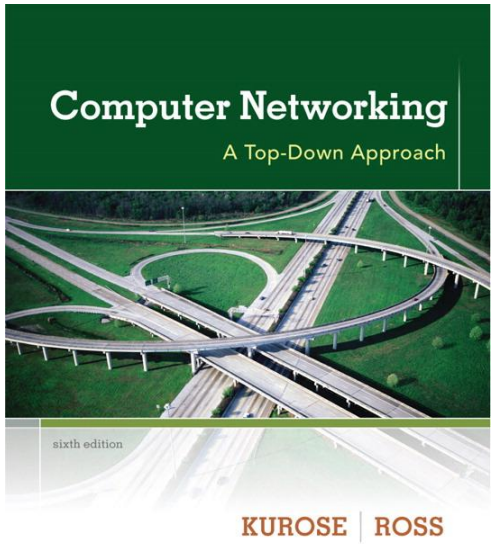
Chapter 5
Link Layer



Computer Networking: A Top Down Approach
6th edition
Jim Kurose, Keith Ross
Addison-Wesley
March 2012

Chapter 5: Link layer

our goals:

- understand principles behind link layer services:
  - error detection, correction
  - sharing a broadcast channel: multiple access
- understand link layer addressing
- internetworking devices: know differences between routers & switches
- local area networks: basics of Ethernet and WiFi technologies

Link layer, LANs: outline

5.1 introduction, services

5.2 error detection, correction

5.3 multiple access protocols

5.4 LANs

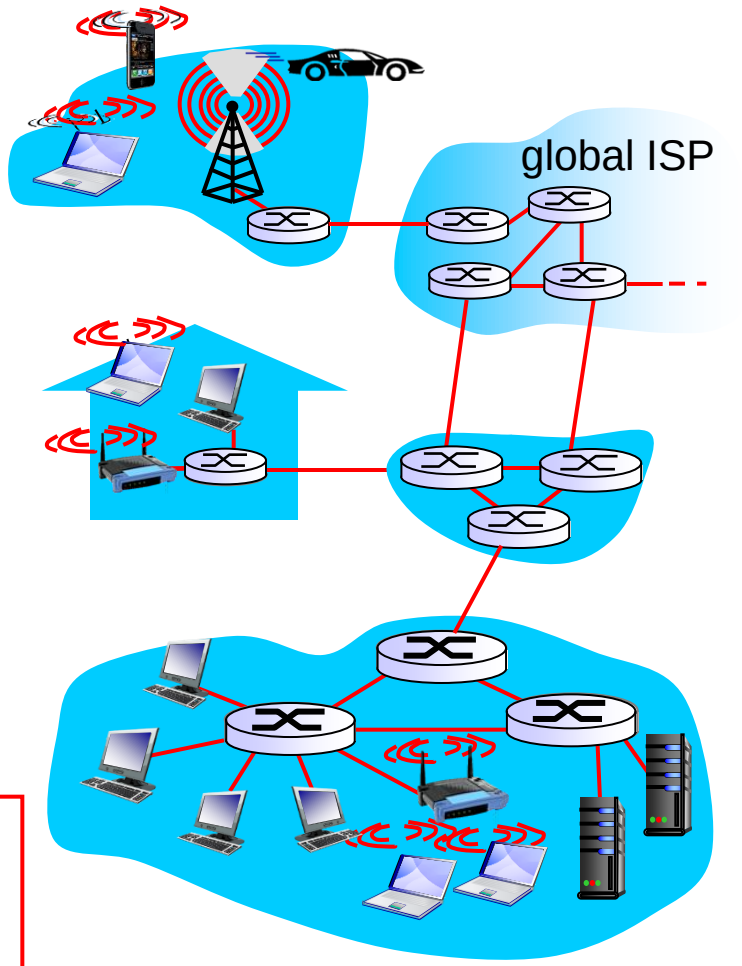
- addressing, ARP (Labs)
- Internetworking Devices
- Ethernet
- WiFi

5.5 a day in the life of a web request

Link layer: introduction

terminology:

- hosts and routers: nodes
- communication channels that connect adjacent nodes along communication path: links
  - wired links
  - wireless links
  - LANs
- layer-2 packet: frame, encapsulates datagram



data-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., Ethernet on first link, frame relay on intermediate links, 802.11 on last link
- each link protocol provides different services
  - e.g., may or may not provide reliable data transfer (rdt) over link

transportation analogy:

- trip from Princeton to Lausanne
  - limo: Princeton to JFK
  - 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

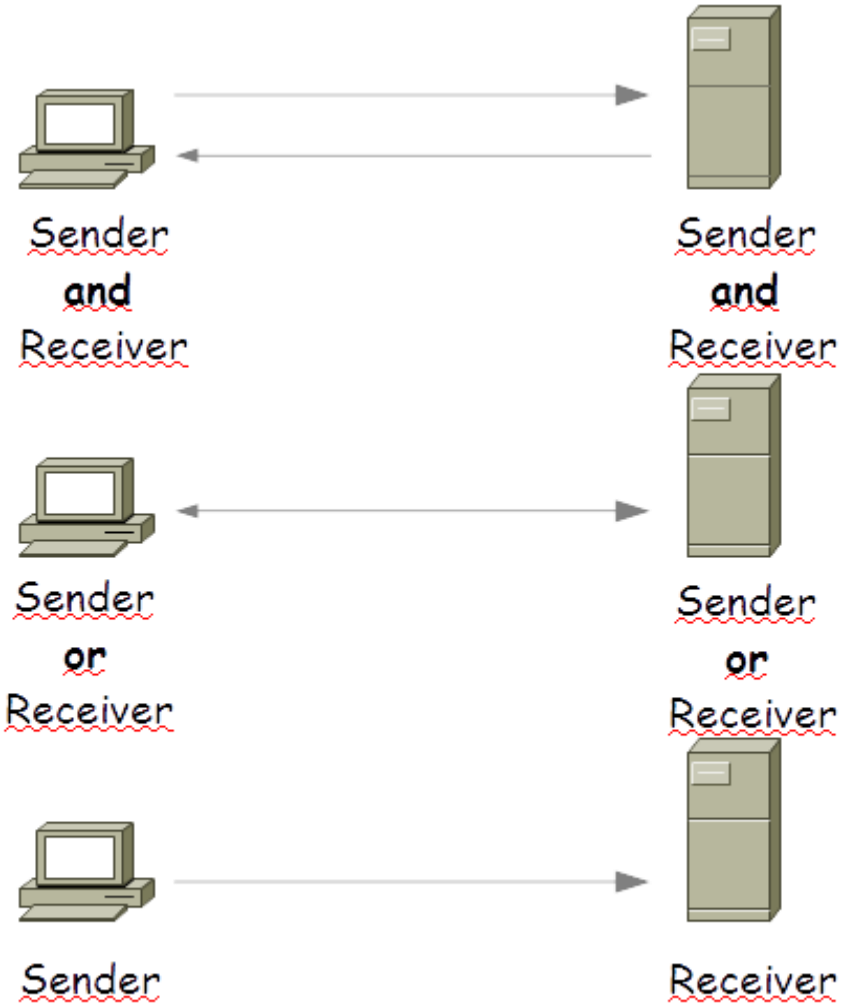
- *framing, link access:*
  - encapsulate datagram into frame, adding header, trailer
  - channel access if shared medium
  - "MAC" addresses used in frame headers to identify source, dest
    - different from IP address!
- *reliable delivery between adjacent nodes*
  - we learned how to do this already (chapter 3)!
  - seldom used on low bit-error link (fiber, some twisted pair)
  - wireless links: high error rates
    - *Q:* why both link-level and end-end reliability?

Link layer services (more)

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

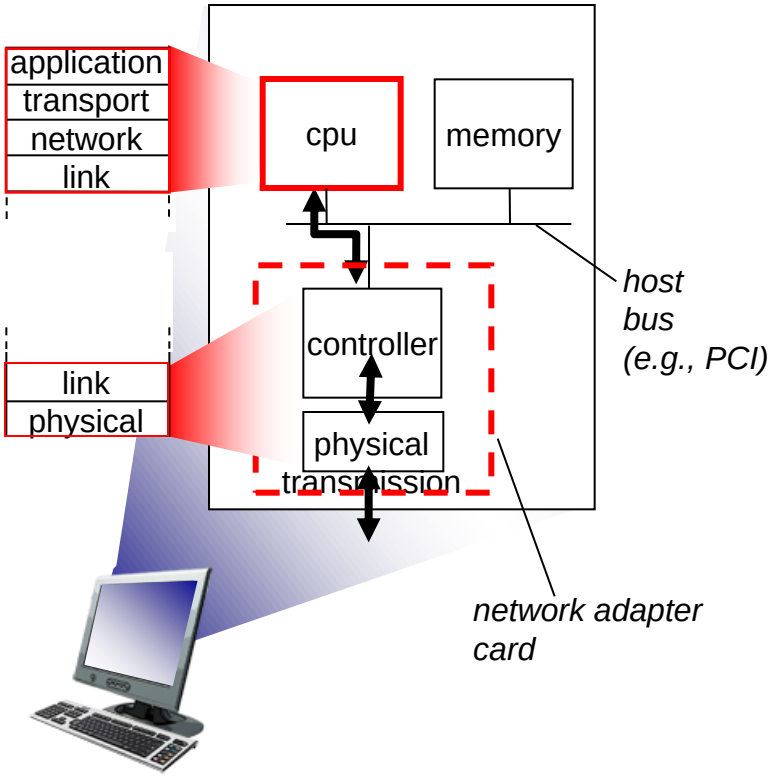
Full-duplex, Half-duplex, simplex

- full-duplex
- half-duplex
- simplex

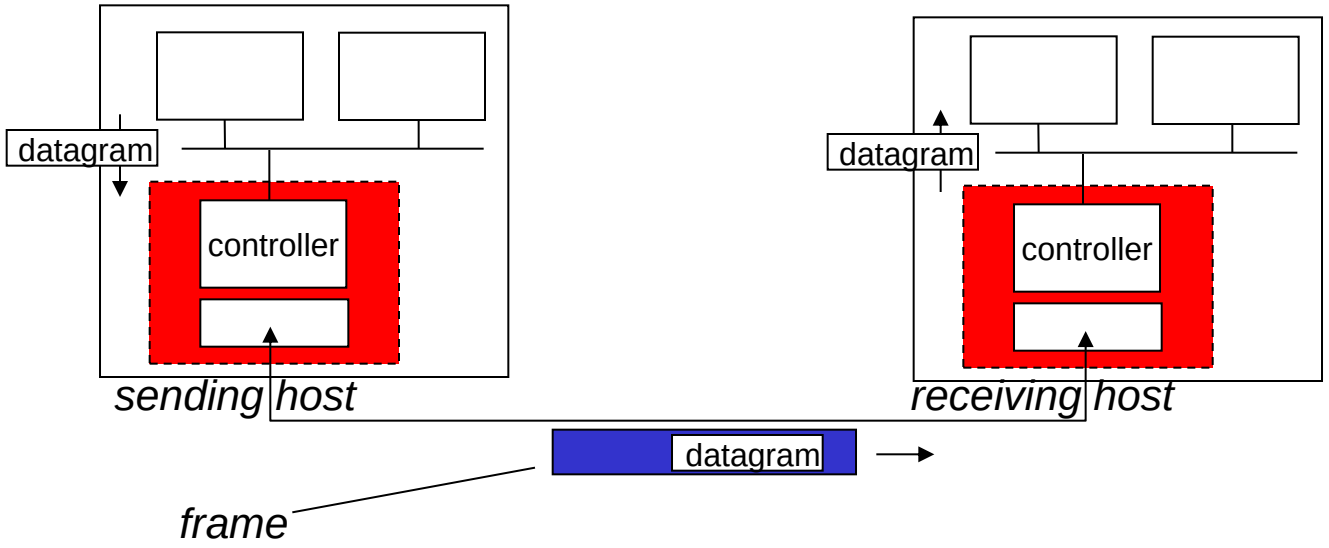


Where is the link layer implemented?

- in each and every host
- link layer implemented in "adaptor" (aka *network interface card* NIC) or on a chip
  - Ethernet card, 802.11 card; Ethernet chipset
  - implements link, physical layer
- attaches into host's system buses
- combination of hardware, software, firmware



Adaptors communicating



- sending side:
  - encapsulates datagram in frame
  - adds error checking bits, rdt, flow control, etc.
- receiving side
  - looks for errors, rdt, flow control, etc
  - extracts datagram, passes to upper layer at receiving side

Link layer, LANs: outline

5.1 introduction, services

5.2 error detection, correction

5.3 multiple access protocols

5.4 LANs

- addressing, ARP
- Ethernet
- switches
- VLANS

5.5 link virtualization: MPLS

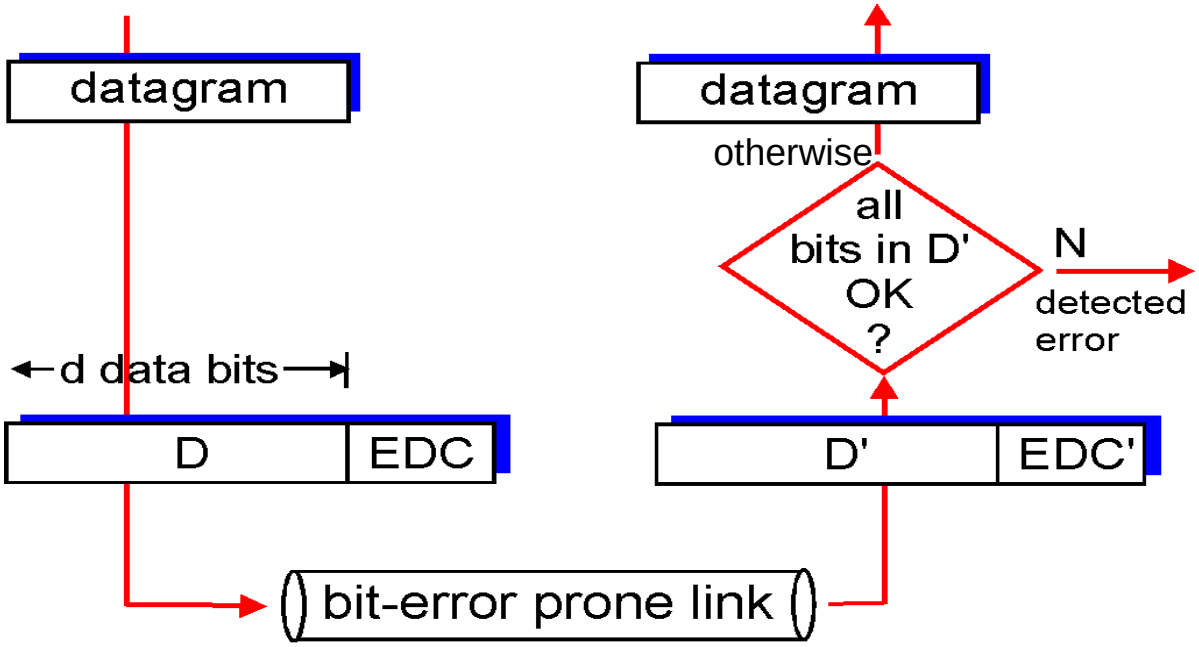
5.6 data center networking

5.7 a day in the life of a web request

Error detection

EDC= Error Detection and Correction bits (redundancy)
D = Data protected by error checking, may include header fields

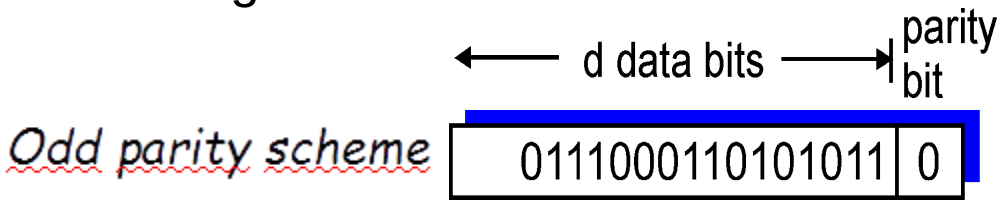
- Error detection not 100% reliable!
  - protocol may miss some errors, but rarely
  - larger EDC field yields better detection and correction



Parity checking

single bit parity:

- detect single bit errors



- Parity Bit = 1 or 0 to make:
  - Even / odd parity : even / odd number of 1s in each byte
    - Parity bit value is chosen such that number of 1's send is even / odd.
  - Example:

Information to transmit: 0100101

with even parity: 01001011

with odd parity: 01001010

Internet checksum (review)

goal: detect "errors" (e.g., flipped bits) in transmitted packet (note: used at transport layer *only*)

sender:

- treat segment contents as sequence of 16-bit integers
- checksum: addition (1's complement sum) of segment contents
- sender puts checksum value into UDP checksum field

receiver:

- compute checksum of received segment
- check if computed checksum equals checksum field value:
  - NO - error detected
  - YES - no error detected. *But maybe errors nonetheless?*

Binary Data	CheckSum Value
0001	1
0010	2
0011	3
0001	1
Total	7

Binary Data	CheckSum Value
0011	3
0000	0
0001	1
0011	3
Total	7

Checksum example

- Given a 48-bit package, divide it into three 16-bit words:

  0110011001100110
  0101010101010101
  0000111100001111

- The sum of the first two words would be:

  0110011001100110
  0101010101010101

  1011101110111011

- Adding the third "word" to the previous result:

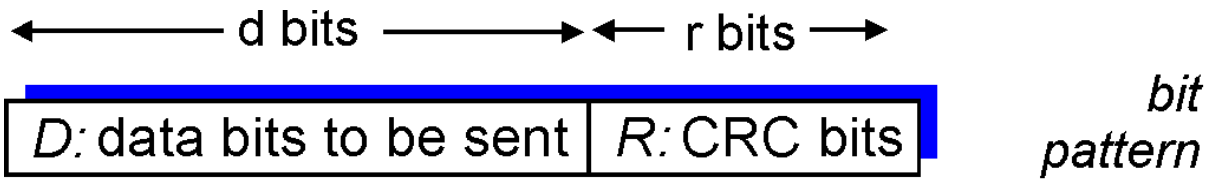
  1011101110111011
  0000111100001111

  1100101011001010

- The Checksum would be the1's complement sum (swapping 0s for 1s and vice versa): 0011010100110101
- When arriving at the receiver the four 16-bit word including the checksum are added and the result should be 1111111111111111. If one of the bits is zero, an error has been detected.

Cyclic redundancy check (CRC)

- more powerful error-detection coding
- view data bits, D, as a binary number
- choose r+1 bit pattern (generator), G
- goal: choose r CRC bits, R, such that
  - <D,R> exactly divisible by G (modulo 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, ATM)



$D * 2^r \quad XOR \quad R$ — *mathematical formula*

CRC example

want:

$D \cdot 2^r$ XOR R = nG

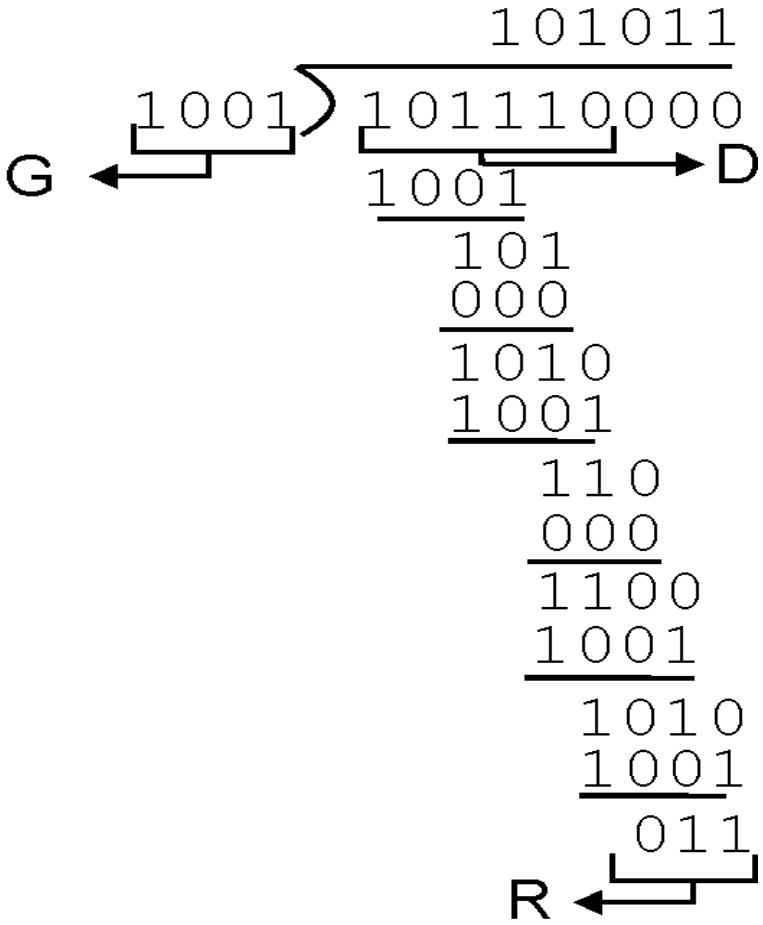
equivalently:

$D \cdot 2^r$ = nG XOR R

equivalently:

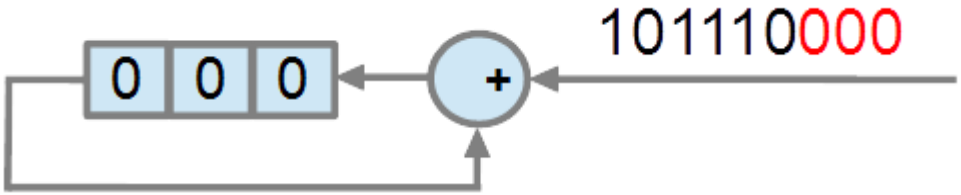
if we divide $D \cdot 2^r$ by G, want remainder R to satisfy:

$$R = remainder\left[\frac{D \cdot 2^r}{G}\right]$$

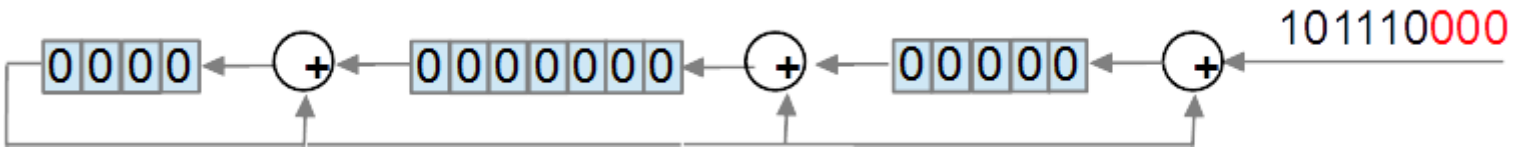


HW implementation of CRC

- The mathematical foundations of CRC are complex ... but hardware implementation is simple
  - Simply with shift registers and XOR gates
  - For the previous example the hardware implementation would be:



  - Complex example: Generator= $x^{16} + x^{12} + x^5 + 1$:



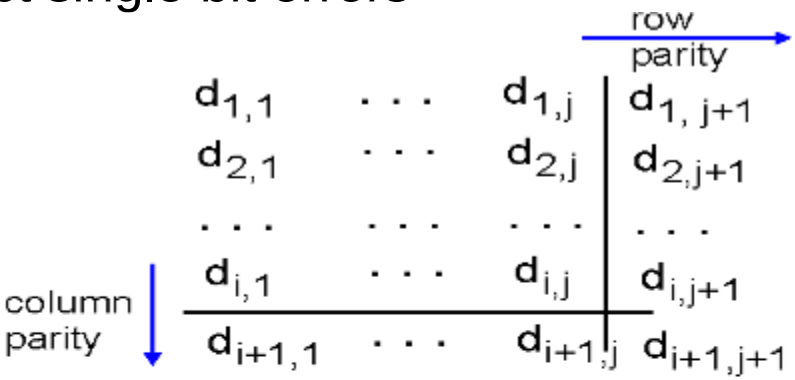
Error Correction

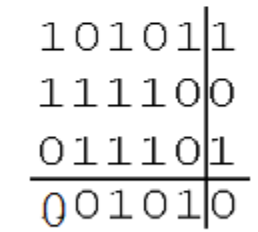
- In order to correct the mistakes you can use two strategies:
  - **FEC *(Forward Error Correction)***
    - Adding enough information that the receiver can recover the correct information
    - Detection + Recovery
  - **ARQ *(Automatic Repeat reQuest)***
    - The receiver asks the transmitter to forward the correct information
    - Detection + Forward

FEC (Forward Error Correction)

- Example:

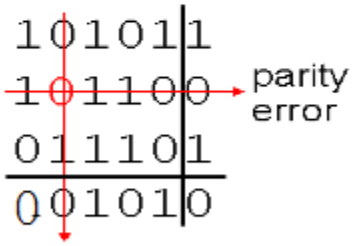
  two-dimensional bit parity:
  - detect and correct single bit errors





no errors

(even parity)



correctable single bit error

ARQ (*Automatic Repeat reQuest*)

- Frames with errors must be discarded and sent again
- The techniques are based on reliable transmission mechanisms
  - Acknowledgements (ACK)
  - timeout

↖ Studied in Unit 5 !!!!!

Link layer, LANs: outline

5.1 introduction, services

5.2 error detection, correction

5.3 multiple access protocols

5.4 LANs

- addressing, ARP (Labs)
- Internetworking Devices
- Ethernet
- WiFi

5.5 a day in the life of a web request

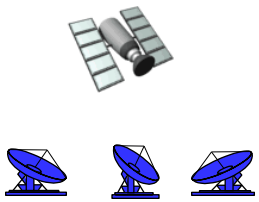
Multiple access links, protocols

two types of "links":

- point-to-point
  - PPP for dial-up access
  - point-to-point link between Ethernet switch, host
- *broadcast (shared wire or medium)*
  - old-fashioned Ethernet
  - upstream HFC
  - 802.11 wireless LAN

shared wire (e.g., cabled Ethernet)

shared RF (e.g., 802.11 WiFi)



shared RF (satellite)

humans at a cocktail party (shared air, acoustical)

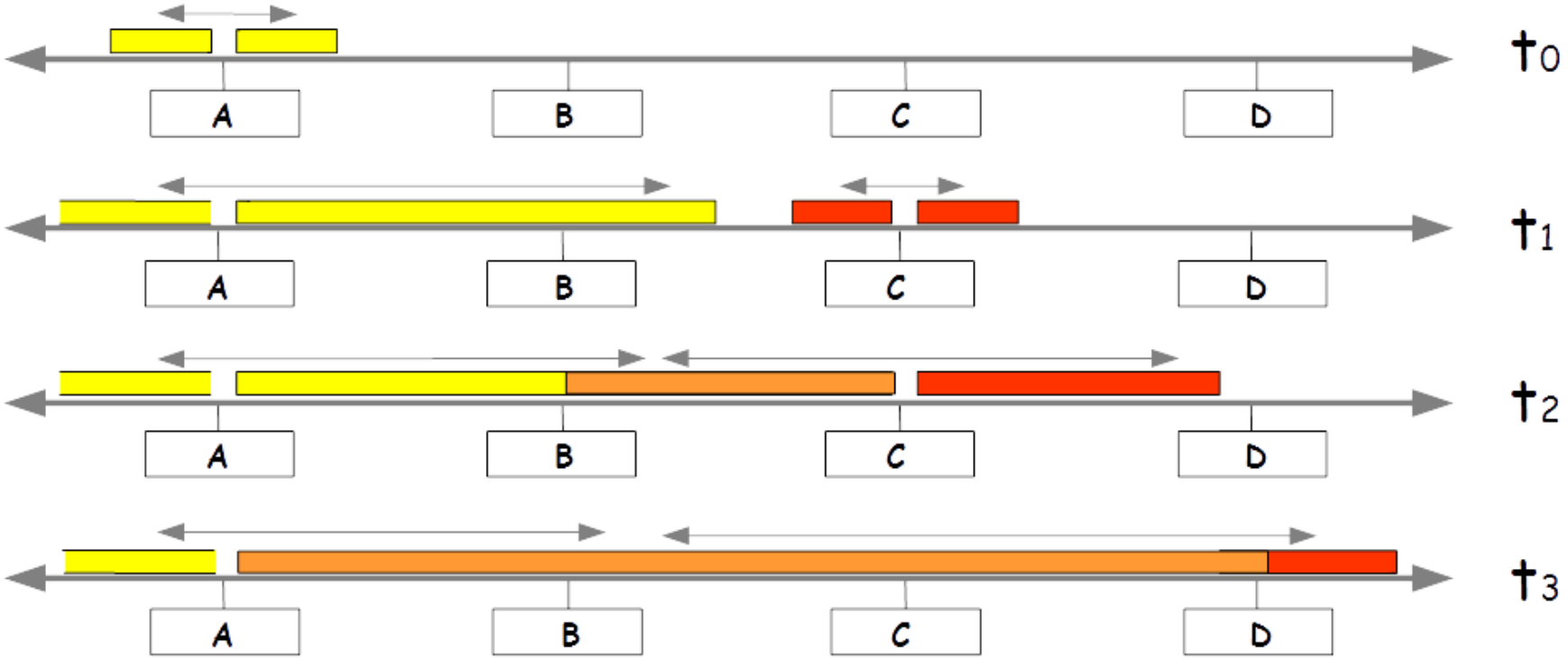
Multiple access protocols

- 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

Frame collision



- all the frames involved in the collision are lost
- the broadcast channel is wasted during the collision interval

An ideal multiple access protocol

given: broadcast channel of rate R bps

desiderata:

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

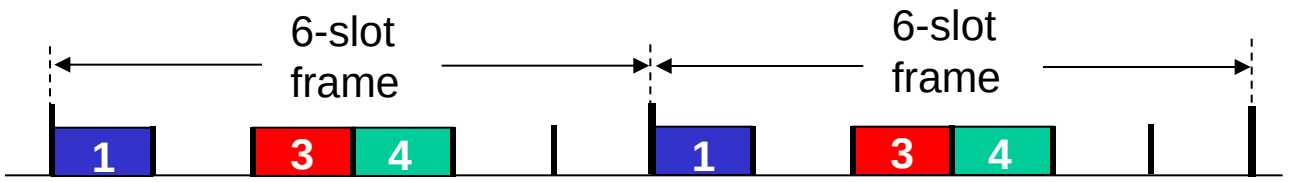
three broad classes:

- *channel partitioning*
  - divide channel into smaller "pieces" (time slots, frequency, code)
  - allocate piece to node for exclusive use
- *random access*
  - channel not divided, allow collisions
    - "recover" from collisions
  - nodes try to obtain the channel without any control
- *"taking turns"*
  - nodes take turns, but nodes with more to send can take longer turns

Channel partitioning MAC protocols: TDMA

TDMA: time division multiple access

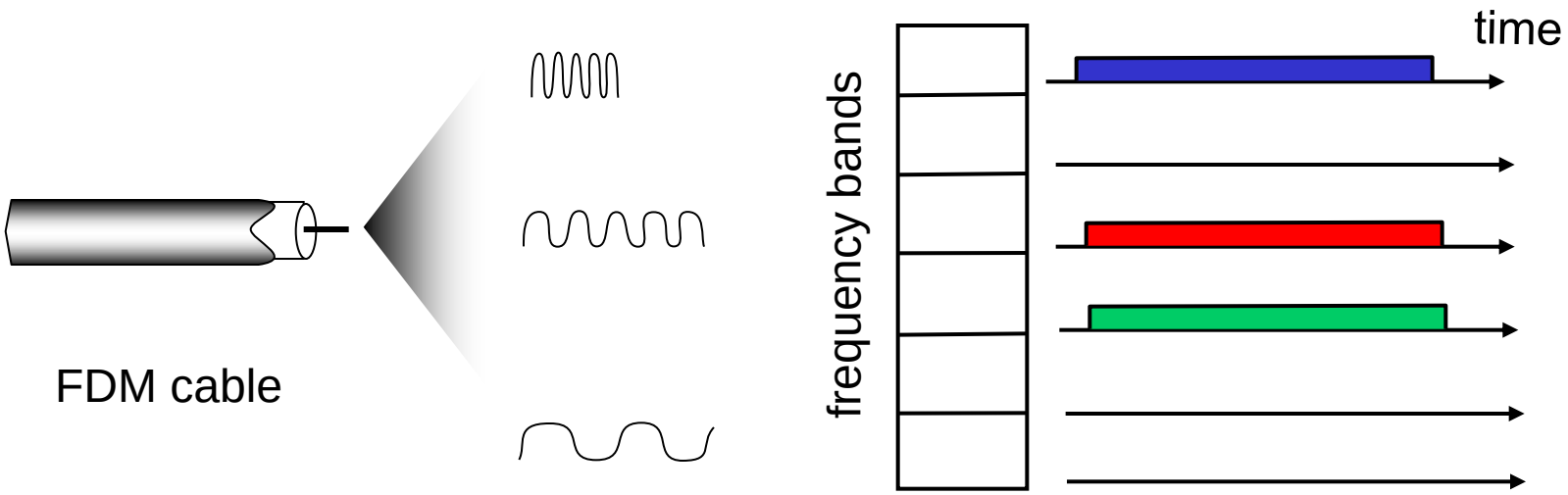
- Share the broadcast channel in time
- access to channel in "rounds"
- each station gets fixed length slot (length = pkt trans time) in each round
- unused slots go idle
- example: 6-station LAN, 1,3,4 have pkt, slots 2,5,6 idle



Channel partitioning MAC protocols: FDMA

FDMA: frequency division multiple access

- channel spectrum divided into frequency bands
- each station assigned fixed frequency band
- unused transmission time in frequency bands go idle
- example: 6-station LAN, 1,3,4 have pkt, frequency bands 2,5,6 idle



Channel partitioning MAC protocols: TDMA & FDMA

- Advantages:
  - It eliminates collisions
  - It divides the bandwidth fairly among the N nodes
    - each node gets a dedicated transmission rate of R/N bps
    - R is the transmission rate of the channel
    - N is the number of nodes that can transmit
- Drawbacks:
  1. A node is limited to an average rate of R/N bps
  2. A node must always wait for its turn in the transmission sequence
  - 1 and 2 happen even when a node is the only node with packets to transmit

Random access protocols

- a node's decision to transmit is made independently of the activity of other nodes attaches to the broadcast channel
- when node has packet to send
  - transmit at full channel data rate R.
  - no *a priori* coordination among nodes
- two or more transmitting nodes ➜ "collision",
- random access MAC protocol specifies:
  - how to detect collisions
  - how to recover from collisions (e.g., via delayed retransmissions)
- examples of random access MAC protocols:
  - slotted ALOHA,
  - ALOHA
  - CSMA, CSMA/CD, CSMA/CA

CSMA/CD (collision detection)

CSMA: listen before transmit:

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

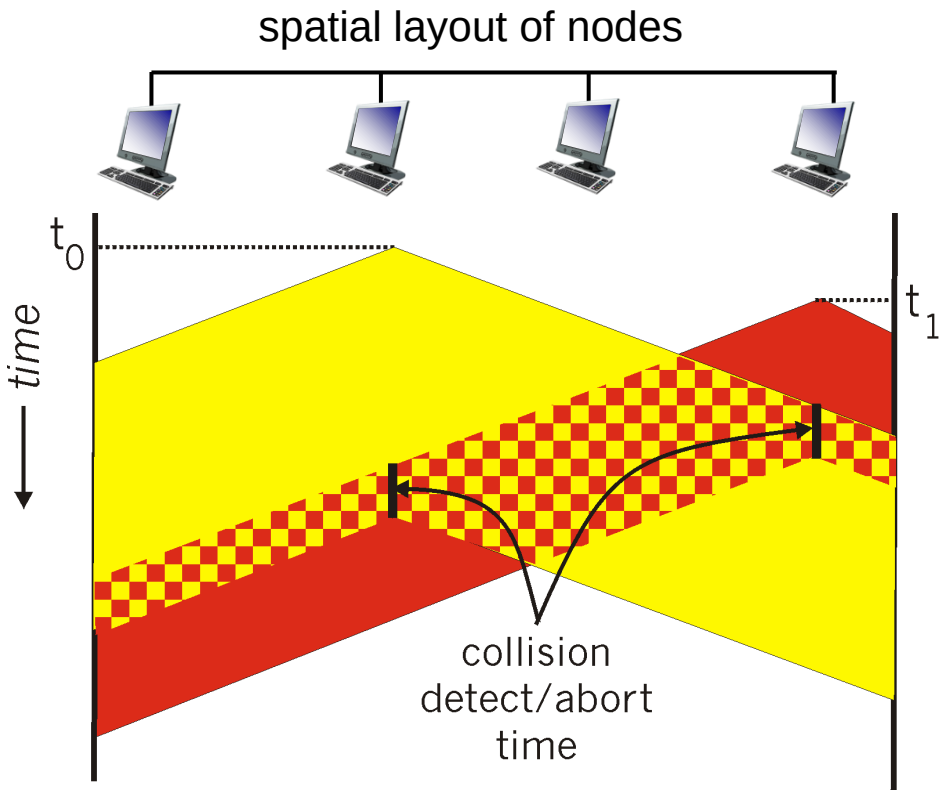
CSMA/CD: carrier sensing, deferral as in CSMA

- collisions *detected* within short time
- colliding transmissions aborted, reducing channel wastage

- collision detection:
  - easy in wired LANs: measure signal strengths, compare transmitted, received signals
  - difficult in wireless LANs: received signal strength overwhelmed by local transmission strength
- human analogy: the polite conversationalist

CSMA/CD (collision detection)

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



Ethernet CSMA/CD algorithm

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

Binary exponential backoff algorithm:

- After the nth collision of a frame
- NIC waits = K * 512 bit times
  - K is chosen at random from $\{0,1,2, \ldots, 2^m-1\}$
  - m = min (n, 10)
- Example:
  - A node in a 10 Mbps Ethernet network tries to transmit a frame for the second time, and a collision occurs
    - bit time is 1bit/10^6 bps = 0.1 μs
    - m=min(2,10)
    - NIC chooses *K* at random from {0,1,2,3}, p.e: K=2
    - Timeout before transmitting again the frame = 2* 512 * 0,1= 102,4 μs

CSMA/CD efficiency

- 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

"Taking turns" MAC protocols

channel partitioning MAC protocols:

- share channel *efficiently* and *fairly* at high load
- inefficient at low load: delay in channel access, 1/N bandwidth allocated even if only 1 active node!

random access MAC protocols

- efficient at low load: single node can fully utilize channel
- high load: collision overhead

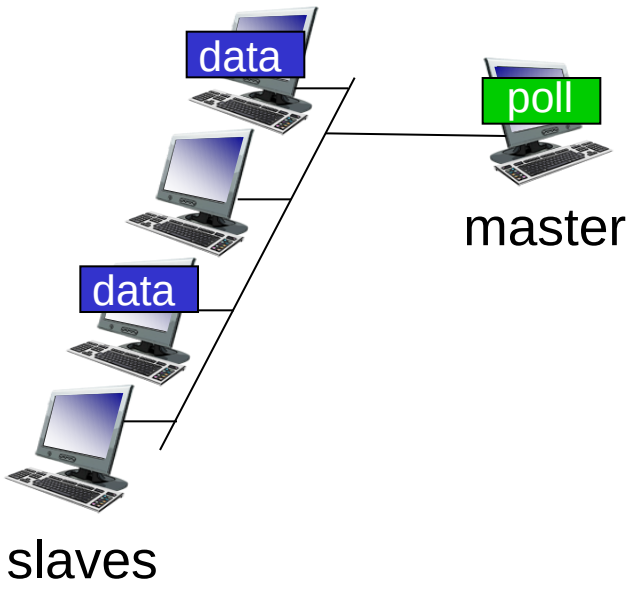
"taking turns" protocols

look for best of both worlds!

"Taking turns" MAC protocols

polling:

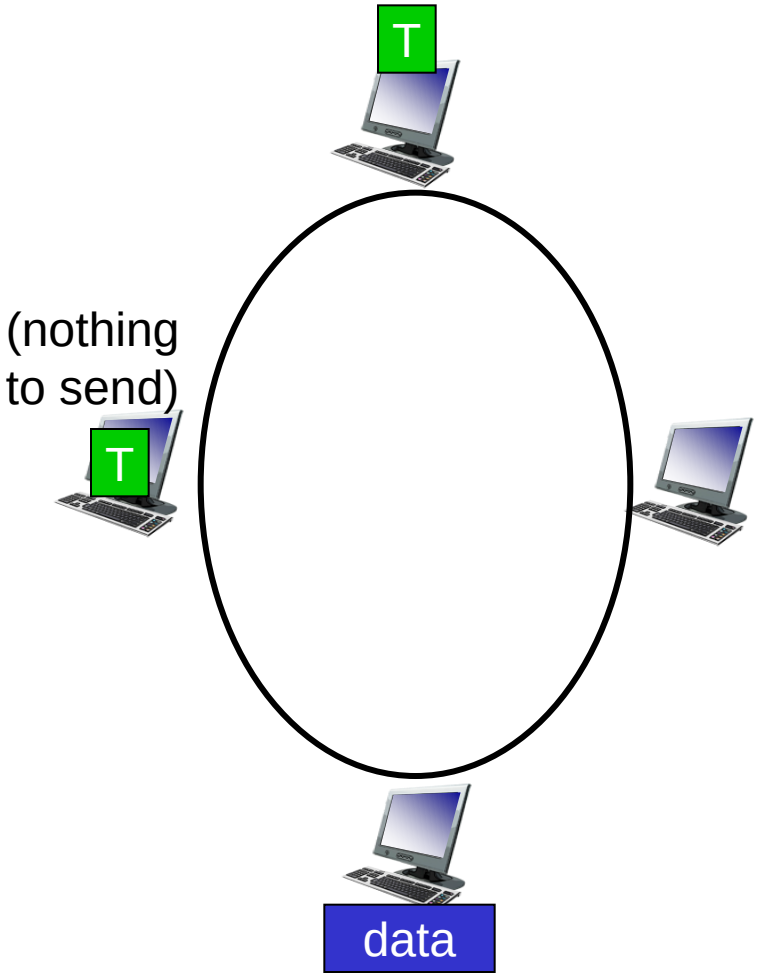
- master node "invites" slave nodes to transmit in turn
- typically used with "dumb" slave devices
- concerns:
  - polling overhead
  - latency
  - single point of failure (master)



"Taking turns" MAC protocols

token passing:

- control *token* passed from one node to next sequentially.
- token message
- concerns:
  - token overhead
  - latency
  - single point of failure (token)



Summary of MAC protocols

- *channel partitioning,* by time, or frequency
  - Time Division, Frequency Division
- *random access* (dynamic),
  - ALOHA, S-ALOHA, CSMA, CSMA/CD
  - carrier sensing: easy in some technologies (wire), hard in others (wireless)
  - CSMA/CD used in Ethernet
  - CSMA/CA used in 802.11
- *taking turns*
  - polling from central site, token passing
  - bluetooth, FDDI, token ring

Link layer, LANs: outline

5.1 introduction, services

5.2 error detection, correction

5.3 multiple access protocols

5.4 LANs

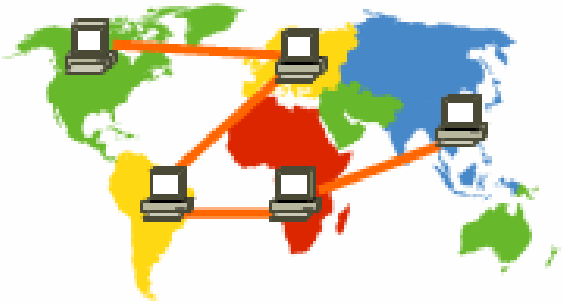
- addressing, ARP (Labs)
- Internetworking Devices
- Ethernet
- WiFi

5.5 a day in the life of a web request

Why LANs?

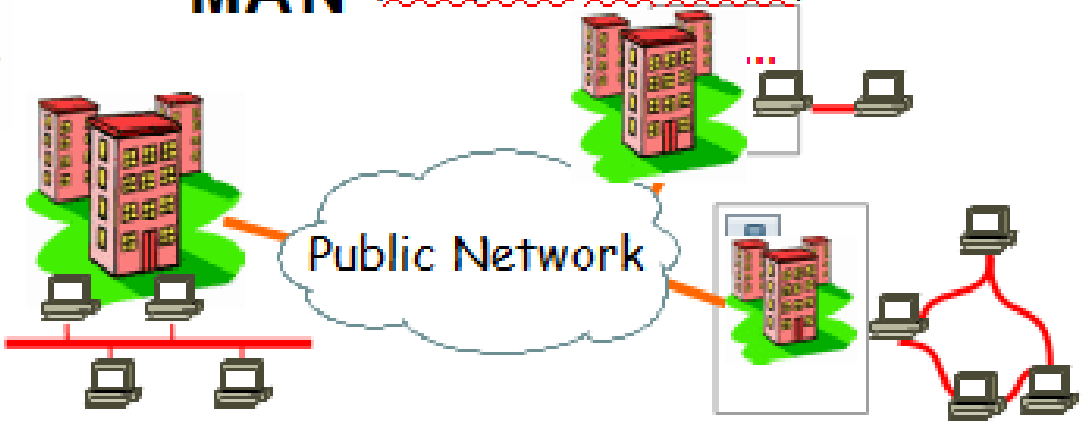
- The need to share data is a compelling reason for interconnection
- To provide ubiquitous access to shared resources (e.g., printers, databases, file systems...)
- To allow remote users to communicate (e.g., email, IP telephony, social networks)
- To do transactions (banking, e-commerce, stock trading)
- The need to share expensive resources (e.g., printers, storage devices, video equipment)

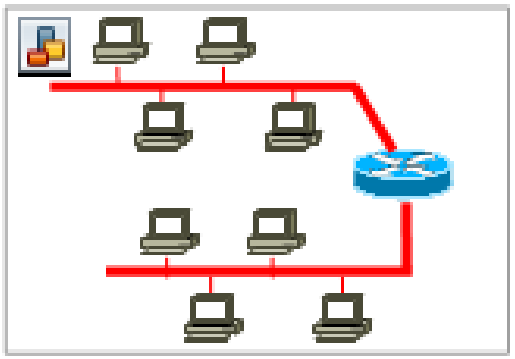
Networks classification



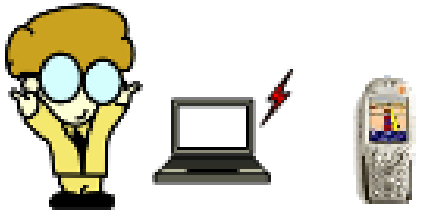
WAN (Wide Area Network)

MAN (Metropolitan Area Network)





LAN (Local Area Network)



PAN (Personal Area Network)

LANs

- Cover a moderate size geographical area
- LANs typically connect systems located within a single room, multiple floors of a building, a warehouse, or a campus consisting of several buildings
- LANs broadcasts data at high data transfer rates with very low error rates
- LANs can even incorporate sites located at great distance from one another (Virtual Private Networks (VPN))
- LAN is owned, used and operated by the same organization that owns the attached devices
- The main LAN technologies are 802.3 Family (Ethernet) and 802.11 (Wi-Fi)

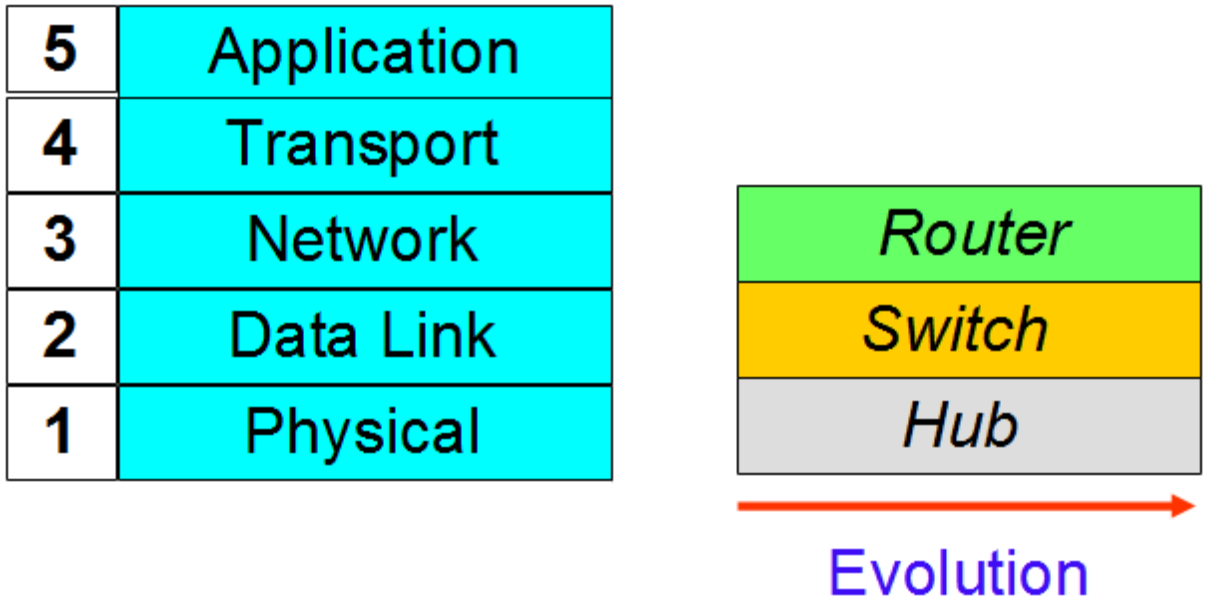
MANs (Metropolitan Area Networks)

- Cover greater distances at higher data rates than LANs
- A MAN usually interconnects a number of LANs using a high-capacity backbone technology

WANs (Wide Area Networks)

- Cover a large geographical area
  - Such a state, province or country
- Often connect multiple smaller networks, such as local area networks (LANs) or metro area networks (MANs)
- Typical WAN technologies are SONET (Synchronous Optical Network), Frame Relay, and ATM
- The world's most popular WAN is the Internet

Interconnection Devices



Collision and Broadcast Domain

- Collision Domain:

  Set of stations that are affected in a collision (whether they participate in it or no)

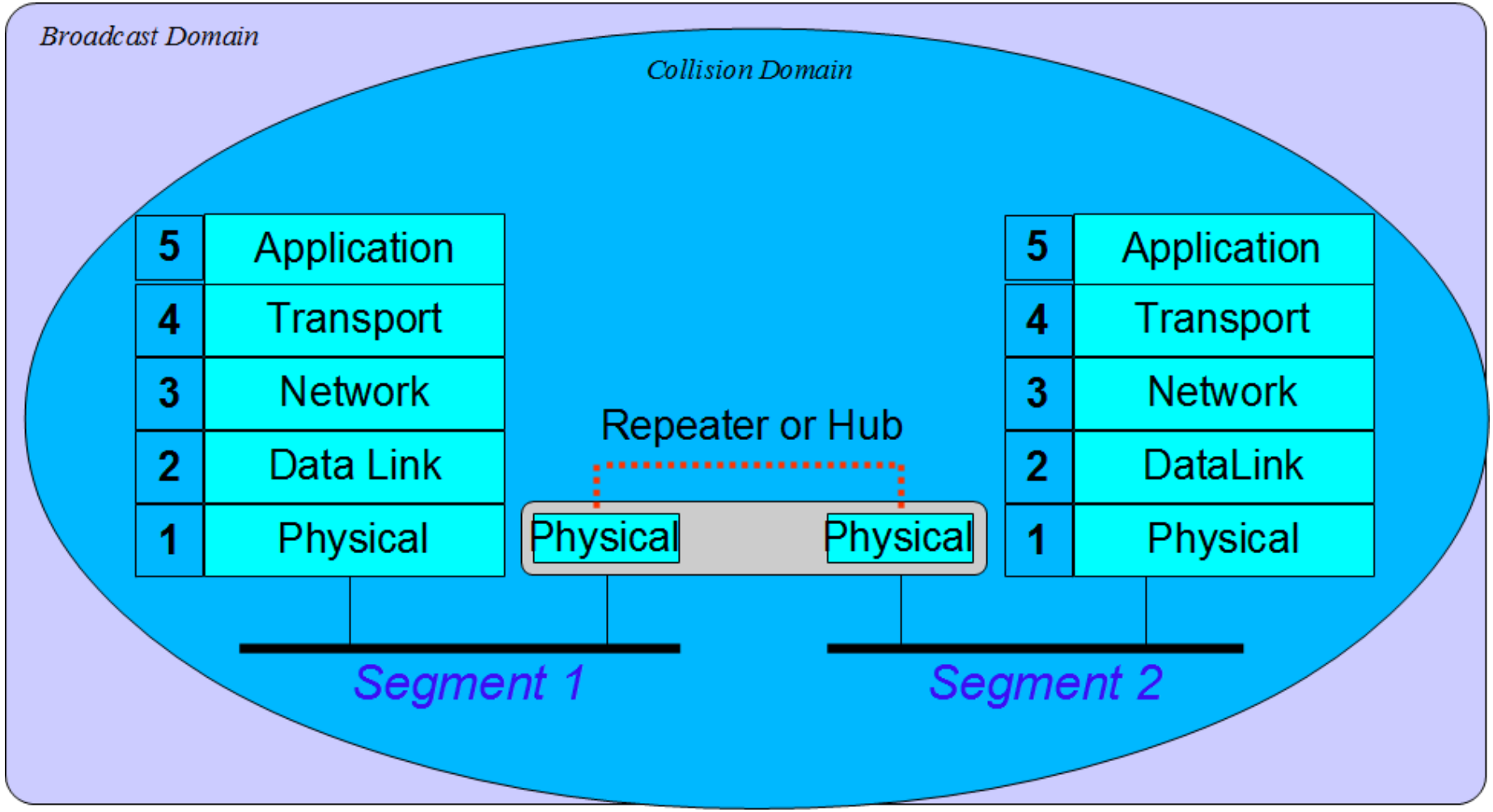
- Broadcast Domain:

  Set of stations that receive a broadcast made by any of them

Repeaters/*Hubs*

- The transmitted signal is attenuated with distance
- A repeater is an electronic device that regenerates the signal to reach farther
  - Interconnect two or more LAN segments
  - The repeater does not understand the format of the frame, or physical addresses: copy any electrical signal (also collisions)
- Hubs are multiport repeaters
- Repeaters and Hubs not separate collision domains
- Cannot support multiple LAN technologies
  - Repeaters/hubs do not buffer or interpret frames
  - So, can't interconnect between different rates or formats
  - E.g., no mixing 10 Mbps Ethernet & 100 Mbps Ethernet

Hubs


Broadcast Domain
Collision Domain
5 Application
4 Transport
3 Network
2 Data Link
1 Physical
Repeater or Hub
Physical
Physical
5 Application
4 Transport
3 Network
2 DataLink
1 Physical
Segment 1
Segment 2

Bridge/Switch

- Interconnect segments of a single LAN
- Separate collision domains but not broadcast domains
- Switches are multiport bridges
- The main function of a switch is to forward any frame whose source and destination are on different sides of the switch
- The switches do not analyze the frame data, only the physical addresses
- When a switch forward a frame, it uses the original source address
- The switch is called “transparent” because you can plug it in and forget it. It is effective, and its actions are invisible to user

Switches Learning

- How does a switch know where a station with a particular MAC address is located?
  - The switch watches all the traffic at each of its ports
  - The switch notes the source MAC address of each frame and the port at which the frame was observed
  - The switch adds what it learns to a table called *filtering table*. Learned entries also are called *dynamic* entries
- The processing steps are:
  - If frame's destination is in the filtering table and is reached through the arrival port, the switch discard the frame
  - If the frame's destination is in the filtering table and its exit port is different from the arrival port, the switch forward the frame through the exit port
  - If the frame's destination in not in the filtering table, the switch forwards the frame through all ports other than the arrival port

Example

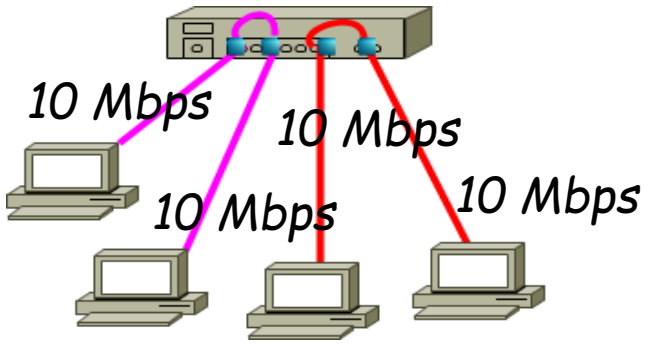
Destination MAC Address	Transmit Port	Status
00-06-08-1E-AE-42	2	Learned
00-60-08-BD-7D-1A	3	Learned
00-90-27-AE-B9-1D	1	Age out

- *If a frame with destination MAC address 00-60-08-1E-AE-42 arrives from switch port 2, the switch will ignore it.*
- *If a frame with this destination arrives from any other port, the switch will transmit the frame out port 2*
- *Every time a switch observes a MAC address, the switch restarts a timer associated with its entry.*
- *If the station with this address stop sending frames, its entry will time out and be removed from the table (Age out)*
- *The usual default age-out time is 5 minutes, but this can be changed by the LAN administrator*

Switch

- A switch makes it possible to build mixed-speed LAN
  - For example, traffic must be switched between a 10Mbps segment and a 100Mbps segment

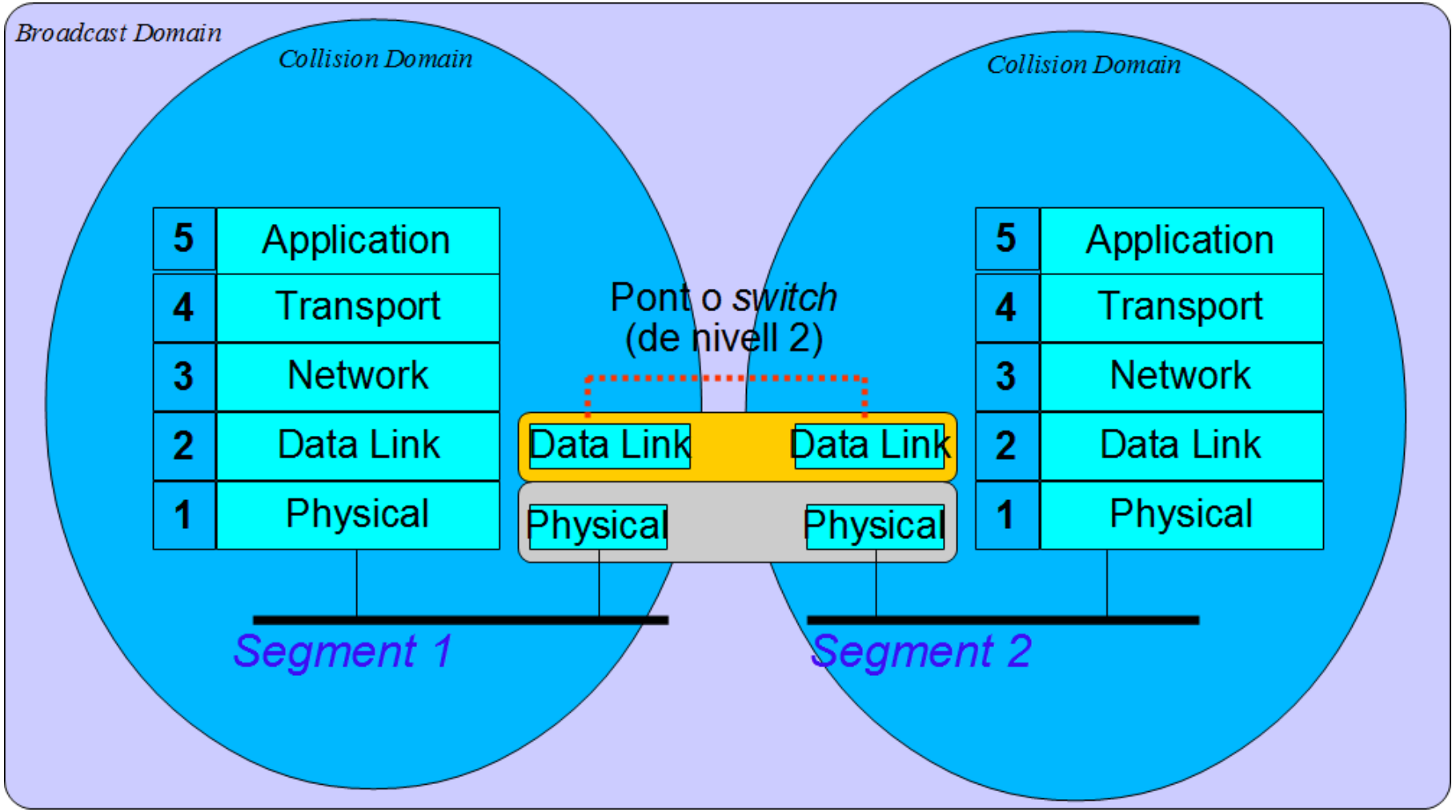
- Switch can amplify the bandwidth available
  - In the best case (and with end systems that support full-duplex transfer) the effective LAN bandwidth is equal to multiply the link bandwidth by the number of ports



Switch limitations

- The switch does not separate broadcast domains
  - Limited scalability
  - The number of broadcasts increases with the number of hosts connected to the switch
  - Broadcasts interrupt to all hosts
- Switches have to build a *spanning* tree topology because they forward packets according to MAC addresses which are not structured and they do not detect frames that loop
- The spanning tree algorithm reduces the active topology to a tree

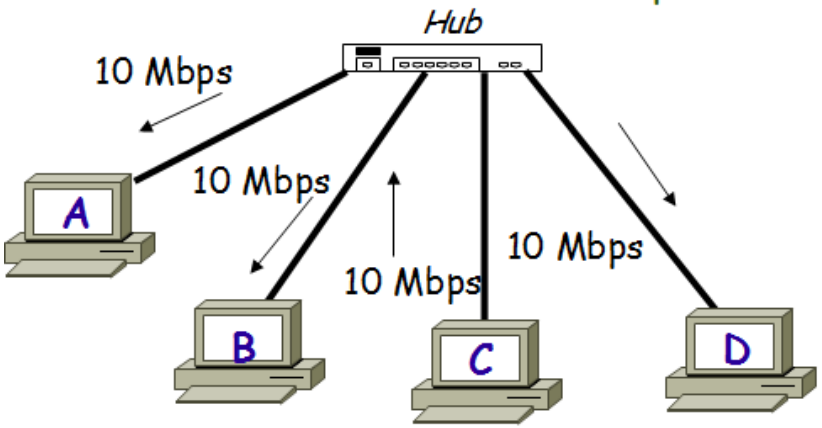
Switch


Broadcast Domain
Collision Domain
Collision Domain
5 Application
4 Transport
3 Network
2 Data Link
1 Physical
Pont o switch
(de nivell 2)
Data Link
Data Link
Physical
Physical
5 Application
4 Transport
3 Network
2 Data Link
1 Physical
Segment 1
Segment 2

Hub vs Switch

Half-duplex

Maximum LAN Bandwidth =10 Mbps

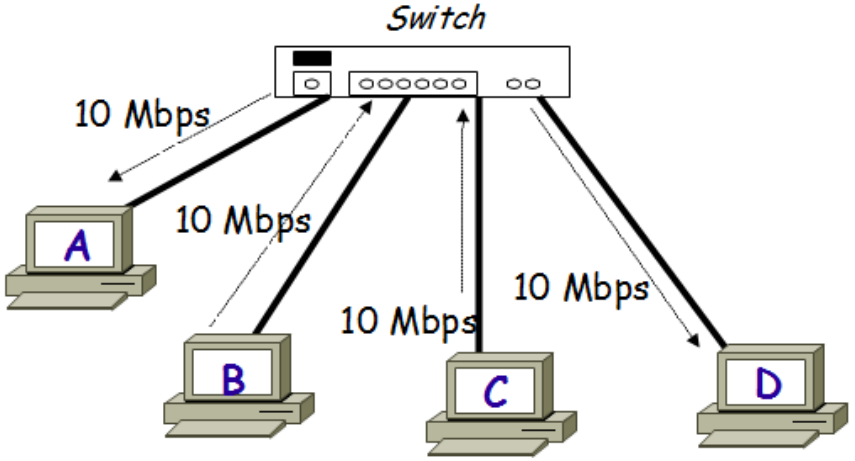


C sends a frame and A, B and D receive the frame

Full-duplex

B sends a frame to D and at the same time C sends a frame to A

Maximum LAN Bandwidth = # Ports * 10 Mbps



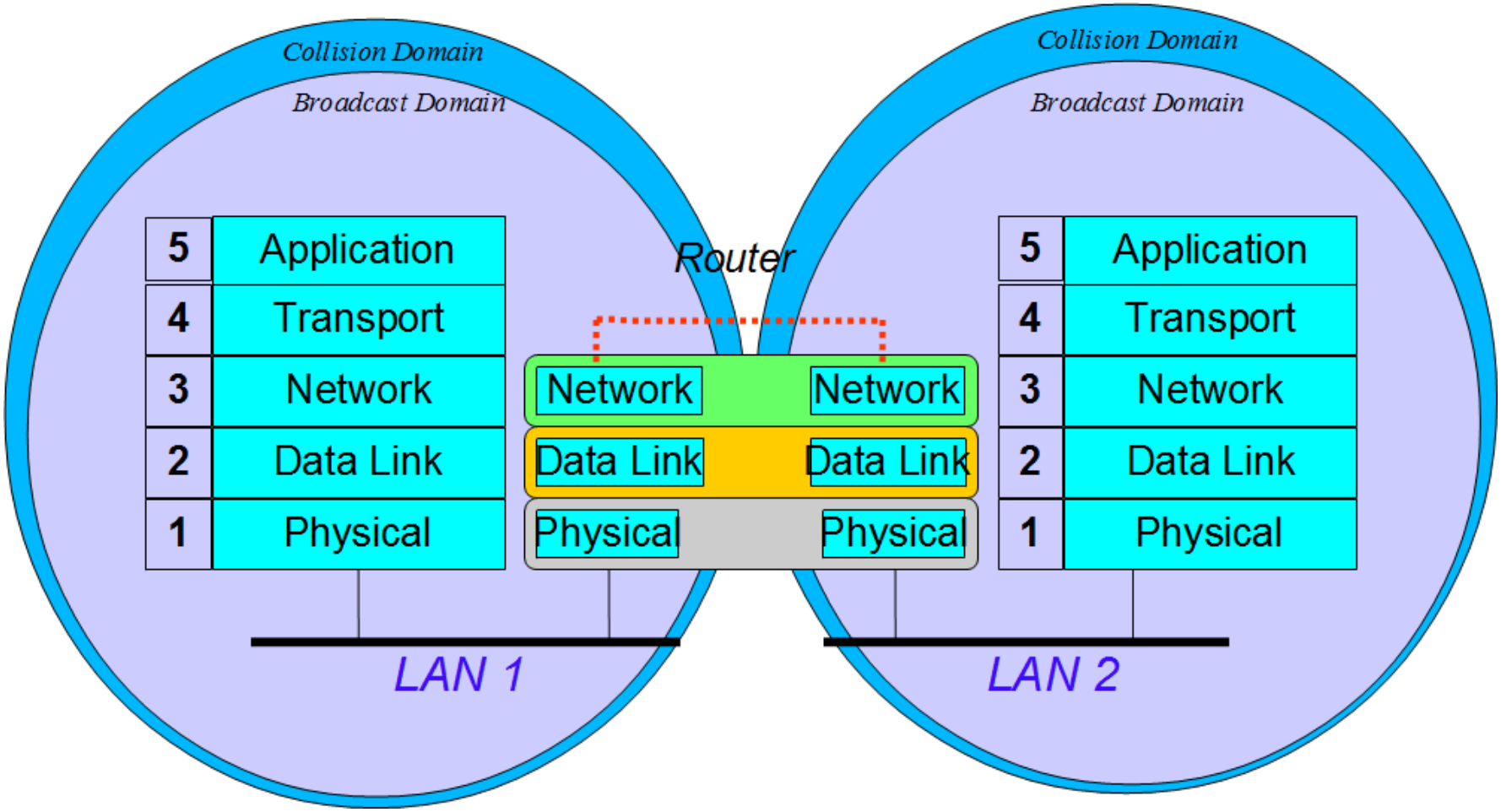
Routers

- Carry traffic to and from LANs
- Provide total flexibility in building a network topology
  - Routers do not have to build a spanning tree since they forward packets according to IP addresses which are structured and eventually discard packets that loop
- Routing decisions are taken based on IP addresses
  - IP network addresses that start with the same prefix belong to the same LAN
  - An IP system decides whether a destination is on its LAN by comparing its own address prefix with the destination's address prefix
    - If the source and destination address prefix match, both systems are on the same LAN. The next step is to discover the destination's link layer MAC address
    - An IP system discovers the MAC address of a destination on its LAN by broadcasting an Address Resolution Protocol (ARP) message that asks the owner of a specific IP address to respond

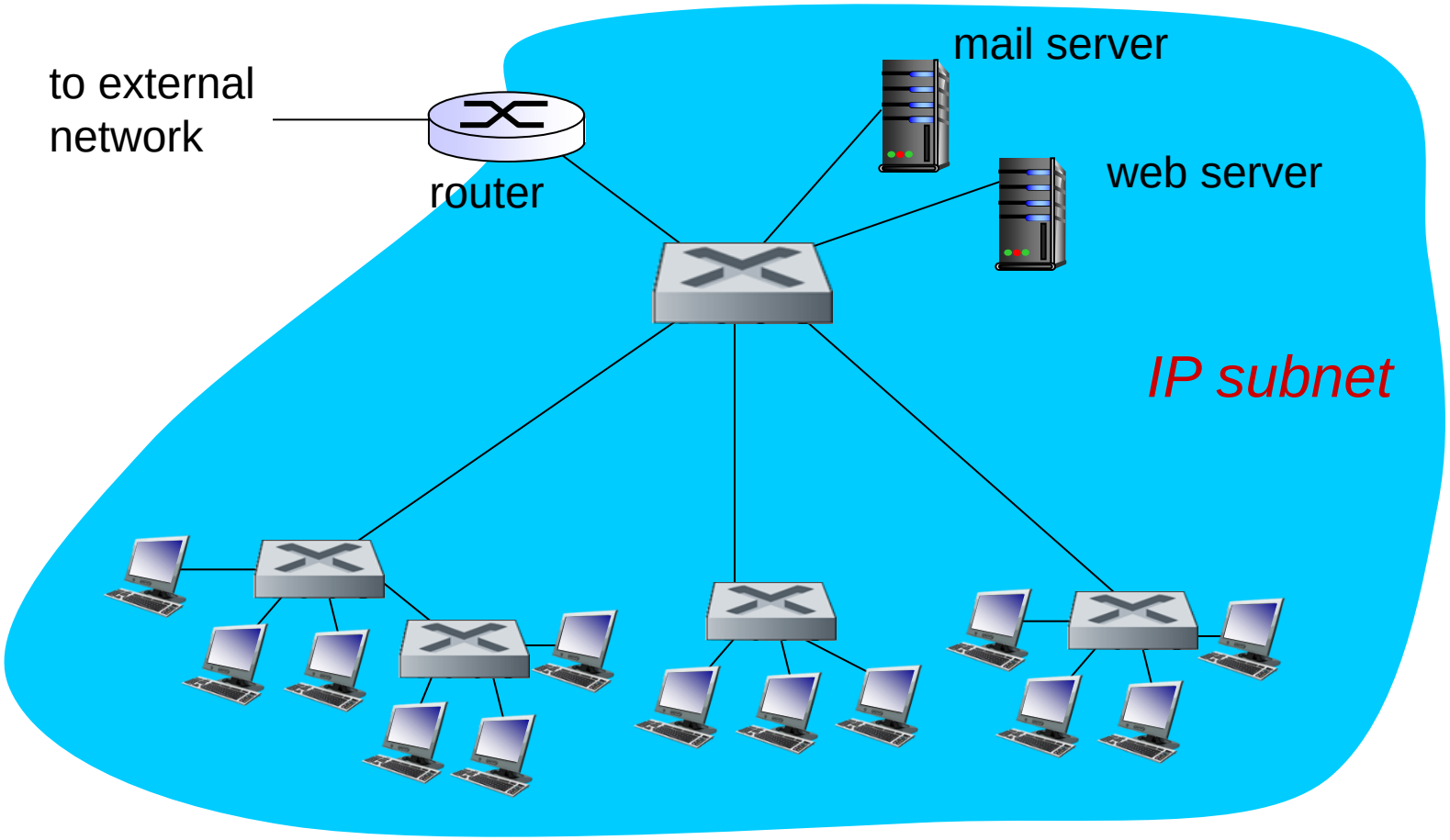
Routers

- Routers separate collision and broadcast domain
- Each router port is a broadcast domain
- Perform software processing of received packets:
  - Modify IP header:
    - Decrease TTL field, checksum calculation, fragmentation
  - Routers can generate ICMP packets
  - Routers run routing algorithms
  - Router generates a new frame and changes the source MAC address of the received frame by its own MAC address

Router


Collision Domain
Broadcast Domain
Collision Domain
Broadcast Domain
Router
5 Application
4 Transport
3 Network
2 Data Link
1 Physical
Network
Network
Data Link
Data Link
Physical
Physical
5 Application
4 Transport
3 Network
2 Data Link
1 Physical
LAN 1
LAN 2

Institutional network



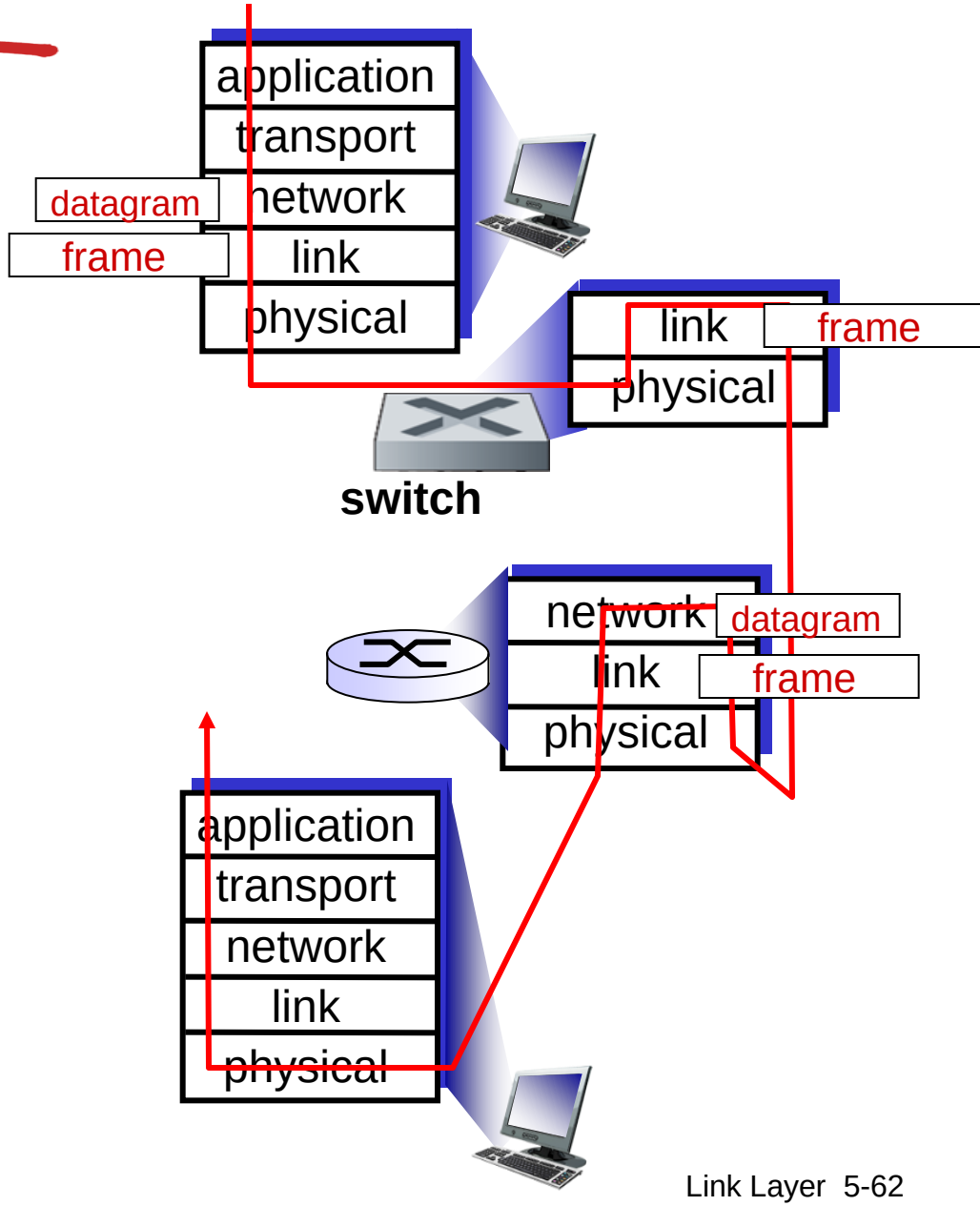
Switches vs. routers

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



Link layer, LANs: outline

5.1 introduction, services

5.2 error detection, correction

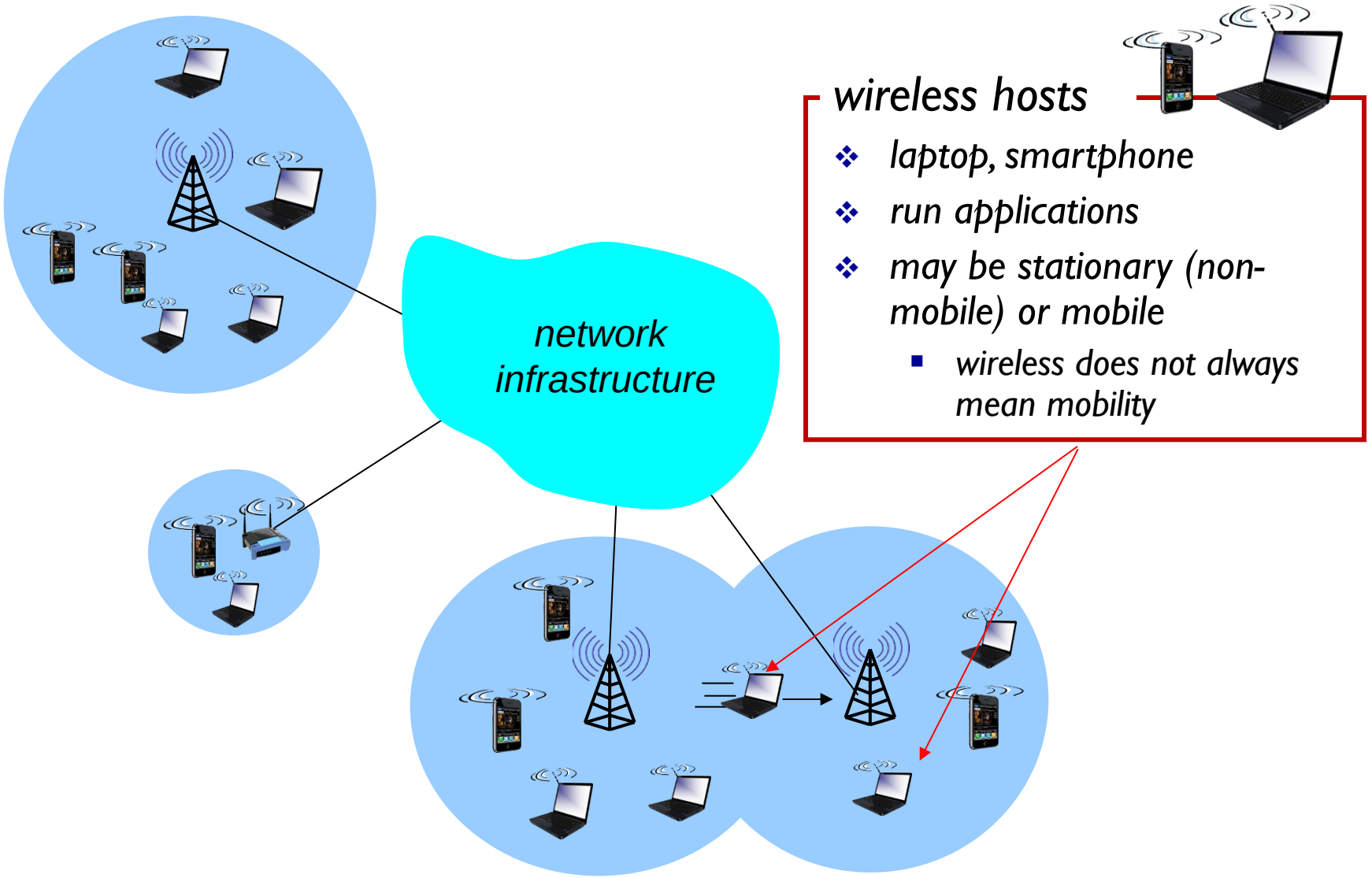
5.3 multiple access protocols

5.4 LANs

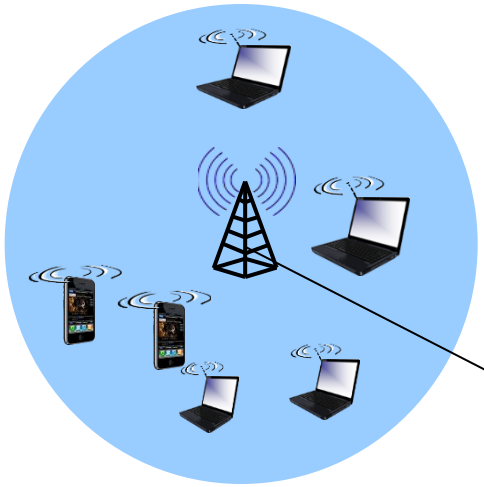
- addressing, ARP (Labs)
- Internetworking Devices
- WiFi
- Ethernet

5.5 a day in the life of a web request

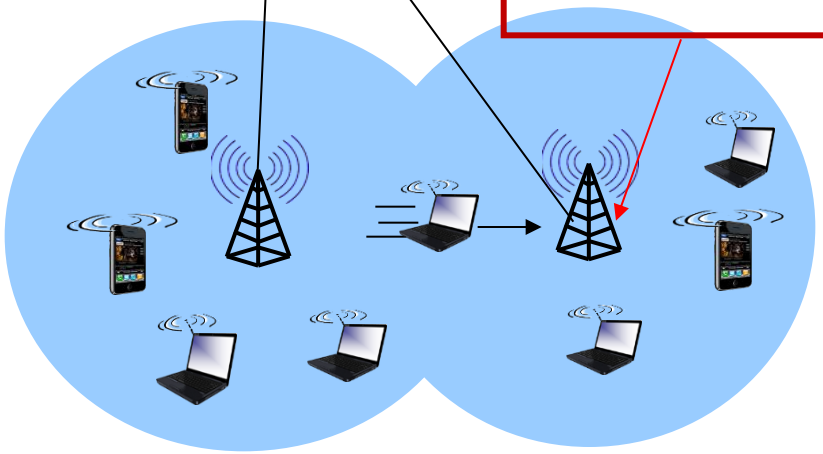
Elements of a wireless network



Elements of a wireless network



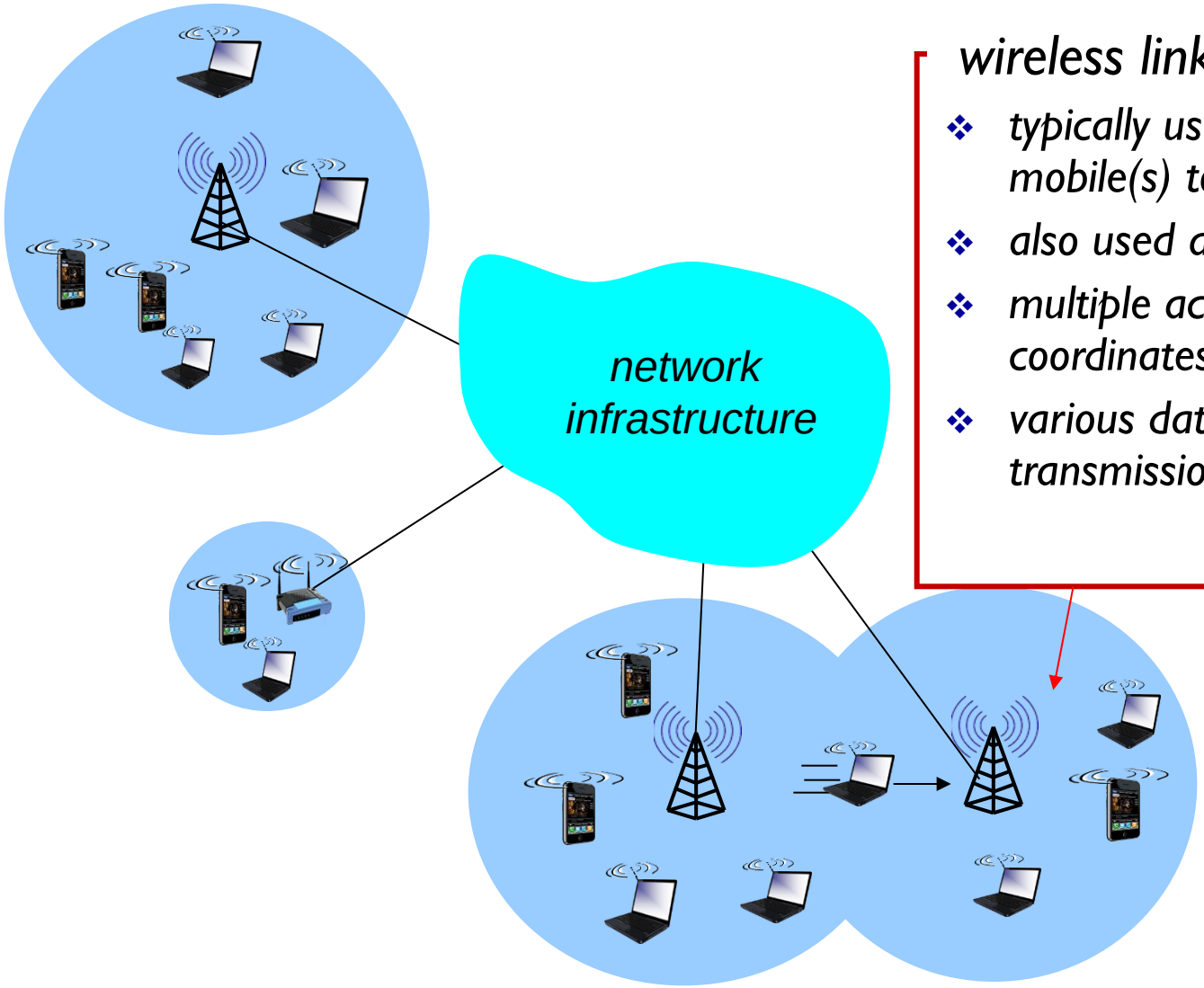
network infrastructure



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*

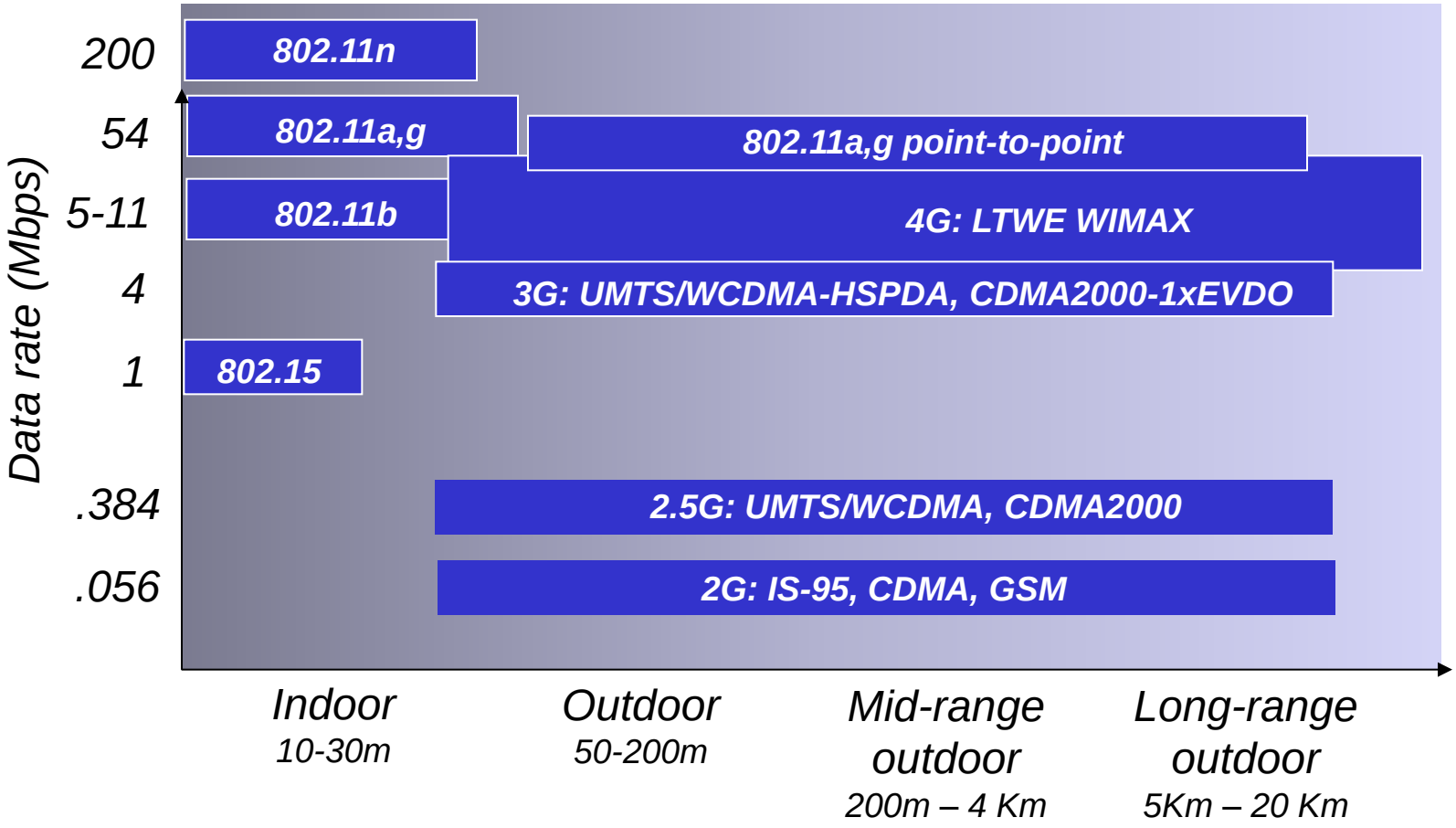
Elements of a wireless network



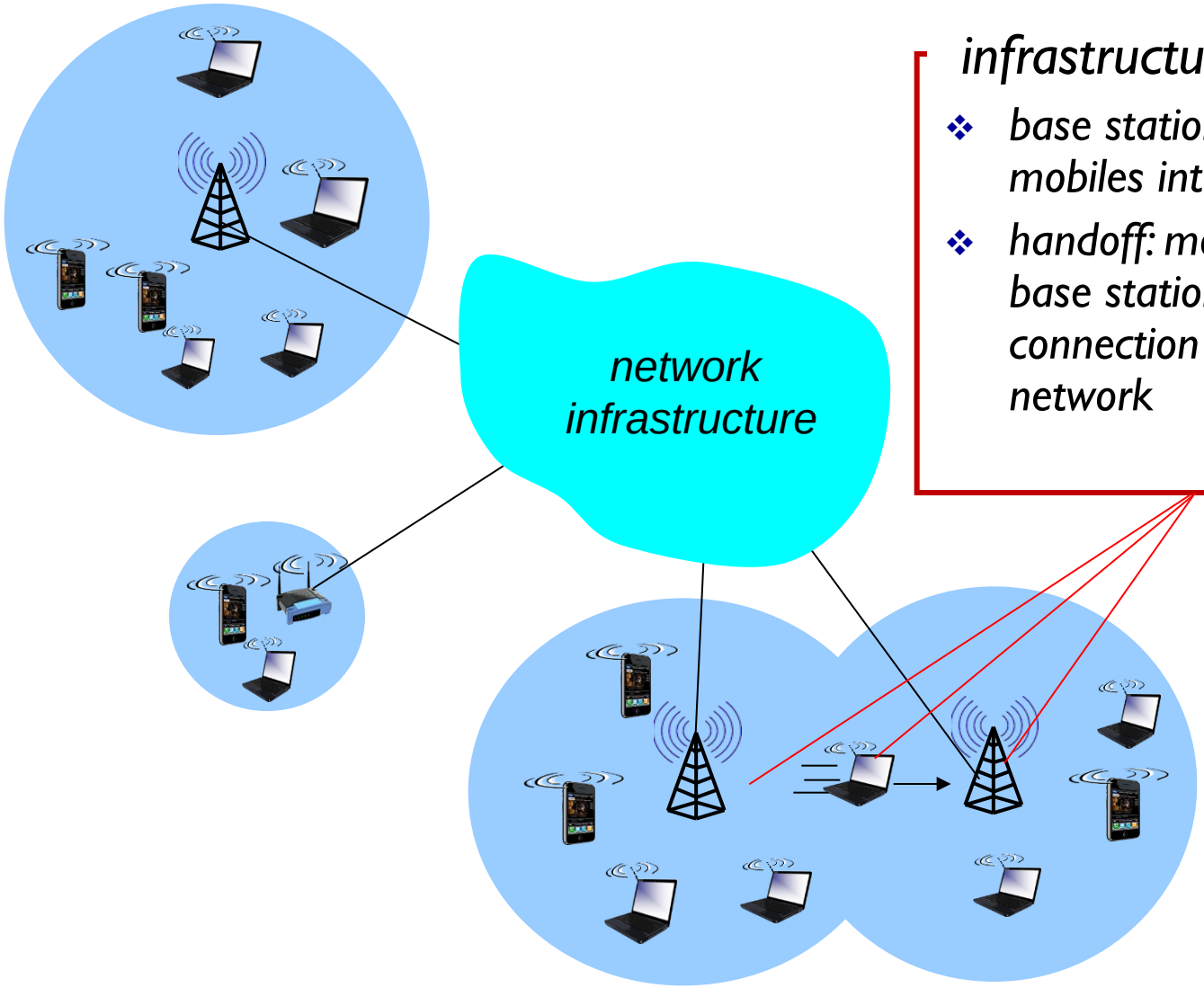
wireless link

- *typically used to connect mobile(s) to base station*
- *also used as backbone link*
- *multiple access protocol coordinates link access*
- *various data rates, transmission distance*

Characteristics of selected wireless links



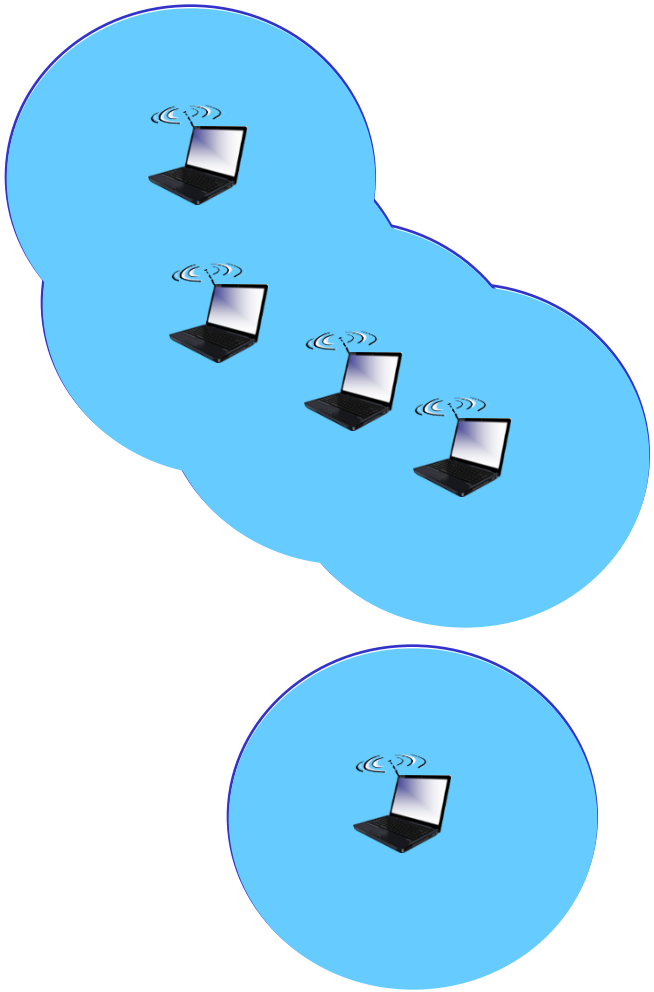
Elements of a wireless network



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: route among themselves*

Wireless network taxonomy

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

Wireless Link Characteristics (1)

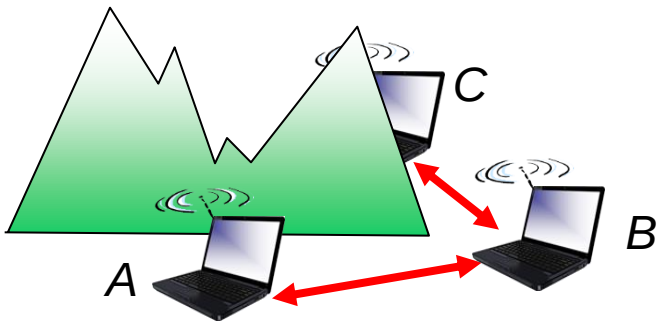
important differences from wired link ….

- *decreased signal strength:* radio signal attenuates as it propagates through matter (path loss)
- *interference from other sources:* standardized wireless network frequencies (e.g., 2.4 GHz) shared by other devices (e.g., phone); devices (motors) interfere as well
- *multipath propagation:* radio signal reflects off objects ground, arriving ad destination at slightly different times

…. make communication across (even a point to point) wireless link much more "difficult"

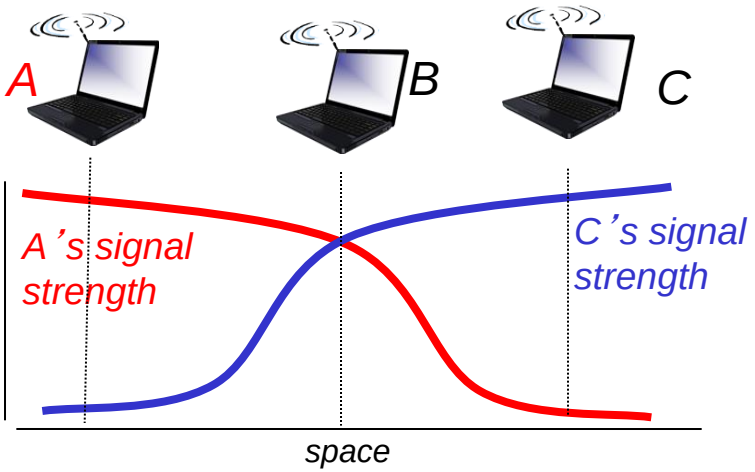
Wireless network characteristics

Multiple wireless senders and receivers create additional problems (beyond multiple access):



Hidden terminal problem

- *B, A hear each other*
- *B, C hear each other*
- *A, C can not hear each other means A, C unaware of their interference at B*



Signal attenuation:

- *B, A hear each other*
- *B, C hear each other*
- *A, C can not hear each other interfering at B*

IEEE 802.11 Wireless LAN

802.11b

- 2.4-5 GHz unlicensed spectrum
- up to 11 Mbps
- direct sequence spread spectrum (DSSS) in physical layer
  - all hosts use same chipping code

802.11a

- 5-6 GHz range
- up to 54 Mbps

802.11g

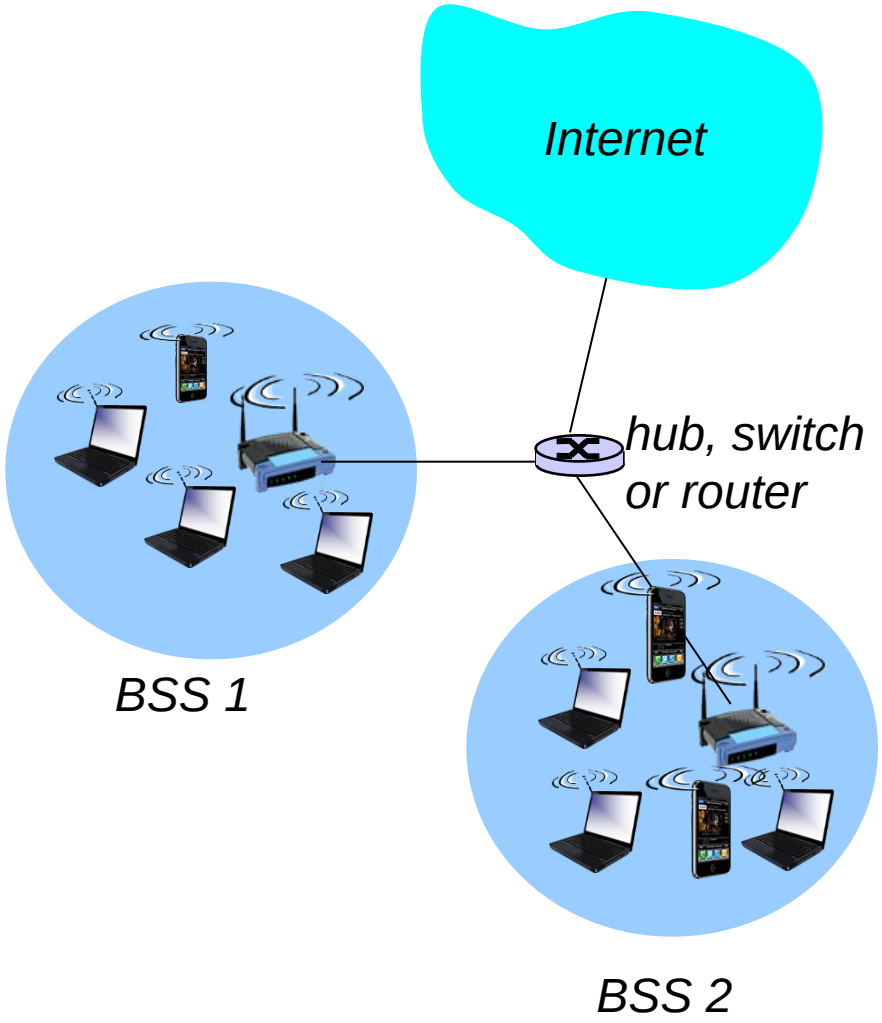
- 2.4-5 GHz range
- up to 54 Mbps

802.11n: multiple antennae

- 2.4-5 GHz range
- up to 200 Mbps

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

802.11 LAN architecture

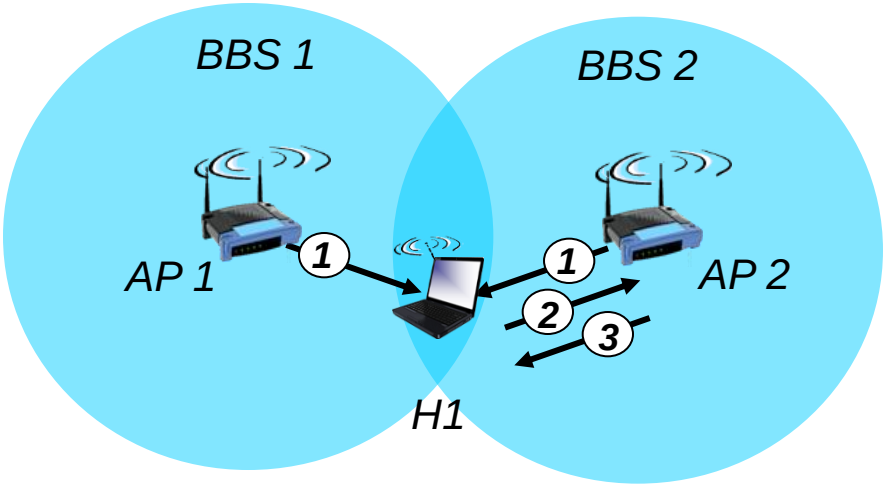


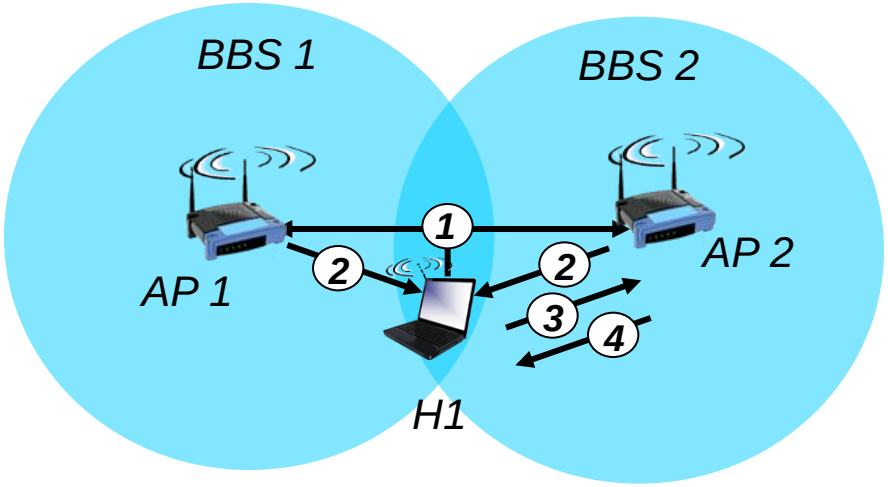
- *wireless host communicates with base station*
  - *base station = access point (AP)*
- *Basic Service Set (BSS) (aka "cell") in infrastructure mode contains:*
  - *wireless hosts*
  - *access point (AP): base station*

802.11: Channels, association

- 802.11b: 2.4GHz-2.485GHz spectrum divided into 11 channels at different frequencies
  - AP admin chooses frequency for AP
  - interference possible: channel can be same as that chosen by neighboring AP!
- 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
  - may perform authentication [Chapter 8]
  - will typically run DHCP to get IP address in AP's subnet

802.11: passive/active 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

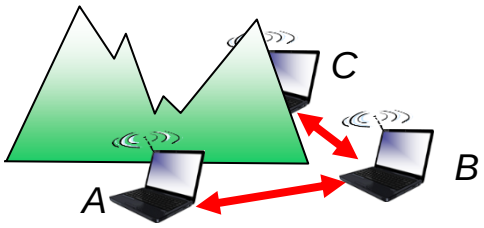
(2) Probe Response frames sent from APs

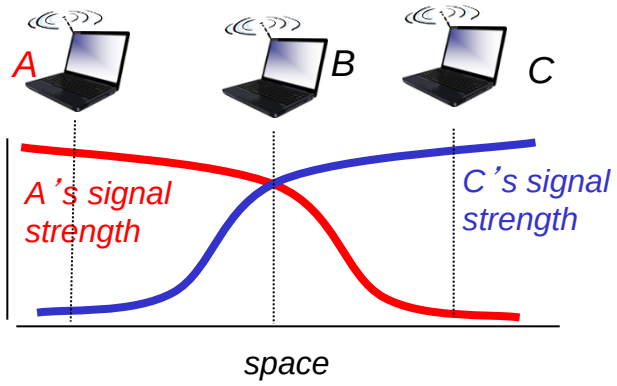
(3) Association Request frame sent: H1 to selected AP

(4) Association Response frame sent from selected AP to H1

IEEE 802.11: multiple access

- avoid collisions: 2^+ nodes transmitting at same time
- 802.11: CSMA - sense before transmitting
  - don't collide with ongoing transmission by other node
- 802.11: *no* collision detection!
  - difficult to receive (sense collisions) when transmitting due to weak received signals (fading)
  - can't sense all collisions in any case: hidden terminal, fading
  - goal: *avoid collisions:* CSMA/C(ollision)A(voidance)





IEEE 802.11 MAC Protocol: CSMA/CA

802.11 sender

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

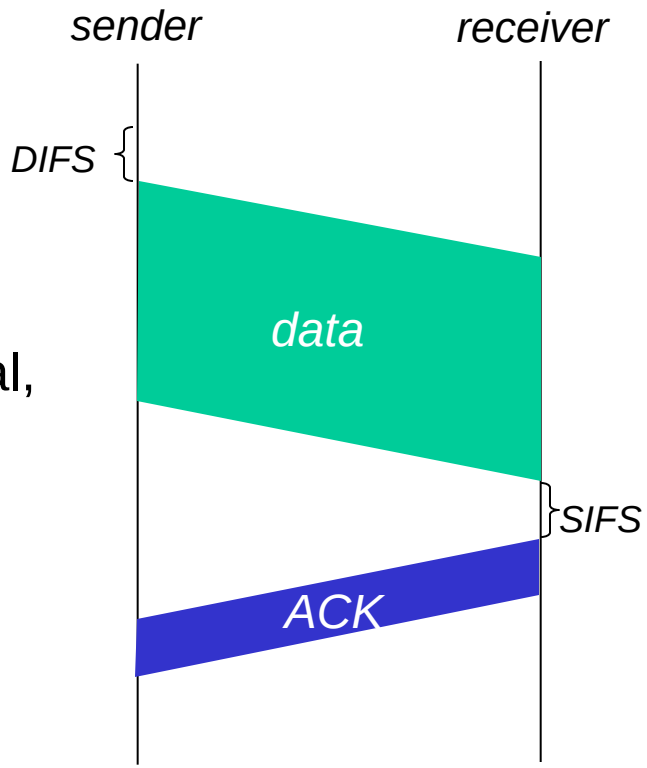
transmit when timer expires

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)



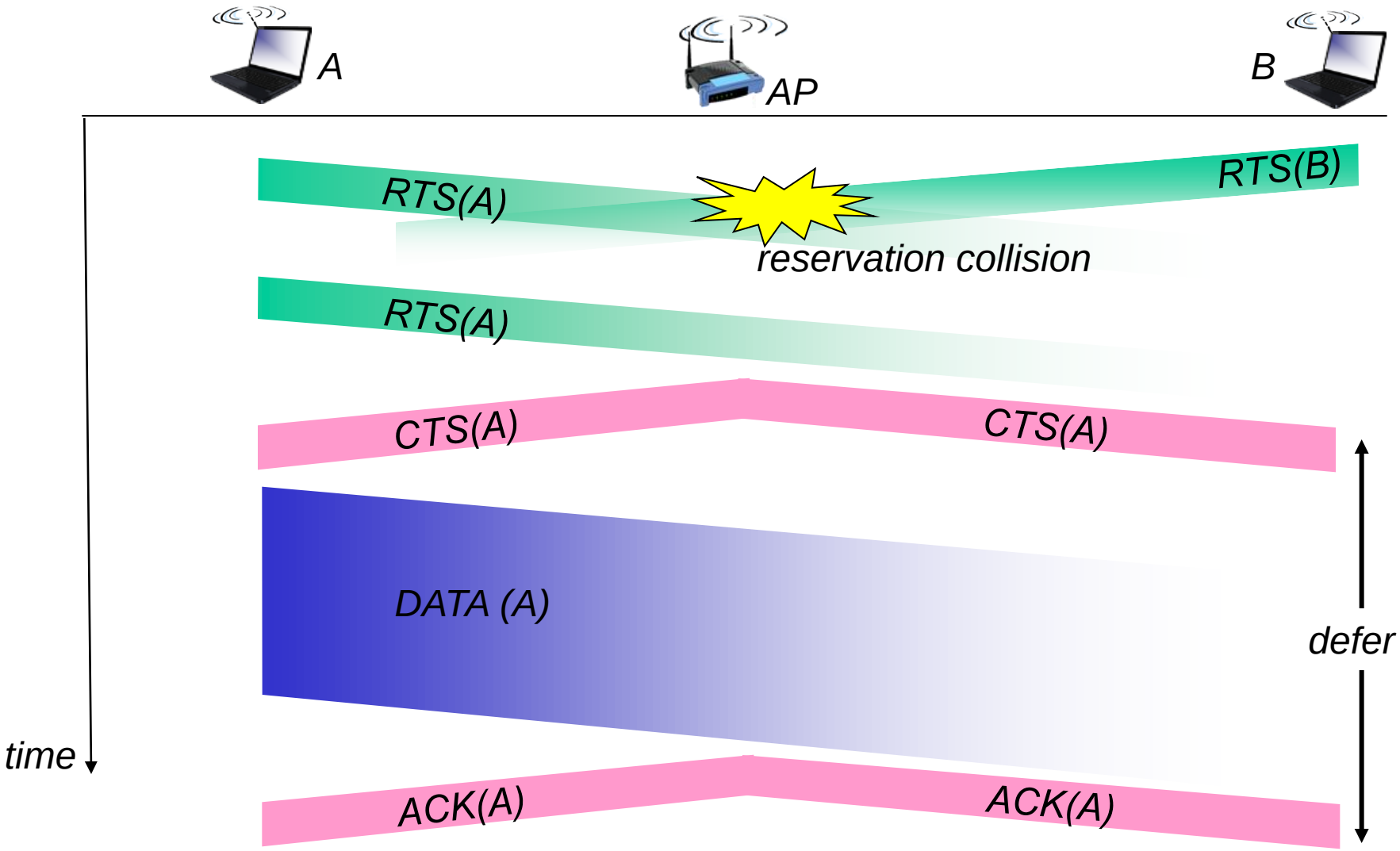
Avoiding collisions (more)

idea: allow sender to "reserve" channel rather than random access of data frames: avoid collisions of long data frames

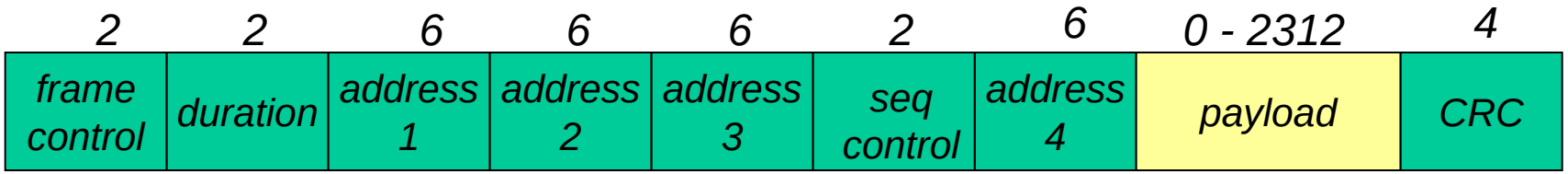
- sender first transmits *small* request-to-send (RTS) packets to BS using CSMA
  - RTSs may still collide with each other (but they're short)
- BS broadcasts clear-to-send CTS in response to RTS
- CTS heard by all nodes
  - sender transmits data frame
  - other stations defer transmissions

avoid data frame collisions completely using small reservation packets!

Collision Avoidance: RTS-CTS exchange



802.11 frame: addressing



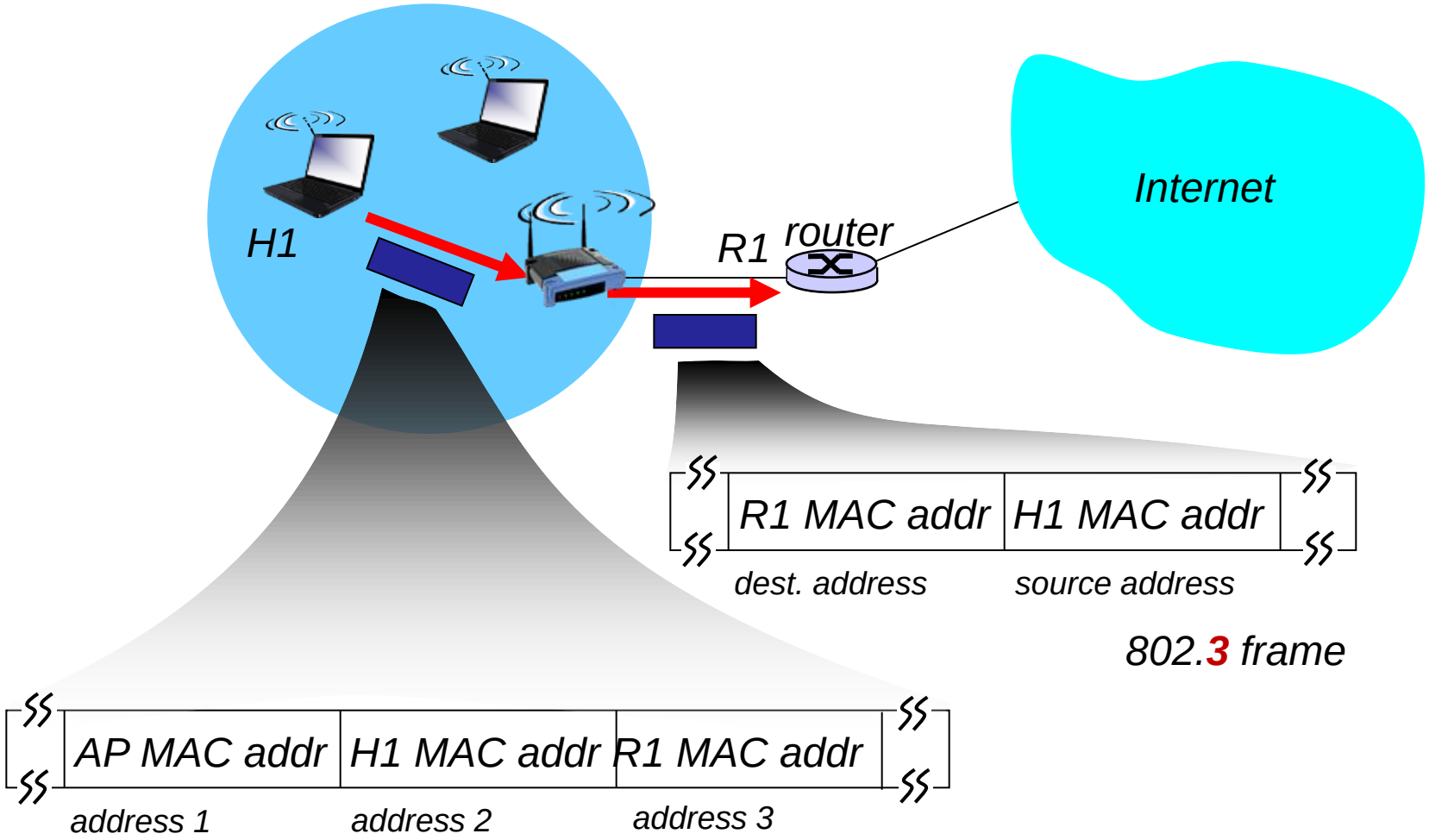
Address 1: MAC address of wireless host or AP to receive this frame

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

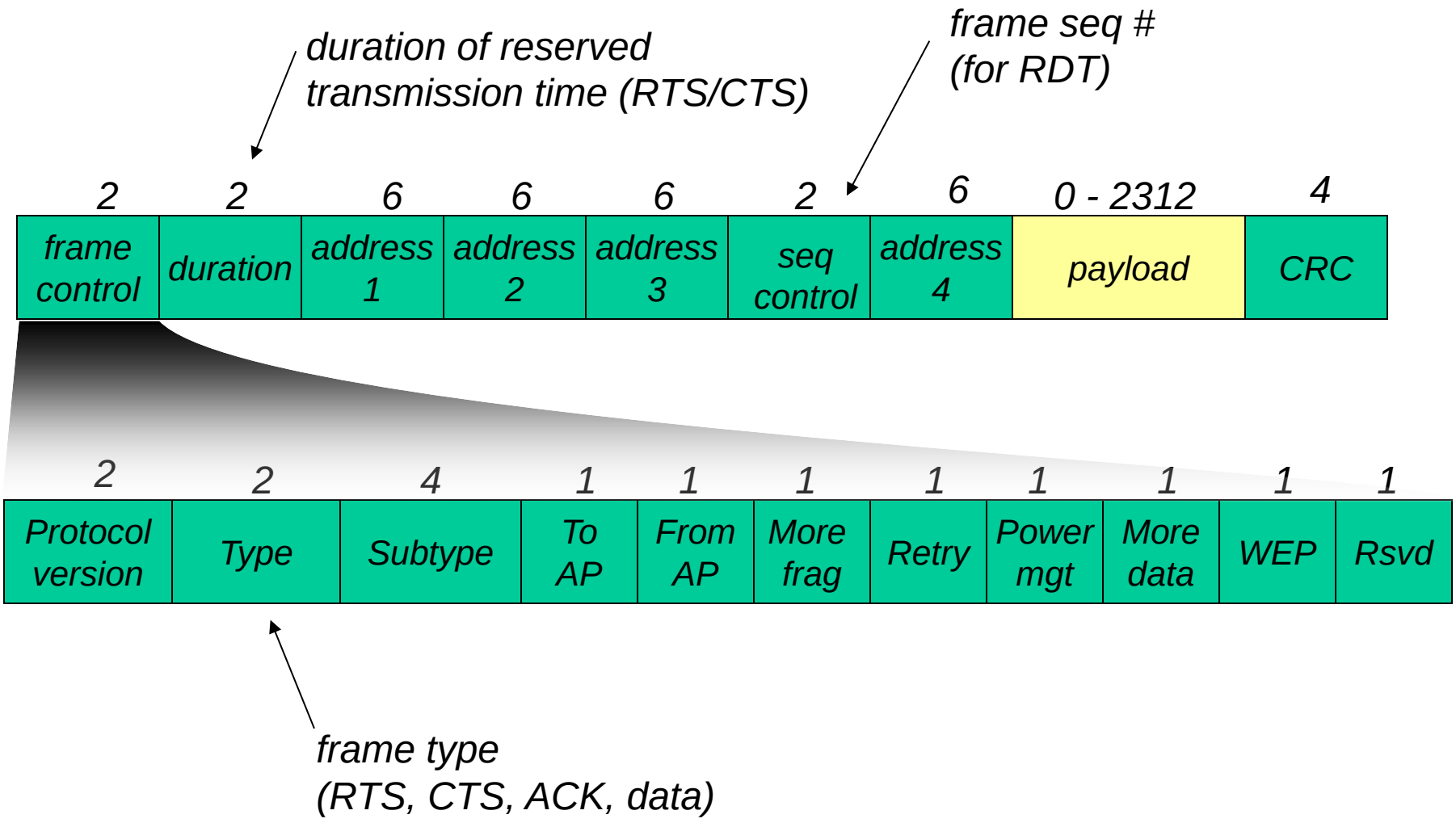
Address 3: MAC address of router interface to which AP is attached

Address 4: used only in ad hoc mode

802.11 frame: addressing



802.11 frame: more



802.11 Addresses

To DS	From DS	Address 1	Address 2	Address 3	Address 4
0	0	RA = DA	TA = SA	BSSID	N/A
0	1	RA = DA	TA = BSSID	SA	N/A
1	0	RA = BSSID	TA = SA	DA	N/A
1	1	RA	TA	DA	SA

RA: Receiver Address;
TA: Transmitter Address;
DA: Destination Address;
SA: Source Address; BSSID: BSS Identifier . The BSSID is the MAC address of the AP (access point) in the case of an infrastructure BSS, while it is randomly chosen in the case of an IBSS ((Independent Basic Service Set, also called ad-hoc networks) *by the station initializing the IBSS;*
WDS: Wireless Distribution System
DS: Distribution System

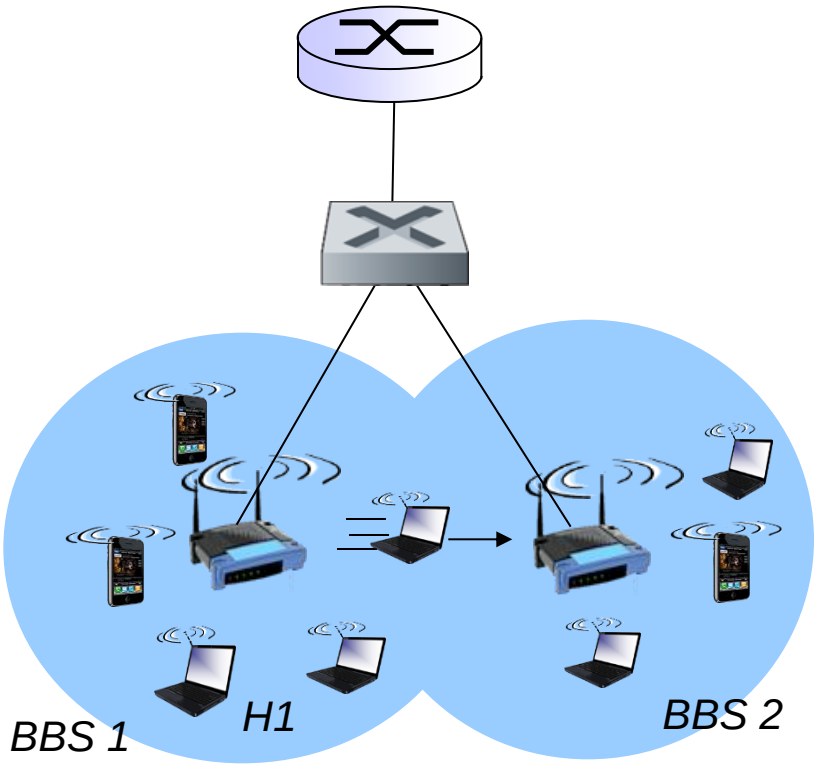
- ***To DS = 0 & From DS = 0****, such as a transmission within an IBSS: Address 1 = RA = DA and Address 2 = TA = SA, while Address 3 = BSSID, identifying the BSS. This combination is used for transmissions in an IBSS as well as for management and control type frames in an infrastructure BSS. It is also used for a direct link in an infrastructure BSS running IEEE 802.11e.*
- ***To DS = 0 &From DS = 1*** *(i.e., a downlink transmission): Address 1 = RA = DA and Address 2 = TA = BSSID, while Address 3 = SA is the MAC address of the source in the subnet, which could be either a router or another non-AP station.*
- ***To DS = 1 &From DS = 0*** *(i.e., an uplink transmission): Address 1 = RA = BSSID and Address 2 = TA = SA, while Address 3 = DA is the MAC address of the destination in the subnet, which could be either a router or another non-AP station.*
- ***To DS = 1 & From DS = 1*** *(i.e., a transmission within a WDS): Address 1 = RA and Address 2 = TA, while Address 3 = DA and Address 4 = SA. Note that for a transmission within a WDS (e.g., a wireless system connecting multiple APs), the destination and the source should be different from the receiver and the transmitter, respectively. Note that in WDS, both transmitter and receiver should be basically APs.*

Types of Frames

- *Control Frames*
  - *RTS/CTS/ACK*
  - *CF-Poll/CF-End*
- *Management Frames*
  - ***Beacons** : beacon frame announce the existence of a network, used in passive scanning*
  - ***Probe Request/Response**: used in active scanning*
  - ***Association Request/Response***
  - ***Dissociation/Reassociation***
  - ***Authentication/Deauthentication***
- *Data Frames*

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. 5): switch will see frame from H1 and "remember" which switch port can be used to reach H1



Link layer, LANs: outline

5.1 introduction, services

5.2 error detection, correction

5.3 multiple access protocols

5.4 LANs

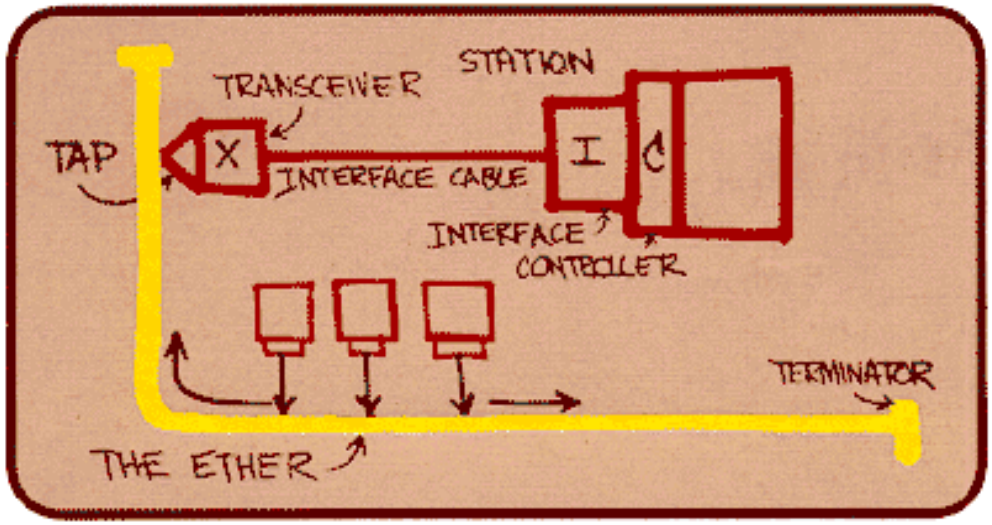
- addressing, ARP (Labs)
- Internetworking Devices
- WiFi
- Ethernet

5.5 a day in the life of a web request

Ethernet (IEEE 802.3 standards)

"dominant" wired LAN technology:

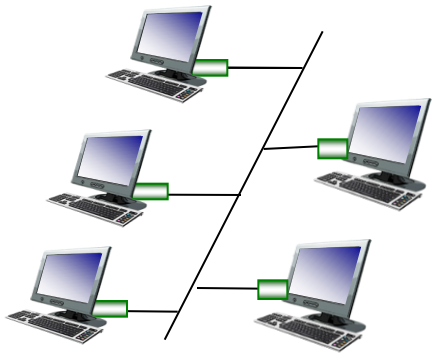
- cheap $20 for NIC
- first widely used LAN technology
- simpler, cheaper than token LANs and ATM
- kept up with speed race: 10 Mbps – 10 Gbps



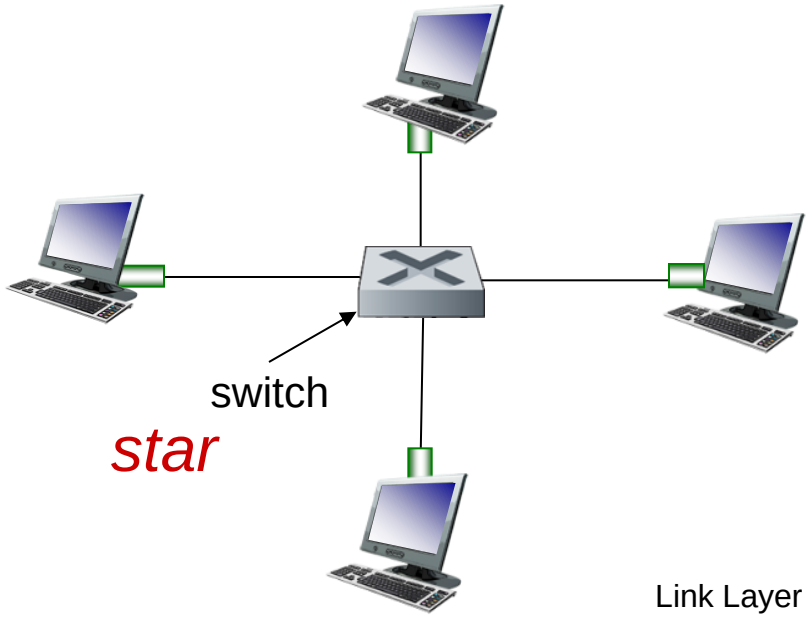
Metcalfe's Ethernet sketch

Ethernet: physical topology

- *bus:* popular through mid 90s
  - all nodes in same collision domain (can collide with each other)
- *star:* prevails today
  - active *switch* in center
  - each "spoke" runs a (separate) Ethernet protocol (nodes do not collide with each other)



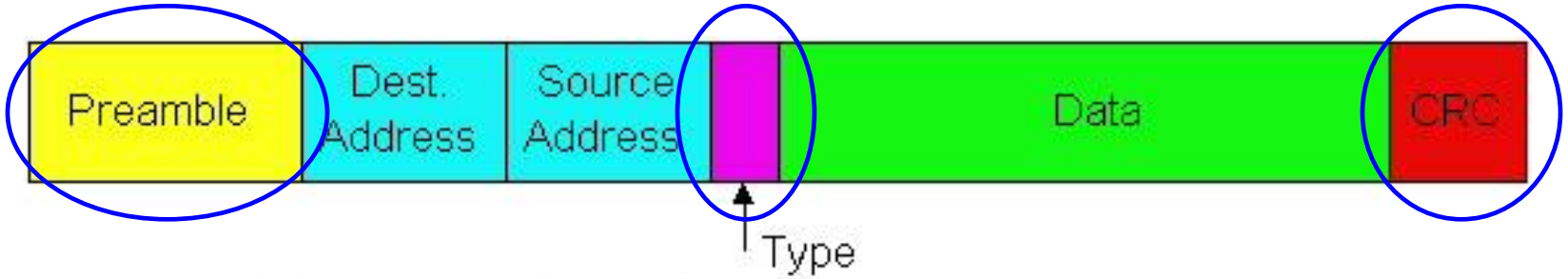
bus: coaxial cable



star

802.3 frame structure

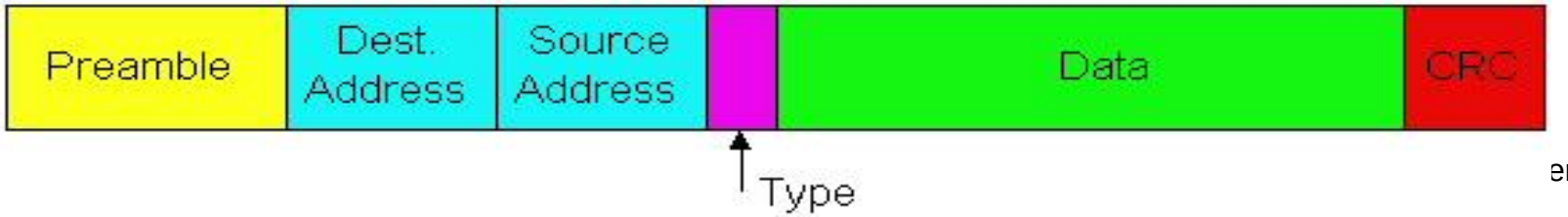
sending adapter encapsulates IP datagram (or other network layer protocol packet) in *Ethernet frame*



- *Preamble*: synchronization
  - Seven bytes with pattern 10101010, followed by one byte with pattern 10101011
  - Used to synchronize receiver & sender
- *Type*: indicates the higher layer protocol
  - Usually IP (but also Novell IPX, AppleTalk, …)
- *CRC*: cyclic redundancy check
  - Receiver checks & simply drops frames with errors

802.3 frame structure

- *Addresses*: 48-bit source and destination MAC addresses
  - The source address is the unicast address of the station that sent the frame
  - Receiver's adaptor passes frame to network-level protocol
    - If destination address matches the adaptor's
    - Or the destination address is the broadcast address (ff:ff:ff:ff:ff:ff)
    - Or the destination address is a multicast group receiver belongs to
    - Or the adaptor is in promiscuous mode
  - Addresses are globally unique
- *Data*:
  - Maximum: 1,500 bytes
  - Minimum: 46 bytes (+14 bytes header + 4 byte trailer = 512 bits)

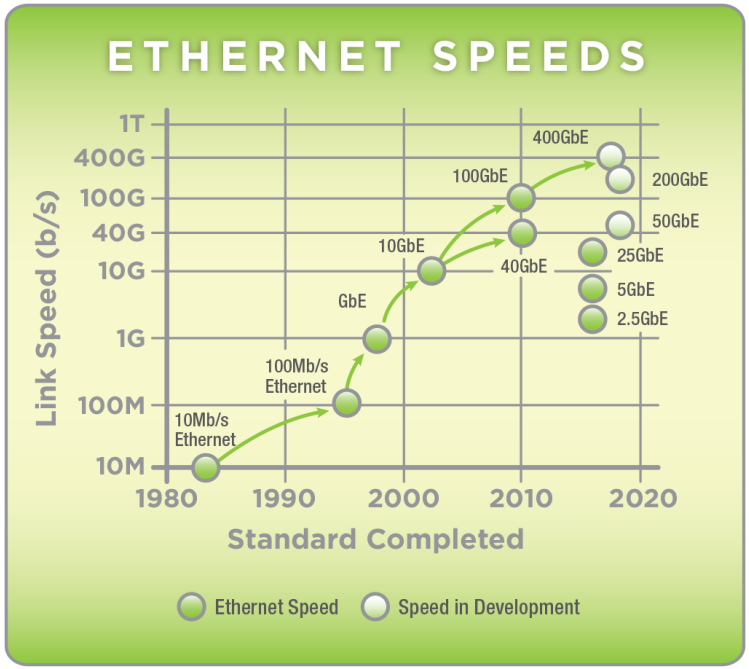


Ethernet: unreliable, connectionless

- *connectionless:* no handshaking between sending and receiving NICs
- *unreliable:* receiving NIC doesnt send acks or nacks 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 & physical layers

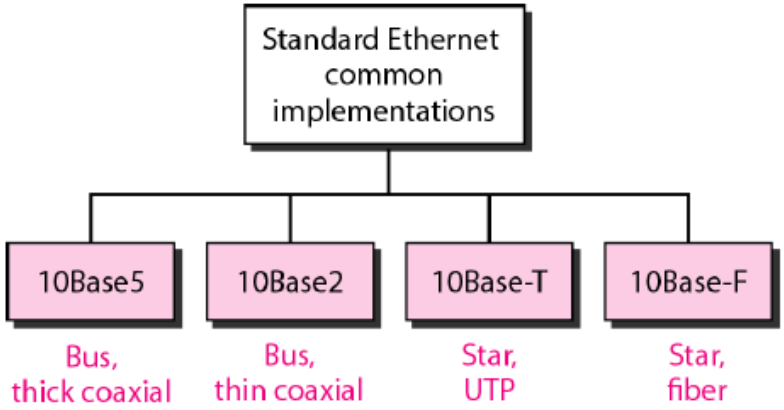
- *many* different Ethernet standards
  - Common characteristics
    - Distributed media access control (when required)
      - Use CSMA / CD as MAC protocol
    - Best-effort delivery Service
      - Unreliable delivery service
    - Same frame format
  - different speeds

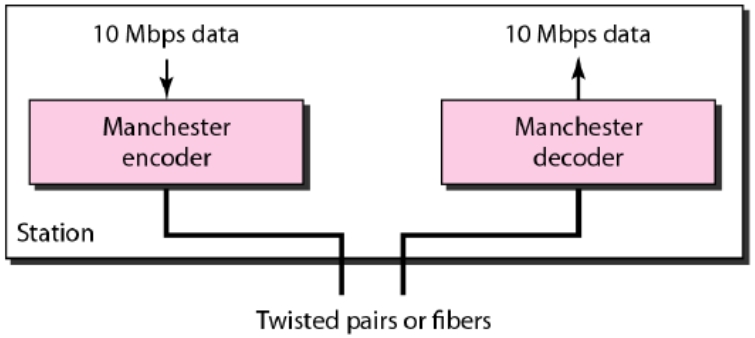


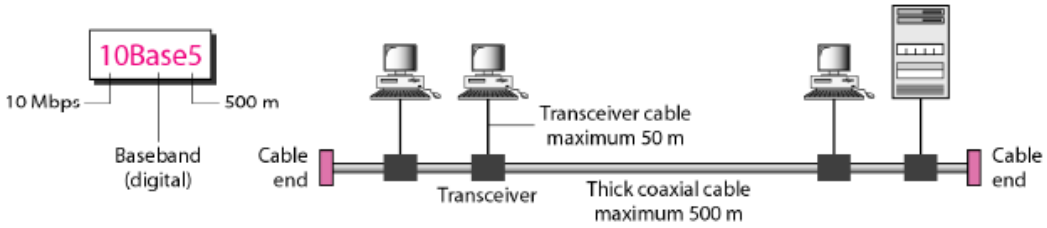
802.3 Ethernet standards: link & physical layers

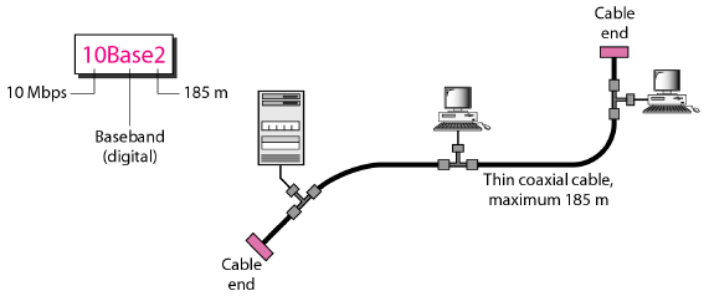
10Mbps Ethernet : IEEE 802.3

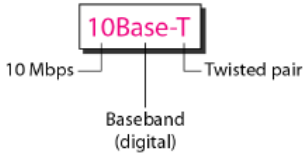
- different physical layer media: fiber, cable

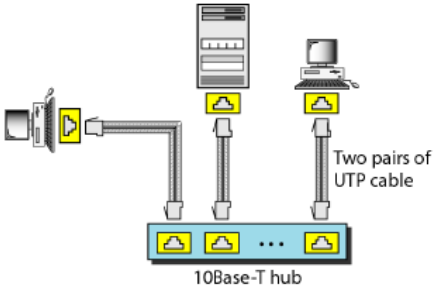


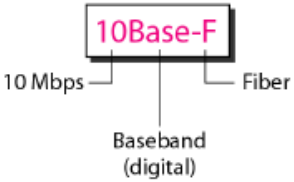


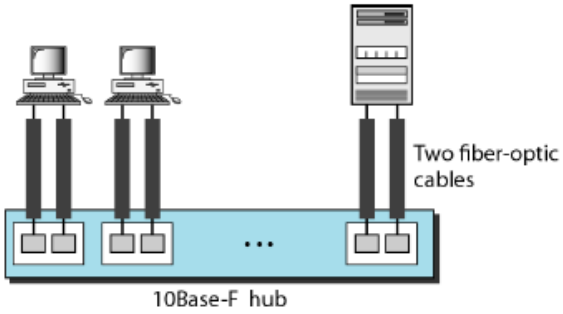






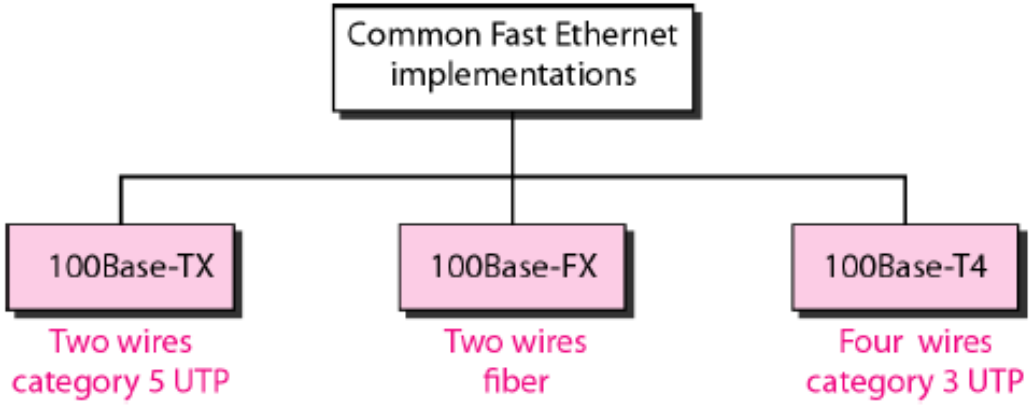


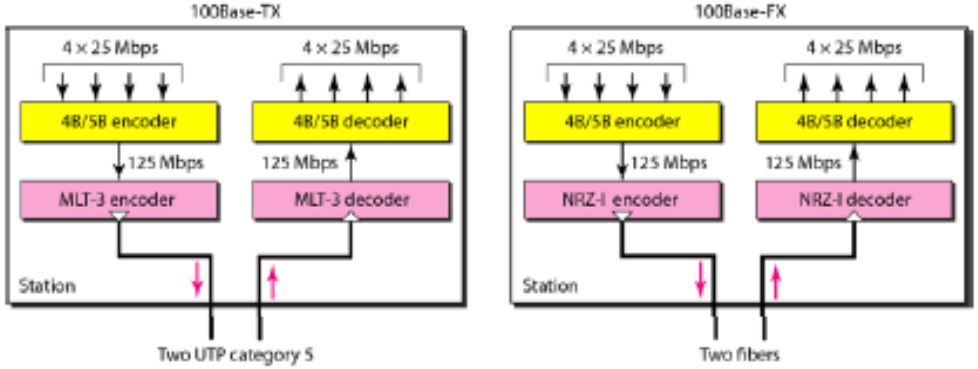


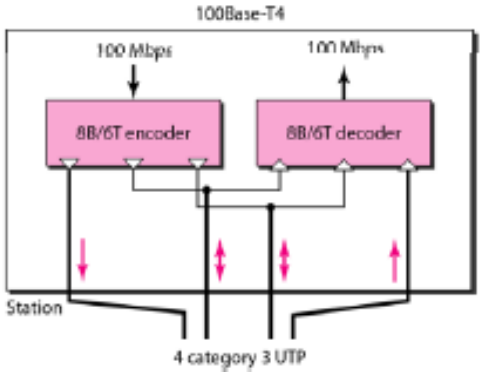


802.3 Ethernet standards: link & physical layers

100Mbps Ethernet : IEEE 802.3u Fast Ethernet

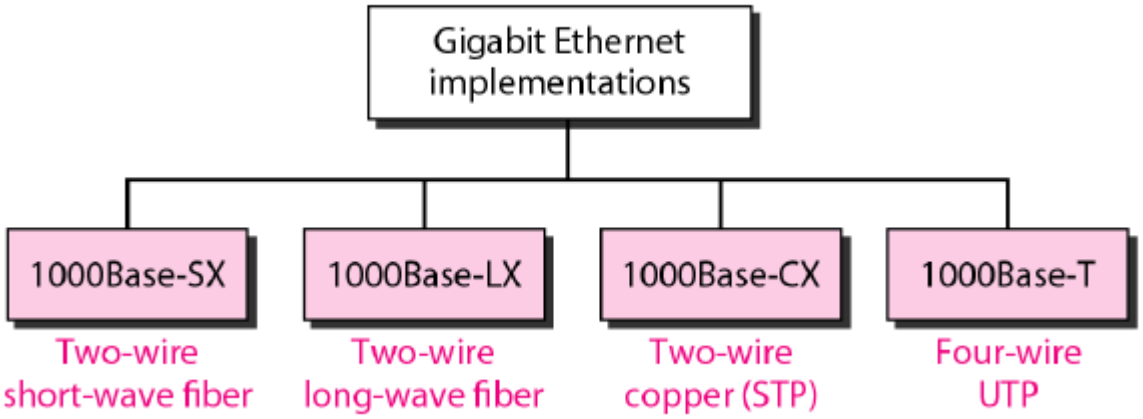


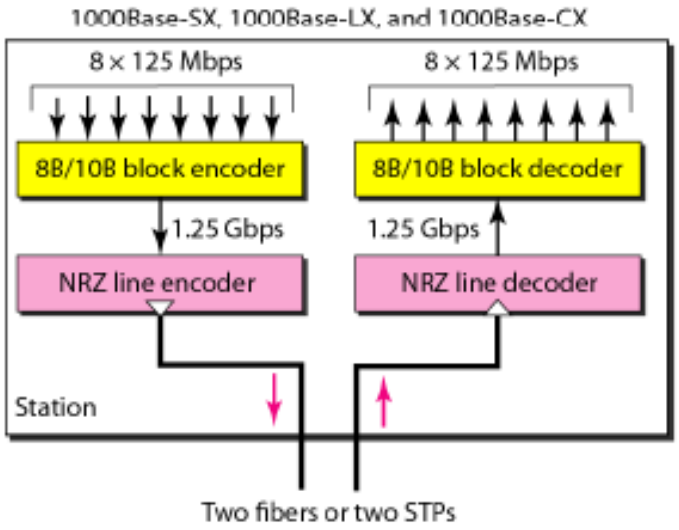


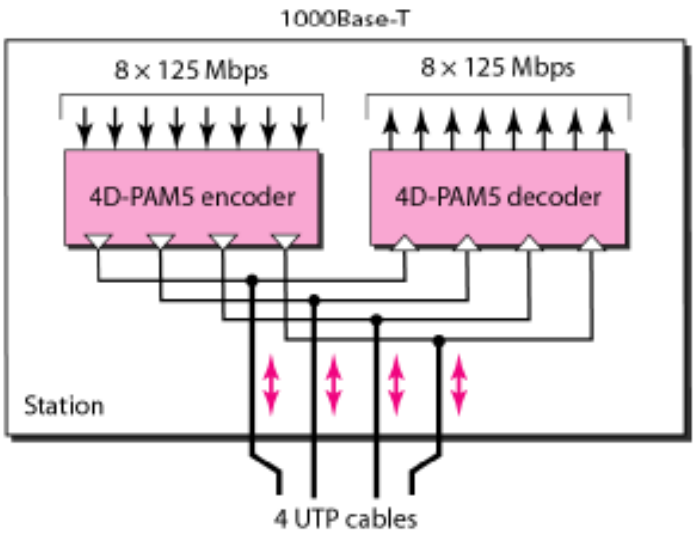


802.3 Ethernet standards: link & physical layers

1000Mbps Ethernet : IEEE 802.3z Gigabit Ethernet







Link layer, LANs: outline

5.1 introduction, services

5.2 error detection, correction

5.3 multiple access protocols

5.4 LANs

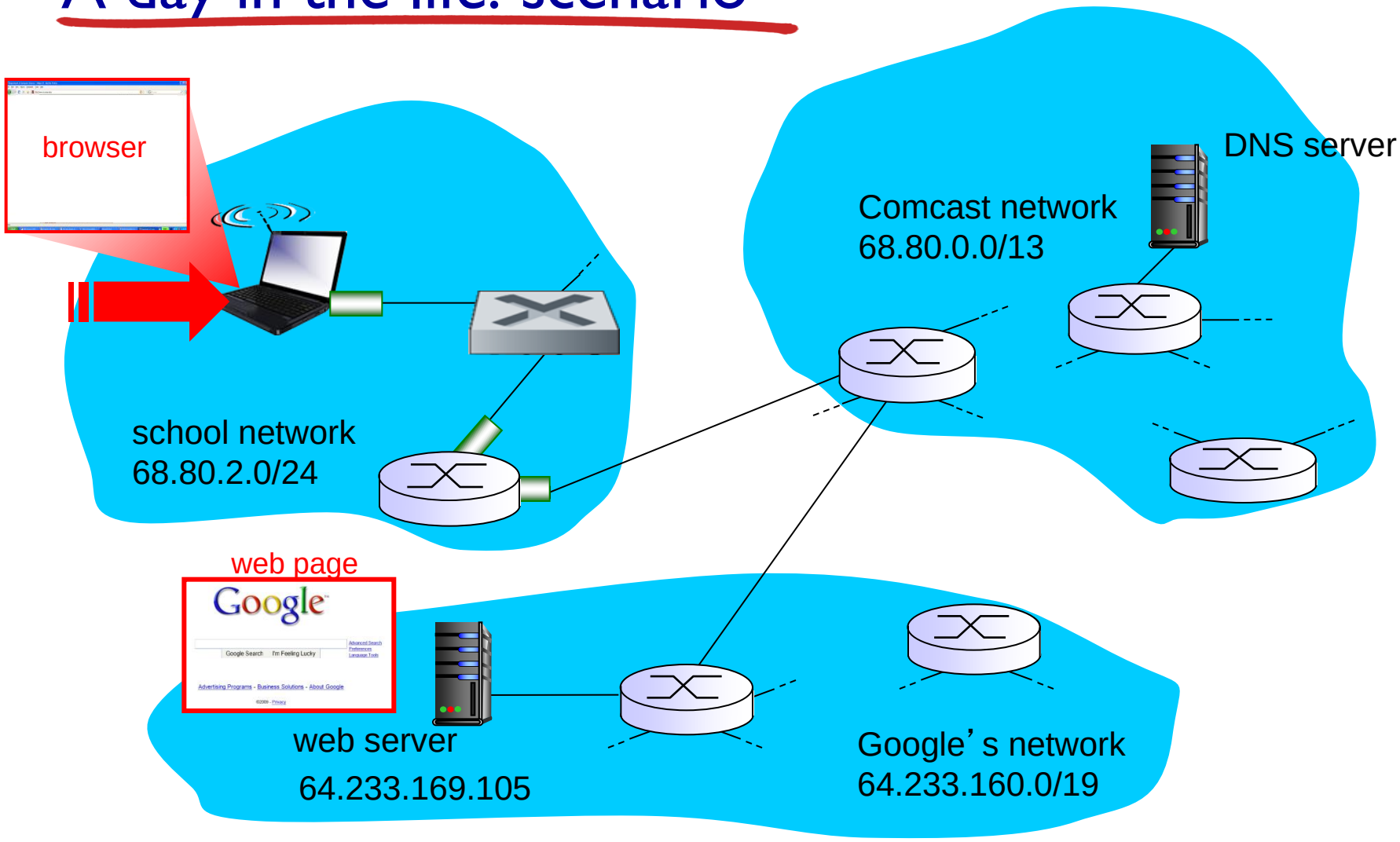
- addressing, ARP (Labs)
- Internetworking Devices
- WiFi
- Ethernet

5.5 a day in the life of a web request

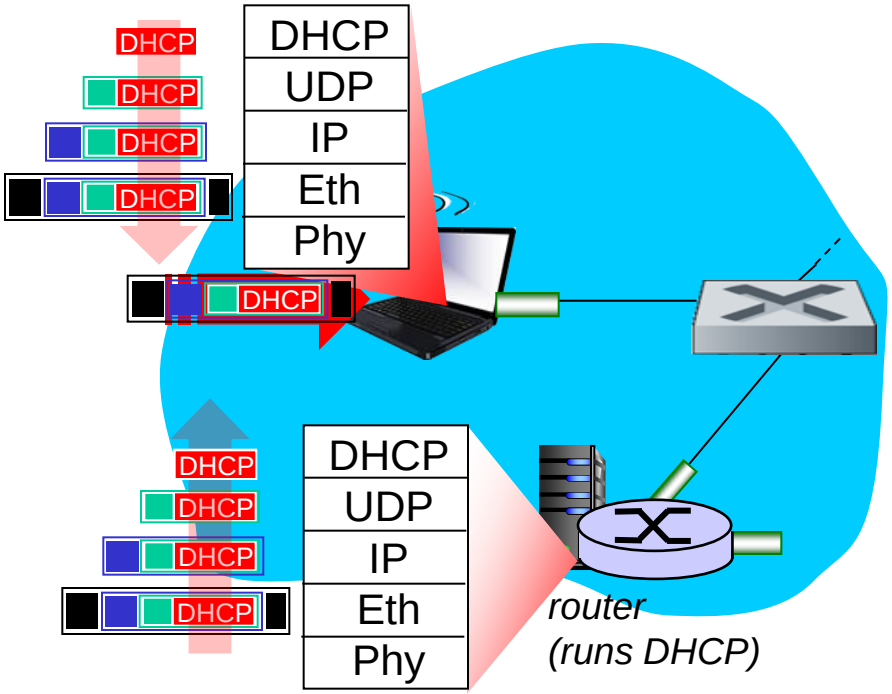
Synthesis: a day in the life of a web request

- journey down protocol stack 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

A day in the life: scenario

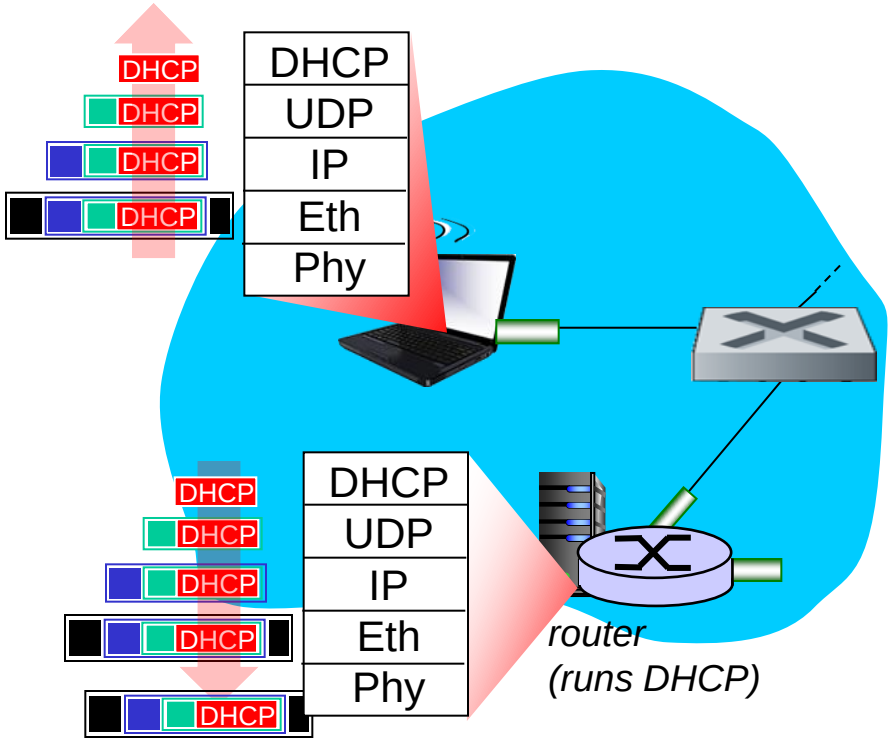


A day in the life… connecting to the Internet



- 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

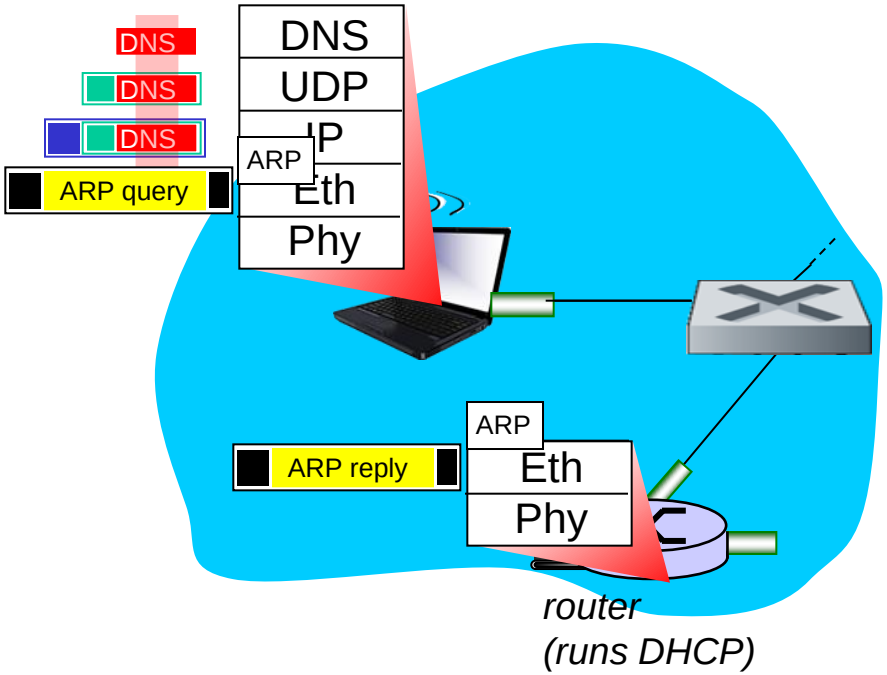
A day in the life… connecting to the Internet



- 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

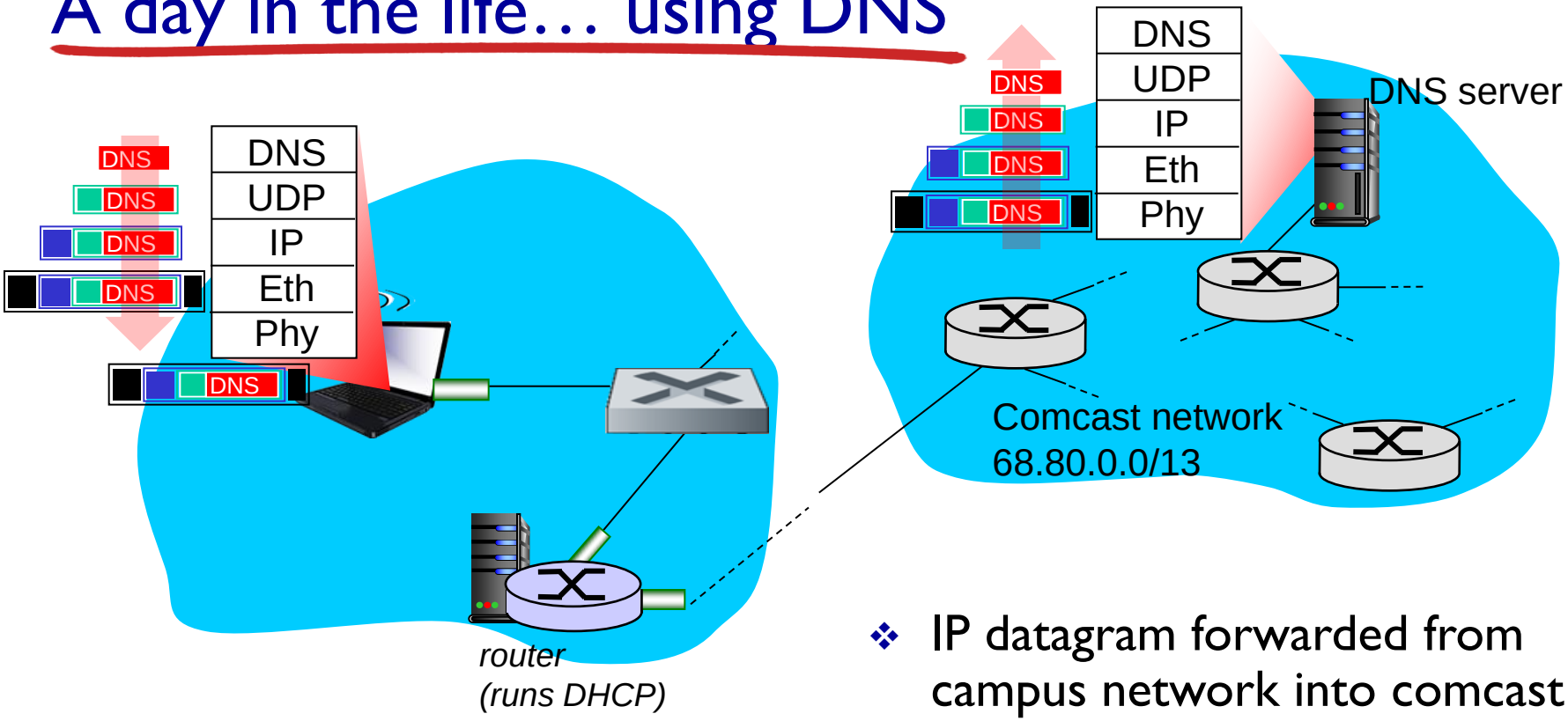
Client now has IP address, knows name & addr of DNS server, IP address of its first-hop router

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



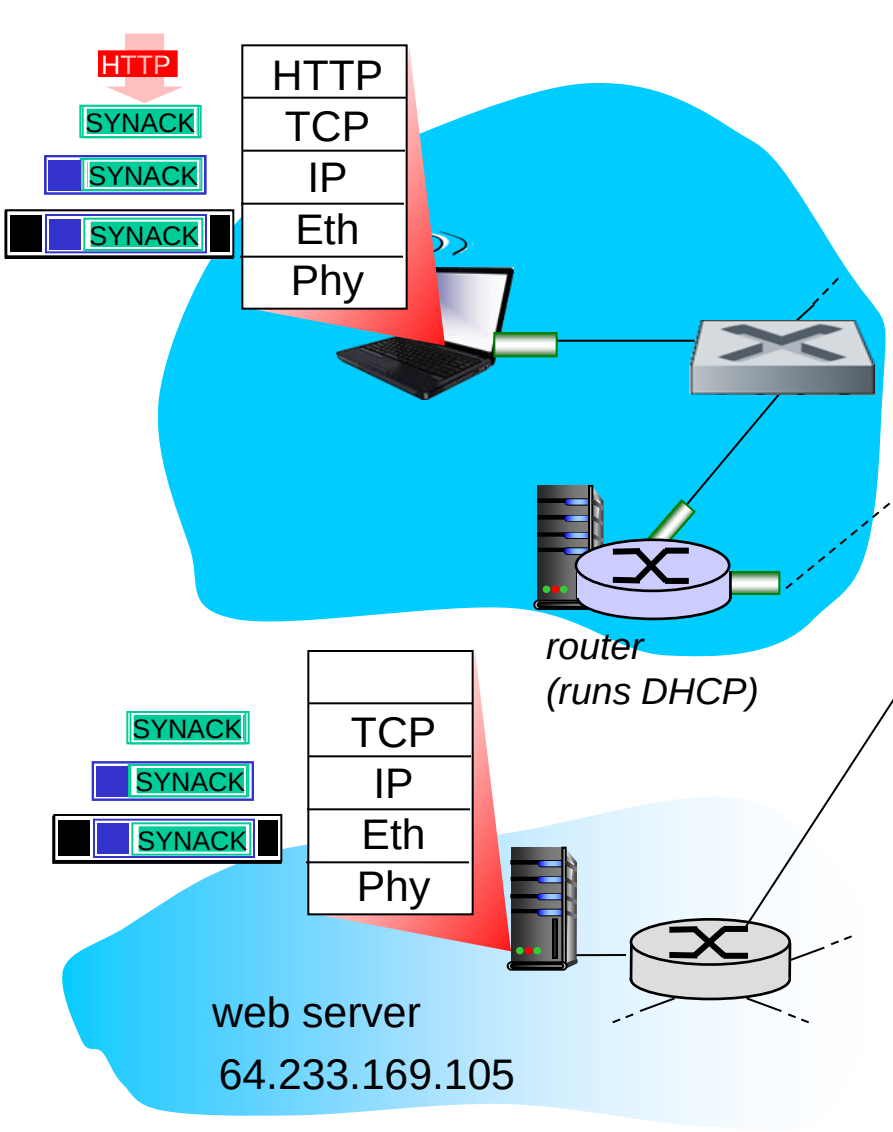
- before sending *HTTP* request, need IP address of www.google.com: *DNS*
- DNS query created, encapsulated in UDP, encapsulated in IP, encapsulated in Eth. To send frame to router, need MAC address of router interface: *ARP*
- *ARP query* broadcast, received by router, which replies with *ARP reply* giving MAC address of router interface
- client now knows MAC address of first hop router, so can now send frame containing DNS query

A day in the life… using DNS



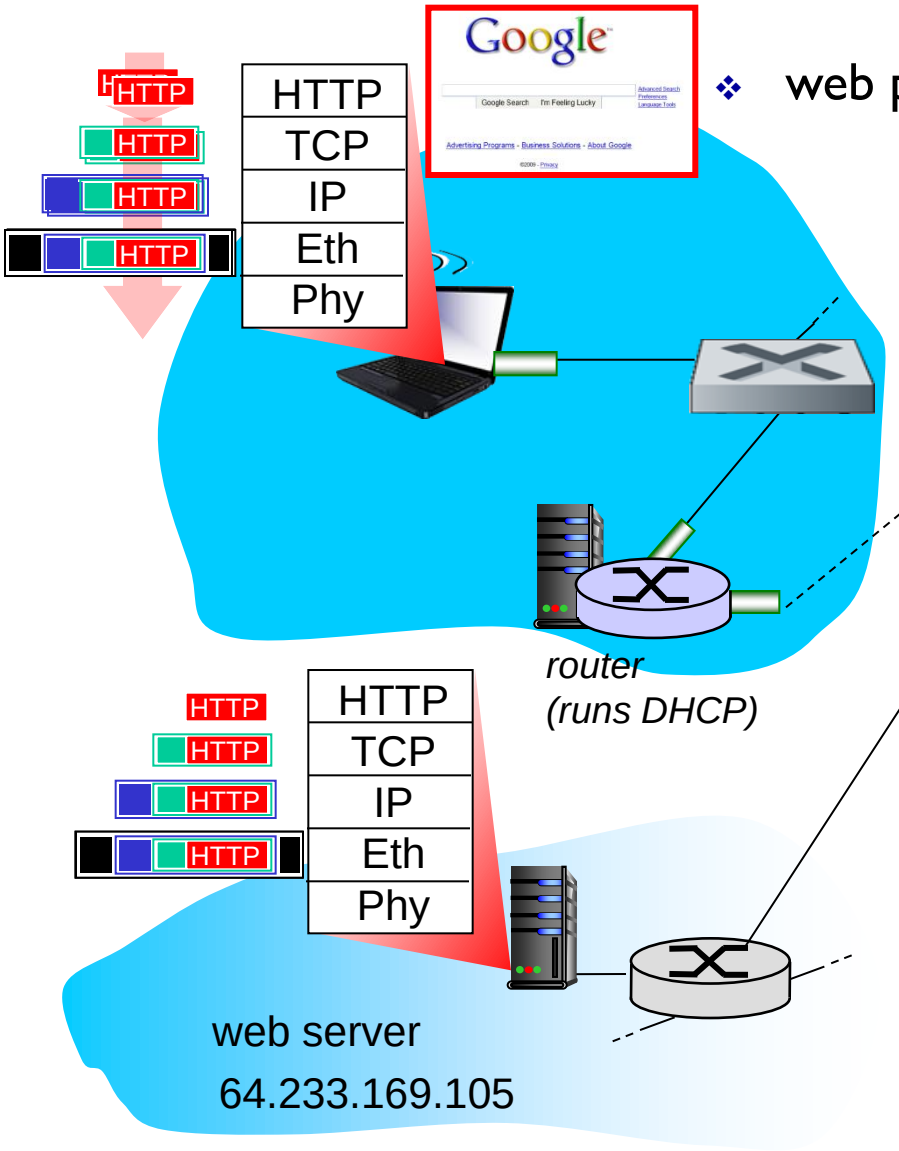
- IP datagram containing DNS query forwarded via LAN switch from client to 1st hop router
- IP datagram forwarded from campus network into comcast network, routed (tables created by *RIP, OSPF, IS-IS* and/or *BGP* routing protocols) to DNS server
- demux'ed to DNS server
- DNS server replies to client with IP address of www.google.com

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 3-way handshake) *inter-domain routed* to web server
- web server responds with *TCP SYNACK* (step 2 in 3-way handshake)
- TCP *connection established!*

A day in the life… HTTP request/reply



- web page *finally (!!!)* displayed

- *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

Chapter 5: Summary

- principles behind data link layer services:
  - error detection, correction
  - sharing a broadcast channel: multiple access
  - link layer addressing
- instantiation and implementation of various link layer technologies
  - Interconnection Devices: hub, switch , and router
  - LANs
    - Wireless network: IEEE 802.11 WIFI
    - Wired network: IEEE 802.3 Ethernet
- synthesis: a day in the life of a web request