Computer Networking

COMP 177 | Fall 2020 | University of the Pacific | Jeff Shafer

Ethernet

Recap

Past Topics

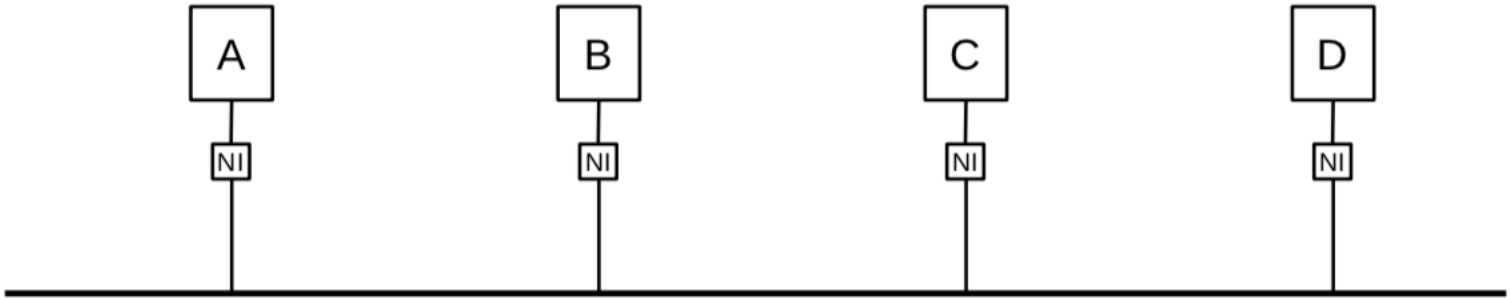
- An overview of computer networking
- Wireshark

Today's Topics

- Ethernet
- Hubs
- Switches
- Packet format in Ethernet
- MAC addresses

Classic Ethernet

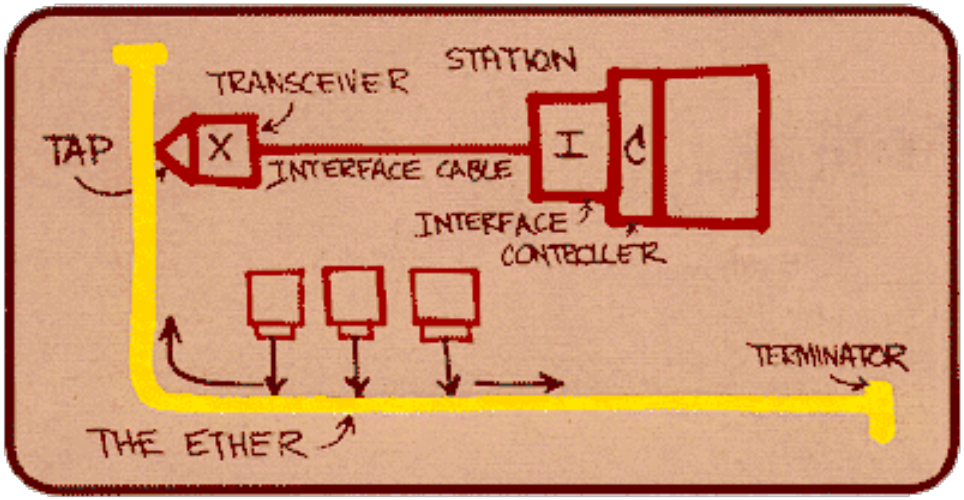
- Ethernet is an IEEE standard for wired LANs - IEEE 802.3
- A *data link* layer protocol, proposed in 1970s
- Initially was a *bus topology* for LANs
  - a long cable to which all devices are attached signal could be attenuated
  - Solution: coaxial cable rather than twisted pair copper wire!



Classic Ethernet

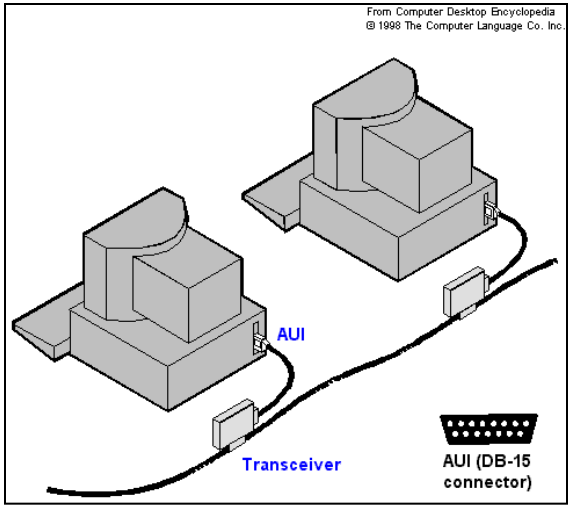
- Each device is connected to the bus through a hardware component called *network interface controller* (*NIC*)
- Each node in the LAN *broadcasts* its packet (called *Ethernet frame*) over the bus
- All NICs on a LAN can receive a transmitted packet
- A NIC decides whether to send the received packet to the operating system. How?
  - Each Ethernet packet includes physical address of the destination NIC
  - Upon receiving a packet, NIC checks whether that address matches its own
  - If so, passes the packet to OS. Otherwise, drops it!
- Ethernet physical addresses are called *MAC addresses*
  - Medium Access Control

Classic Ethernet



Original picture drawn by Bob Metcalfe, inventor of Ethernet
(1972 – Xerox PARC)

Ether – 19th century name for media enabling the propagation of light



Collision Detection

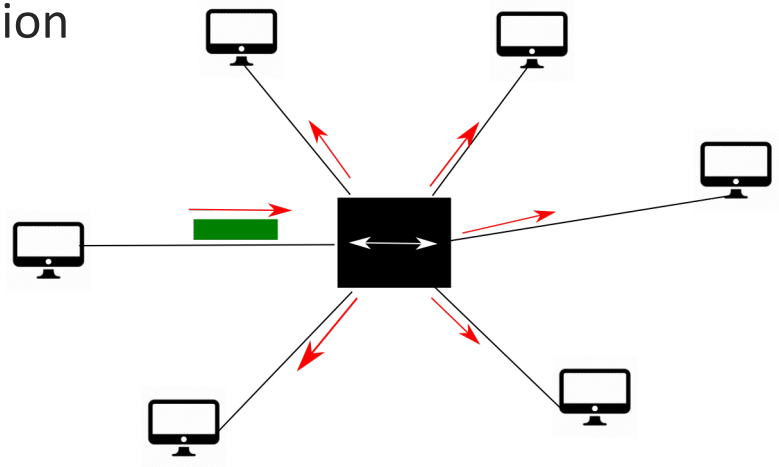
- Broadcasting packets in a shared medium, e.g., a bus, may end in collision
  - If two packets collide, both transmissions fail

Animation from http://www.datacottage.com/nch/eoperation.htm

- Solution: *Carrier Sense, Multiple Access, with Collision Detection* (CSMA/CD)
  - Before transmission, wait for the line to be quiet
  - While transmitting, monitor the line
  - If collision detected, wait ("back off"), and then retransmit when quiet again

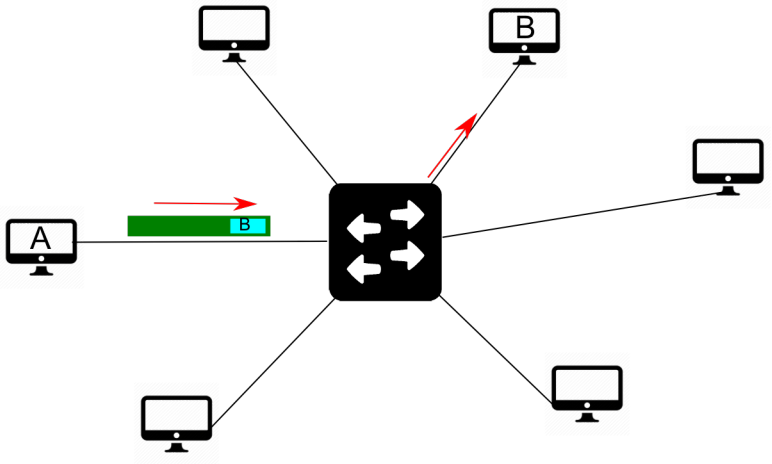
Ethernet: Hubs

- Hubs are *physical-layer* devices that interconnect machines within a LAN
- With the rise of hubs, the *bus topology* for LANs diminished
- Hubs provided a cheaper solution for LANs
  - Twisted pair copper wire replaced coaxial cable
  - Shorter distances with less attenuation
  - Cheaper 10BASE-T standard
    - 10 Mbps / Twisted Pair
- Collisions could still occur in hubs
  - CSMA/CD is used!

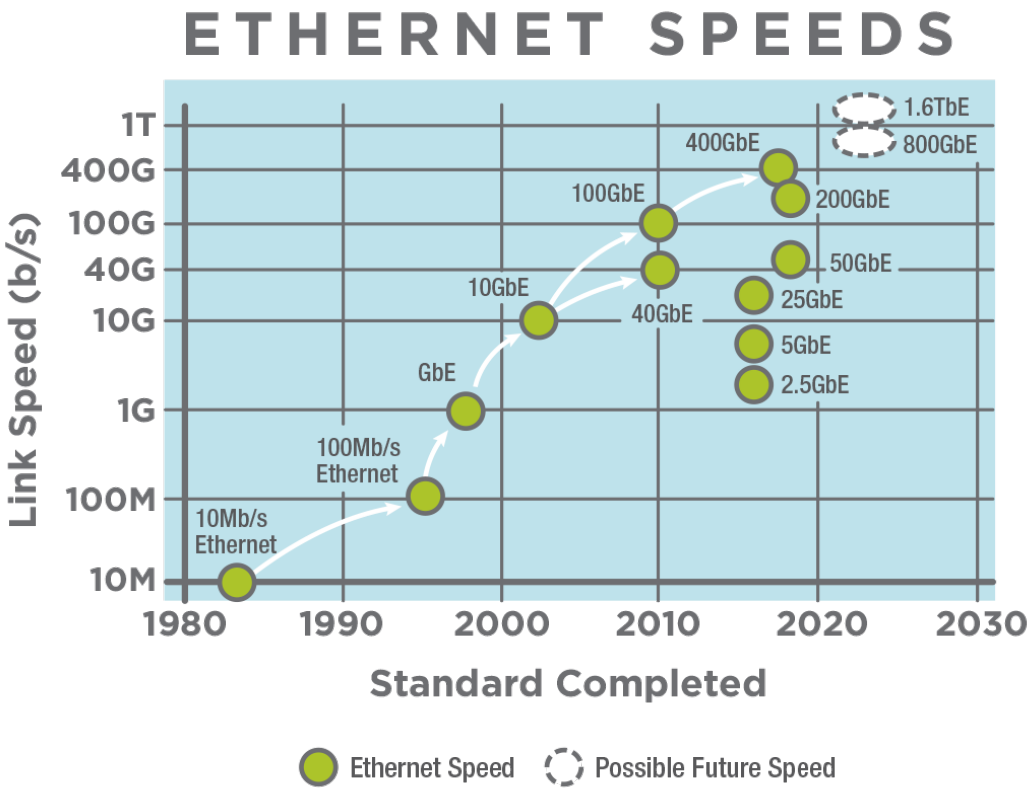


Ethernet: Switches

- *Switches* (aka *bridges*) have both *physical* and *data link* layers
- Switches process the incoming frame's destination physical address
  - Then, transmit the frame from the corresponding egress port
  - Reduces collisions & performance hit of CSMA/CD in LANs
- Switched LANs use twisted pair copper wire
- Moving from hubs to switches is straightforward



Ethernet Standards



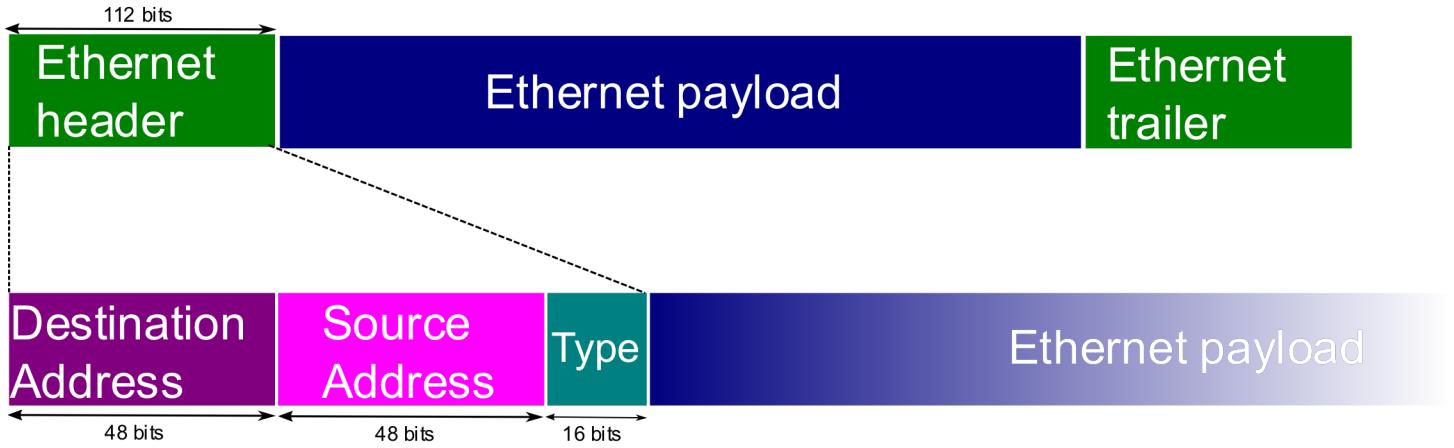
https://ethernetalliance.org/technology/2019-roadmap/

Ethernet Frame Format

- An Ethernet frame consists of three components
  - *Header*, comprised of some fields
    - 112 bits long
  - *Payload* is the network layer packet, e.g., an IP datagram
    - Up to 1500 bytes long
  - *Trailer*, comprises of a single Ethernet field
    - 32 bits long

Ethernet Frame Format: Header

- Ethernet header has three fields:
  - *Dest. MAC address* (48 bits) – physical addr. of NIC in *receiving* host
  - *Source MAC address* (48 bits) – physical addr. of NIC in *sending* host
  - *Type* (16 bits) stores the upper layer protocol, i.e., the protocol used in the Ethernet payload
    - IPv4: `0x0800` IPv6: `0x86DD` ARP: `0x0806`



Ethernet Frame Format: Payload, Trailer

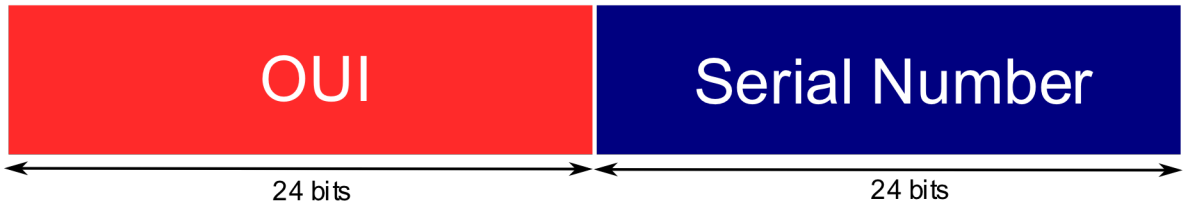
- Ethernet frame *payload*
  - The packet coming from upper layer, i.e., network layer
  - Payload size was limited to 1500 bytes in 10 Mbps LANs due to technological constraints
  - 1500 bytes became the de facto maximum network layer packet size in the Internet
    - Jumbo Frames (non-standard): 9000 bytes
- Ethernet frame *trailer*
  - Consists of a single 32-bit field: Cyclic redundancy check (CRC)
    - Computed based on the entire Ethernet frame
    - Used to identify bit flips (errors due to noise) during frame transmission

Promiscuous Mode

- By *default* upon receiving a frame
  - NIC reads the destination MAC address in frame header
  - If that address matches NIC's own address, then NIC sends the frame payload to the upper layer process
  - Otherwise, discards the frame
- *Promiscuous mode*:
  - NIC accepts all frames!
  - Independent of what the destination MAC address is, the payload is passed to the upper layer process
  - Allows machine to sniff all of frames transmitted in a LAN
  - Used for diagnostic purposes (e.g. *Wireshark*)

MAC Addresses

- MAC addresses are 48 bits long
- Represented usually by sequence of 6 hex numbers separated by colon
  - Example: **08:00:27:A8:69:6C**
- Higher 24 bits refer to *manufacturer ID*
  - Called *Organizationally Unique Identifier* (OUI)
  - Managed by *IEEE*
- Lower 24 bits refer to the *serial number* of NIC
  - Assigned by manufacturer of NIC



Broadcast & Multicast MAC Addresses

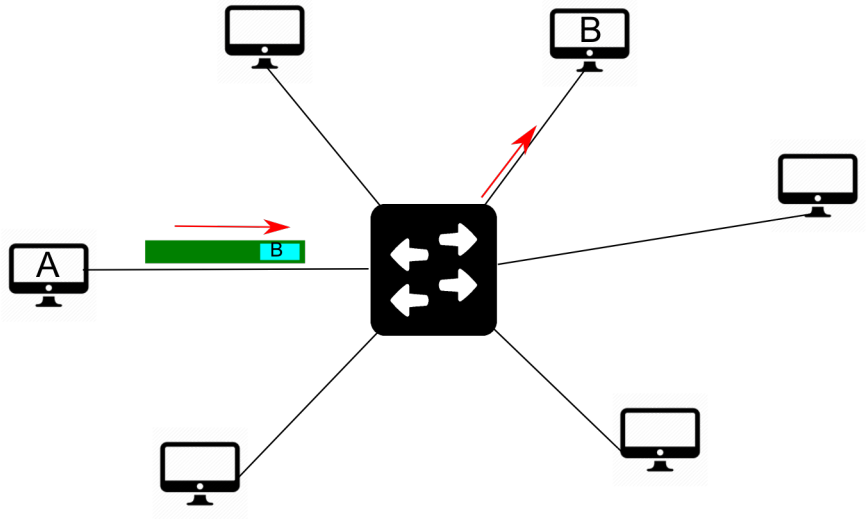
- Broadcast MAC address
  - 48 bits of 1: **`FF:FF:FF:FF:FF:FF`**
  - NIC accepts all frames with destination broadcast address
- Multicast MAC address
  - To transmit packets to a predefined set of receivers
  - The host needs to register the multicast address in its NIC
  - If NIC receives a frame with the already-registered multicast destination address, accepts it
  - Lowest bit in the first byte of address
    - 0: physical (unicast) address
    - 1: multicast address

NIC and Destination MAC Address

- NIC *accepts* an Ethernet frame according to its destination MAC if
  - Destination MAC *address* is the same as NIC's MAC address, or
  - Destination MAC address is *broadcast* MAC address, or
  - Destination MAC address is an already-registered *multicast* address, or
  - NIC is in promiscuous mode
- Otherwise, the NIC *drops* the frame

Switch Forwarding Table

- A switch has multiple Ethernet interfaces
- Upon receiving a frame, the switch
  - Examines the destination MAC address in the frame's header
  - Sends the packet through the appropriate Ethernet interface to the device with that destination MAC address

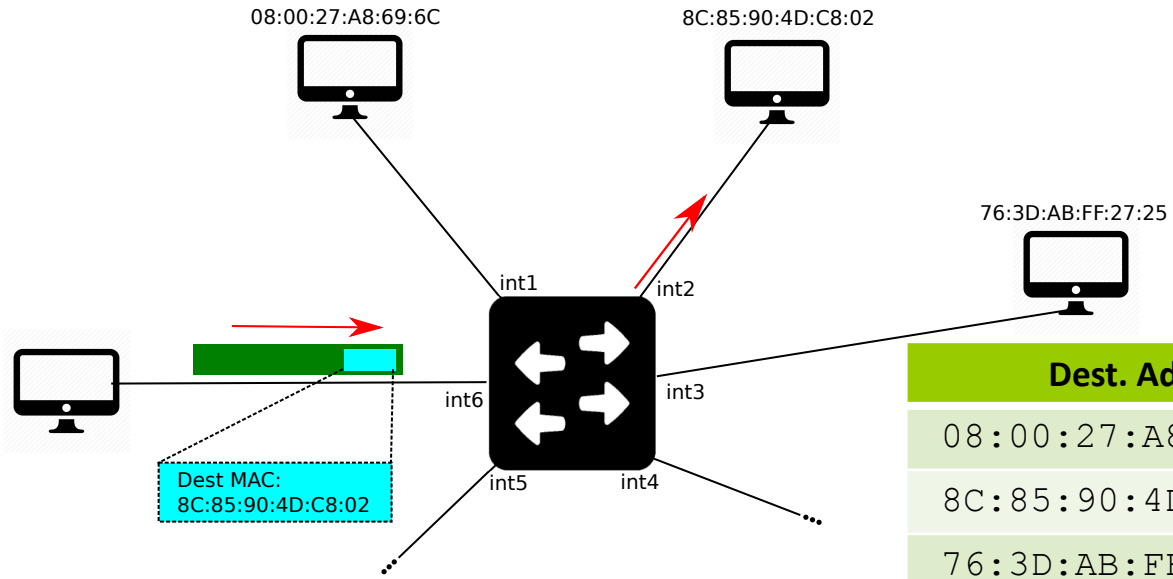


Question: How does a switch identify the egress interface?

Answer: By maintaining a forwarding table!

Switch Forwarding Table

- A *switch forwarding table* maps
  - The MAC address of a device connected to the switch, to
  - An Ethernet interface of that switch



Dest. Addr	Interface
08:00:27:A8:69:6C	int1
8C:85:90:4D:C8:02	int2
76:3D:AB:FF:27:25	int3
. .	…

Switch Forwarding Table

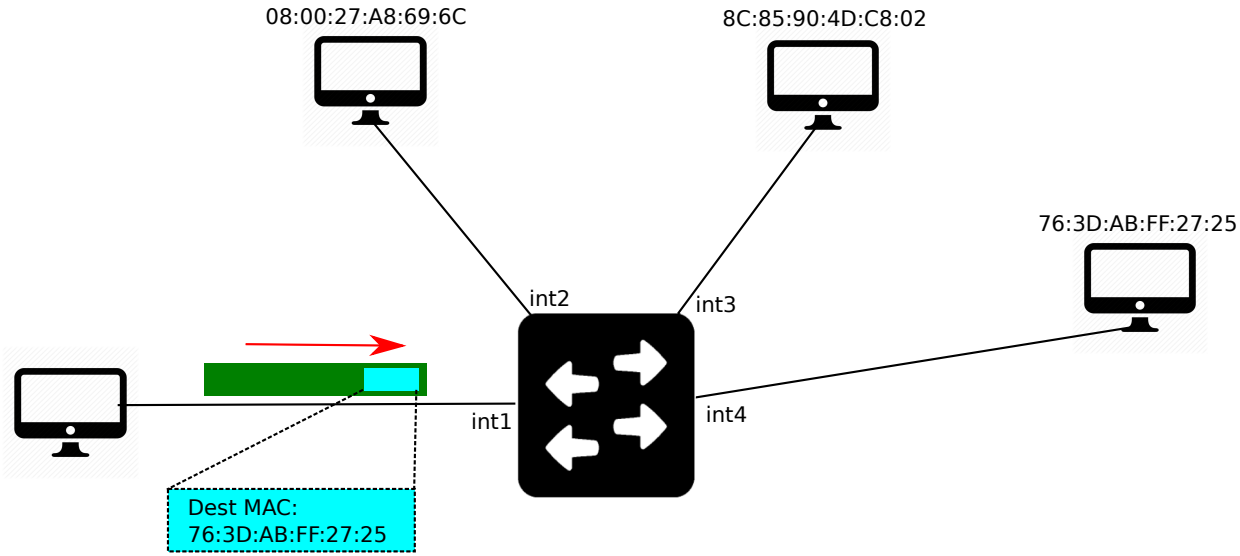
- Upon receiving a frame, the switch
  - Extracts the destination MAC address in the frame header
  - Searches the forwarding table for a match
- **Match found?**
  - Switch *forwards* the frame through that interface
- **Match not found?**
  - Switch *floods* the frame
  - Transmitting the frame from every Ethernet interface *except* for the one the frame was received from

Example: Switch Flooding

- Consider this forwarding table for a switch:

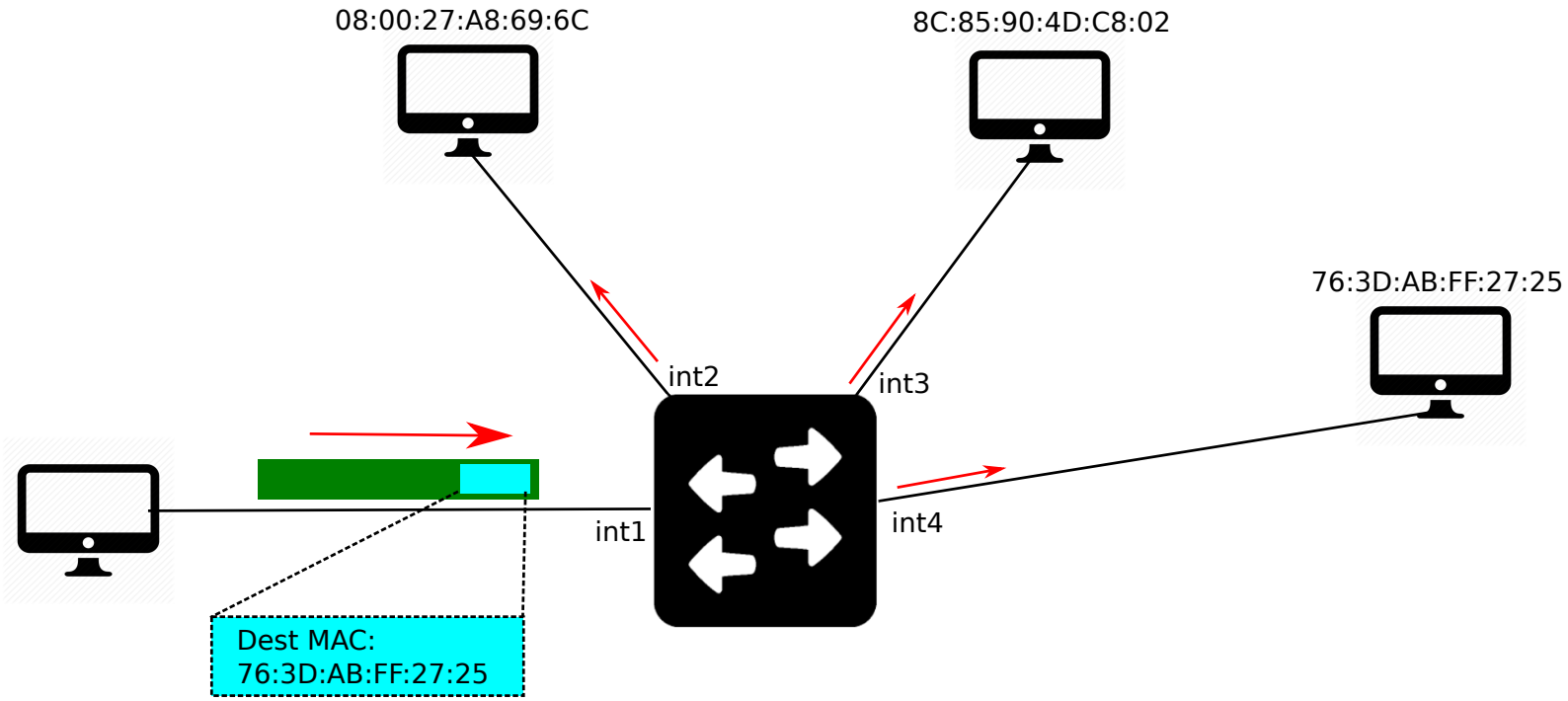
Dest. Addr	Interface
08:00:27:A8:69:6C	int2
8C:85:90:4D:C8:02	int3

- **How does the switch forward the frame in this LAN?**



Example: Switch Flooding

- Answer: **It Floods!**



Switch Forwarding Table

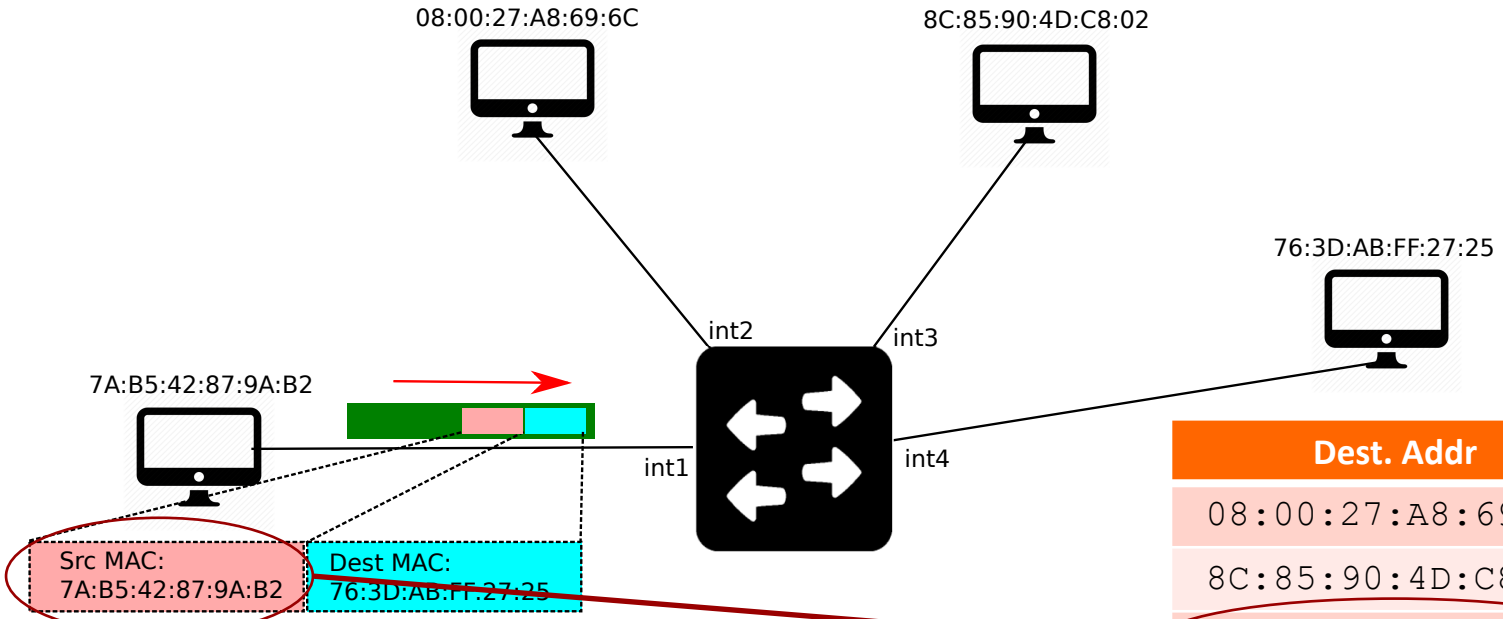
- A switch forwarding table on boot up is empty
- Switches do not need manual configuration of their forwarding tables
- *Question:* How is a switch forwarding table populated?
- *Answer:* A switch gradually learns about the topology of the network and populates the table
  - The switch maintains a per-interface list of all *source* MAC addresses received
  - Assumption: If a frame destined to that MAC address appears, it must be *reachable* through that interface

Example: Populating Switch Forwarding Table

- Consider the following forwarding table for a switch

Dest. Addr	Interface
08:00:27:A8:69:6C	int2
8C:85:90:4D:C8:02	int3

- **How does the switch forwarding table change upon receiving the packet in the following LAN?**

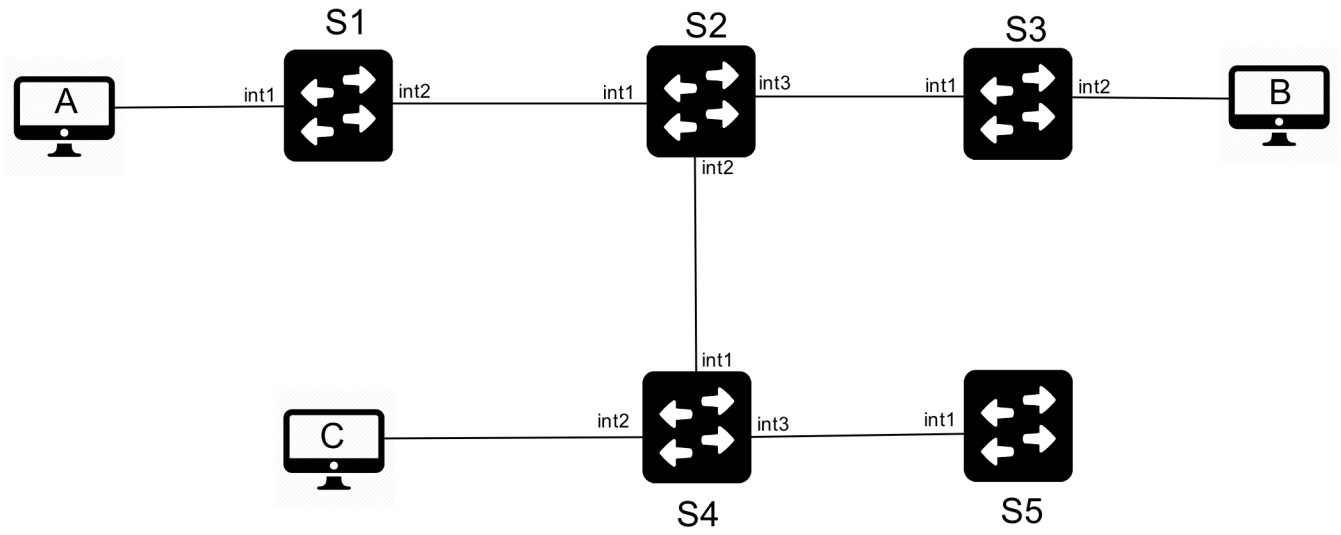


Dest. Addr	Interface
08:00:27:A8:69:6C	int2
8C:85:90:4D:C8:02	int3
7A:B5:42:87:9A:B2	int1

Example: Switch Forwarding Tables

- Consider the following LAN
  - Assume that initially all forwarding tables are empty
  - A sends a frame to B
  - B responds to A

- **How are the forwarding tables are updated in each switch?**
  How do switches forward these two frames?



Closing Thoughts

Recap

- Today we discussed
  - Ethernet protocol in data link layer with different standards
  - MAC addresses and their structure
  - Hubs and switches
  - How switches do frame forwarding

Next Class

- WiFi (802.11)

Class Activity

CA.2 – Ethernet & Wireshark

Due tonight at 11:59pm