

Chapter 5

Link Layer and LANs

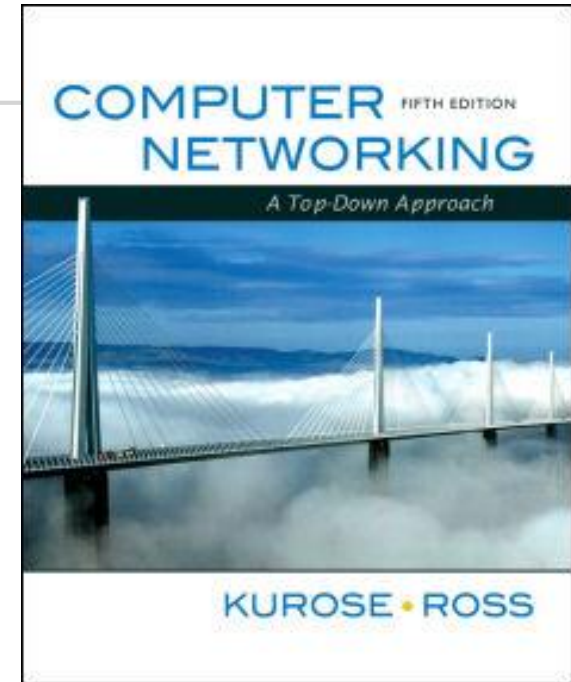
A note on the use of these ppt slides:

We're making these slides freely available to all (faculty, students, readers). They're in PowerPoint form so you can add, modify, and delete slides (including this one) and slide content to suit your needs. They obviously represent a *lot* of work on our part. In return for use, we only ask the following:

- If you use these slides (e.g., in a class) in substantially unaltered form, that you mention their source (after all, we'd like people to use our book!)
- If you post any slides in substantially unaltered form on a www site, that you note that they are adapted from (or perhaps identical to) our slides, and note our copyright of this material.

Thanks and enjoy! JFK/KWR

All material copyright 1996-2010
J.F Kurose and K.W. Ross, All Rights Reserved

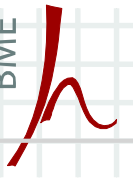


**Computer Networking: A
Top Down Approach
Featuring the Internet,
5th edition.
Jim Kurose, Keith Ross
Pearson Addison-Wesley,
2009.**

Chapter 5: The data link layer

Our goals:

- Understand principles behind data link layer services:
 - Error detection, correction
 - Sharing a broadcast channel: multiple access
 - Link layer addressing
 - Reliable data transfer, flow control: *done!*
- Instantiation and implementation of various link layer technologies



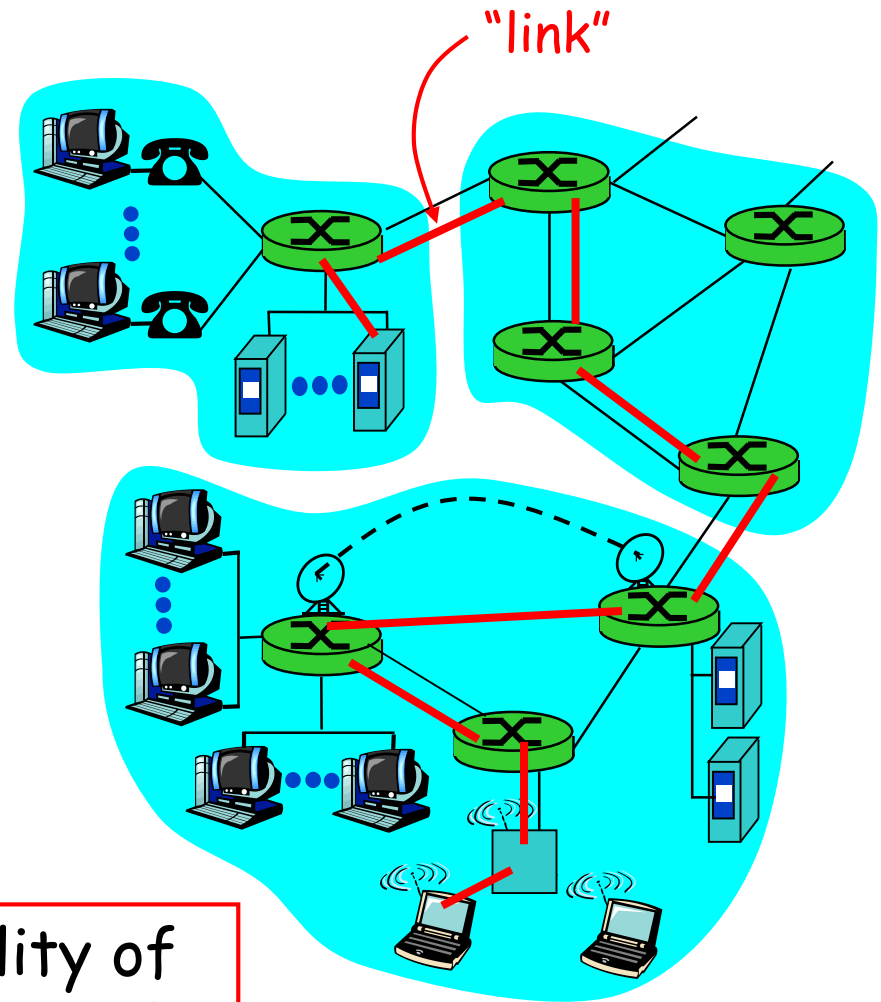
Chapter 5 outline

- 5.1 Introduction and services
- 5.2 Multiple access protocols
- 5.3 Link-Layer Addressing
- 5.4 Ethernet
- 5.5 Hubs and switches

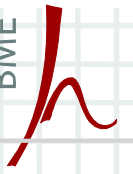
Link layer: Introduction

Some terminology:

- Hosts and routers are **nodes**
- Communication channels that connect adjacent nodes along communication path are **links**
 - Wired links
 - Wireless links
 - LANs
- Layer-2 packet is a **frame**, encapsulates datagram



Data-link layer has responsibility of transferring datagram from one node to 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 rdt over link

Transportation analogy:

- Trip from Washington to Debrecen
 - limo: Washington to Airport
 - plane: Airport to Budapest
 - train: Budapest to Debrecen
- 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

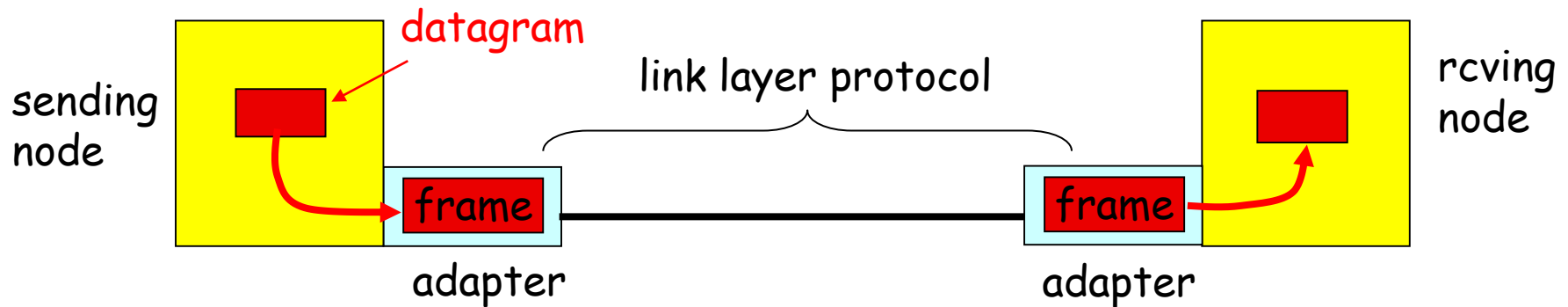
- We learned how to do this already (chapter 3)!
- Seldom used on low bit error link (fiber, some twisted pair)
- Wireless links: high error rates
 - **Q: why both link-level and end-end reliability?**
 - **A: if there is more than one lossy link, only the end systems will know which packets are really to be retransmitted and end-to-end reliability will such minimize the extra work to handle losses.**



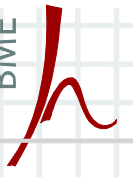
Link layer services (more)

- *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

Adaptors communicating



- Link layer implemented in “adaptor” (aka NIC)
 - Ethernet card, PCMCIA card, 802.11 card
- Sending side
 - Encapsulates datagram in a frame
 - Adds error checking bits, rdt, flow control, etc.
- Receiving side
 - Looks for errors, rdt, flow control, etc
 - Extracts datagram, passes to rcving node
- Adapter is semi-autonomous
- Link & physical layers



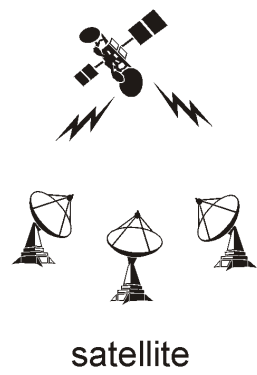
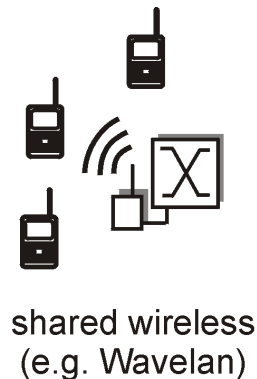
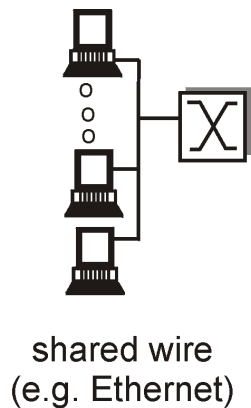
Chapter 5 outline

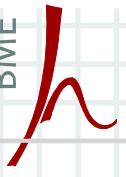
- 5.1 Introduction and services
- 5.2 Multiple access protocols
- 5.3 Link-Layer Addressing
- 5.4 Ethernet
- 5.5 Hubs and switches

Multiple access links and protocols

Two types of “links”:

- Point-to-point
 - PPP for dial-up access
 - Point-to-point link between Ethernet switch and host
- **Broadcast** (shared wire or medium)
 - Old-fashioned Ethernet
 - Upstream HFC
 - 802.11 wireless LAN





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



Ideal multiple access protocol

Broadcast channel of rate R bps

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

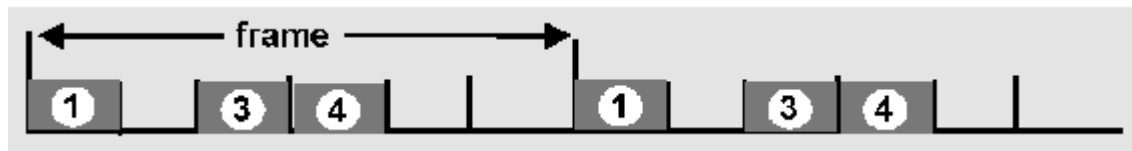
MAC protocols: A taxonomy

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
- **“Taking turns”**
 - Nodes take turns, but nodes with more to send can take longer turns

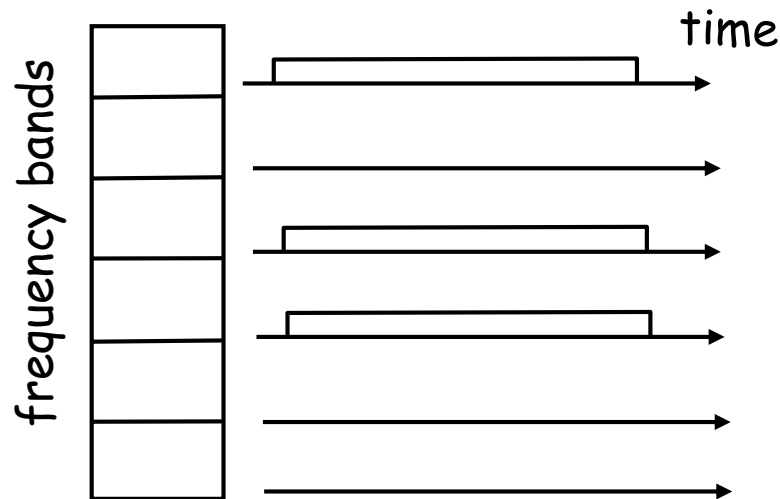
TDMA: Time Division Multiple Access

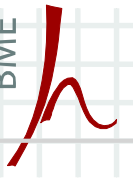
- Access to channel in "rounds"
- Each station gets fixed length slot (length = pkt trans time) in each round
- Unused slots go idle
- Example: 6-station LAN, 1,3,4 have pkt, slots 2,5,6 idle



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





Random access protocols

- 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



Slotted ALOHA

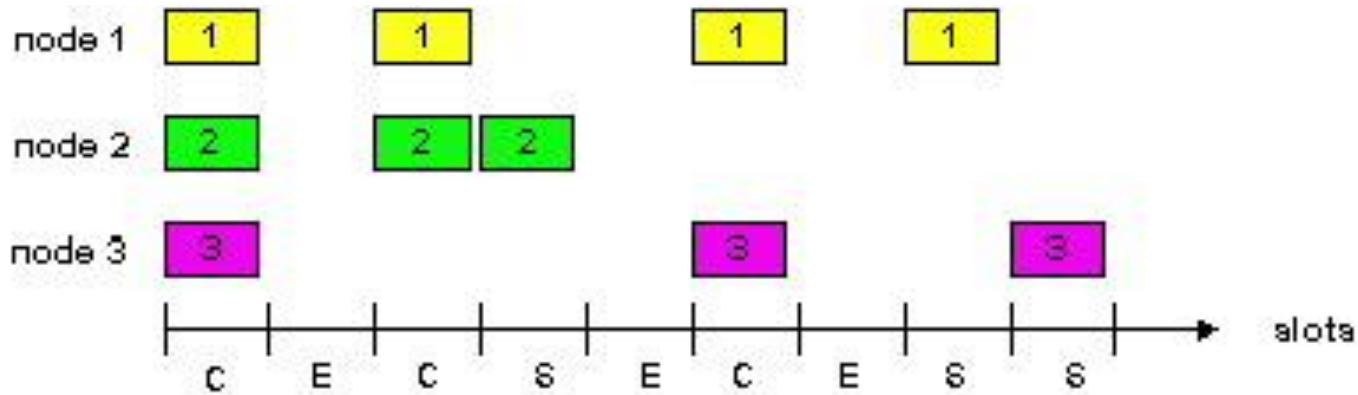
Assumptions:

- All frames same size
- Time is divided into equal size slots, time to transmit 1 frame
- Nodes start to transmit frames only at beginning of slots
- Nodes are synchronized
- If 2 or more nodes transmit in slot, all nodes detect collision

Operation:

- When node obtains fresh frame, it transmits in next slot
- No collision, node can send new frame in next slot
- If collision, node retransmits frame in each subsequent slot with prob. p until success

Slotted ALOHA

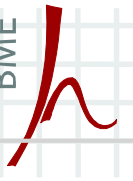


Pros

- Single active node can continuously transmit at full rate of channel
- Highly decentralized: only slots in nodes need to be in sync
- Simple

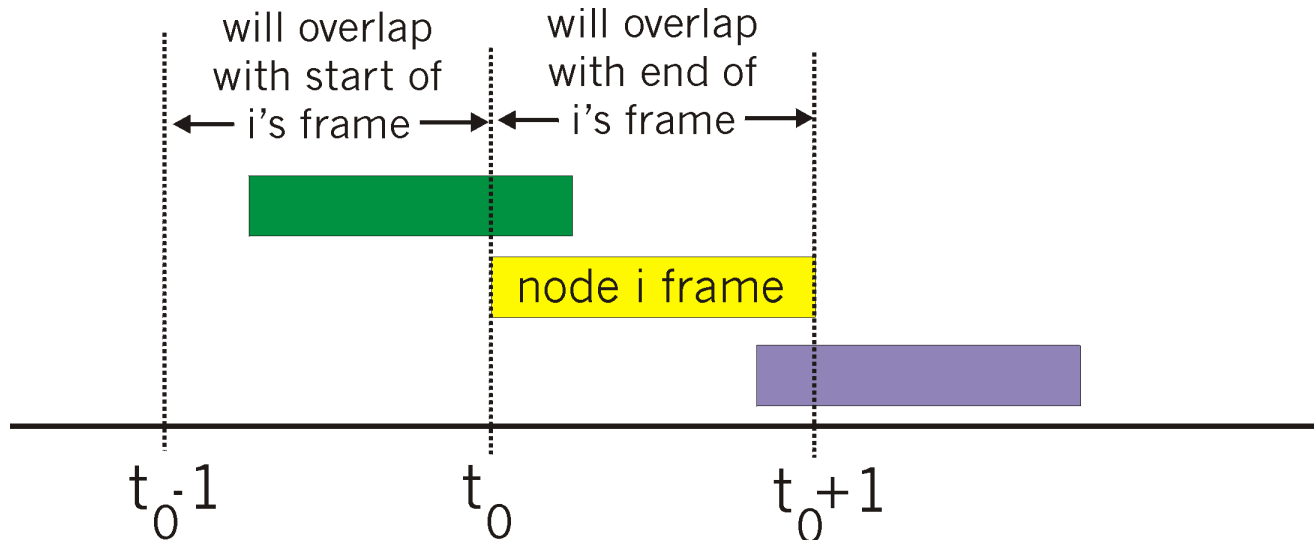
Cons

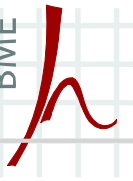
- Collisions, wasting slots
- Idle slots
- Nodes may be able to detect collision in less than time to transmit packet
- Clock synchronization



Pure (unslotted) ALOHA

- Unslotted Aloha: simpler, no synchronization
- When frame first arrives
 - Transmit immediately
- Collision probability increases
 - Frame sent at t_0 collides with other frames sent in $[t_0-1, t_0+1]$

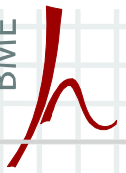




CSMA (Carrier Sense Multiple Access)

CSMA: Listen before transmit

- If channel sensed idle → transmit entire frame
- If channel sensed busy, defer transmission
- Human analogy: Don't interrupt others!



CSMA collisions

Collisions *can* still occur:

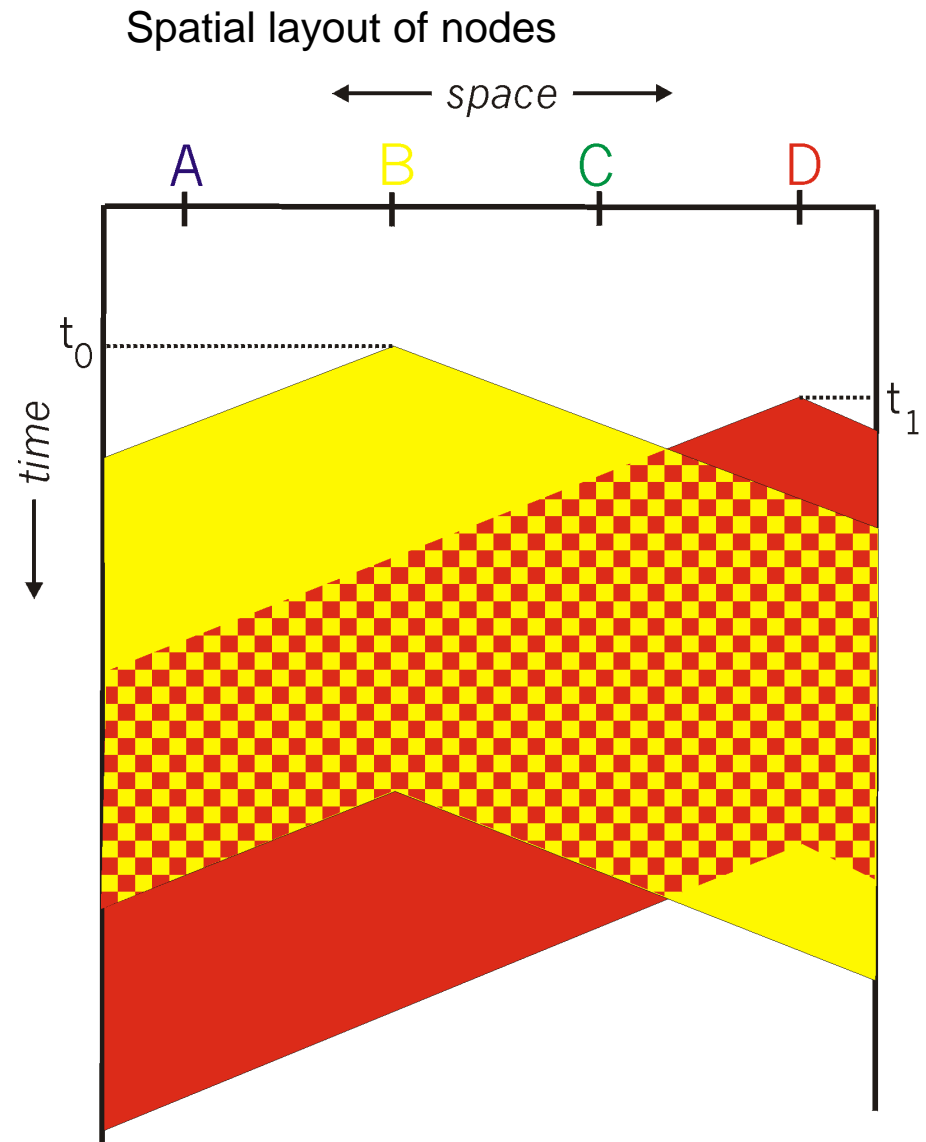
Propagation delay means two nodes may not hear each other's transmission

Collision:

Entire packet transmission time wasted

Note:

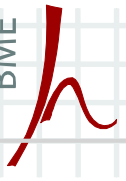
Role of distance & propagation delay in determining collision probability



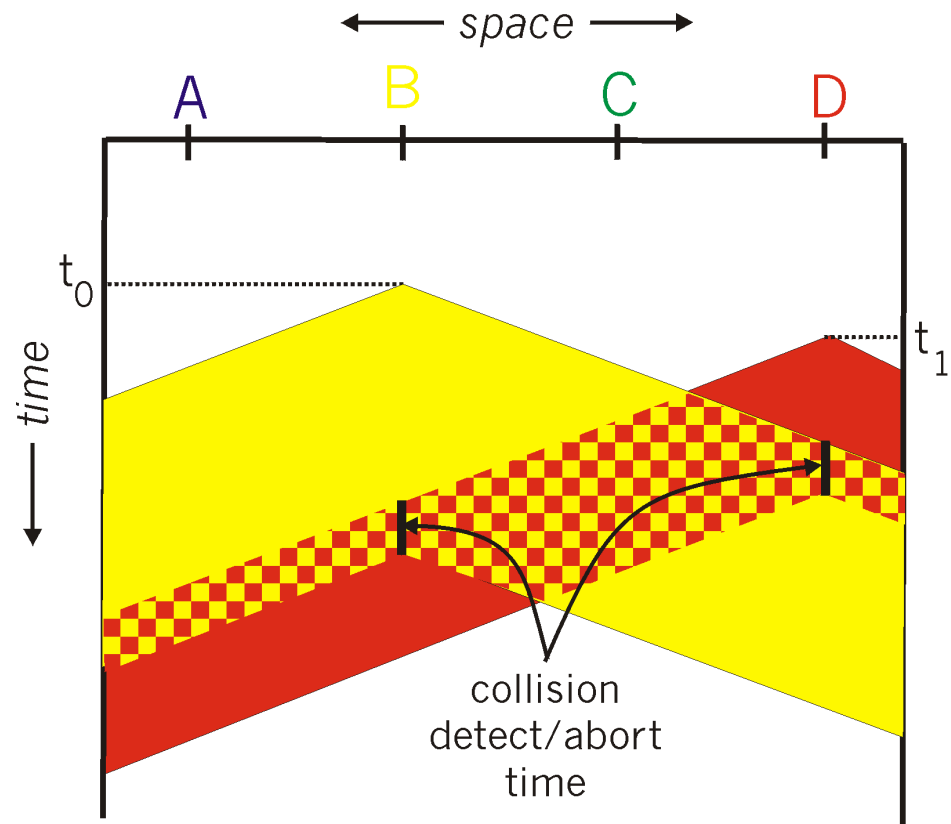
CSMA/CD (Collision Detection)

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: receiver shut off while transmitting
- Human analogy: the polite conversationalist



CSMA/CD collision detection





“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!

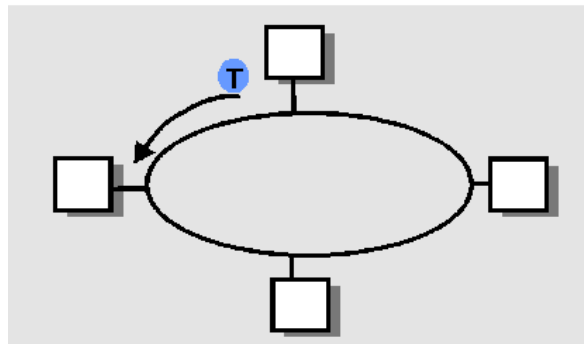
“Taking turns” MAC protocols

Polling:

- Master node “invites” slave nodes to transmit in turn
- Concerns:
 - Polling overhead
 - Latency
 - Single point of failure (master)

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

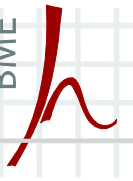
- What do you do with a shared media?
 - Channel partitioning, by time, frequency or code
 - Time Division, Frequency Division
 - Random partitioning (dynamic)
 - ALOHA, S-ALOHA, CSMA, CSMA/CD
 - Carrier sensing: easy in some technologies (wire), hard in others (wireless)
 - CSMA/CD used in Ethernet
 - CSMA/CA used in 802.11
 - Taking turns
 - Polling from a central site, token passing

Data link layer so far:

- Services, error detection/correction, multiple access

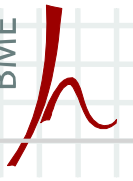
Next: LAN technologies

- Addressing
- Ethernet
- Hubs, switches
- PPP



Chapter 5 outline

- 5.1 Introduction and services
- 5.2 Multiple access protocols
- 5.3 Link-Layer Addressing
- 5.4 Ethernet
- 5.5 Hubs and switches

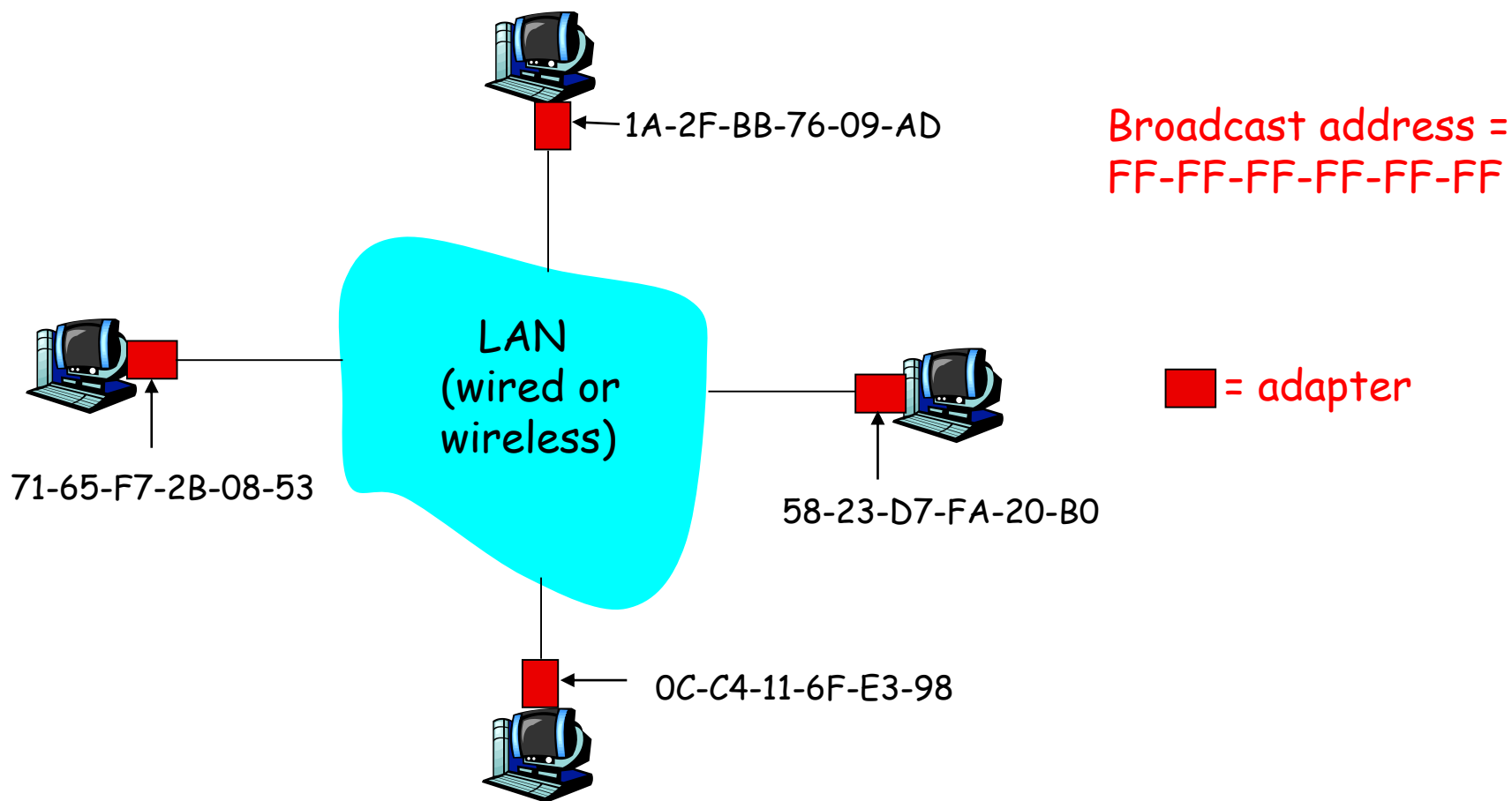


MAC addresses and ARP

- 32-bit IP address
 - *Network-layer* address
 - Used to get datagram to destination IP subnet
- MAC (or LAN or physical or Ethernet) address
 - Used to get frame from one interface to another physically-connected interface (same network)
 - 48 bit MAC address (for most LANs)
burned in the adapter ROM

LAN addresses and ARP

Each adapter on LAN has unique LAN address





LAN addresses (more)

- MAC address allocation administered by IEEE
- Manufacturer buys portion of MAC address space (to assure uniqueness)
- Analogy:
 - (a) MAC address: like Social Security Number
 - (b) IP address: like postal address
- MAC flat address → portability
 - Can move LAN card from one LAN to another
- IP hierarchical address NOT portable
 - Depends on IP subnet to which node is attached

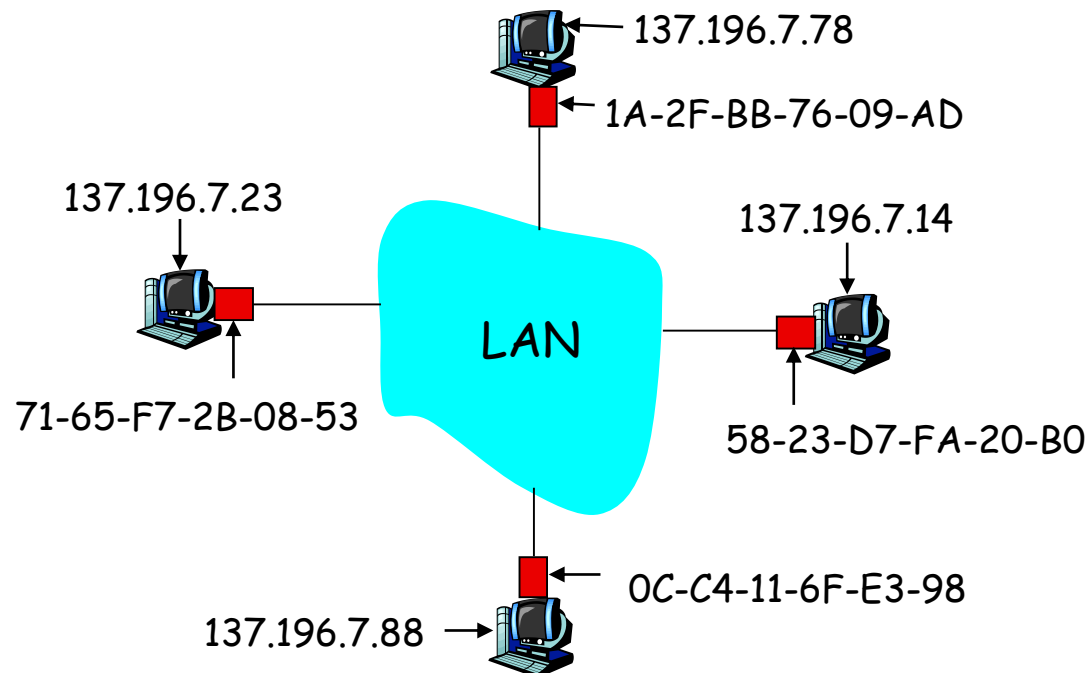
ARP: Address Resolution Protocol

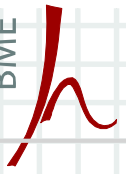
Question: how to determine MAC address of B knowing B's IP address?

- Each IP node (Host, Router) on LAN has an **ARP** table
- ARP 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: Same LAN (network)

- A wants to send datagram to B, and B's MAC address not in A's ARP table
- A **broadcasts** ARP query packet, containing B's IP address
 - Dest MAC address = FF-FF-FF-FF-FF-FF
 - All machines on LAN receive ARP query
- B receives ARP packet, replies to A with its (B's) MAC address
 - Frame sent to A's MAC address (unicast)
- A caches (saves) IP-to-MAC address pair in its ARP table until information becomes old (times out)
 - Soft state: information that times out (goes away) unless refreshed
- ARP is “plug-and-play”
 - Nodes create their ARP tables without intervention from net administrator

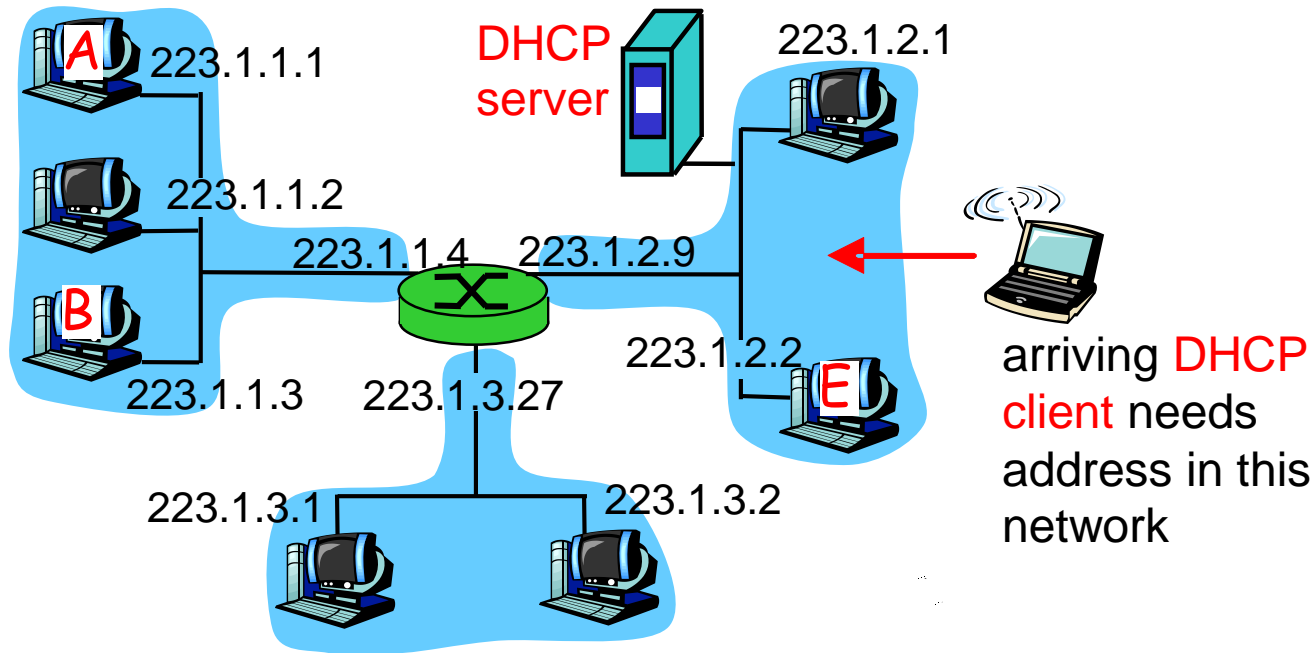


DHCP: Dynamic Host Configuration Protocol

Goal: allow host to *dynamically* obtain its IP address from network server when it joins network

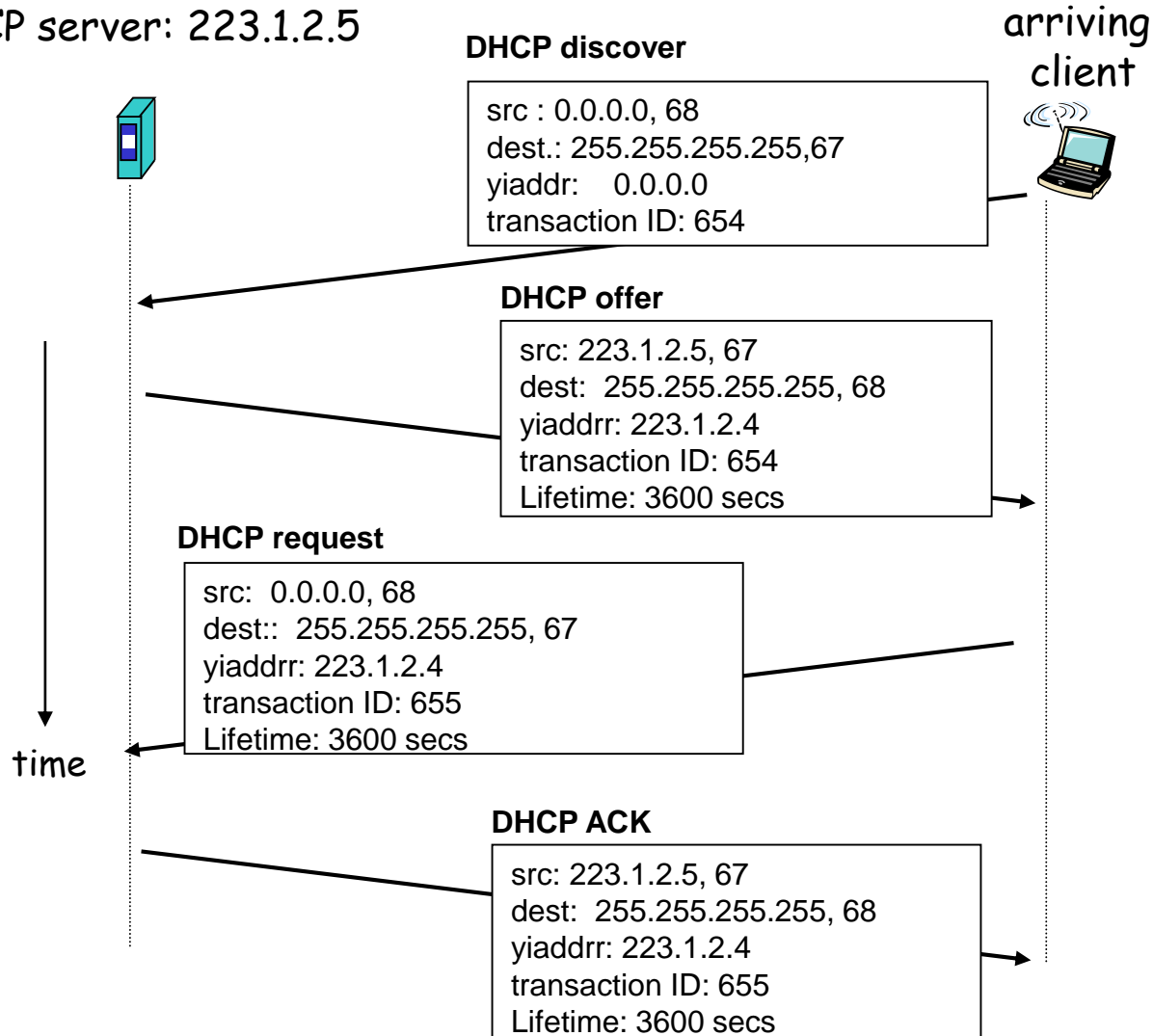
- Can renew its lease on address in use
 - Allows reuse of addresses (only hold address while connected)
 - Support for mobile users who want to join network
- DHCP overview
- Host broadcasts “DHCP discover” msg
 - DHCP server responds with “DHCP offer” msg
 - Host requests IP address: “DHCP request” msg
 - DHCP server sends address: “DHCP ack” msg

DHCP client-server scenario



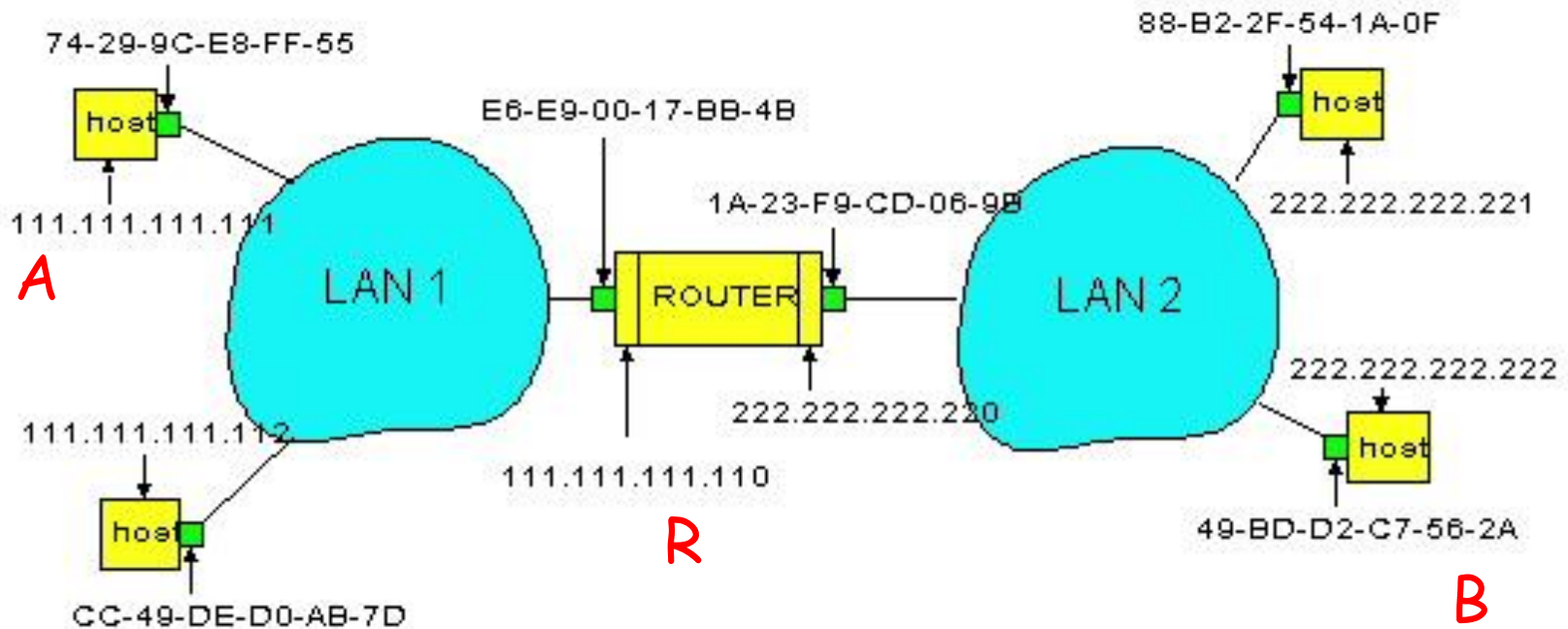
DHCP client-server scenario (cont'd)

DHCP server: 223.1.2.5



Routing to another LAN

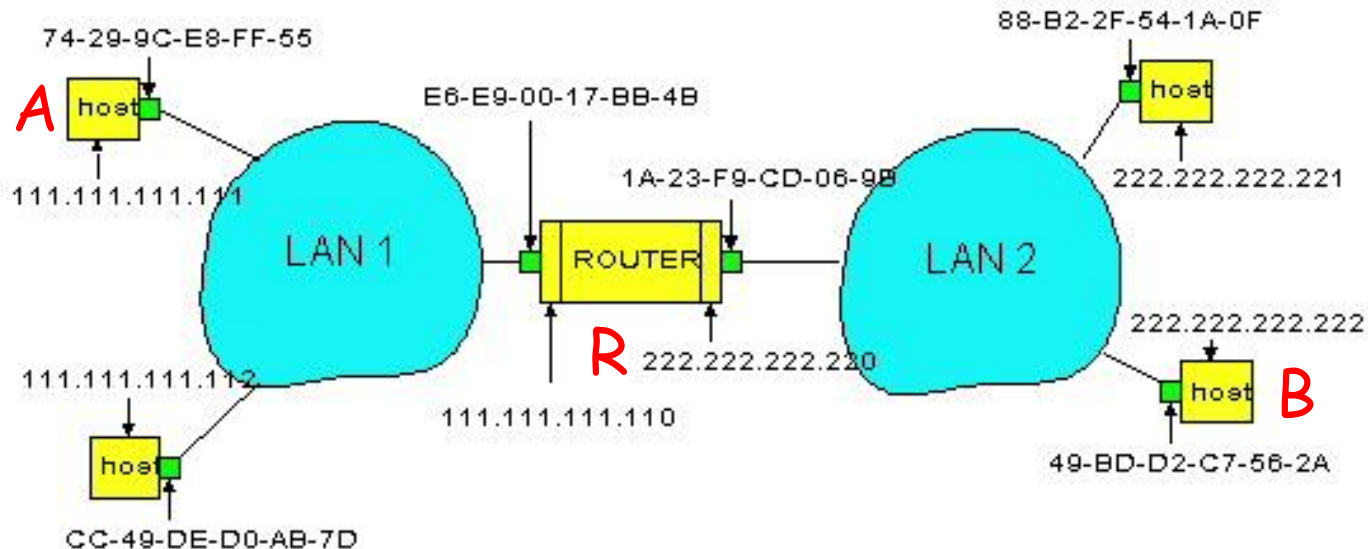
Walkthrough: send datagram from A to B via R
assume A knows B's IP address

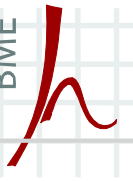


- Two ARP tables in router R, one for each IP network (LAN)
- In routing table at source Host, find router 111.111.111.110
- In ARP table at source, find MAC address E6-E9-00-17-BB-4B, etc.

Routing to another LAN (cont'd)

- A creates datagram with source A, destination B
- A uses ARP to get R's MAC address for 111.111.111.110
- A creates link-layer frame with R's MAC address as dest, frame contains A-to-B IP datagram
- A's adapter sends frame
- R's adapter receives frame
- R removes IP datagram from Ethernet frame, sees its destined to B
- R uses ARP to get B's MAC address
- R creates frame containing A-to-B IP datagram sends to B



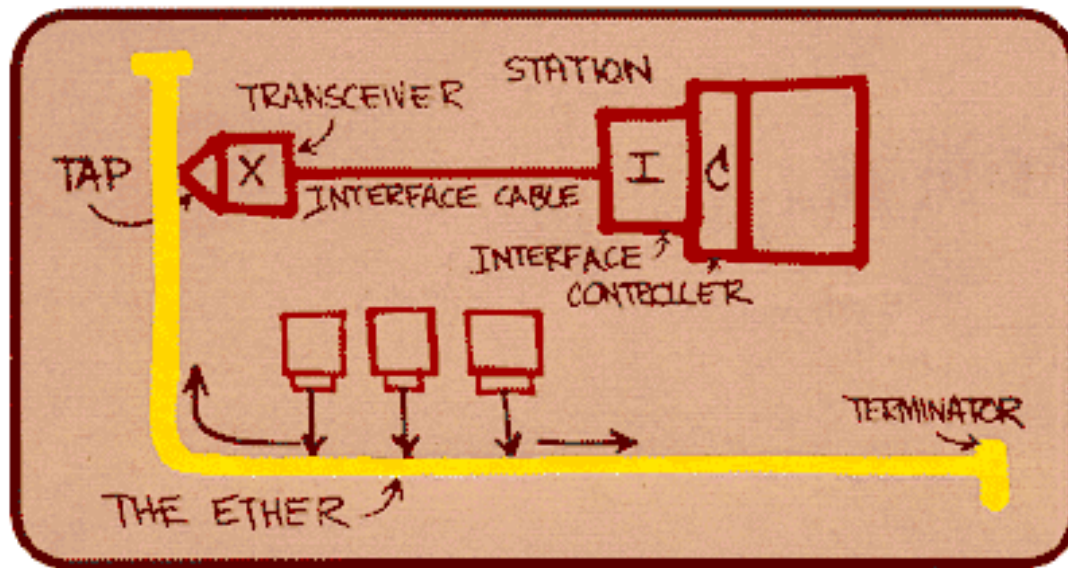


Chapter 5 outline

- 5.1 Introduction and services
- 5.2 Multiple access protocols
- 5.3 Link-Layer Addressing
- 5.4 Ethernet
- 5.5 Hubs and switches

“Dominant” wired LAN technology

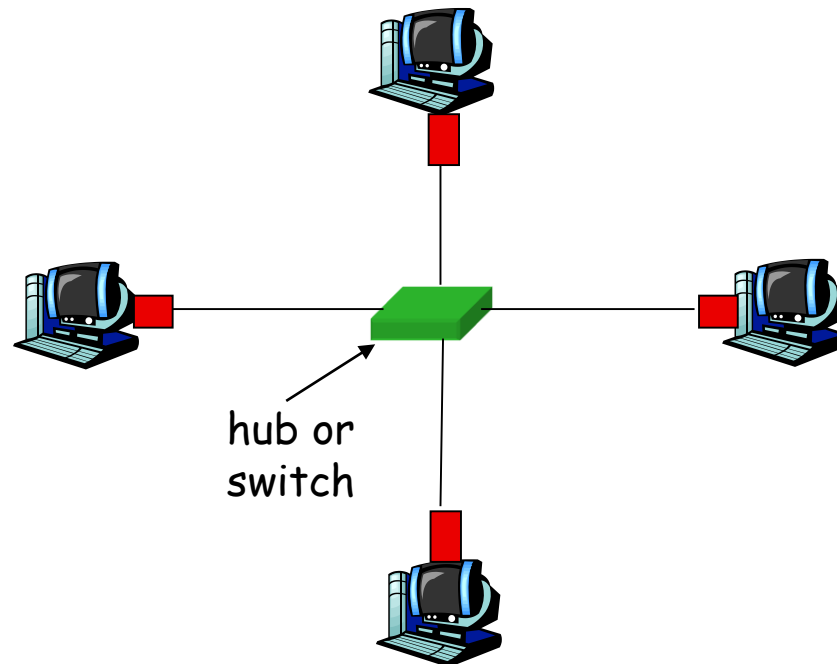
- Cheap \$20 for 100Mbps!
- First widely used LAN technology
- Simpler, cheaper than token LANs and ATM
- Kept up with speed race: 10 Mbps – 10 Gbps



Metcalfe's Ethernet sketch

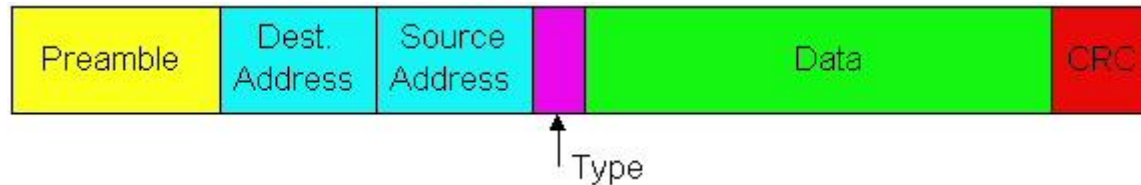
Star topology

- Bus topology popular through mid 90s
- Now star topology prevails
- Connection choices: hub or switch (more later)



Ethernet frame structure

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

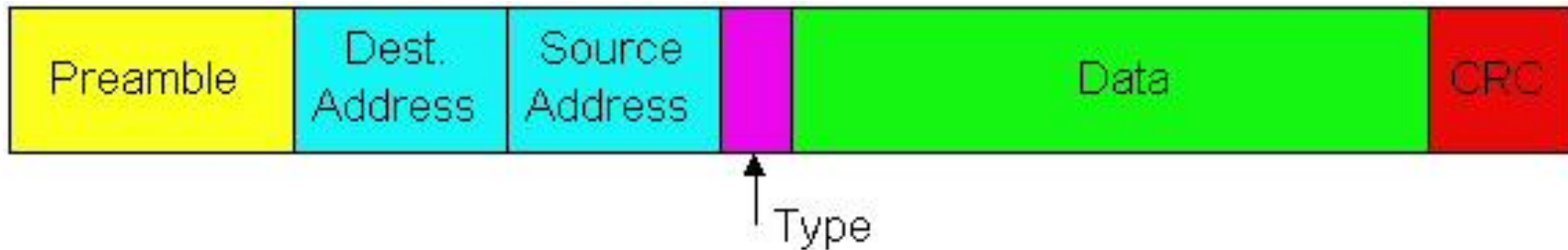


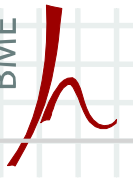
Preamble:

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

Ethernet frame structure (more)

- **Addresses:** 6 bytes
 - If adapter receives frame with matching destination address, or with broadcast address (eg., ARP packet), it passes data in frame to net-layer protocol
 - Otherwise, adapter discards frame
- **Type:** indicates the higher layer protocol (mostly IP but others may be supported such as Novell IPX and AppleTalk)
- **CRC:** checked at receiver, if error is detected, the frame is simply dropped





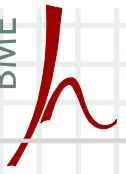
Unreliable, connectionless service

■ Connectionless

- No handshaking between sending and receiving adapter

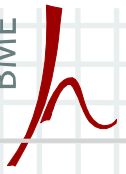
■ Unreliable

- Receiving adapter doesn't send acks or nacks to sending adapter
- Stream of datagrams passed to network layer can have gaps
- Gaps will be filled if app is using TCP
- Otherwise, app will see the gaps



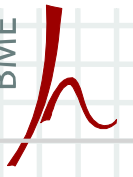
Ethernet uses CSMA/CD

- No slots
- Adapter doesn't transmit if it senses that some other adapter is transmitting, that is, **carrier sense**
- Transmitting adapter aborts when it senses that another adapter is transmitting, that is, **collision detection**
- Before attempting a retransmission, adapter waits a random time, that is, **random access**



Ethernet's CSMA/CD algorithm

1. Adaptor receives datagram from net layer & creates frame
2. If adapter senses channel idle, it starts to transmit frame. If it senses channel busy, waits until channel idle and then transmits
3. If adapter transmits entire frame without detecting another transmission, the adapter is done with frame!
4. If adapter detects another transmission while transmitting, aborts and sends jam signal
5. After aborting, adapter enters **exponential backoff**: after the m^{th} collision, adapter chooses a K at random from $\{0, 1, 2, \dots, 2^m - 1\}$. Adapter waits $K * 512$ bit times and returns to Step 2



Ethernet's CSMA/CD algorithm (more)

Jam signal

- Make sure all other transmitters are aware of collision; 48 bits

Bit time

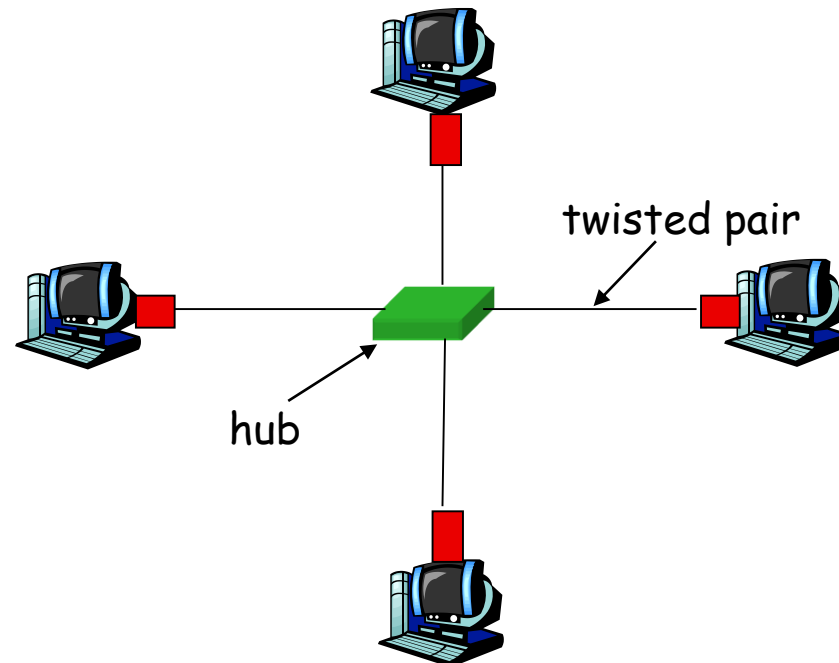
- 0.1 microsec for 10 Mbps Ethernet;
for $K=1023$, wait time is about 50 msec

Exponential backoff

- *Goal*: adapt retransmission attempts to estimated current load
 - Heavy load: random wait will be longer
- First collision: choose K from $\{0,1\}$; delay is $K * 512$ bit transmission times
- After second collision: choose K from $\{0,1,2,3\}$...
- After 10 collisions, choose K from $\{0,1,2,3,4,\dots,1023\}$

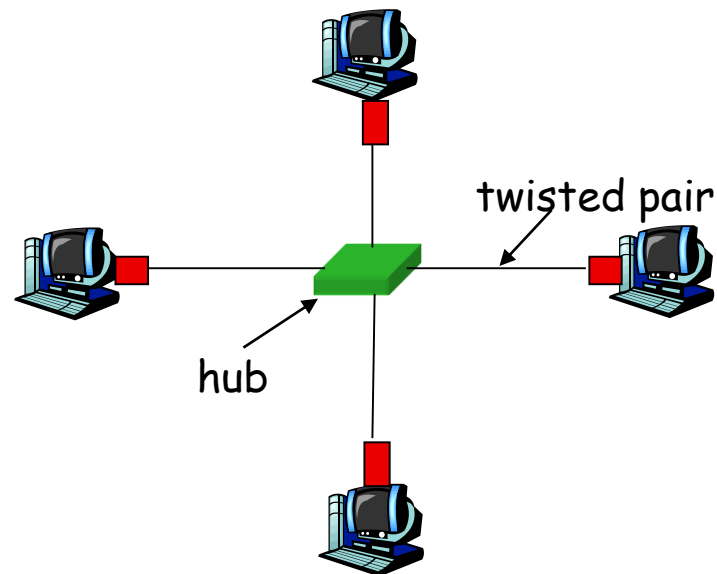
10BaseT and 100BaseT

- 10/100 Mbps rate; latter called “Fast Ethernet”
- T stands for Twisted Pair
- Nodes connect to a hub: “Star topology”; 100 m max. distance between nodes and hub

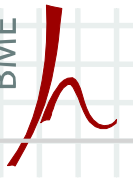


Hubs are essentially physical-layer repeaters

- Bits coming from one link go out all other links
- At the same rate
- No frame buffering
- No CSMA/CD at hub: adapters detect collisions



- Uses standard Ethernet frame format
- Allows for point-to-point links and shared broadcast channels
- In shared mode, CSMA/CD is used; short distances between nodes required for efficiency
- Uses hubs, called here “Buffered Distributors”
- Full-Duplex at 1 Gbps for point-to-point links
- 10 Gbps now!

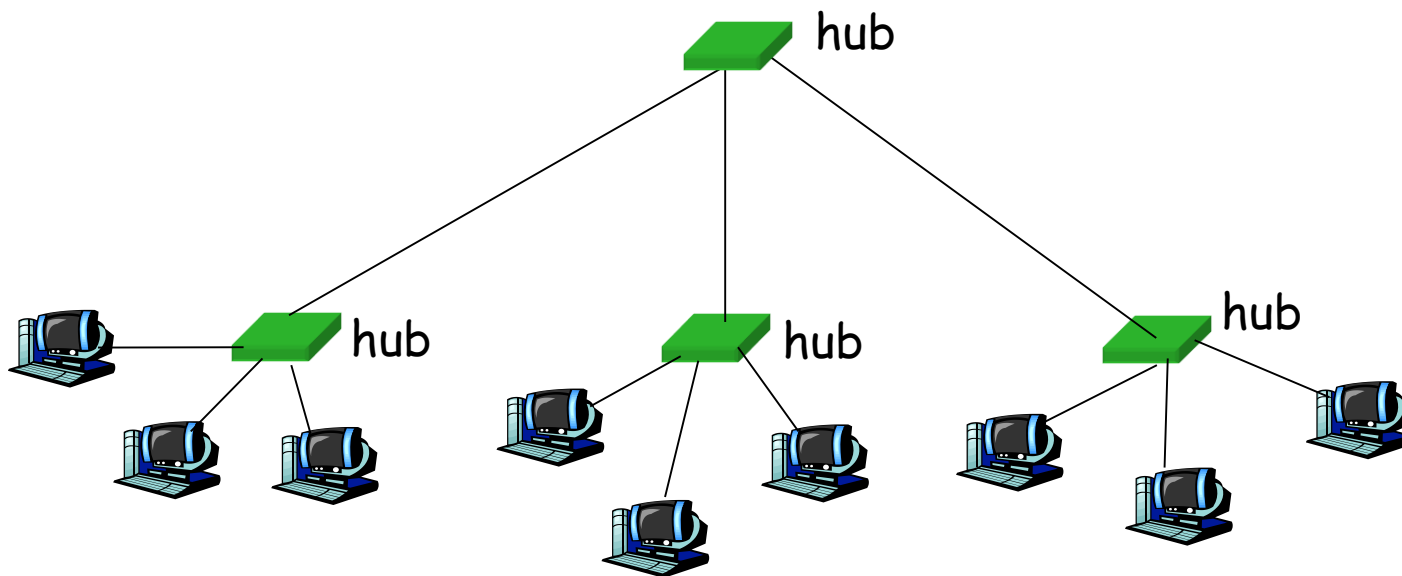


Chapter 5 outline

- 5.1 Introduction and services
- 5.2 Multiple access protocols
- 5.3 Link-Layer Addressing
- 5.4 Ethernet
- 5.5 Hubs and switches

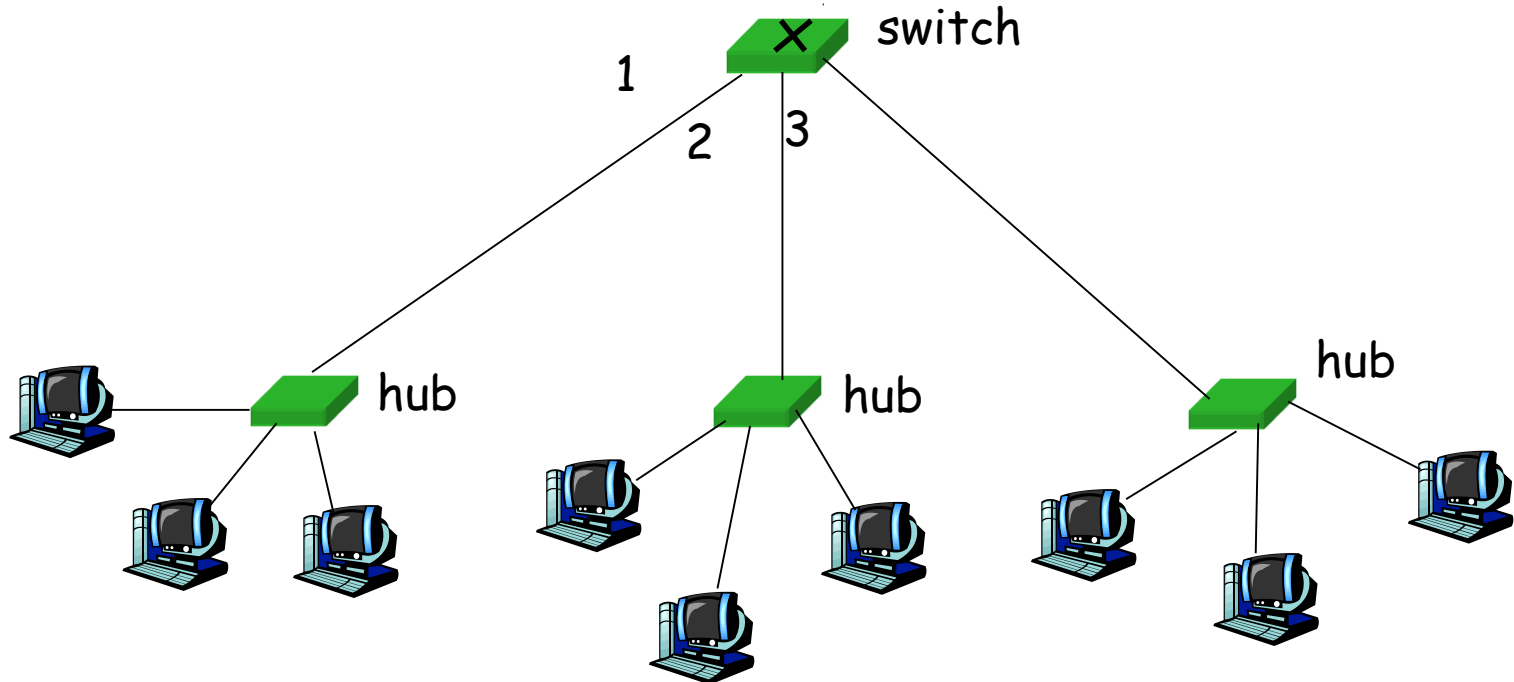
Interconnecting with hubs

- Backbone hub interconnects LAN segments
- Extends max distance between nodes
- But individual segment collision domains become one large collision domain
- Can't interconnect 10BaseT & 100BaseT



- Link layer device
 - Stores and forwards Ethernet frames
 - Examines frame header and **selectively** forwards frame based on MAC dest address
 - When frame is to be forwarded on segment, uses CSMA/CD to access segment
- Transparent
 - Hosts are unaware of presence of switches
- Plug-and-play, self-learning
 - Switches do not need to be configured

Forwarding



- How do determine onto which LAN segment to forward frame?
- Looks like a routing problem...

- A switch has a **switch table**
- Entry in switch table
 - (MAC Address, Interface, Time Stamp)
 - Stale entries in table dropped (TTL can be 60 min)
- 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

Filtering / Forwarding

When switch receives a frame:

index switch table using MAC dest address

if entry found for destination

then{

if dest on segment from which frame arrived

then drop the frame

else forward the frame on interface indicated

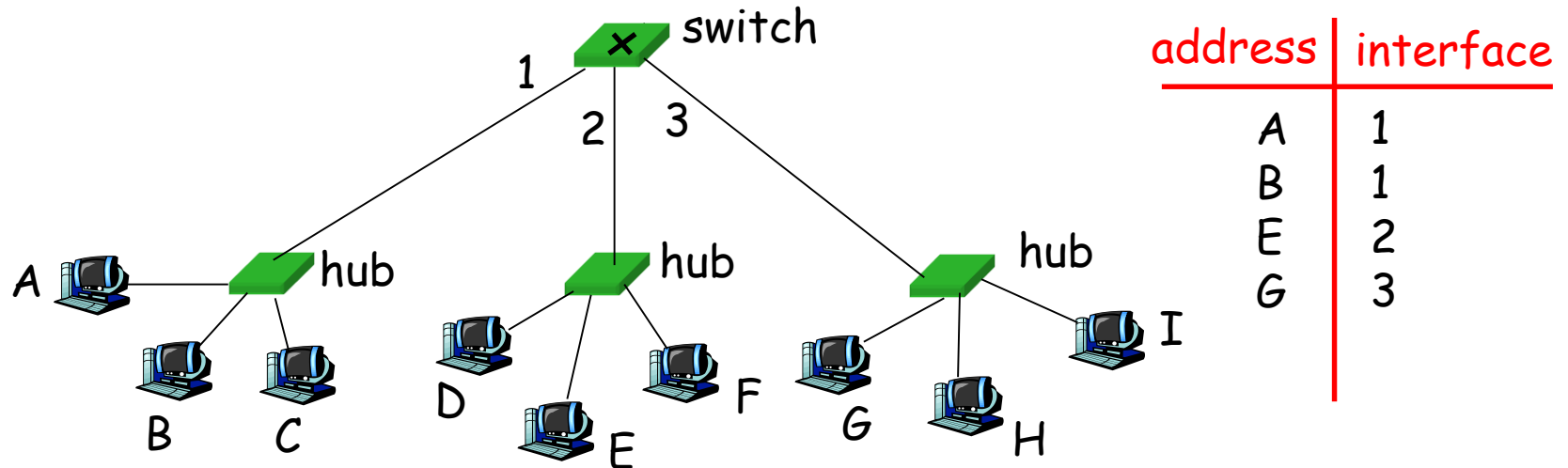
}

else flood

*forward on all but the interface
on which the frame arrived*

Switch example

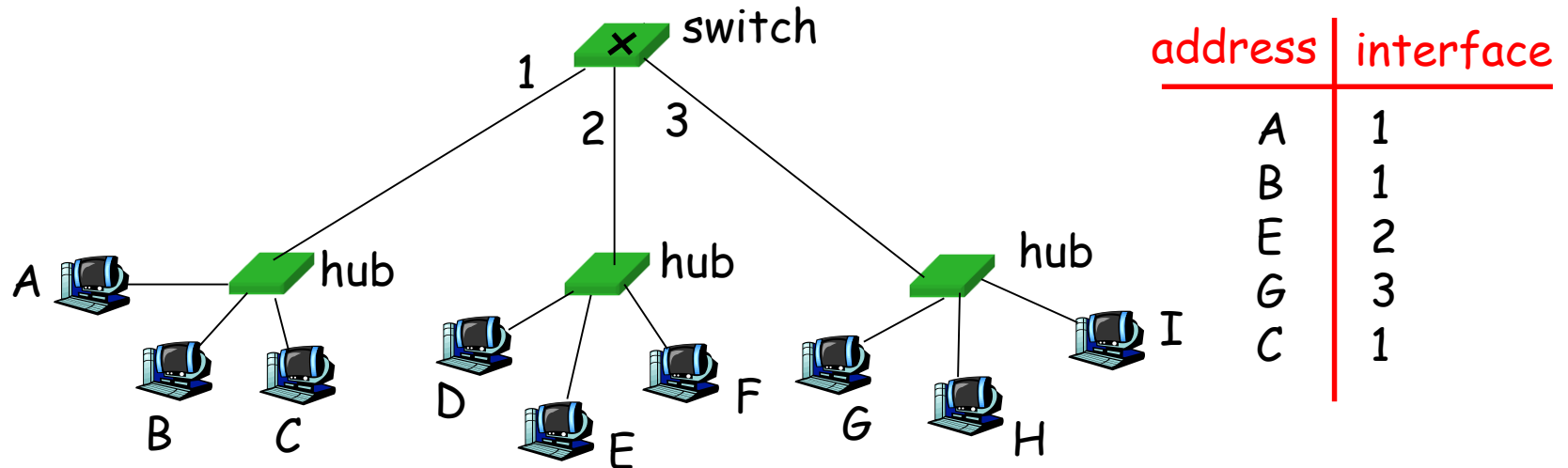
Suppose C sends frame to D



- Switch receives frame from C
 - Notes in bridge table that C is on interface 1
 - Because D is not in table, switch forwards frame into interfaces 2 and 3
- Frame received by D

Switch example (cont'd)

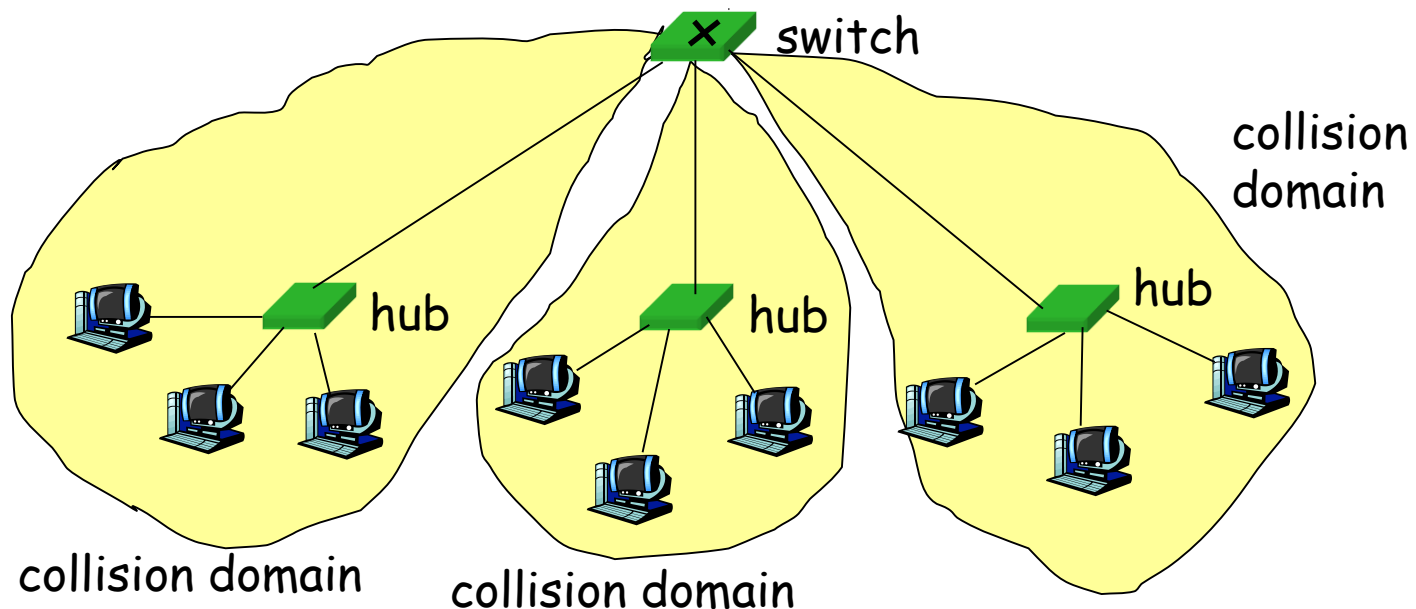
Suppose D replies back with frame to C



- Switch receives frame from D
 - Notes in bridge table that D is on interface 2
 - Because C is in table, switch forwards frame only to interface 1
- Frame received by C

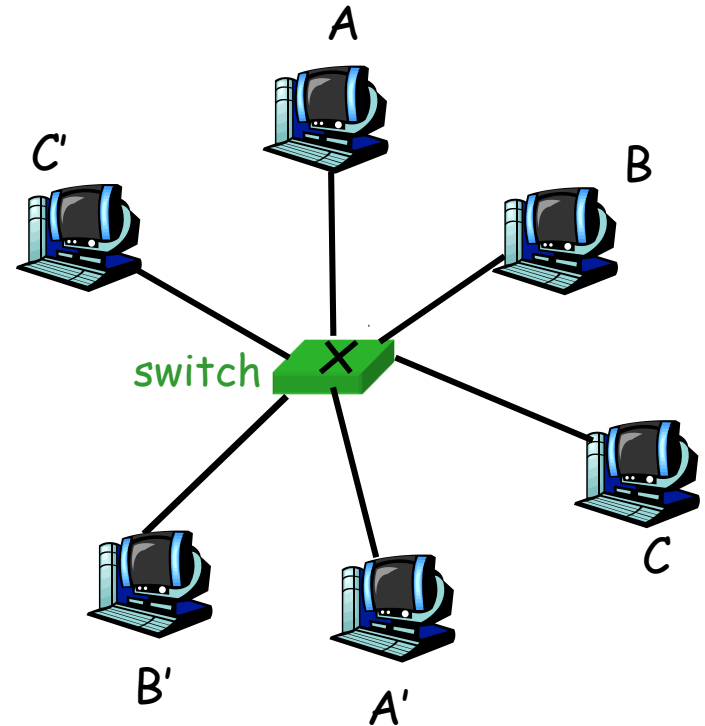
Switch: Traffic isolation

- Switch installation breaks subnet into LAN segments
- Switch **filters** packets
 - Same-LAN-segment frames not usually forwarded onto other LAN segments
 - Segments become separate **collision domains**

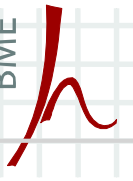


Switches: Dedicated access

- Switch with many interfaces
- Hosts have direct connection to switch
- No collisions; full duplex



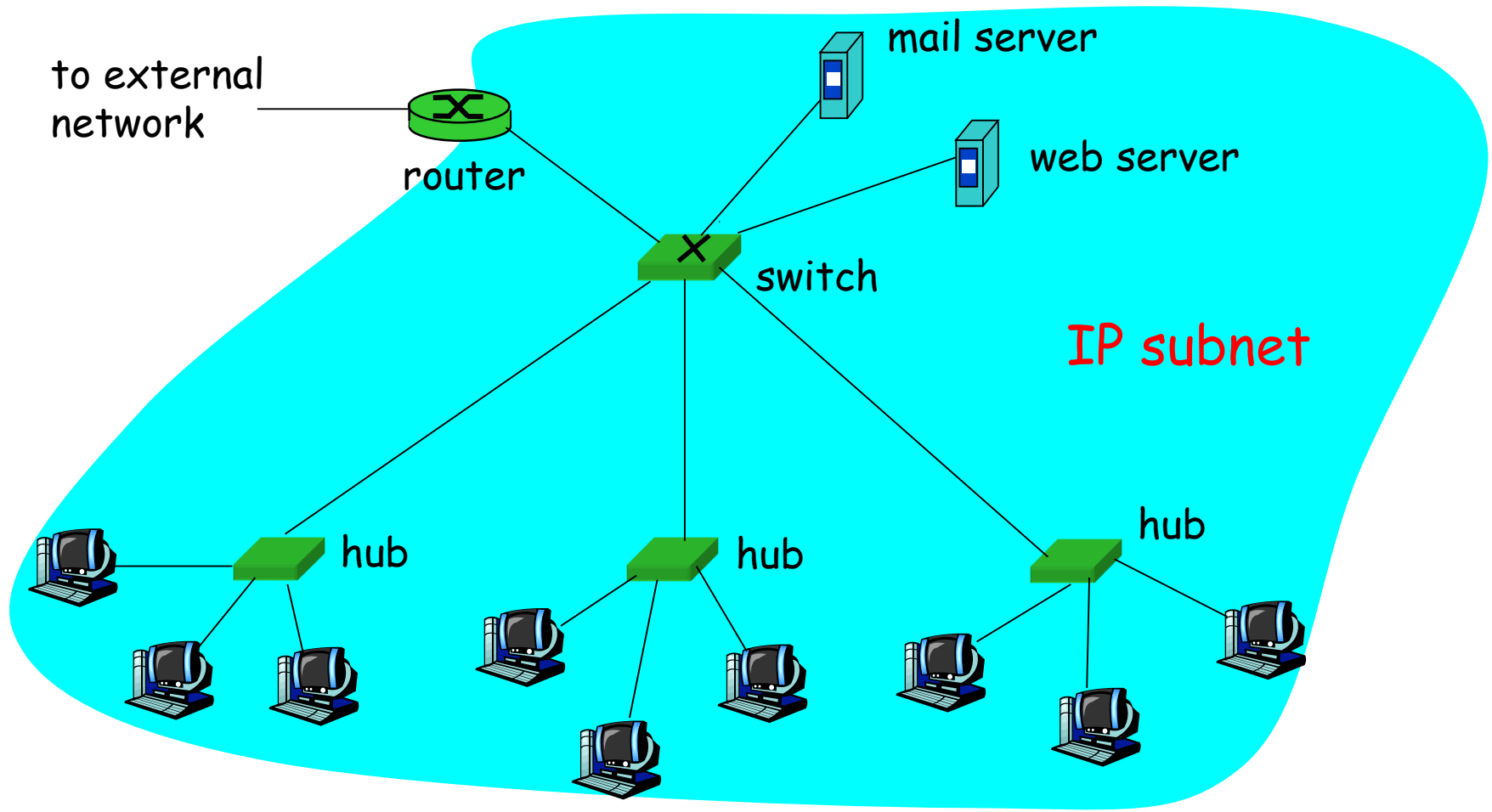
Switching: A-to-A' and B-to-B' simultaneously, no collisions



More on switches

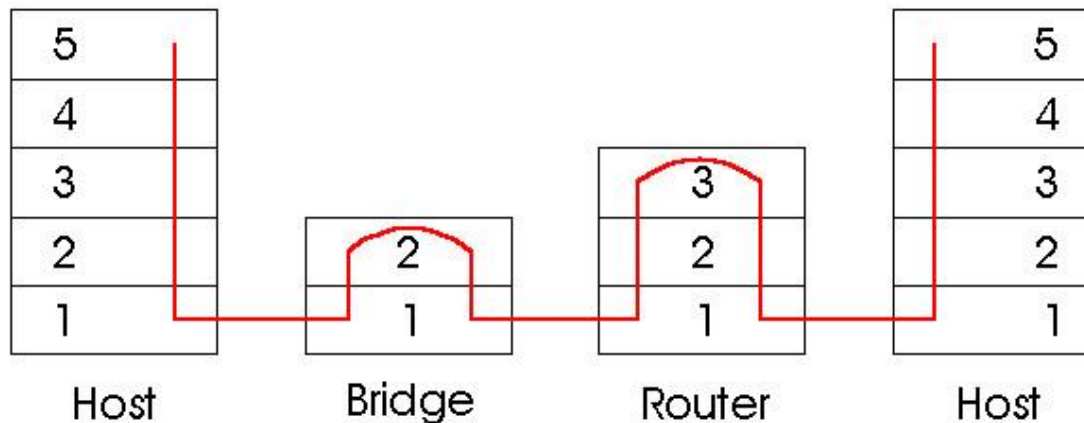
- **Cut-through switching**
 - Frame forwarded from input to output port without first collecting entire frame
 - Slight reduction in latency
- Combinations of shared/dedicated, 10/100/1000 Mbps interfaces

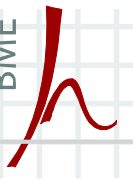
Institutional network



Switches vs. routers

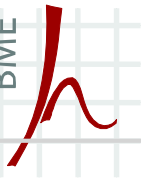
- Both store-and-forward devices
 - Routers: network layer devices (examine network layer headers)
 - Switches are link layer devices
- Routers maintain routing tables, implement routing algorithms
- Switches maintain switch tables, implement filtering, learning algorithms





Summary comparison

| | <u>Hubs</u> | <u>Routers</u> | <u>Switches</u> |
|-------------------|-------------|----------------|-----------------|
| Traffic isolation | no | yes | yes |
| Plug & play | yes | no | yes |
| Optimal routing | no | yes | no |
| Cut through | yes | no | yes |



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
 - Ethernet
 - Switched LANS