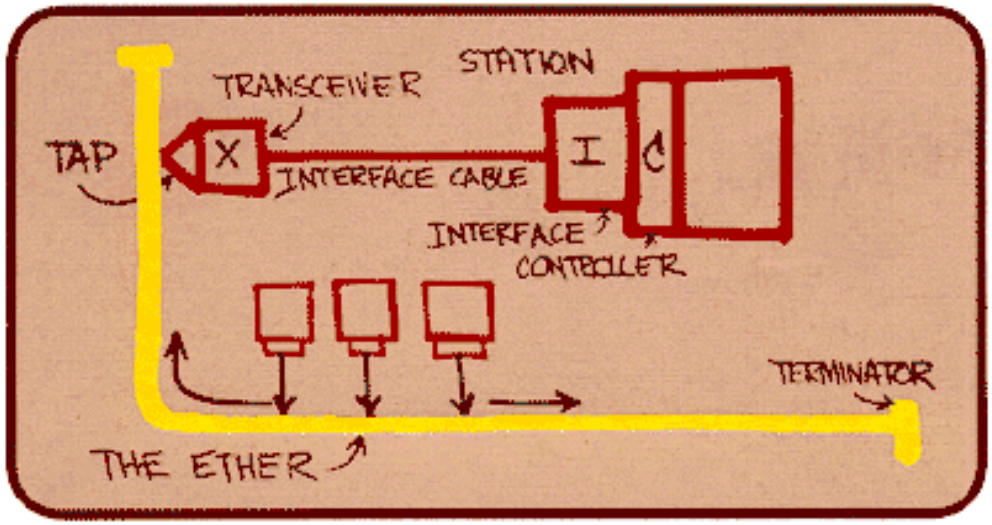
Local Area Networks

Richard T. B. Ma
School of Computing
National University of Singapore

CS 3103: Compute Networks and Protocols

Ethernet

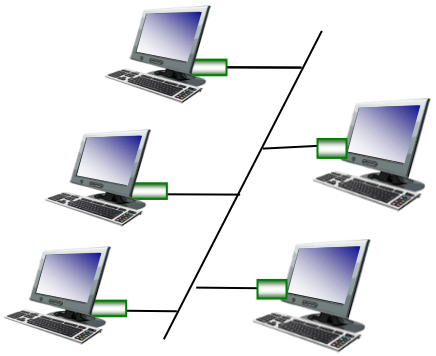
- invented mid-70s'
- competed with
  - token ring
  - FDDI
  - ATM
- "dominant" wired LAN technology:
  - cheap $20 for NIC
  - first widely used LAN technology
  - simpler, cheaper than token LANs and ATM
  - kept up with speed race: 10 Mbps – 10 Gbps



Metcalfe's Ethernet sketch

Ethernet: physical topology

- *Bus topology:* popular through mid 90s
  - Broadcast LAN: all nodes in same collision domain (can collide with each other)
  - Need MAC protocol: unslotted *CSMA/CD wth binary backoff*

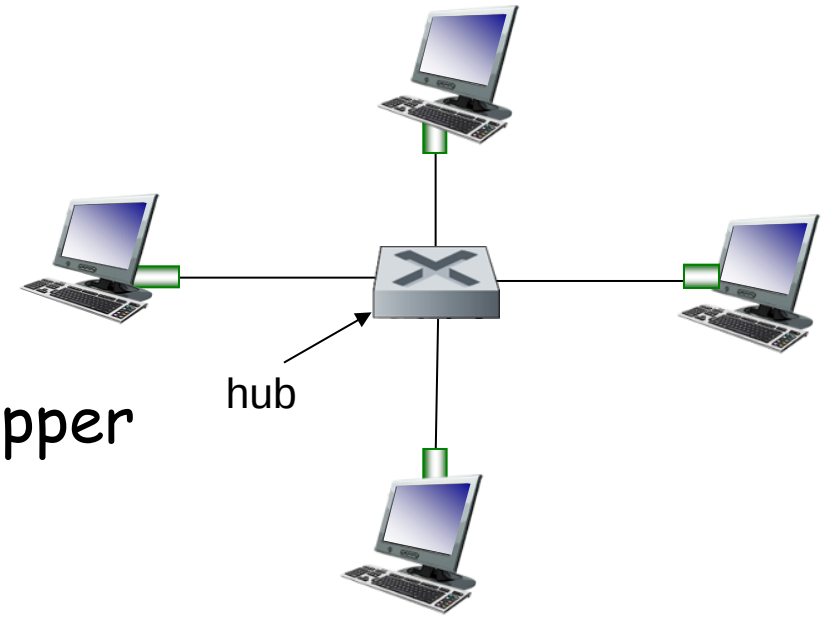


bus: coaxial cable

Ethernet: physical topology

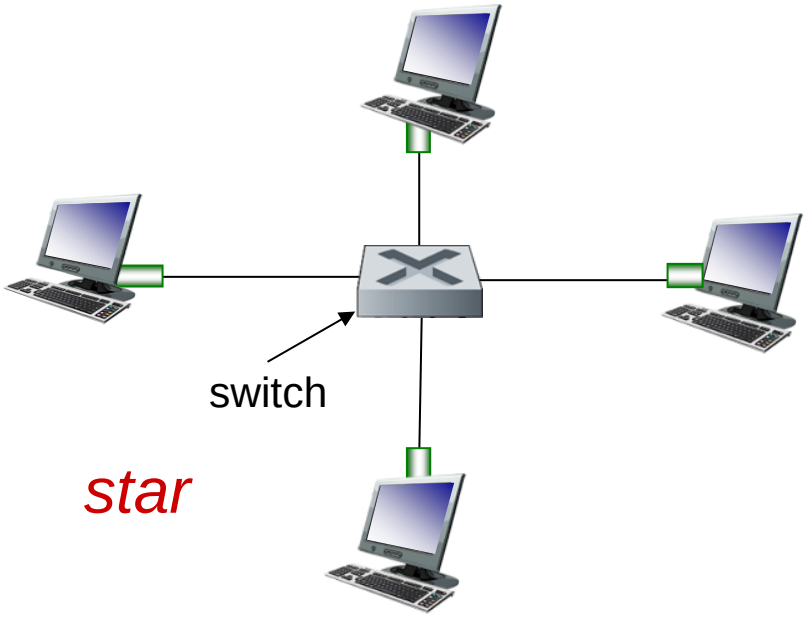
- *Star topology with Hub:* before 2000
  - passive hub in center
  - repeater: re-creates bits, boosts energy strength, and transmits bits onto all the other interfaces
  - acts on individual bits rather than frames

Star: twisted-pair copper



Ethernet: physical topology

- *Star topology with Switch:* prevails today
  - active *switch* in center
  - store-and-forward packets
  - each "spoke" runs a (separate) Ethernet protocol (nodes do not collide with each other)

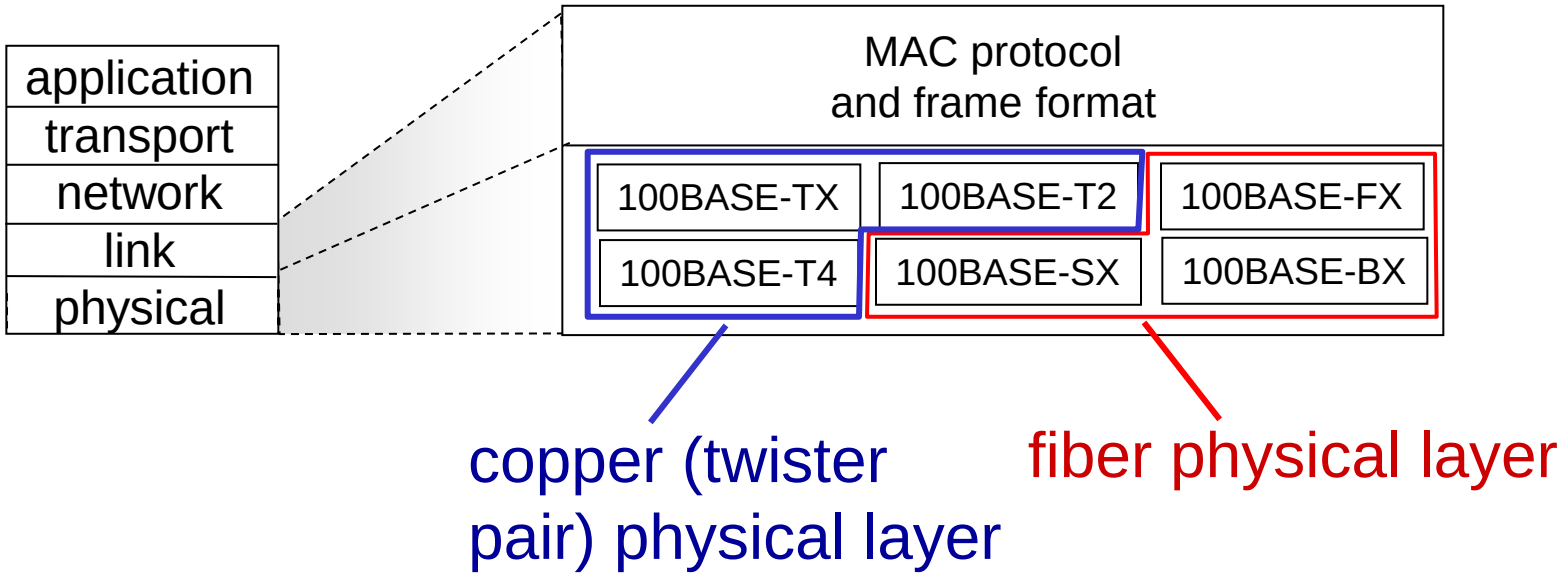


Ethernet: unreliable, connectionless

- *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

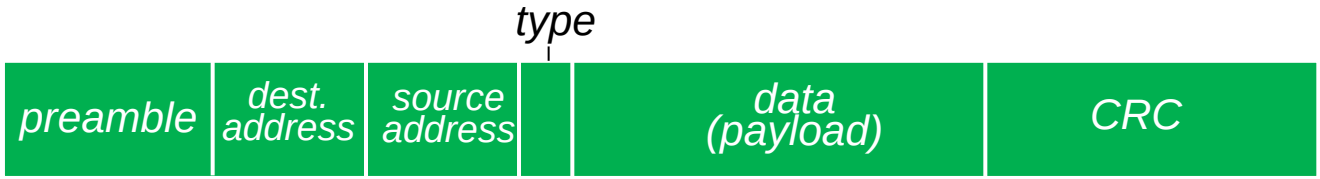
802.3 Ethernet standards: link & physical layers

- *many* different Ethernet standards
  - common MAC protocol and frame format
  - different speeds: 2 Mbps, 10 Mbps, 100 Mbps, 1Gbps, 10G bps
  - different physical layer media: fiber, cable



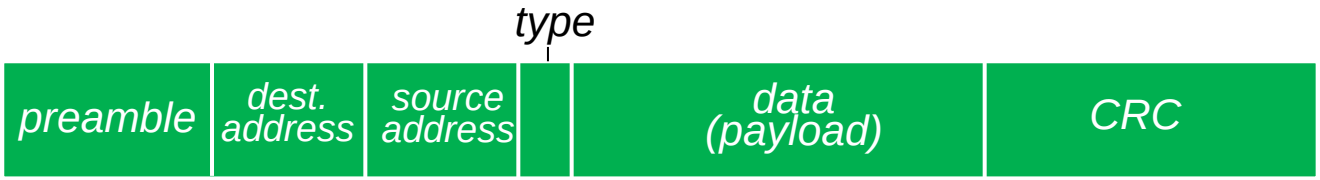
Ethernet frame structure

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



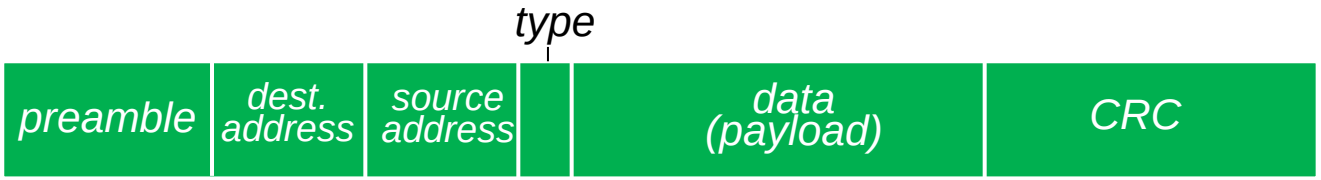
- *preamble:* 7 bytes with pattern 10101010 followed by one byte with pattern 10101011
  - used to synchronize receiver/sender clock rates
- *CRC:* 4-byte cyclic redundancy check
  - error detected at receiver: frame is dropped

Ethernet frame structure (more)



- *addresses:* 6-byte source, destination MAC
  - if receives frame with matching destination address, or with broadcast address (e.g. ARP packet), it passes data in frame to network layer
  - otherwise, adapter discards frame
- *type:* 2-byte indicates higher layer protocol
  - multiplex network-layer protocols
  - (mostly IP but others possible, e.g., Novell IPX)
  - Also ARP has its own type number

Ethernet frame structure (more)



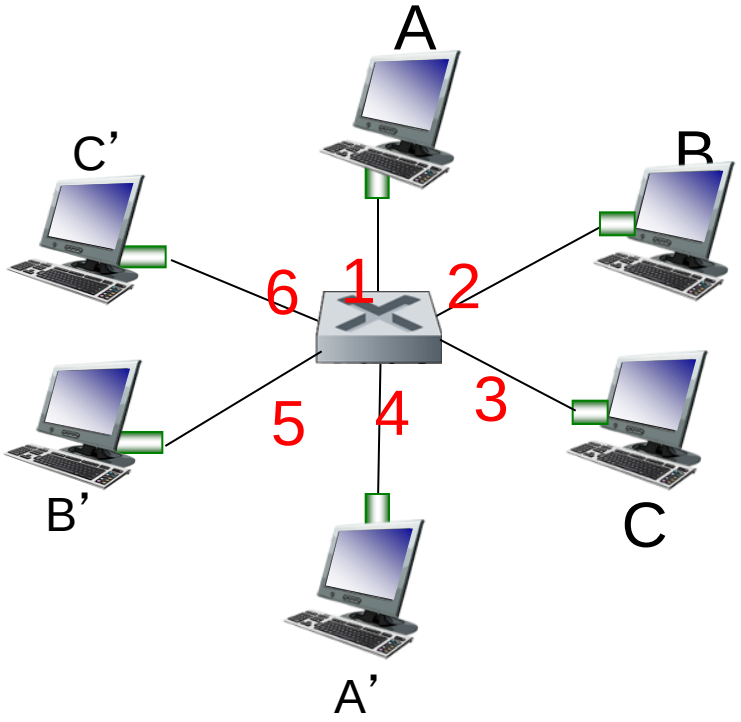
- *data:* carries IP datagram (46 to 1,500 bytes)
  - The maximum transmission unit (MTU) of Ethernet is 1,500 bytes. If the IP datagram exceeds 1,500 bytes, then the host has to fragment the datagram
  - The minimum size of the data field is 46 bytes. If the IP datagram is less than 46 bytes, the data field has to be "stuffed" to fill it out to 46 bytes
  - The network layer uses the length field in the IP datagram header to remove the stuffing

Ethernet switch

- link-layer device: takes an *active* role
  - store, forward Ethernet frames
  - examine incoming frame's MAC address, selectively forward frame to one-or-more outgoing links when frame is to be forwarded on segment
- *transparent*
  - hosts are unaware of presence of switches
- *plug-and-play, self-learning*
  - switches do not need to be configured

Link Layer Switch: *multiple* simultaneous transmissions

- hosts have dedicated, direct connection to switch
- switches buffer packets
- Ethernet protocol used on *each* incoming link, but no collisions; full duplex
  - each link is its own collision domain
- *switching:* A-to-A' and B-to-B' can transmit simultaneously, without collisions



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

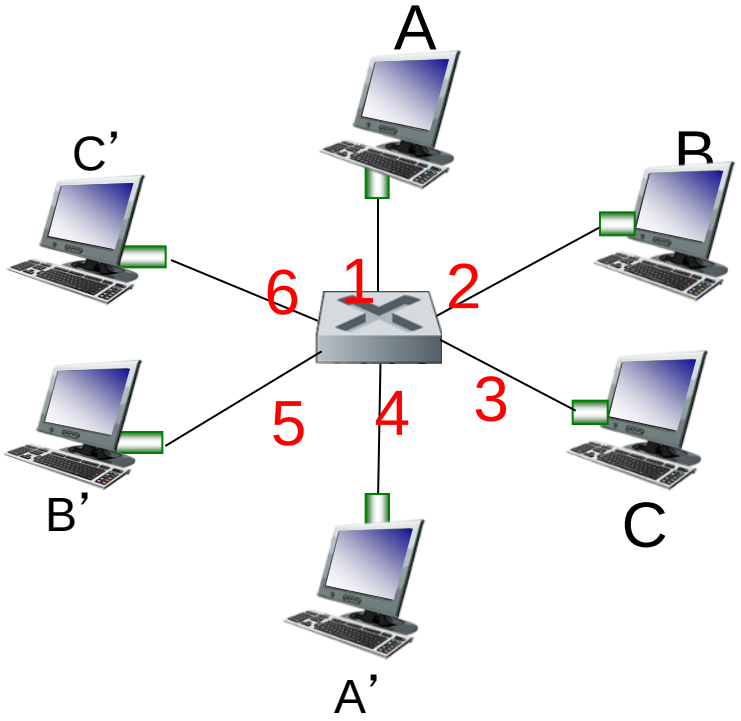
Switch forwarding table

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

- A: each switch has a switch table, each entry:
  - (MAC address of host, interface to reach host, time stamp)
  - looks like a routing table!

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

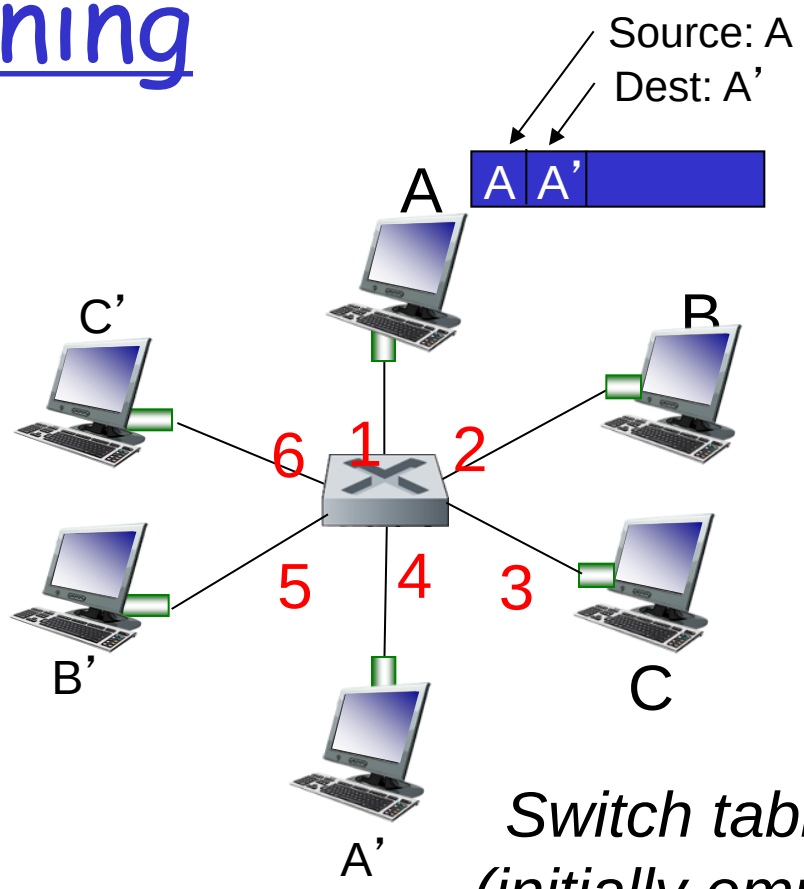
- something like a routing protocol?



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

Switch: self-learning

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



Switch table (initially empty)

MAC addr	interface	TTL
A	*1*	*60*

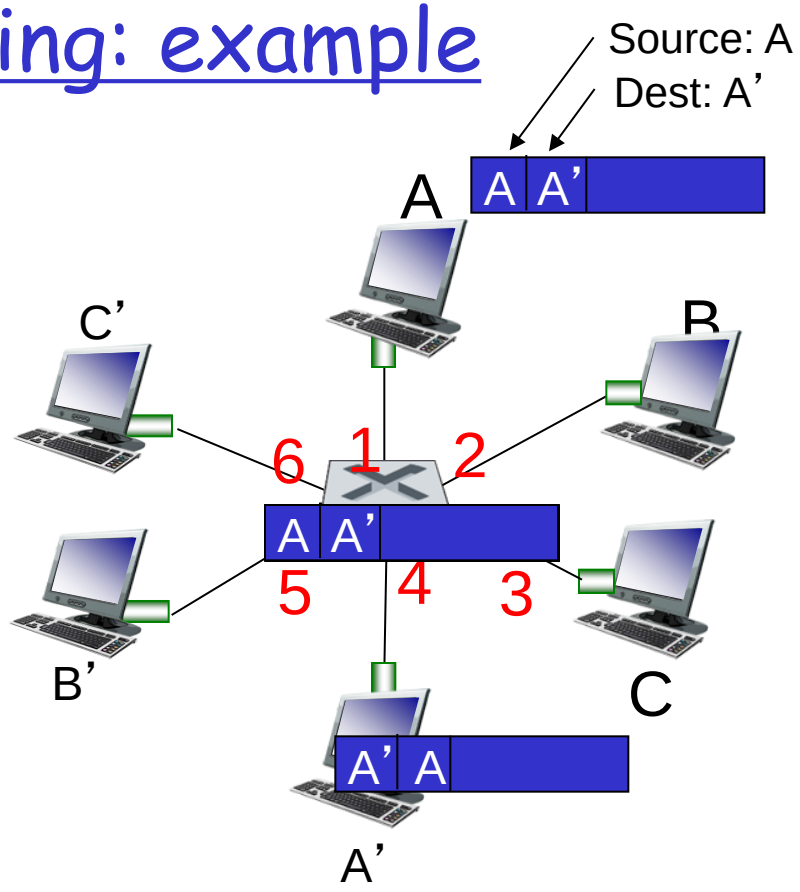
Switch: frame filtering/forwarding

when frame received at switch:

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

Self-learning, forwarding: example

- frame destination, A', location unknown:
  - ➔ *flood*
- destination A location known:
  - ➔ selectively send on just one link

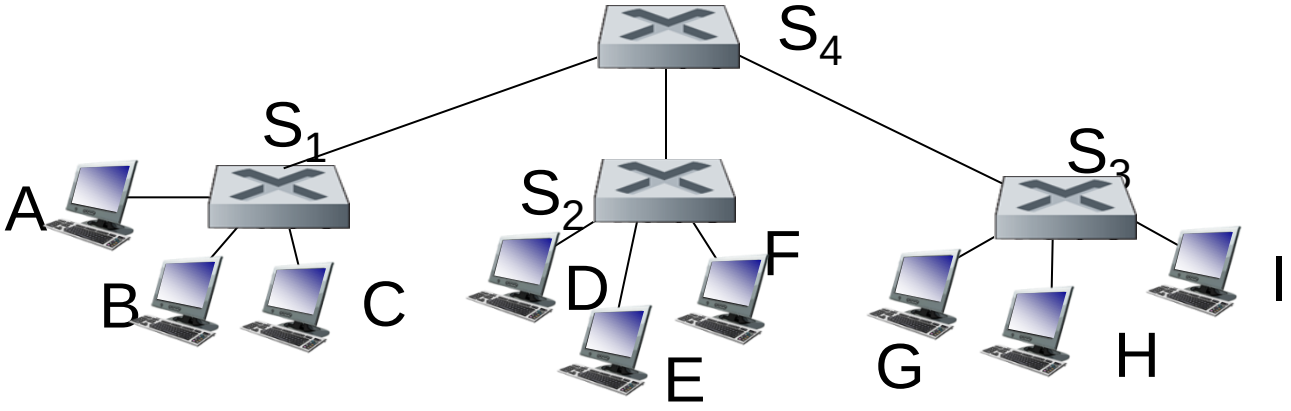


MAC addr	interface	TTL
A	1	60
A'	4	60

switch table (initially empty)

Interconnecting switches

- switches can be connected together

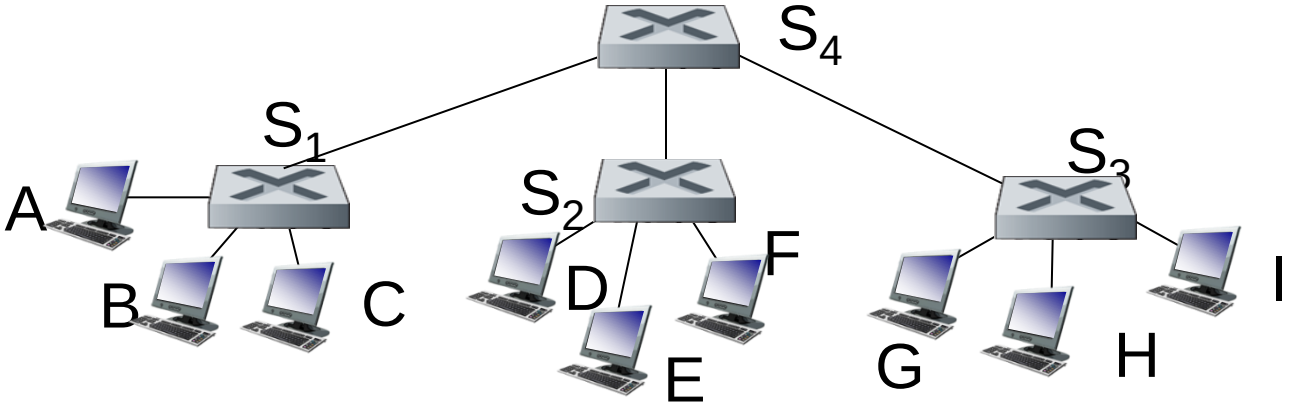


Q: sending from A to G - how does S_1 know to forward frame destined to G via S_4 and S_3?

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

Self-learning multi-switch example

Suppose C sends frame to I, I responds to C



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

Switch's Advantages Over Hub

- Only forwards frames as needed
  - filters frames to avoid unnecessary load
- Extends the geographic span of the network
  - separate segments allow longer distances
- Improves privacy by limiting scope of frames
  - A host can "snoop" the traffic traversing its link
- Self-learning and "plug-and-play"
  - Require no intervention from a network admin
- Can join segments using different technologies
  - Cooper, fiber and other links

Switch's Disadvantages

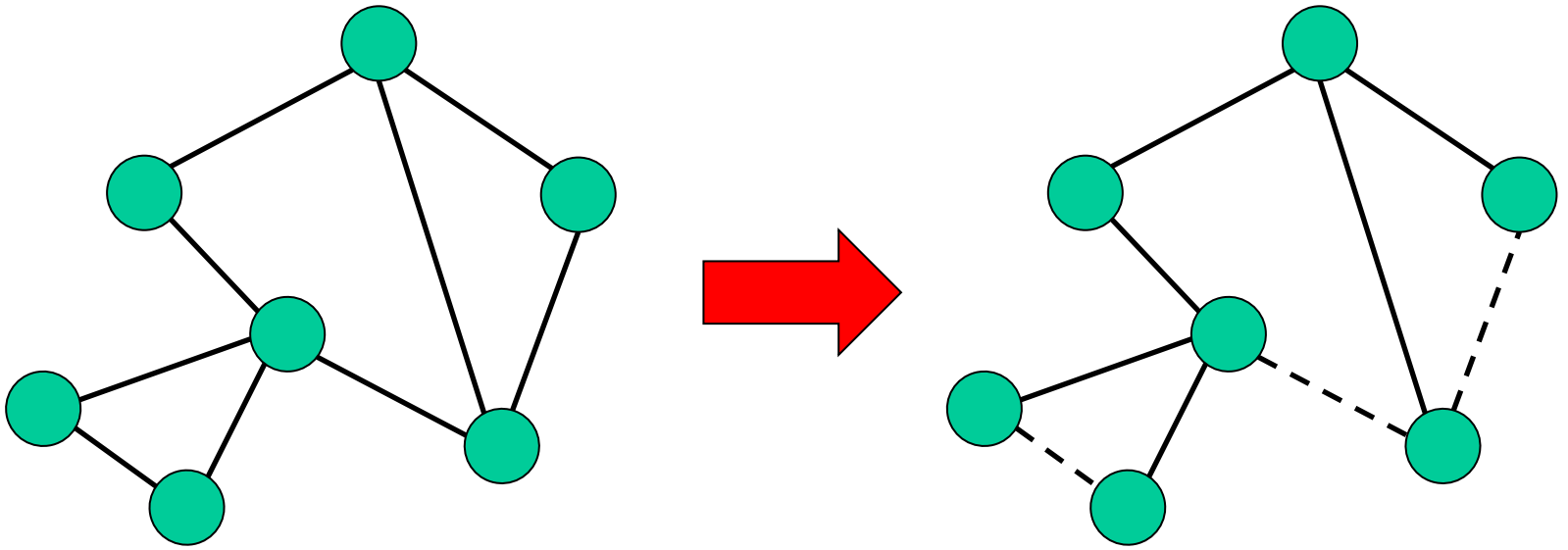
- Delay in forwarding frames
  - switch must receive and parse the frame and perform a look-up to decide where to forward
  - storing and forwarding the packet introduces delay
- Need to learn where to forward frames
  - switch needs to construct a forwarding table
  - Ideally, without intervention from network administrators
- Higher cost
  - More complicated devices that cost more money

Cycles and Broadcast Storm

- LANs may form cycles
  - either accidentally, or by design, for higher reliability
  - use of flooding can lead to forwarding loops
  - causing "braodcast storm"
- To prevent broadcast storm, switches need to avoid some links when flooding, so as not to form a loop
- Switches compute a spanning tree of the network

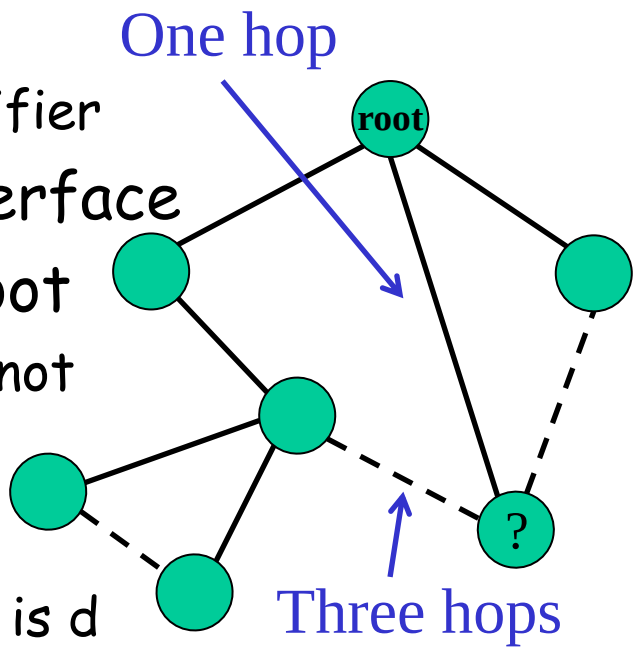
Spanning Tree

- A sub-graph that covers all nodes, but contains no cycle
- To avoid loops, links not in the spanning tree do not forward frames



Constructing a Spanning Tree

- Need a distributed algorithm
  - switches cooperate to build the spanning tree
  - and adapt automatically when failure occur
- Key ingredients of the algorithm
  - switches need to elect a "root"
    - the switch with the smallest identifier
  - each switch identifies if its interface is on the shortest path from root
    - exclude from the spanning tree if not
  - Messages (Y, d, X)
    - from node X
    - claiming Y is the root and distance is d

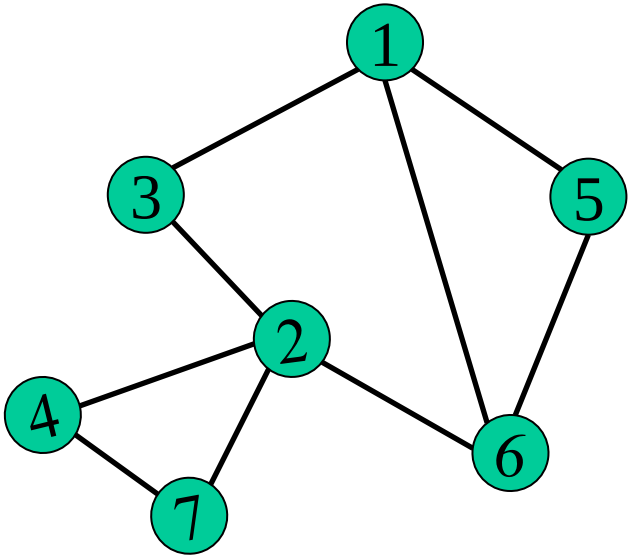


Steps in Spanning Tree Algorithm

- Initially, each switch thinks it is the root
  - Switch sends a message out every interface identifying itself as the root with distance 0
- Switches update their view of the root
  - Upon receiving a message, check the root id
  - If the new id is smaller, view that switch as root
- Switches compute their distance from root
  - Add 1 to the distance received from a neighbor
  - Identify interfaces not on a shortest path to the root and exclude them from the spanning tree

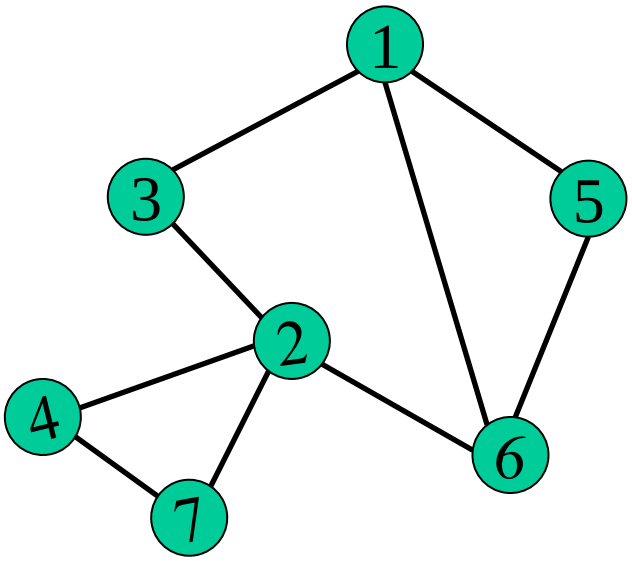
Example from Switch 4's Viewpoint

- Switch 4 thinks it is the root
  - Sends (4, 0, 4) message to 2 and 7
- Then, switch 4 hears from 2
  - Receives (2, 0, 2) message from 2, thinks that 2 is the root and realizes it is just one hop away
- Then, switch 4 hears from 7
  - Receives (2, 1, 7) from 7 and realizes this is a longer path
  - Thus prefers its own one-hop path and removes 4-7 link from the tree



Example from Switch 4's Viewpoint

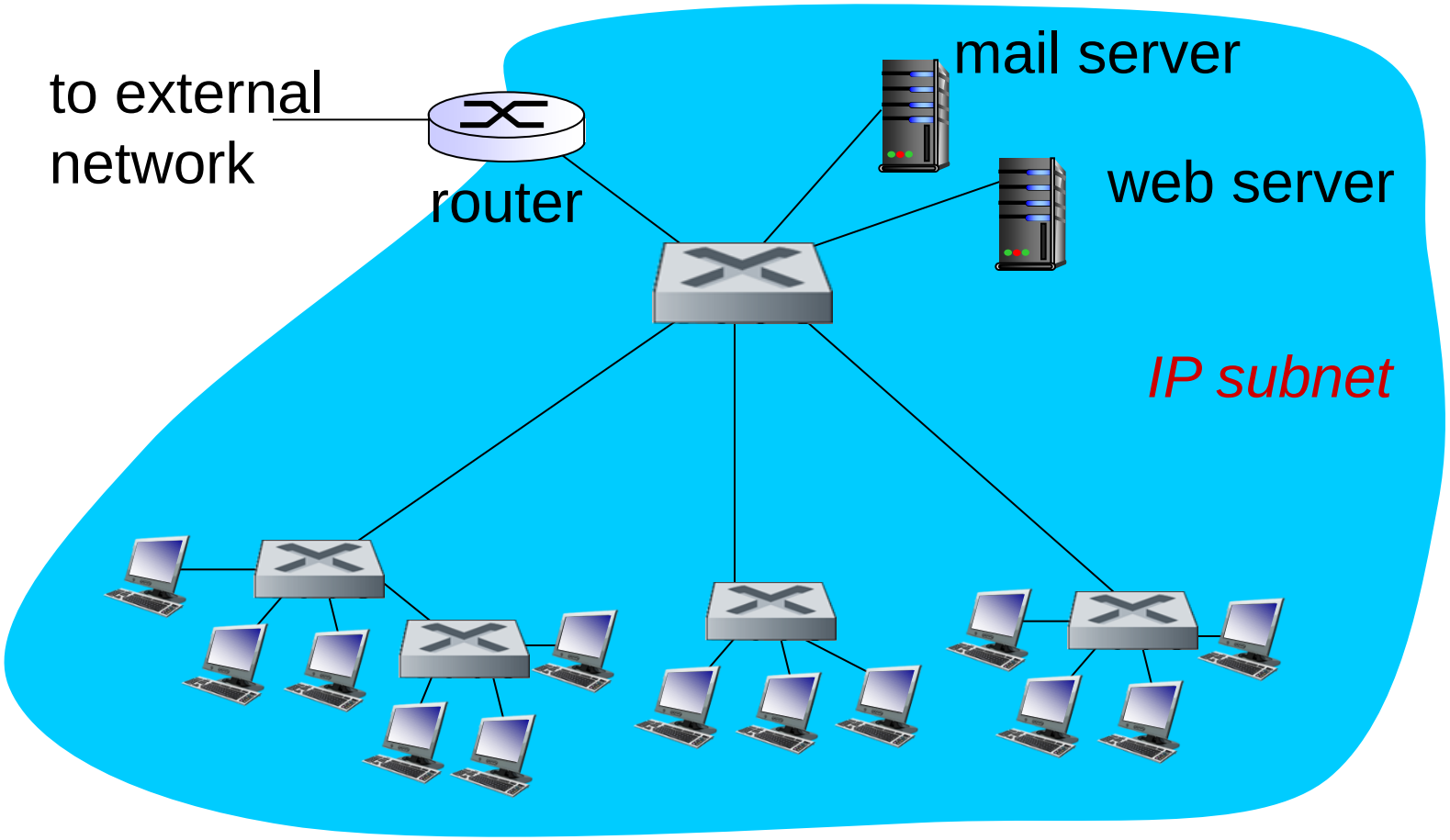
- Switch 2 hears about switch 1
  - Switch 2 hears (1, 1, 3) from 3
  - Switch 2 starts treating 1 as root and sends (1, 2, 2) to neighbors
- Switch 4 hears from switch 2
  - Switch 4 starts treating 1 as root and sends (1, 3, 4) to neighbors
- Switch 4 hears from switch 7
  - Switch 4 receives (1, 3, 7) from 7 and realizes this is a longer path
  - So, prefers its own three-hop path and removes 4-7 link from the tree



Robust Spanning Tree Algorithm

- Algorithm must react to failures
  - Failure of the root node
    - Need to elect a new root, with the next lowest identifier
  - Failure of other switches and links
    - Need to re-compute the spanning tree
- Root switch continues sending messages
  - Periodically announcing itself as the root (1, 0, 1)
  - Other switches continue forwarding messages
- Detecting failures through
  - Switch waits to hear from others
  - Eventually times out and claims to be the root

Institutional network



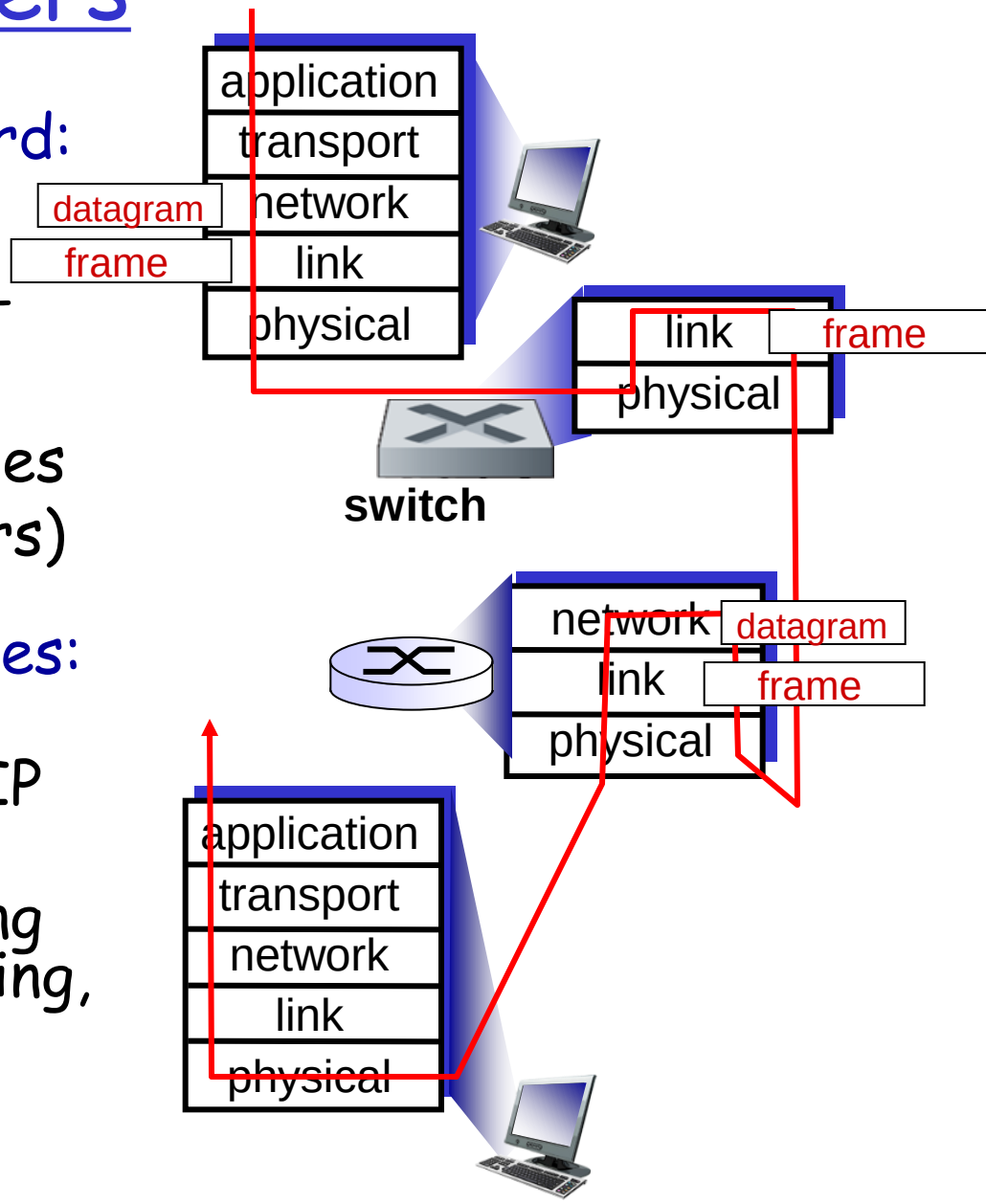
Switches vs. routers

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

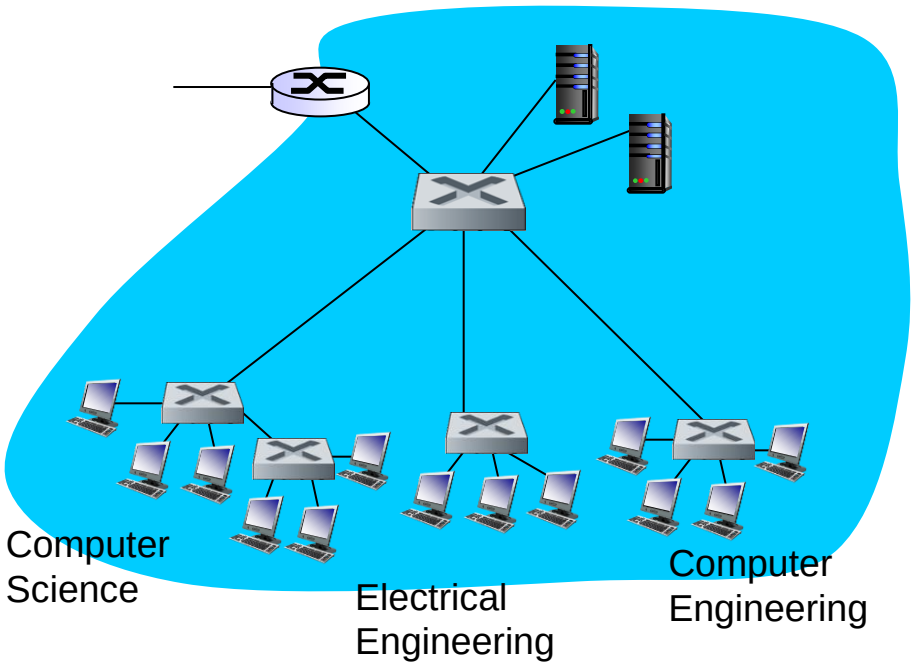


Hubs Vs. Switches Vs. Routers

- Comparison of the typical features of popular interconnection devices

	Hubs	Routers	Switches
Traffic isolation	No	Yes	Yes
Plug and play	Yes	No	Yes
Optimal routing	No	Yes	No

VLANs: motivation



consider:

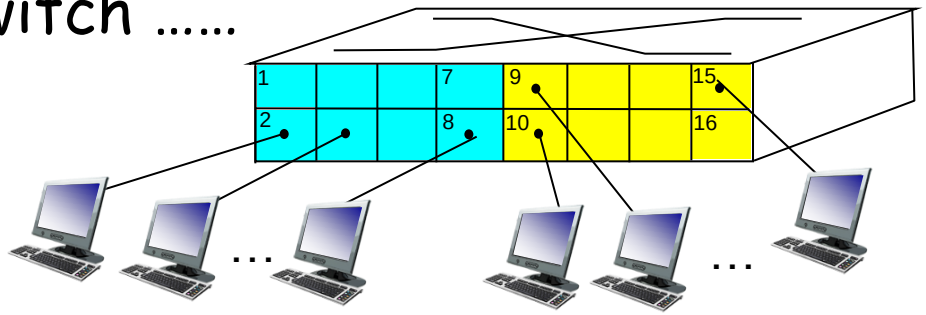
- CS user moves office to EE, but wants connect to CS switch?
- single broadcast domain:
  - all layer-2 broadcast traffic (ARP, DHCP, unknown location of destination MAC address) must cross entire LAN
  - security/privacy, efficiency issues

VLANs

Virtual Local Area Network

switch(es) supporting VLAN capabilities can be configured to define multiple ***virtual*** LANS over single physical LAN infrastructure.

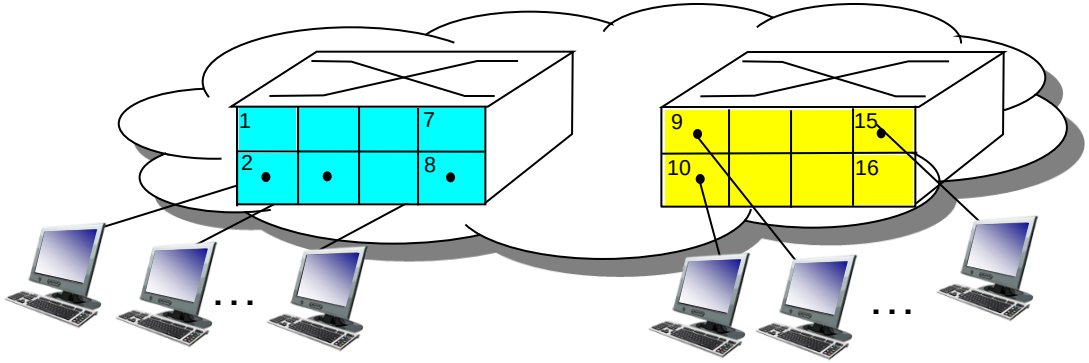
port-based VLAN: switch ports grouped (by switch management software) so that *single* physical switch ……



Electrical Engineering (VLAN ports 1-8)

Computer Science (VLAN ports 9-15)

… operates as *multiple* virtual switches

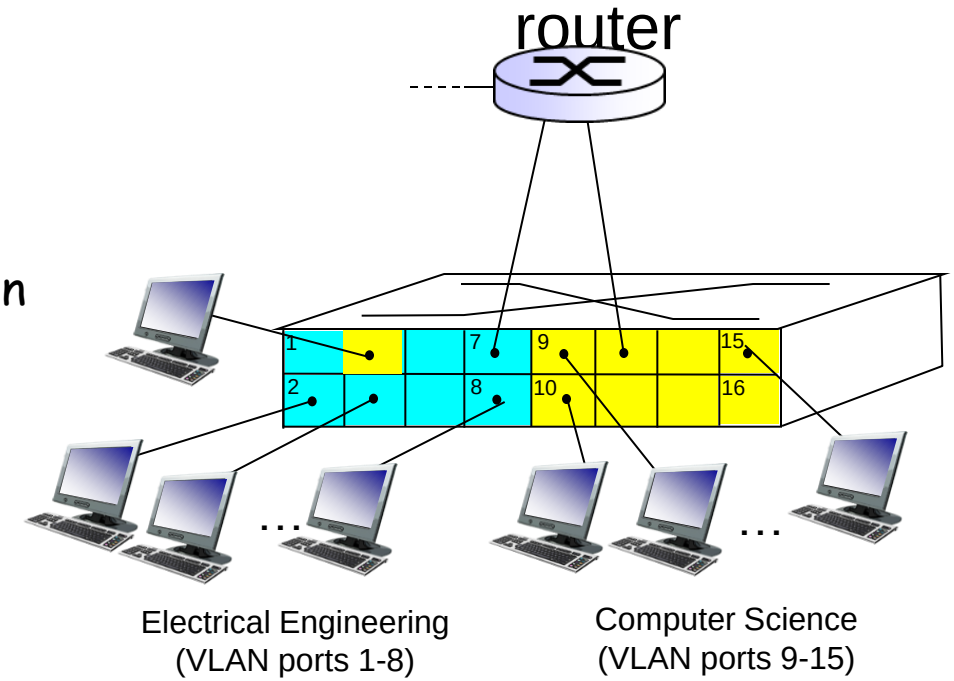


Electrical Engineering (VLAN ports 1-8)

Computer Science (VLAN ports 9-16)

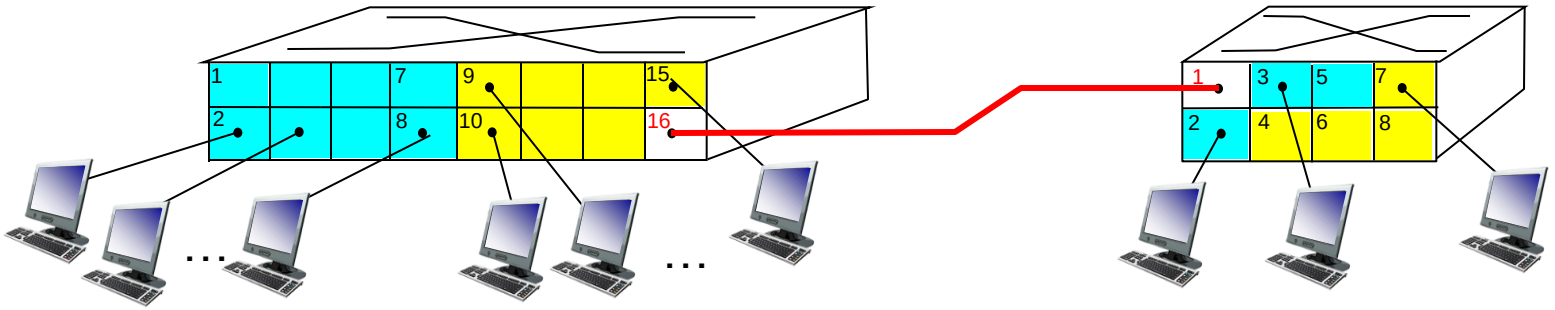
Port-based VLAN

- *traffic isolation:* frames to/from ports 1-8 can *only* reach ports 1-8
  - can also define VLAN based on MAC addresses of endpoints, rather than switch port
- dynamic membership: ports can be dynamically assigned among VLANs



- forwarding between VLANS: done via routing (just as with separate switches)
  - in practice vendors sell combined switches plus routers

VLANS spanning multiple switches



Electrical Engineering
(VLAN ports 1-8)

Computer Science
(VLAN ports 9-15)

Ports 2,3,5 belong to EE VLAN
Ports 4,6,7,8 belong to CS VLAN

- *trunk port:* carries frames between VLANS defined over multiple physical switches
  - frames forwarded within VLAN between switches can't be vanilla 802.1 frames (must carry VLAN ID info)
  - 802.1q protocol adds/removed additional header fields for frames forwarded between trunk ports

802.1Q VLAN frame format