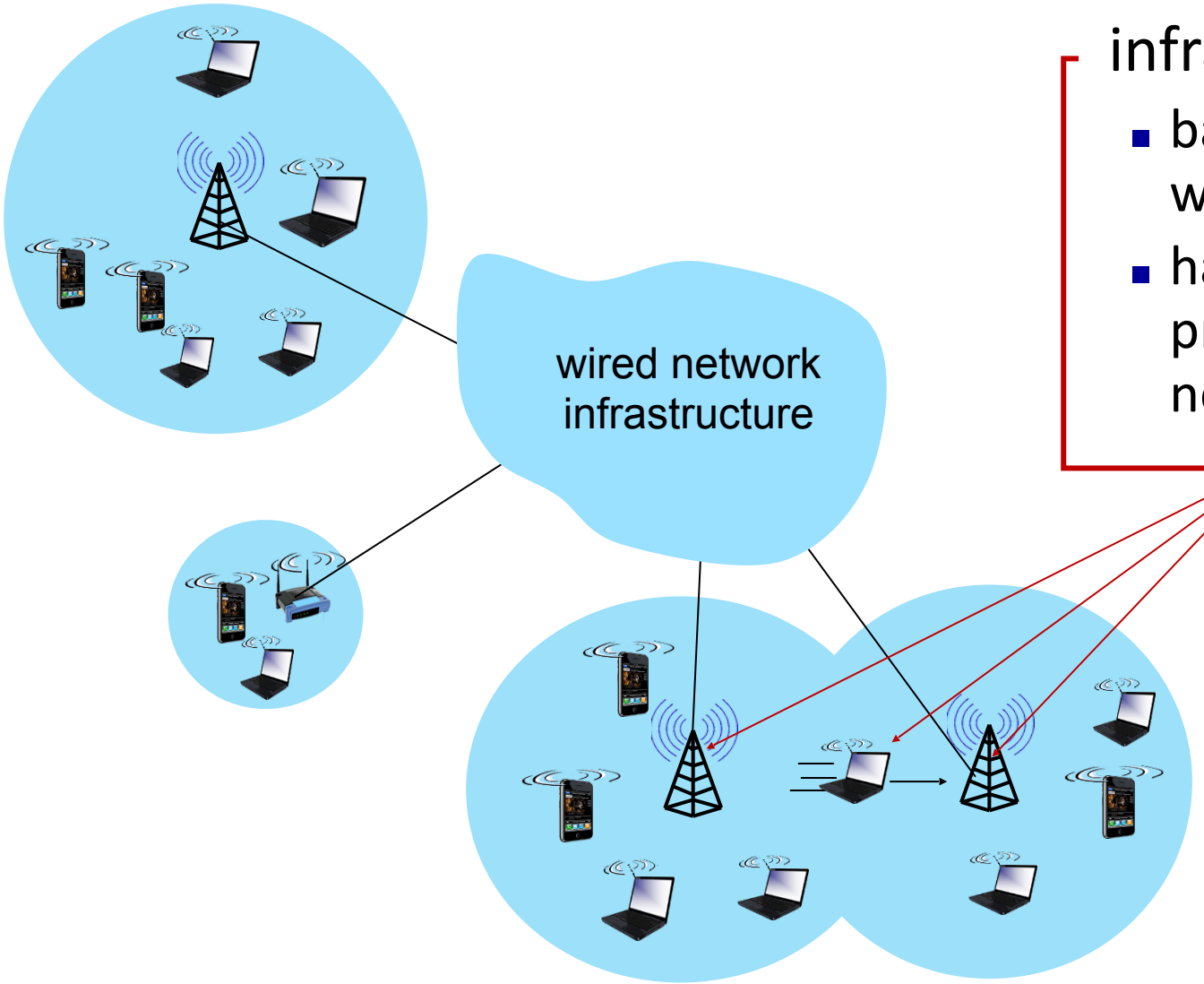
Characteristics of Selected Wireless Links



2 Key Areas:

Coverage area & Link rate

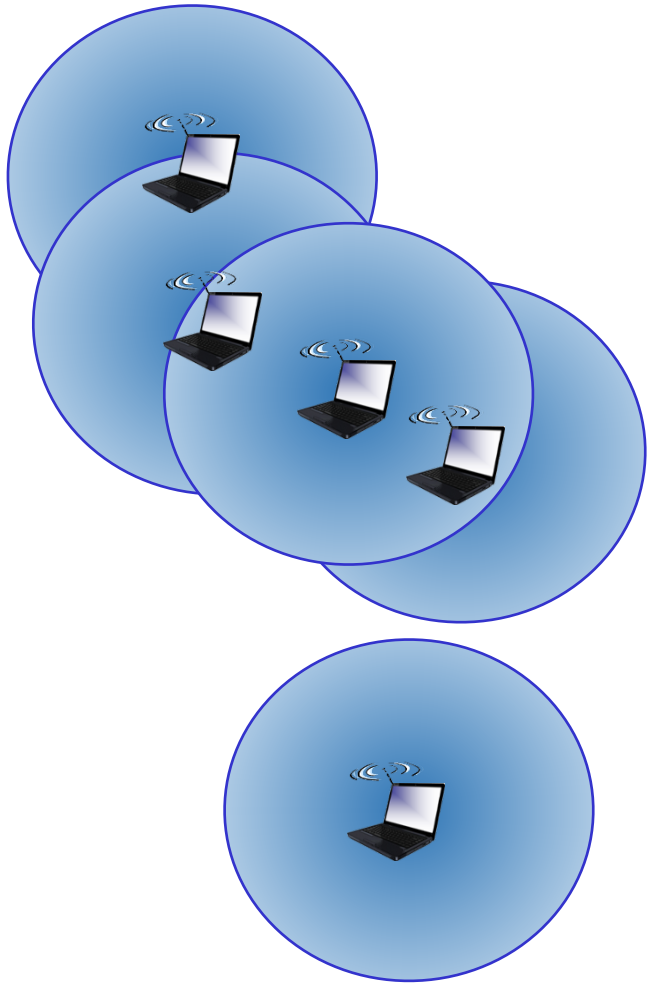
Elements of a Wireless Network



infrastructure mode

- base station connects mobiles into wired network
- handoff: mobile changes base station providing connection into wired network

Elements of a Wireless Network



ad hoc mode

- no base stations
- nodes can only transmit to other nodes within link coverage
- nodes organize themselves into a network: routing, address assignment, DNS-like name translation, and more.

Wireless network taxonomy

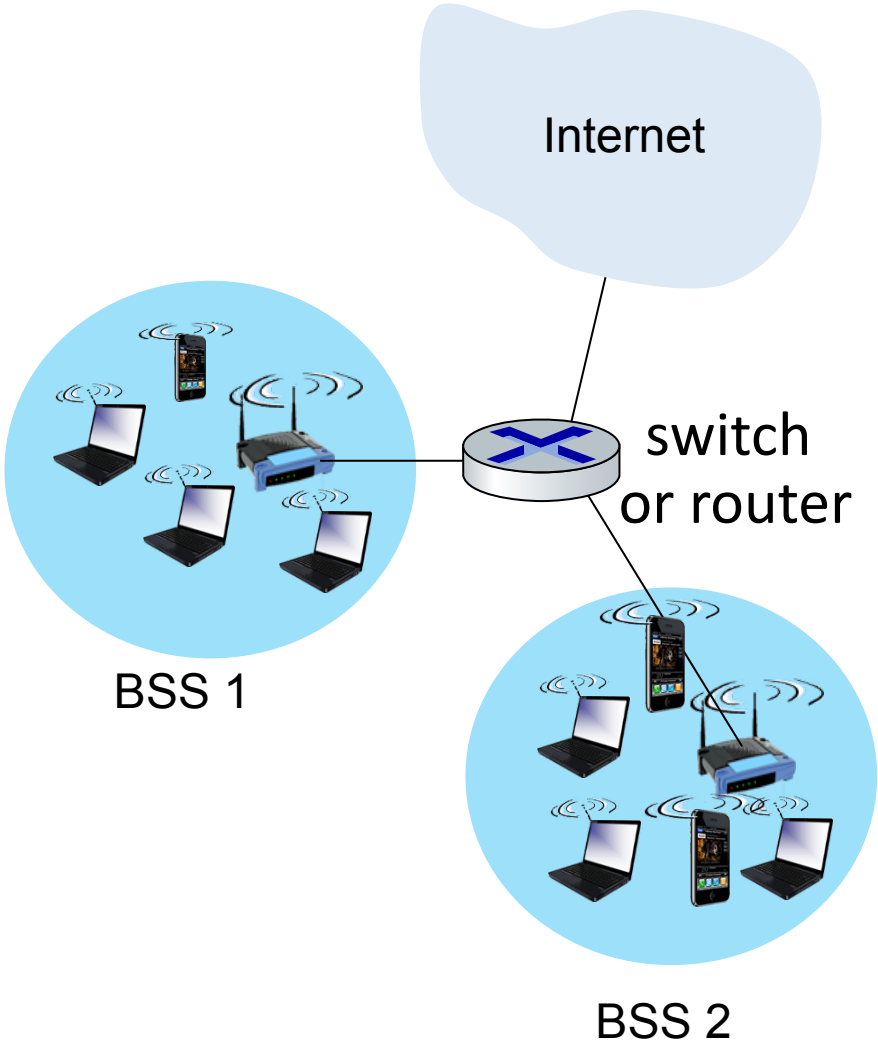
	single hop	multiple hops
infrastructure (e.g., APs)	host connects to base station (WiFi, cellular) which connects to larger Internet. **Eg: 4G LTE**	host may have to relay through several wireless nodes to connect to larger Internet: ***mesh net***
infrastructure less	no base station, no connection to larger Internet. eg: **Bluetooth, ad hoc nets**	no base station, no connection to larger Internet. May have to relay to reach other. **eg: MANET, VANET**

IEEE 802.11 Wireless LAN (WiFi)

IEEE 802.11 standard	Year	Max data rate	Range	Frequency
802.11b	1999	11 Mbps	30 m	2.4 Ghz
802.11g	2003	54 Mbps	30m	2.4 Ghz
802.11n (WiFi 4)	2009	600	70m	2.4, 5 Ghz
802.11ac (WiFi 5)	2013	3.47Gpbs	70m	5 Ghz
802.11ax (WiFi 6)	2020 (exp.)	14 Gbps	70m	2.4, 5 Ghz
802.11af	2014	35 – 560 Mbps	1 Km	unused TV bands (54-790 MHz)
802.11ah	2017	347Mbps	1 Km	900 Mhz

- all use CSMA/CA for multiple access, and have base-station and ad-hoc network versions

The 802.11 LAN architecture



- wireless host communicates with base station
  - base station = access point (AP)
- Basic Service Set (BSS) (aka “cell”) in infrastructure mode contains:
  - wireless hosts
  - access point (AP): base station
  - ad hoc mode: hosts only

COMPUTER NETWORKS

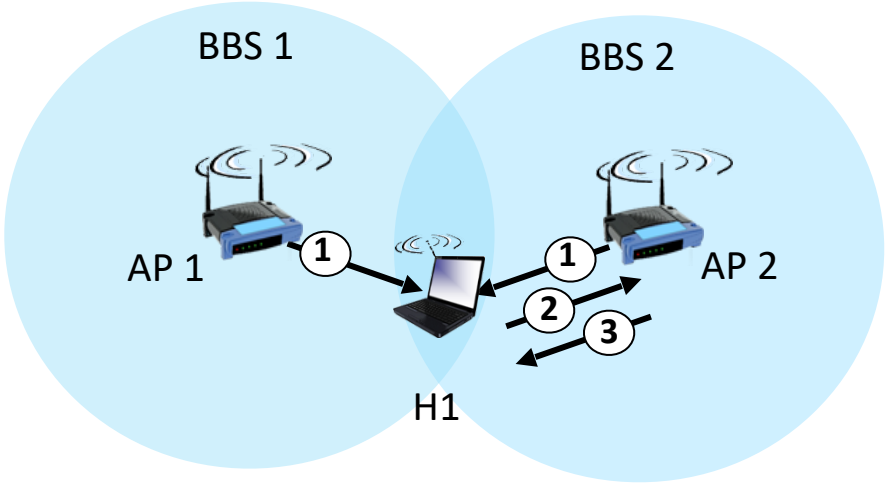
802.11: Channels, Association

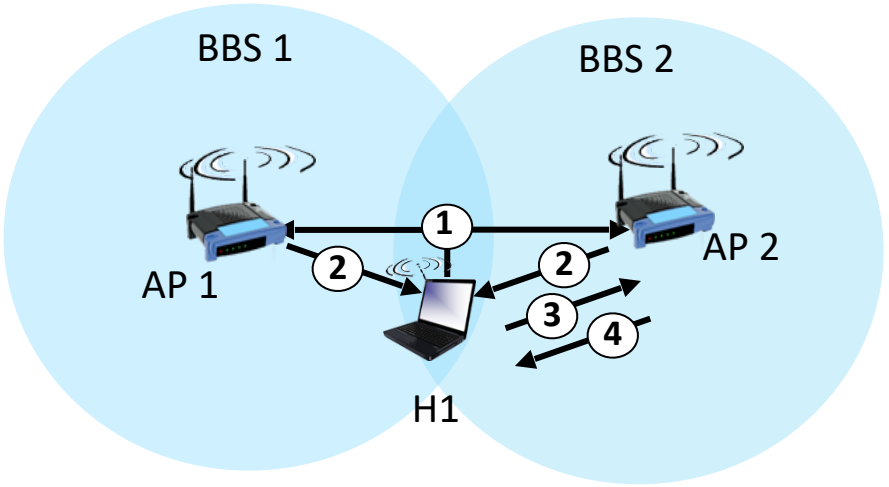
- spectrum divided into channels at different frequencies
  - AP assigns **Service Set ID (SSID)**
  - AP admin chooses frequency for AP (**2.4 GHz to 2.4835 GHz**)
  - interference possible: channel can be same as that chosen by neighboring AP! – **WiFi Jungle**

- arriving host: must associate with an AP
  - scans channels, listening for *beacon frames* containing AP's name (SSID) and MAC address
  - selects AP to associate with
  - then may perform authentication
  - then typically run DHCP to get IP address in AP's subnet

COMPUTER NETWORKS

802.11 active/passive scanning





passive scanning:

(1) beacon frames sent from APs
(2) association Request frame sent: H1 to selected AP
(3) association Response frame sent from selected AP to H1

active scanning:

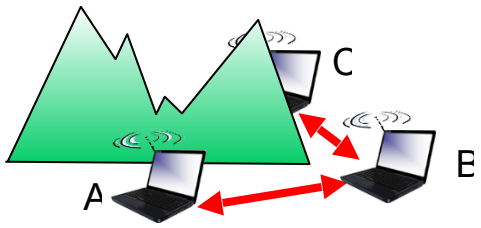
(1) Probe Request frame broadcast from H1
(2) Probe Response frames sent from APs
(3) Association Request frame sent: H1 to selected AP
(4) Association Response frame sent from selected AP to H1

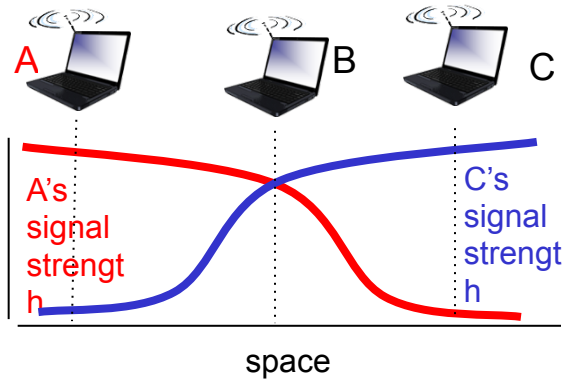
COMPUTER NETWORKS

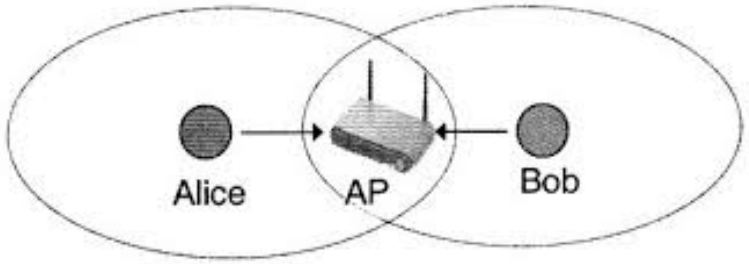
IEEE 802.11: multiple access

- avoid collisions: 2+ nodes transmitting at same time
- 802.11: CSMA - sense before transmitting
  - don't collide with detected ongoing transmission by another node
- 802.11: *no* collision detection!
  - difficult to sense collisions: high transmitting signal, weak received signal due to fading
  - can't sense all collisions in any case: hidden terminal, fading
  - goal: *avoid collisions:* CSMA/CollisionAvoidance

802.11's link-layer acknowledgment scheme!!!







COMPUTER NETWORKS

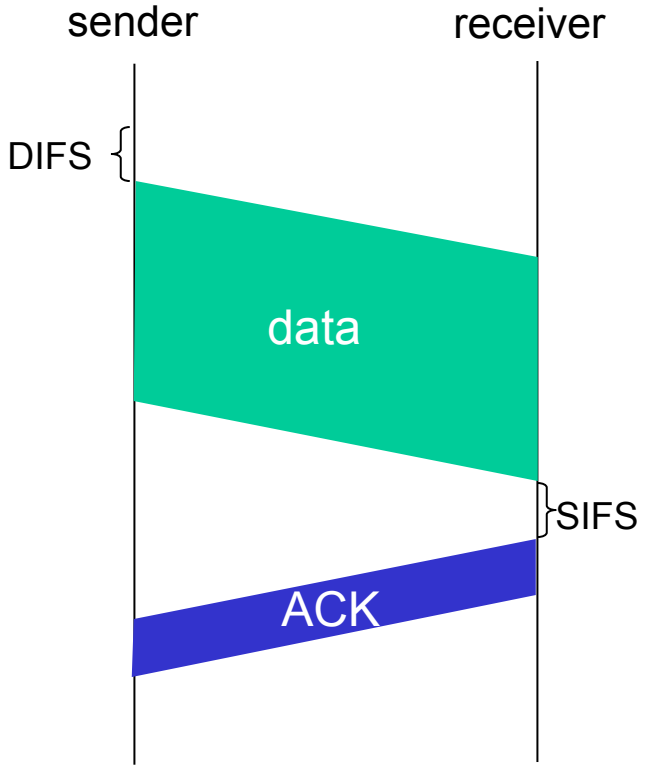
IEEE 802.11 MAC Protocol: CSMA/CA

802.11 sender

Distributed Inter-frame Space (DIFS);

1 if sense channel idle for **DIFS** then
transmit entire frame (no CD)

2 if sense channel busy then
start random backoff time
timer counts down while channel idle
transmit when timer expires
if no ACK, increase random backoff interval, repeat 2



802.11 receiver

if frame received OK
return ACK after **SIFS** (ACK needed due to hidden terminal problem)

Short Inter-frame Spacing (SIFS)

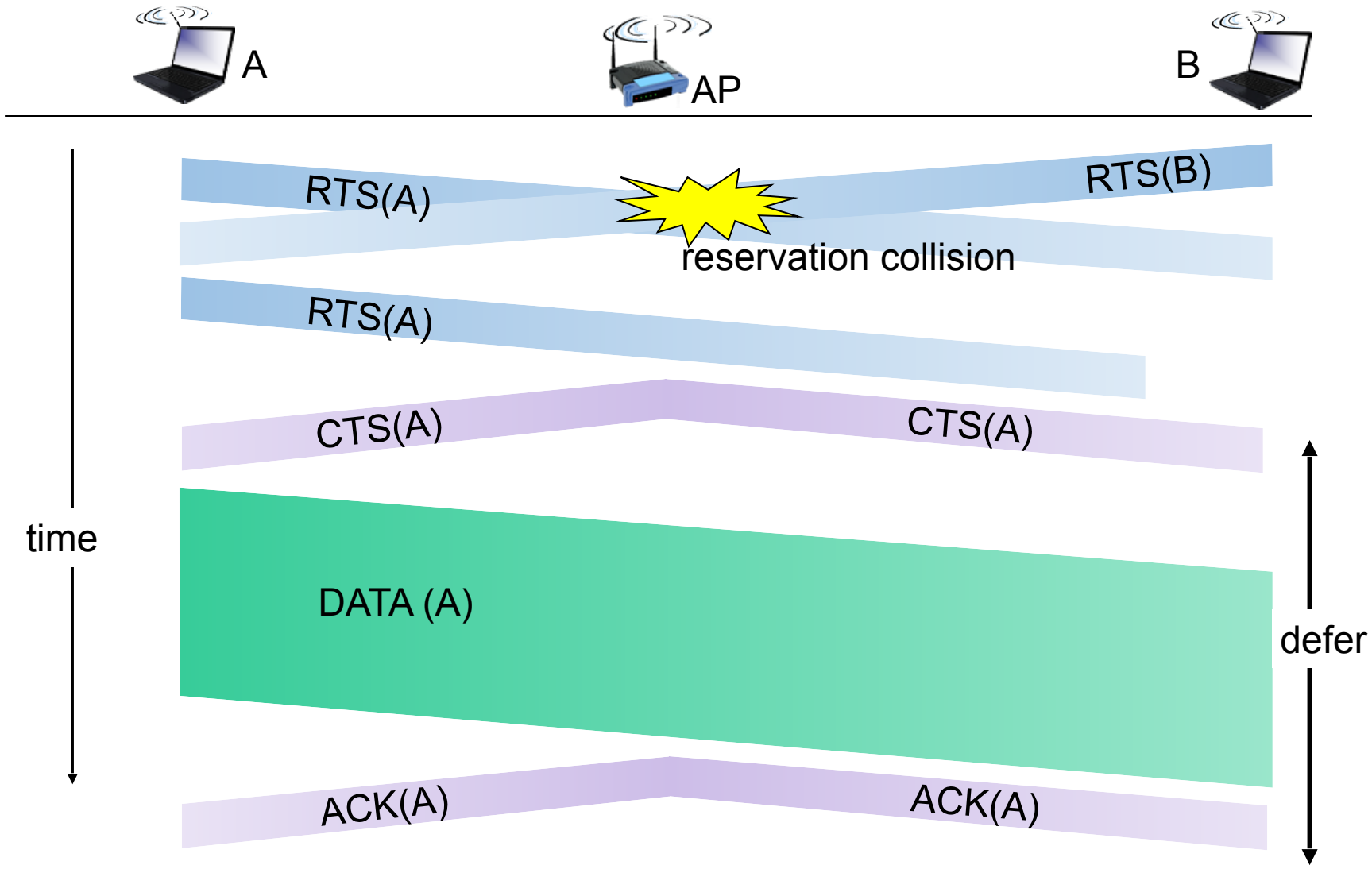
Avoiding Collisions (more)

idea: sender “reserves” channel use for data frames using small reservation packets

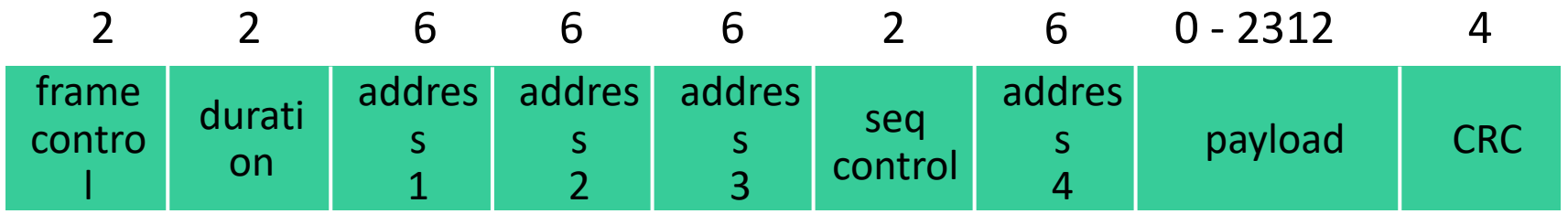
- sender first transmits *small* request-to-send (RTS) packet to BS using CSMA
  - RTSs may still collide with each other (but they’re short)
- BS broadcasts clear-to-send CTS in response to RTS
- CTS heard by all nodes
  - sender transmits data frame
  - other stations defer transmissions

COMPUTER NETWORKS

Collision Avoidance: RTS-CTS exchange



802.11 frame: addressing



Address 1: MAC address of wireless host or AP to receive this frame

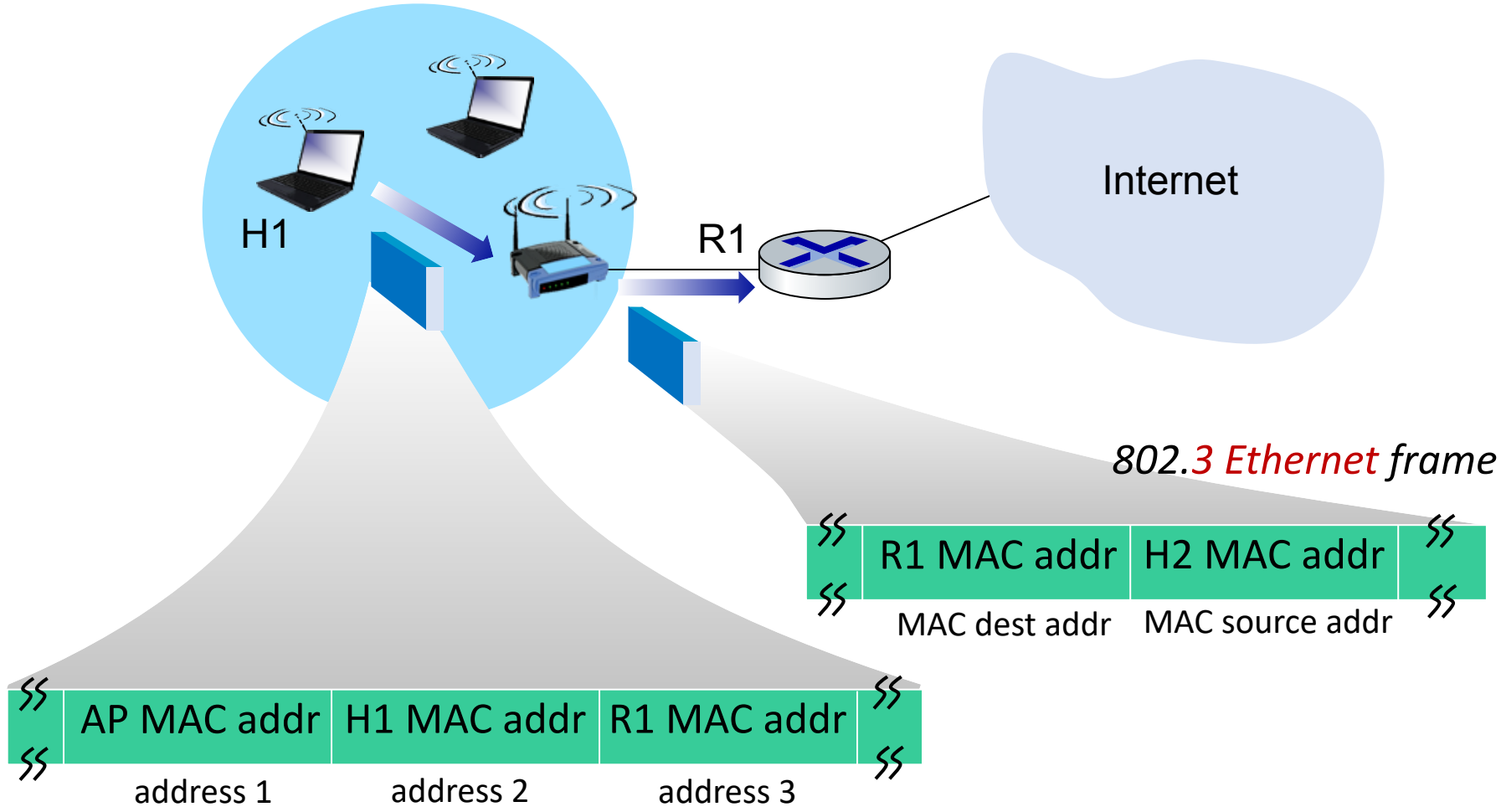
Address 2: MAC address of wireless host or AP transmitting this frame

Address 3: MAC address of router interface to which AP is attached

Address 4: used only in ad hoc mode

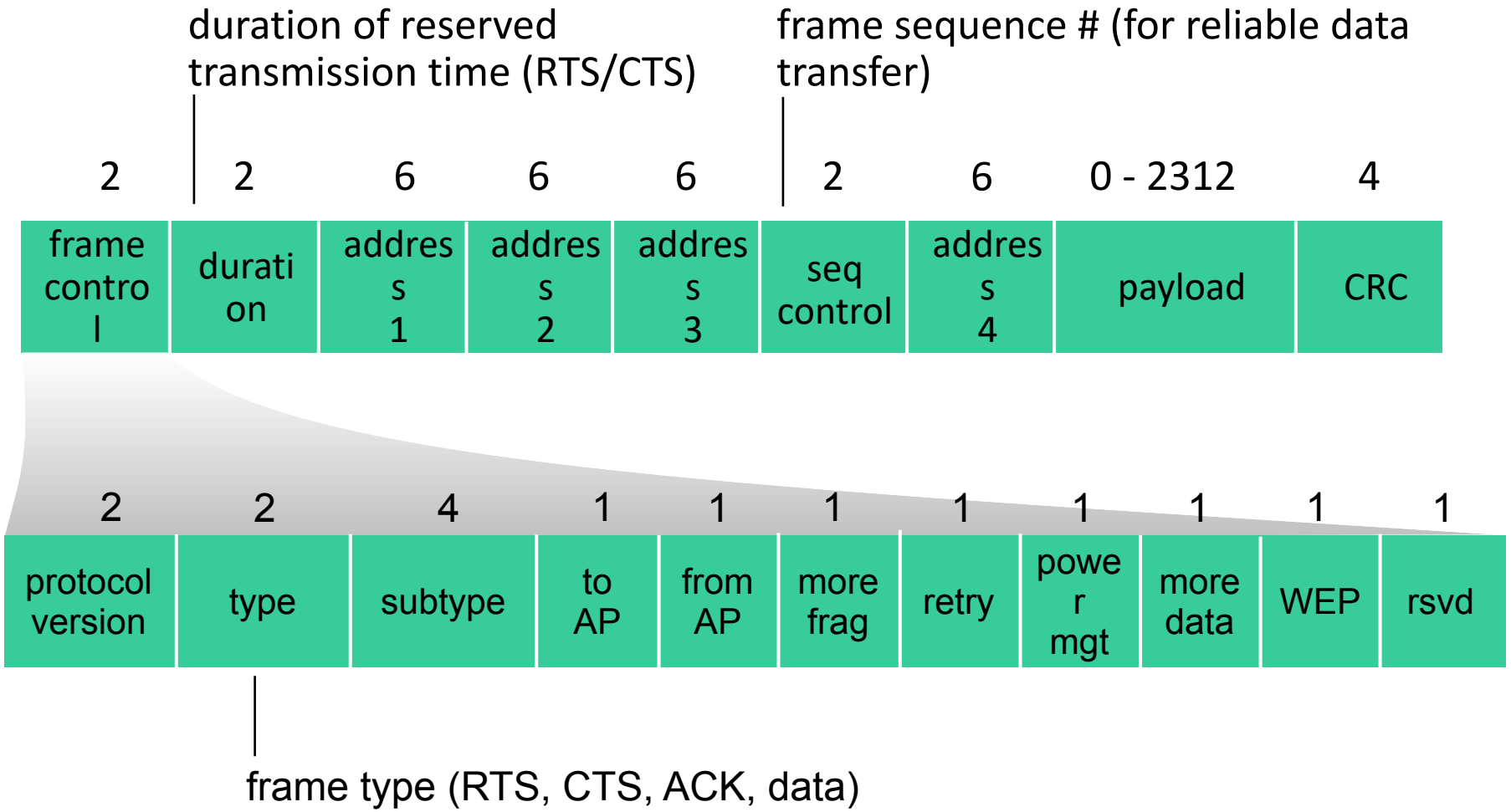
COMPUTER NETWORKS

802.11 frame: addressing



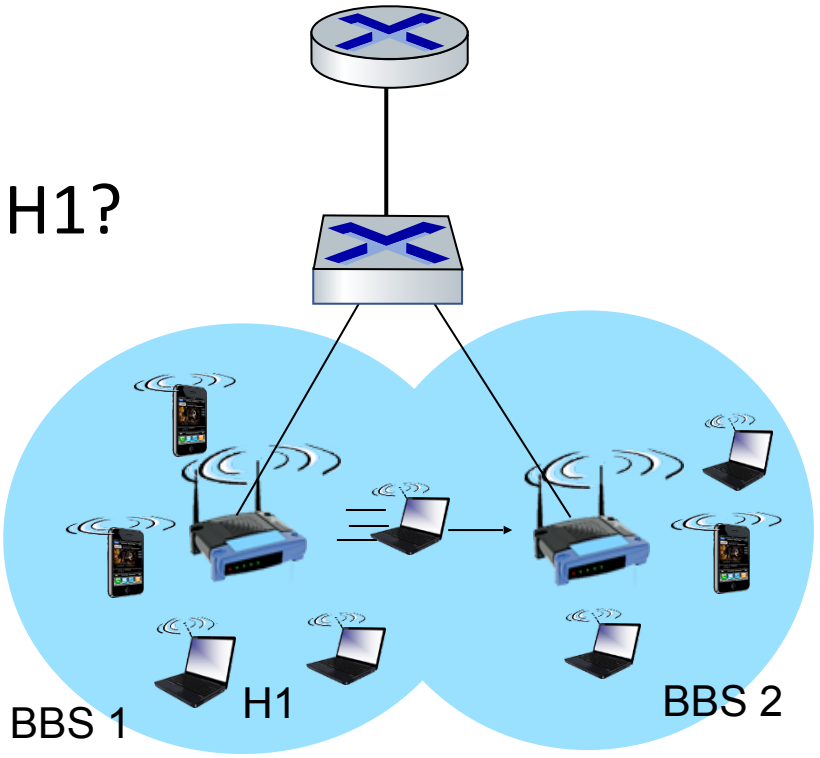
COMPUTER NETWORKS

802.11 frame: addressing



802.11: mobility within same subnet

- H1 remains in same IP subnet: IP address can remain same
- switch: which AP is associated with H1?
  - self-learning (Ch. 6): switch will see frame from H1 and "remember" which switch port can be used to reach H1



Suggested Readings

Thank You
For Your Attention

THANK YOU

TEAM NETWORKS

Department of Computer Science and Engineering