

# Data link layer

## Goals:

- ❑ Principles behind data link layer services
  - Error detection, correction
  - Sharing a broadcast channel: Multiple access
  - Link layer addressing
  - Reliable data transfer, flow control: *Done!*
- ❑ Example link layer technology: Ethernet

# Link layer services

## Framing and link access

- Encapsulate datagram: Frame adds header, trailer
- Channel access – if shared medium
- Frame headers use 'physical addresses' = "MAC" to identify source and destination
  - Different from IP address!

## Reliable delivery (between adjacent nodes)

- Seldom used on low bit error links (fiber optic, co-axial cable and some twisted pairs)
- Sometimes used on high error rate links (e.g., wireless links)

# Link layer services (2.)

## Flow Control

- Pacing between sending and receiving nodes

## Error Detection

- Errors are caused by signal attenuation and noise.
- Receiver detects presence of errors  
signals sender for retrans. 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

# Multiple access links / protocols

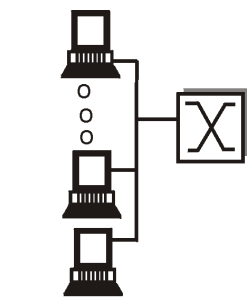
Two types of “links”:

❑ Point-to-point

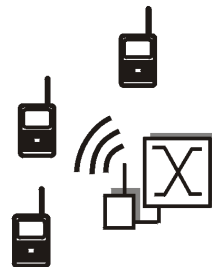
- PPP for dial-up access
- Point-to-point link between Ethernet switch and host

❑ **Broadcast** (shared wire or medium)

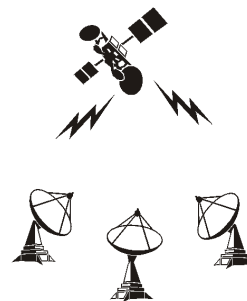
- Traditional Ethernet
- Upstream HFC
- 802.11 wireless LAN



shared wire  
(e.g. Ethernet)



shared wireless  
(e.g. Wavelan)



satellite



ZZZZZZZZZZZZZZZZ



cocktail party

# MAC protocols: Three broad classes

## □ Channel Partitioning

- Divide channel into smaller “pieces” (time slots, frequency)
- Allocate piece to node for exclusive use

## □ Random Access

- Allow collisions
- “Recover” from collisions

## □ “Taking turns”

- Tightly coordinate shared access to avoid collisions

**Goal:** Efficient, fair, simple, decentralized

# Addresses

## IP address (32-bit):

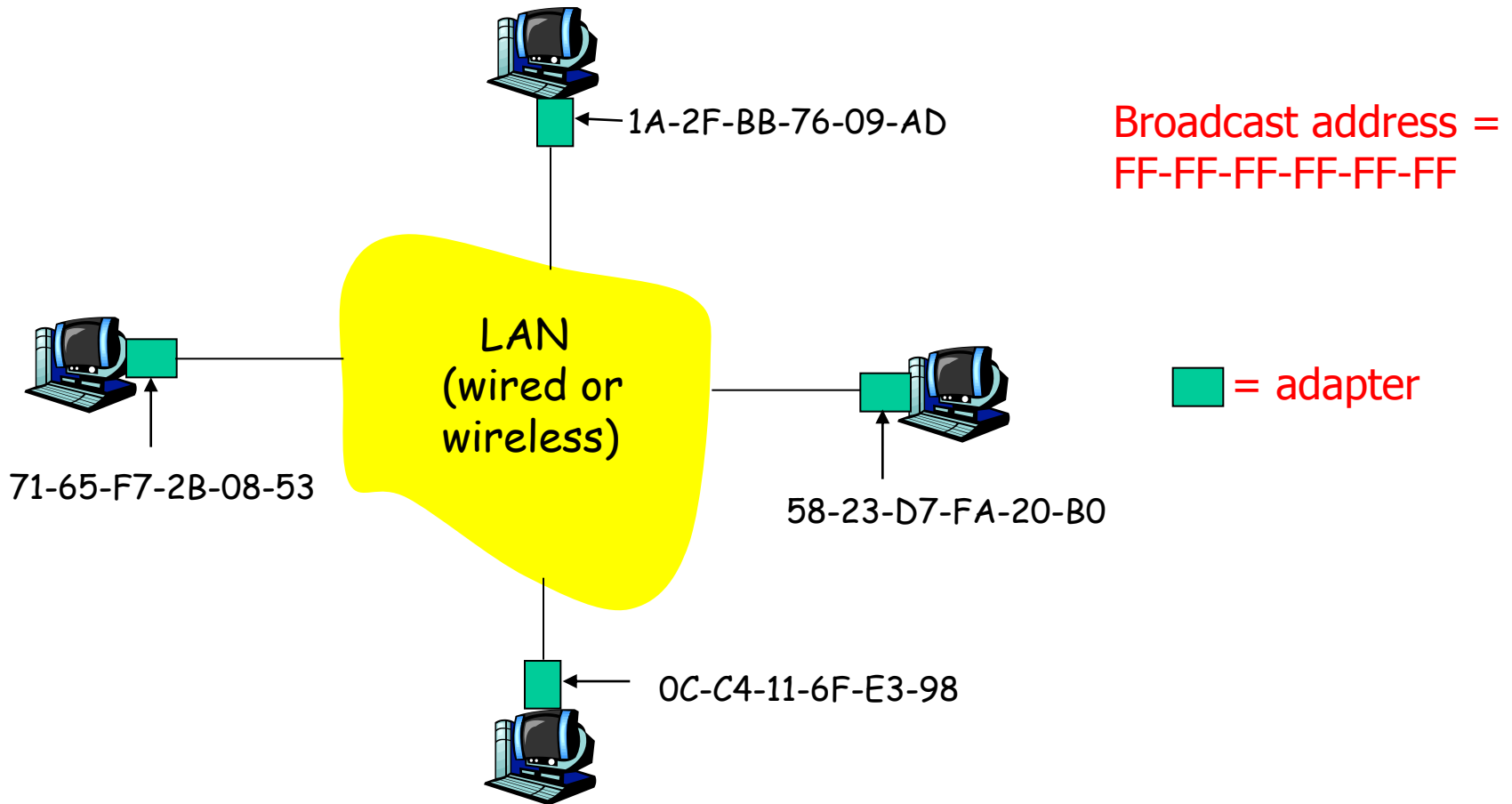
- ❑ Network-layer address
- ❑ Used to get datagram to destination network (recall IP network definition)

## MAC (or LAN or physical or Ethernet) address:

- ❑ Data link-layer address
- ❑ Used to get datagram from one interface to another physically-connected interface (same network)
- ❑ 48 bit MAC address (for most LANs)  
burned in the adapter ROM

# Addresses (2.)

Each adapter on LAN has unique LAN address



# Addresses (3.)

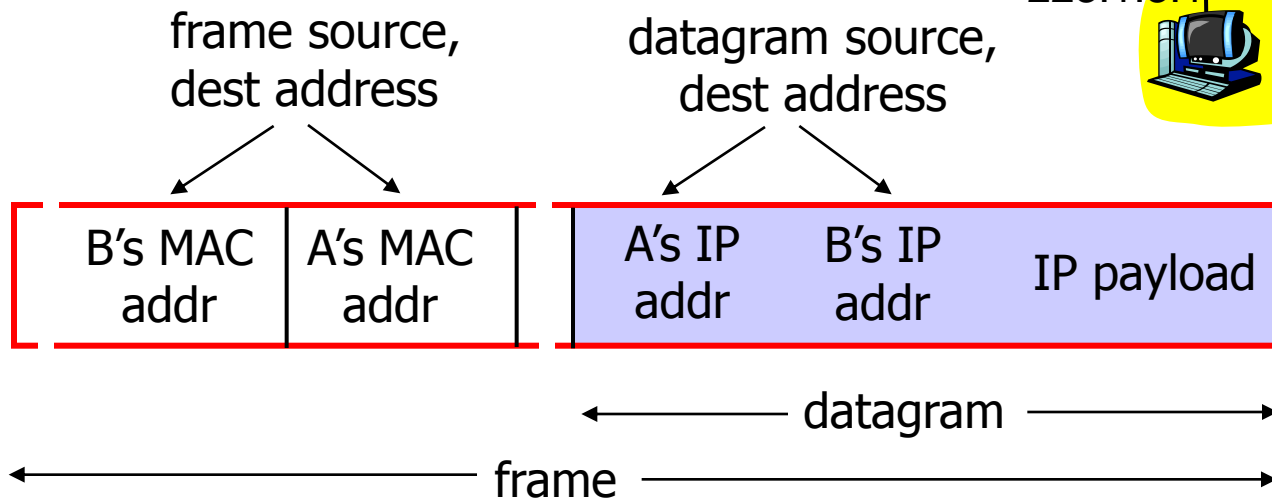
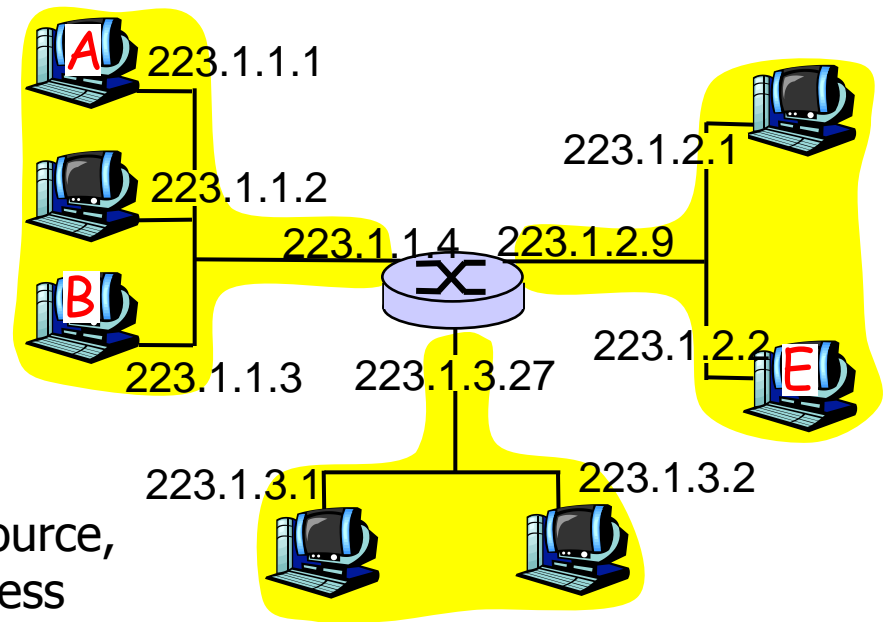
- ❑ MAC address allocation administered by IEEE
- ❑ Manufacturer buys portion of MAC address space (to assure uniqueness)
- ❑ Analogy:
  - MAC address: Like Social Security Number
  - IP address: Like postal address
- ❑ MAC flat address ⇒ portability
  - Can move LAN card from one LAN to another
- ❑ IP hierarchical address NOT portable
  - Depends on network to which one attaches



# Example

Starting at A, given IP datagram addressed to B:

- Look up net. address of B, find B on same net. as A
- **Link layer send datagram to B inside link-layer frame**



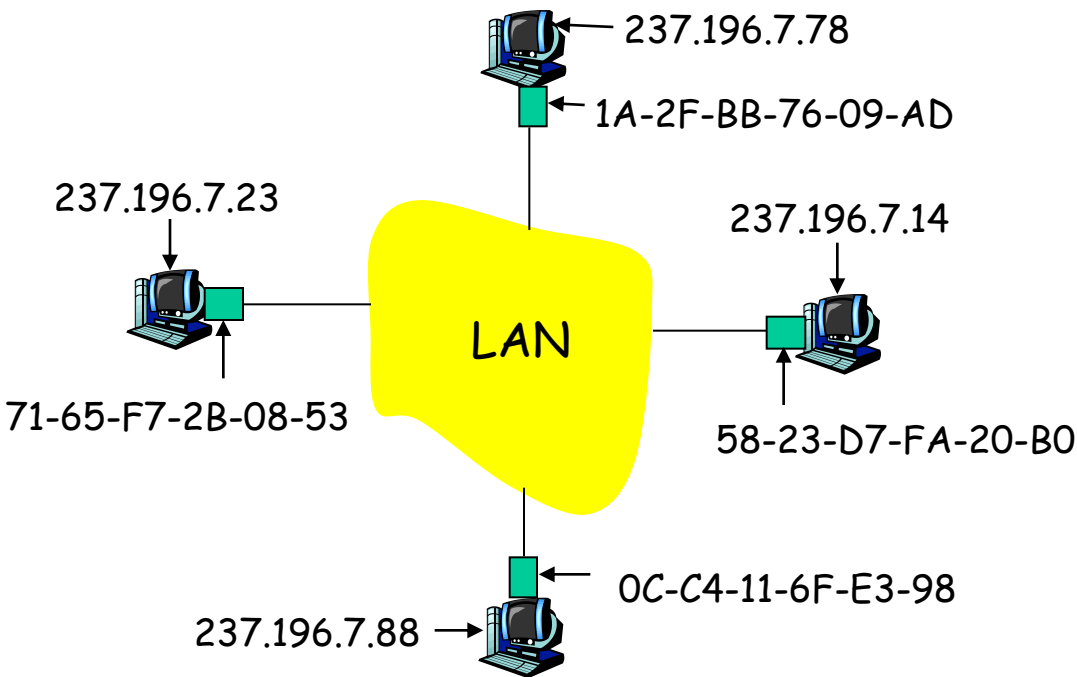
# ARP: Address Resolution Protocol

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

- Each IP node (Host, Router) on LAN has **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)



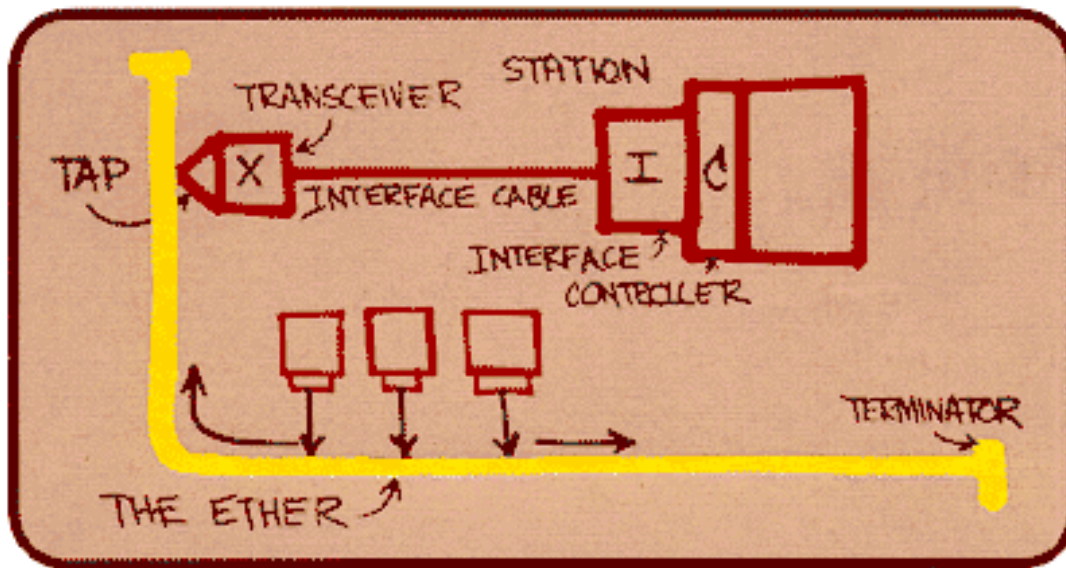
# ARP Protocol: Same LAN (Network)

- ❑ 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

# Ethernet

## “Dominant” 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
- ❑ Shared medium



Metcalfe's Ethernet sketch

# Unreliable, connectionless service

## ❑ Connectionless:

No handshaking between sending and receiving adapter.

## ❑ Unreliable:

Receiving adapter does not 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 does not 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**

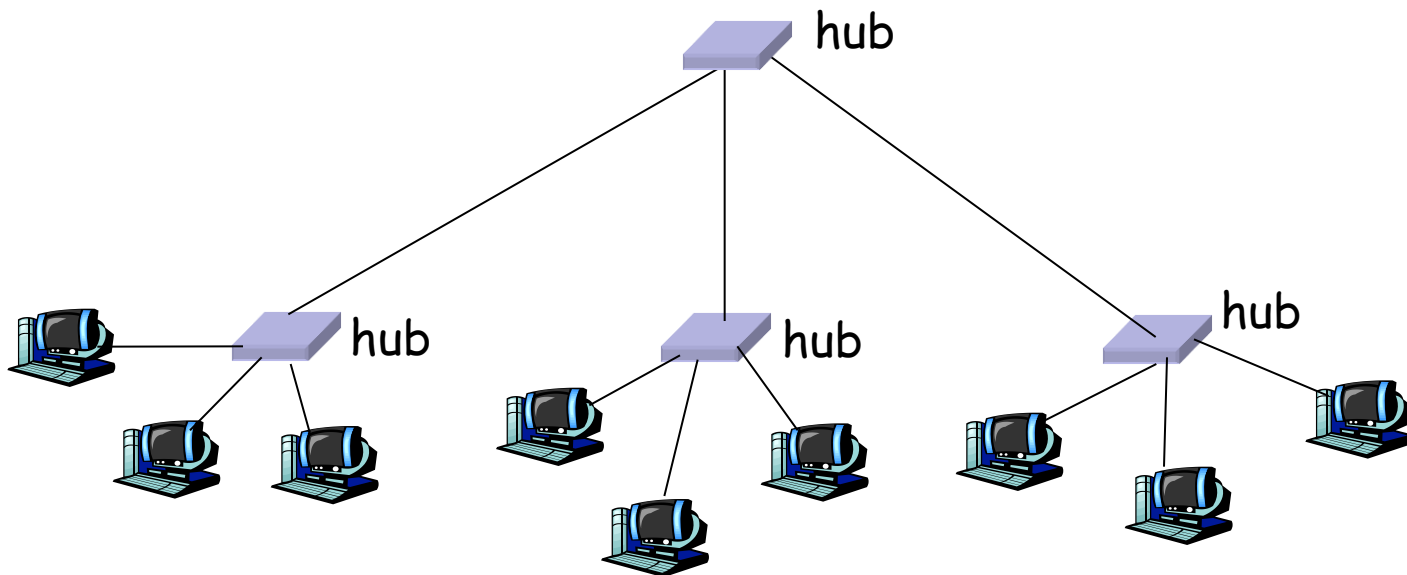
# Interconnecting LANs

Q: Why not just one big LAN?

- ❑ All stations must share bandwidth
- ❑ Limited cable length
- ❑ Large “collision domain” (can collide with many stations)
- ❑ Limited number of stations

# Interconnecting with hubs

- Physical Layer devices:  
Essentially repeaters operating at bit levels:  
Repeat received bits on one interface to all other interfaces
- Hubs can be arranged in a **hierarchy** (or multi-tier design), with **backbone** hub at its top





# Hubs (2.)

- ❑ Each connected LAN referred to as LAN **segment**
- ❑ Hubs **do not isolate** collision domains: Node may collide with any node residing at any segment in LAN
- ❑ Hub Advantages
  - Simple, inexpensive device
  - Multi-tier provides graceful degradation: portions of the LAN continue to operate if one hub malfunctions
  - Extends maximum distance between node pairs (100m per Hub)

# Bridges (switches)

## □ Link Layer devices

- Stores and forwards Ethernet frames
  - Examines frame header and **selectively** forwards frame based on MAC dst address
  - When frame is to be forwarded on segment, uses CSMA/CD to access segment
- ⇒ Bridge **isolates collision** domains: It buffers frames

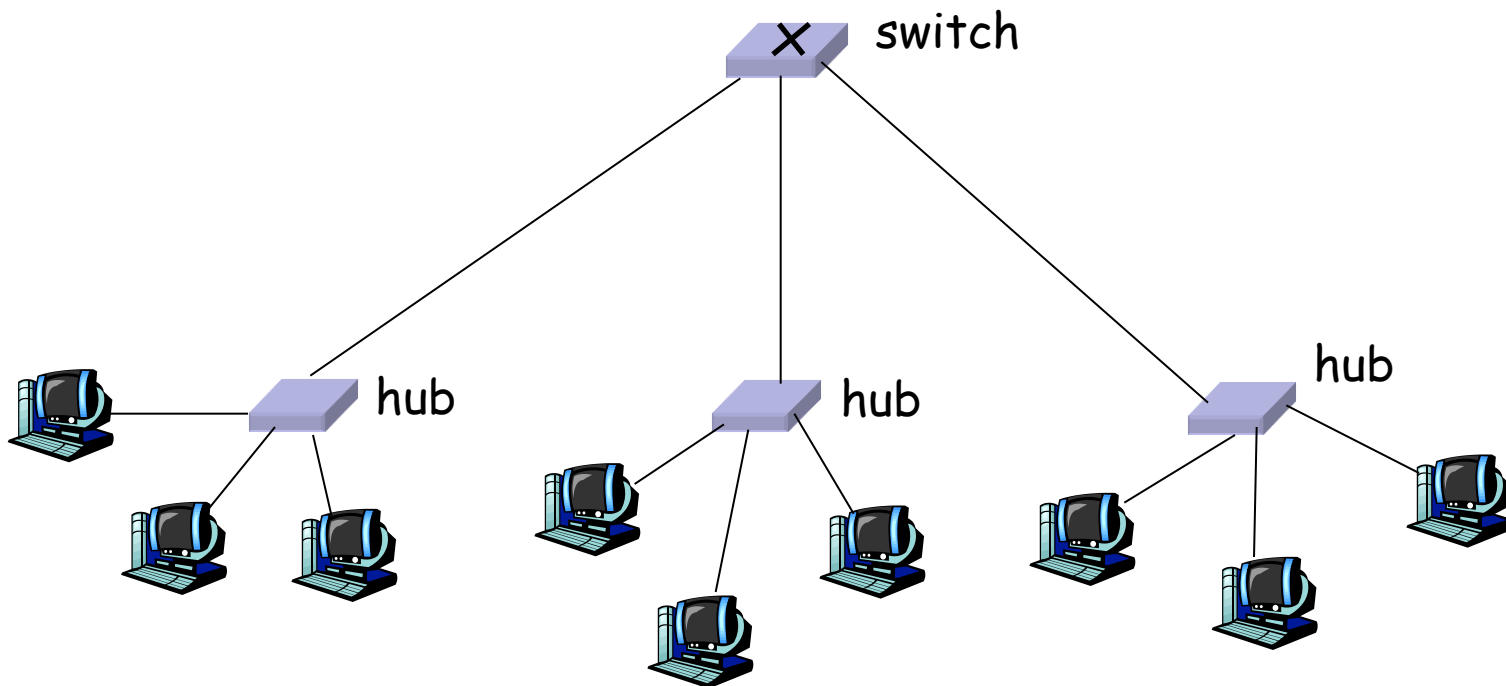
# Bridges/switch: Advantages

- ❑ Higher total max throughput
- ❑ No limit on number of nodes
- ❑ No limit on geographical coverage
- ❑ Can connect different Ethernet types (store and forward)
- ❑ Transparent: Hosts do not need to change LAN adapters
- ❑ Plug-and-play, self-learning
  - Switches do not need to be configured

# Bridges/switch: Forwarding

## □ Forwarding:

- To which LAN segment should a frame be forwarded?
- Looks like a routing problem



# Bridges/switch: Self learning

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

# Bridges/switch: 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

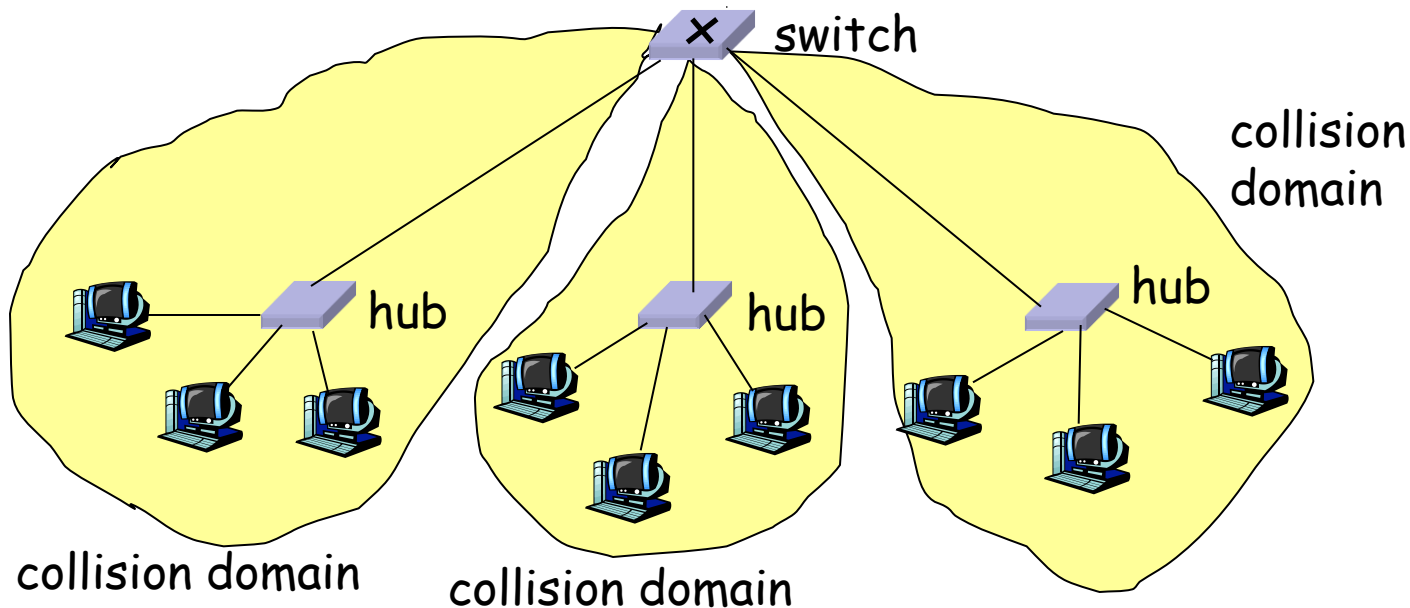
}

**else** flood

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

# Switch: 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**



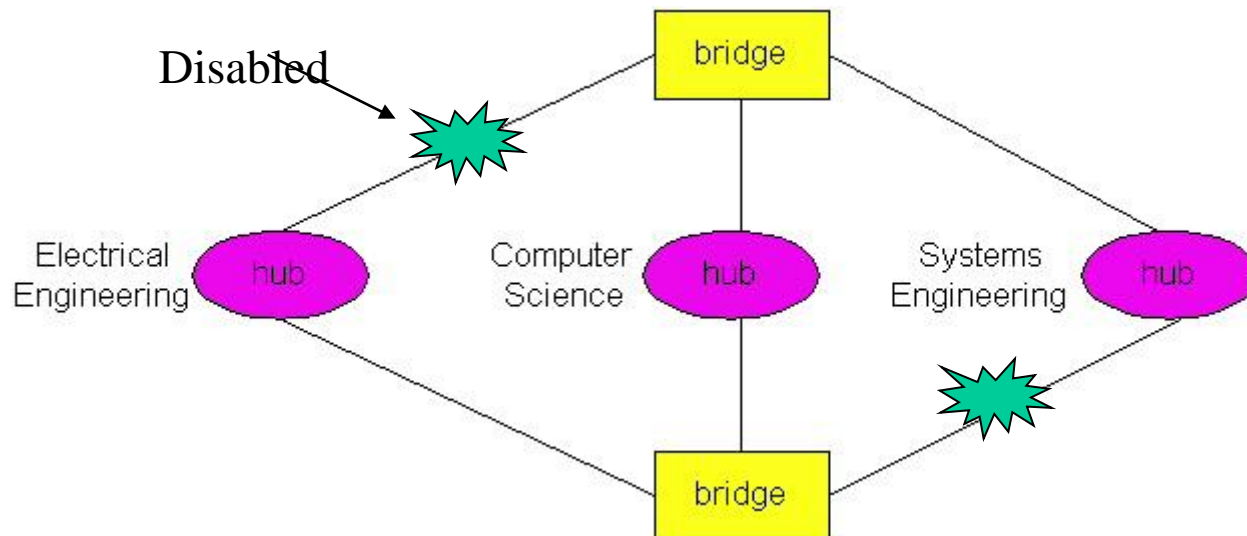
# Redundant networks

- ❑ Network with multiple paths
  - Alternate path for each source, destination pair
- ❑ Advantage
  - Increased reliability
  - Single network failure OK
  - More opportunities for load distribution
- ❑ Disadvantage
  - Added complexity



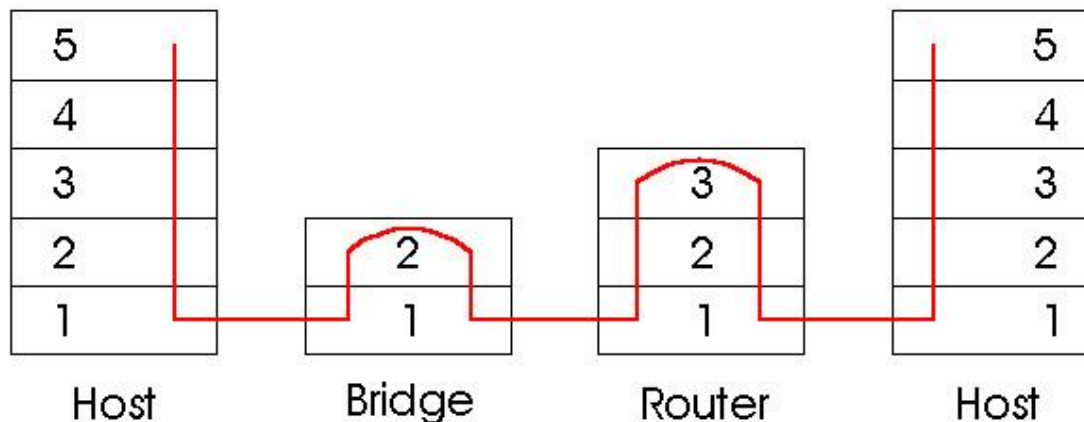
# Bridges spanning tree

- ❑ Avoid cycles
  - Frames may multiply and forwarded forever
- ❑ Organize bridges into spanning tree
  - Disable a subset of interfaces



# Bridges vs. Routers

- ❑ Both store-and-forward devices
  - Routers: Network layer devices (examine network layer headers)
  - Bridges/switches: Link layer devices
- ❑ Use tables
  - Routers: Routing tables via routing algorithms
  - Bridges: Filtering tables via filtering, learning, spanning tree algorithm



# Bridges + and -

- + Simple operation
  - Low processing bandwidth
- Restricted topologies:
  - Spanning tree to avoid cycles
- Single broadcast domain
  - No protection from broadcast storms  
(broadcasts will be forwarded by bridge)

# Routers + and -

## + Arbitrary topologies

Limited cycling (TTL and good routing protocols)

## + Firewalls protection

Against broadcast storms

## - Complex operation

Require IP address configuration (not plug and play)

Require higher processing bandwidth

# Routers vs. Bridges

## ❑ Bridges

- Good in small networks (few hundred hosts)

## ❑ Routers

- Good in large networks (thousands of hosts)

## ❑ Layer 3 switch

- Bridge + router (but usually limited routing table!)

# 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