Data Link Layer - Switching

Dr. T. Venkatesh

Dept of CSE, IIT Guwahati

Slides from lectures of Prof. Srini Seshan, CMU and Prof. Jim Kurose, UMass, Amherst

What Does Data Link Layer Do?

Data link layer has responsibility of transferring frames from one node to adjacent node over a *single* link

- An IP packet from host A to host B may traverses different links using different data link protocols
  - 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 delivery
- Different link protocols are not inter-operable!
  - IP packets are encapsulated/decapsulated with appropriate data link protocol header over each link (forwarding node does this)
  - IP protocol and IP routers glue the links ("physical networks") together and provide end-to-end data delivery!

Data Link Layer Functions

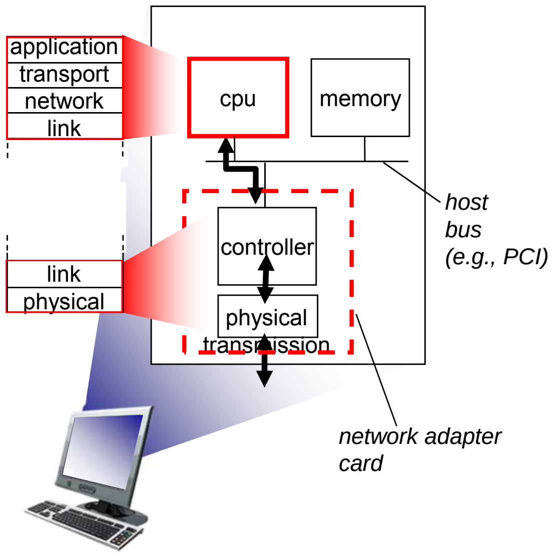
- Framing
  - sender (transmitter): encapsulate datagram into frame, adding header, trailer, transmit frame
  - receiver: detect beginning of frames, receive frame, decapsulate frame, stripping off header, trailer
- Link Access (Media Access Control)
  - determine whether it's okay to transmit over the link
    - particularly important when link shared by many nodes
      - also an issue over "half-duplex" point-to-point link
    - need media access control (MAC)
  - "physical addresses" identify sender/receiver on a link!
    - particularly important when link shared by many nodes, while over point-to-point link, not necessary
    - "physical addresses" often referred to as "MAC" addresses
      - different from IP addresses (which are logical & global)!

Other Data Link Layer Functions

- Error Detection (commonly implemented)
    - errors caused by signal attenuation, noise, etc.
    - sender computes "checksum", attaches to frame
    - receiver detects presence of errors by verifying "checksum"
        - drops corrupted frame, may ask sender for retransmission
    - Commonly used "checksum": cyclic redundancy code (CRC)
- Reliable delivery between adjacent nodes (optional)
    - using, e.g., go-back-N or selective repeat protocol
        - 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?
- Error Correction (optional)
    - receiver identifies *and corrects* bit error(s) without resorting to retransmission, using forward error correction (FEC) codes
- Flow Control (optional)
    - negotiating transmission rates between two nodes

Where is the link layer implemented?

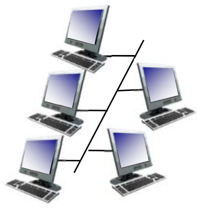
- in each and every host
- implemented in *network interface card* (NIC)
  - Ethernet card, 802.11 card; Ethernet chipset
  - implements link, physical layer
- attaches into host's system buses
- combination of hardware, software, firmware

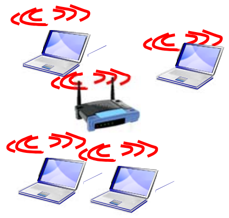


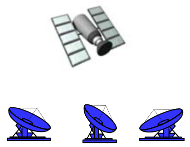
Multiple Access Links and LANs

Two types of "links":

- point-to-point, e.g.,
  - PPP for dial-up access,
  - point-to-point link between Ethernet switch, host
- broadcast (shared wire or medium)
  - traditional Ethernet
  - 802.11 wireless LAN


shared wire (e.g., cabled Ethernet)


shared RF (e.g., 802.11 WiFi)


shared RF (satellite)

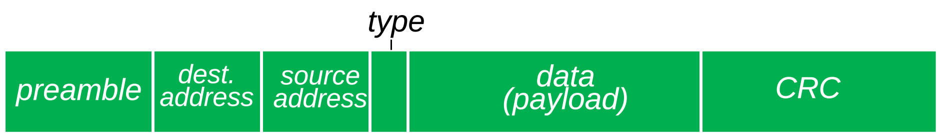
humans at a cocktail party (shared air, acoustical)

MAC (Physical, or LAN) Addresses

- used to get frames from one interface to another physically-connected interface (same physical network)
- 48 bit MAC address (for most LANs)
  - fixed for each adaptor, burned in the adapter ROM
  - MAC address allocation administered by IEEE
    - 1st bit: 0 unicast, 1 multicast.
    - all 1's : broadcast
- MAC flat address -> portability
  - can move LAN card from one LAN to another
- MAC addressing operations on a LAN:
  - each adaptor on the LAN "sees" all frames
  - accept a frame if dest. MAC address matches its own MAC address
  - accept all broadcast (MAC= all1's) frames
  - accept all frames if set in "promiscuous" mode
  - can configure to accept certain multicast addresses (first bit = 1)

Ethernet Frame Structure

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

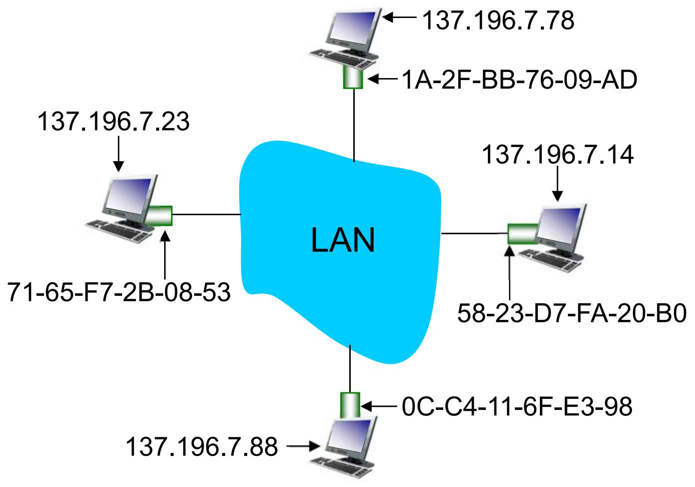


preamble:

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

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; timer>

  - timer: time after which address mapping will be forgotten (typically 20 min)

ARP Protocol

- A wants to send datagram to B, and A knows B's IP address.
- A looks up B's MAC address in its ARP table
- Suppose B's MAC address is not in A's ARP table.
- A broadcasts (why?) ARP query packet, containing B's IP address
  - destination 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

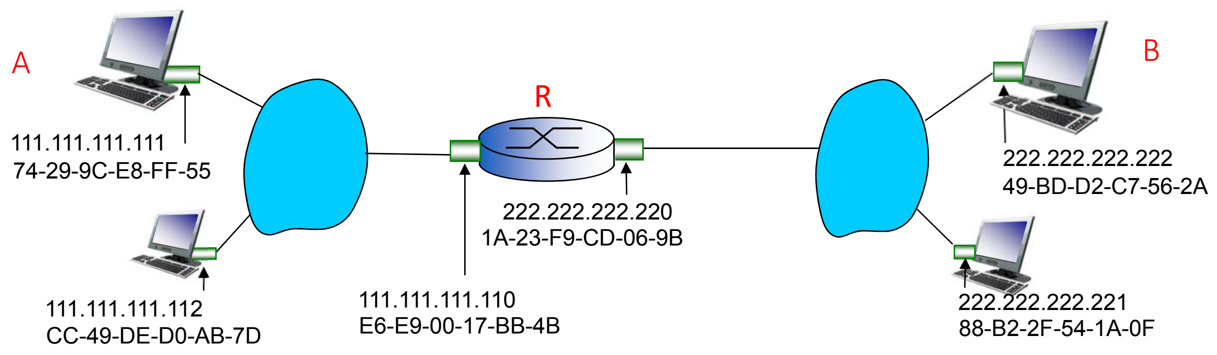
ARP Request & Response Processing

- The requester *broadcasts* ARP request
- The target node *unicasts* ARP reply to requester
  - With its physical address
  - Adds the requester into its ARP table
- On receiving the response, requester
  - updates its table, sets timer
- Other nodes upon receiving the ARP request
  - Refresh the requester entry if already there
  - No action otherwise
- Some questions to think about:
  - Shall requester buffer IP datagram while performing ARP?
  - What shall requester do if never receive any ARP response?

Forwarding to Another LAN

walkthrough: send datagram from A to B via R

- focus on addressing – at IP (datagram) and MAC layer (frame)
- assume A knows B’s IP address
- assume A knows IP address of first hop router, R
- assume A knows R’s MAC address



Ethernet Switch

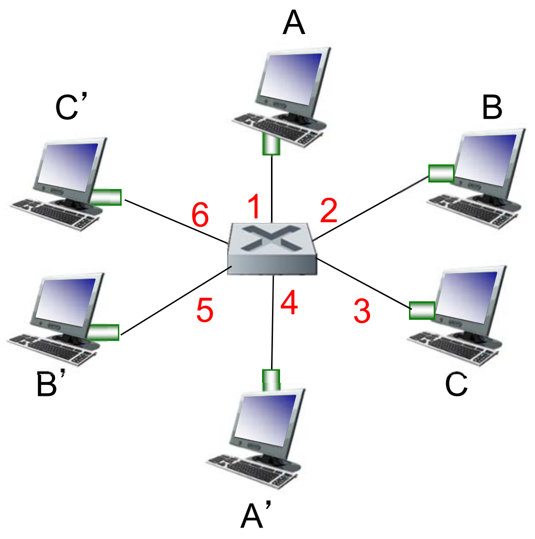
- link-layer device: takes an *active* role
  - store, forward Ethernet frames
  - examine incoming frame's MAC address, selectively forward frame to one-or-more outgoing links when frame is to be forwarded on segment, uses CSMA/CD to access segment
- *transparent*
  - hosts are unaware of presence of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

Ethernet: Unreliable, Connectionless

- *connectionless:* no handshaking between sending and receiving NICs
- *unreliable:* receiving NIC doesn't 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*

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)

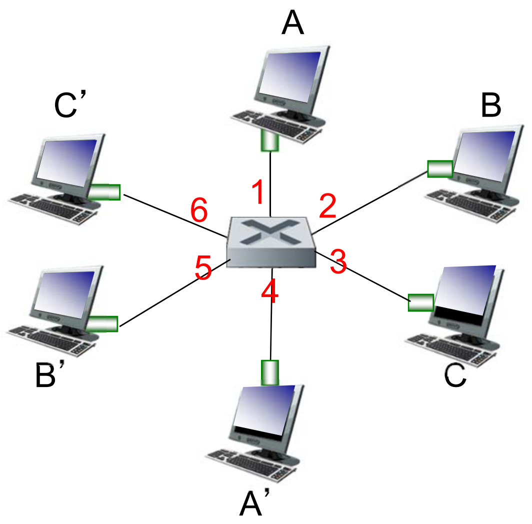
Switch Forwarding Table

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)

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

- something like a routing protocol?



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

Self Learning

- A bridge/switch has a forwarding (or switching) table
- entry in forwarding table:
  - (MAC Address, Interface, Time Stamp)
  - stale entries in table dropped (TTL can be 60 min)
- Bridge/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 forwarding table

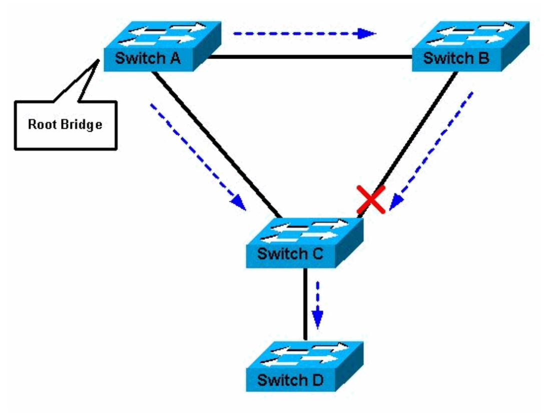
Filtering/Forwarding

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 */
```

Spanning Tree Protocol

- for increased reliability, desirable to have redundant, alternative paths from source to destination
- with multiple paths, cycles result - switches may multiply and forward frame forever
- solution: organize switches in a spanning tree by disabling subset of interfaces



Switch Spanning Tree Algorithm: Poem

I think that I shall never see
A graph more lovely than a tree.
A tree whose crucial property
Is loop-free connectivity.
A tree that must be sure to span
So packets can reach every LAN.
First, the root must be selected.
By ID, it is elected.
Least cost paths from root are traced.
In the tree, these paths are placed.
A mesh is made by folks like me,
Then bridges find a spanning tree

-- Radia Perlman

Advantages of Switches in LAN

- Isolates collision domains resulting in higher total max throughput
- limitless number of nodes and geographical coverage
- Can connect different Ethernet types
- Transparent (“plug-and-play”): no configuration necessary

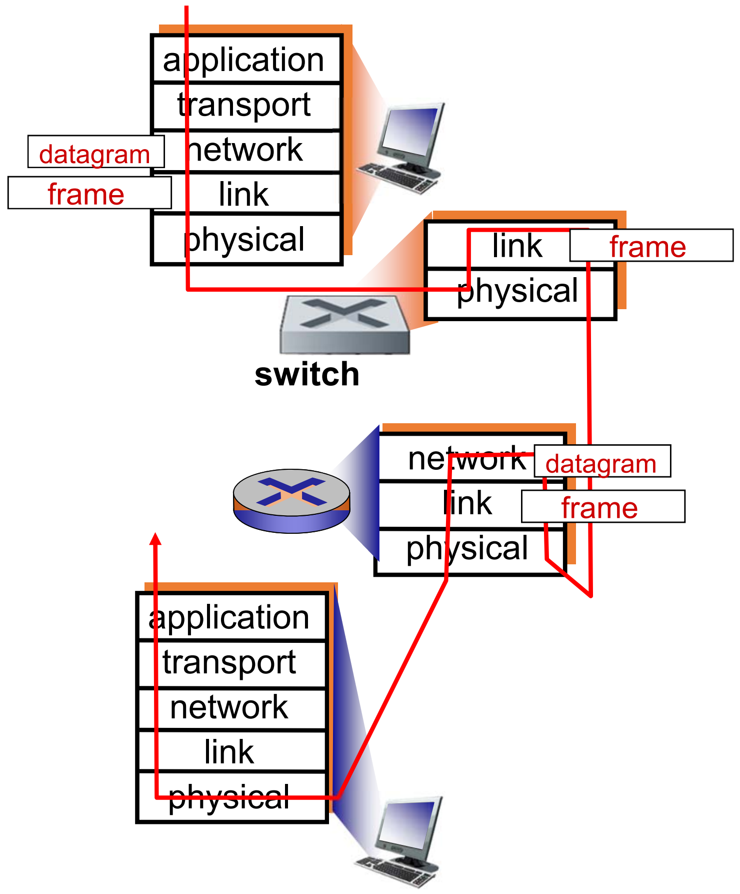
Switches vs. Routers

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



Routers vs. Switches

Switches+ and -

+ Switch operation is simpler requiring less packet processing

+ Switch tables are self learning

- All traffic confined to spanning tree, even when alternative bandwidth is available

- Switches do not offer protection from broadcast storms