

The Link Layer and LANs

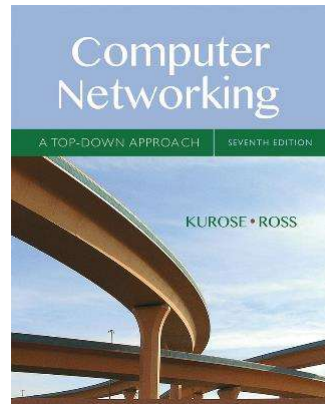
A note on the use of these Powerpoint slides:

We're making these slides freely available to all (faculty, students, readers). They're in PowerPoint form so you see the animations; and 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) that you mention their source (after all, we'd like people to use our book!)
- If you post any slides 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 1996-2016
J.F. Kurose and K.W. Ross, All Rights Reserved



Computer Networking: A Top Down Approach

7th edition

Jim Kurose, Keith Ross

Pearson/Addison Wesley

April 2016

Link Layer and LANs 6-1

1

Link layer, LANs: outline

6.4 LANs

- addressing, ARP
- Ethernet
- switches
- VLANs

6.5 link virtualization: MPLS

6.6 data center networking

6.7 a day in the life of a web request

Link Layer and LANs 6-2

2

MAC addresses and ARP

- 32-bit IP address:
 - network-layer address for interface
 - used for layer 3 (network layer) forwarding
- MAC (or LAN or physical or Ethernet) address:
 - function: *used 'locally' to get frame from one interface to another physically-connected interface (same network, in IP-addressing sense)*
 - 48 bit MAC address (for most LANs) burned in NIC ROM, also sometimes software settable
 - e.g.: 1A-2F-BB-76-09-AD

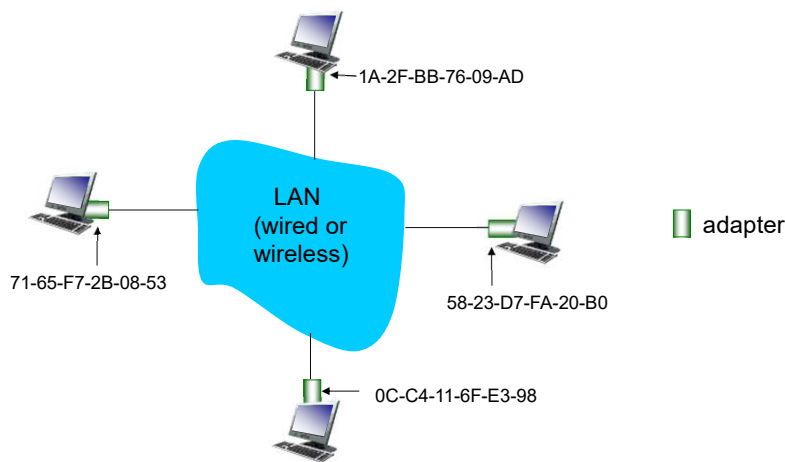
hexadecimal (base 16) notation
(each "numeral" represents 4 bits)

Link Layer and LANs 6-3

3

LAN addresses and ARP

each adapter on LAN has unique **LAN** address



Link Layer and LANs 6-4

4

LAN addresses (more)

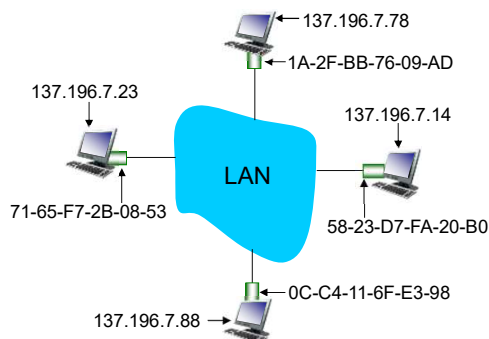
- MAC address allocation administered by IEEE
- manufacturer buys portion of MAC address space (to assure uniqueness)
- analogy:
 - MAC address: like Social Security Number
 - IP address: like postal address
- MAC flat address → portability
 - can move LAN card from one LAN to another
- IP hierarchical address *not* portable
 - address depends on IP subnet to which node is attached

Link Layer and LANs 6-5

5

ARP: address resolution protocol

Question: how to determine interface's MAC address, knowing its IP address?



ARP table: each IP node (host, router) on LAN has 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)

Link Layer and LANs 6-6

6

ARP protocol: same LAN

- A wants to send datagram to B
 - B's MAC address not in A's ARP table.
- A **broadcasts** ARP query packet, containing B's IP address
 - destination MAC address = FF-FF-FF-FF-FF-FF
 - all nodes 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*

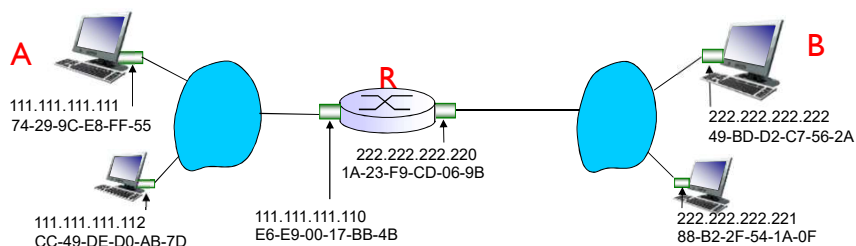
Link Layer and LANs 6-7

7

Addressing: routing 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 (how?)
- assume A knows R's MAC address (how?)

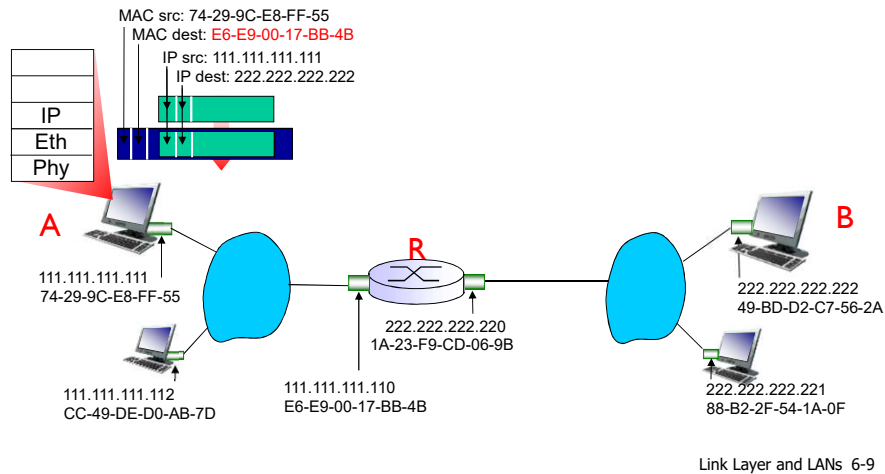


Link Layer and LANs 6-8

8

Addressing: routing to another LAN

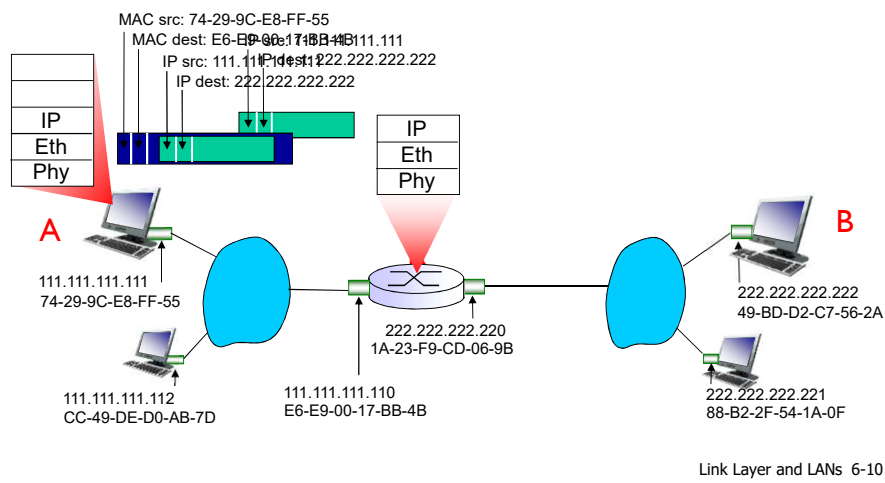
- A creates IP datagram with IP source A, destination B
- A creates link-layer frame with R's MAC address as destination address, frame contains A-to-B IP datagram



9

Addressing: routing to another LAN

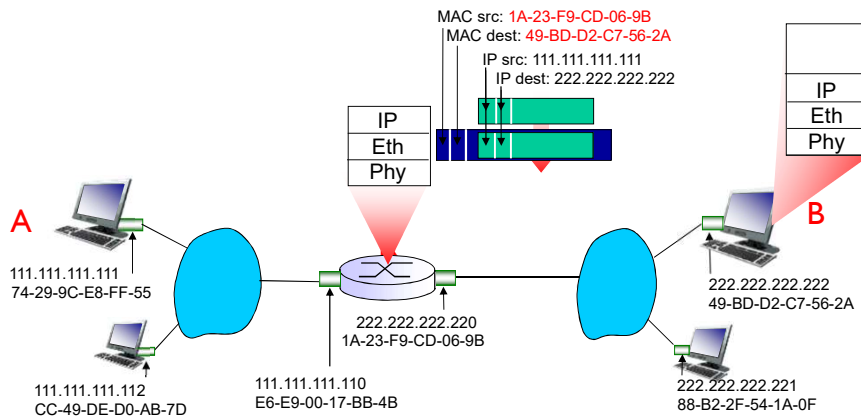
- frame sent from A to R
- frame received at R, datagram removed, passed up to IP



10

Addressing: routing to another LAN

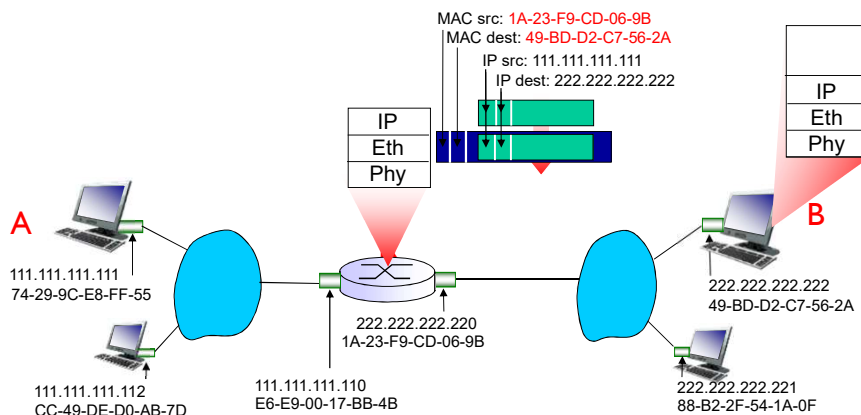
- R forwards datagram with IP source A, destination B
- R creates link-layer frame with B's MAC address as destination address, frame contains A-to-B IP datagram



11

Addressing: routing to another LAN

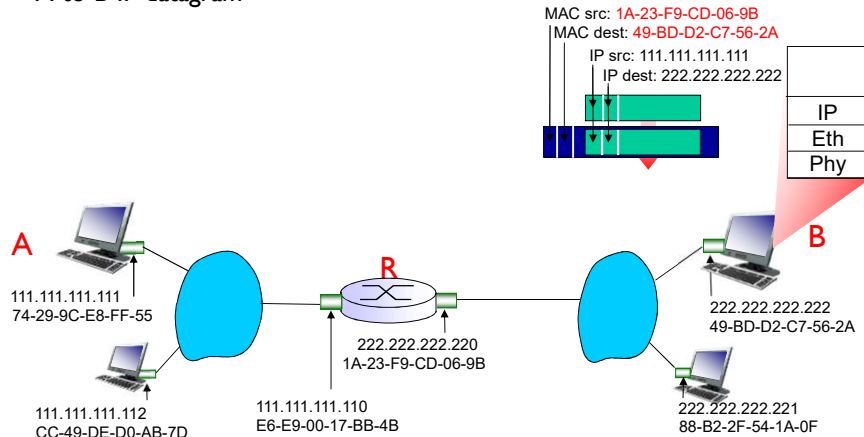
- R forwards datagram with IP source A, destination B
- R creates link-layer frame with B's MAC address as destination address, frame contains A-to-B IP datagram



12

Addressing: routing to another LAN

- R forwards datagram with IP source A, destination B
- R creates link-layer frame with B's MAC address as dest, frame contains A-to-B IP datagram



* Check out the online interactive exercises for more examples: http://gaia.cs.umass.edu/kurose_ross/interactive/

Link Layer and LANs 6-13

13

Link layer, LANs: outline

- 6.1 introduction, services
- 6.2 error detection, correction
- 6.3 multiple access protocols
- 6.4 LANs
 - addressing, ARP
 - Ethernet
 - switches
 - VLANs
- 6.5 link virtualization: MPLS
- 6.6 data center networking
- 6.7 a day in the life of a web request

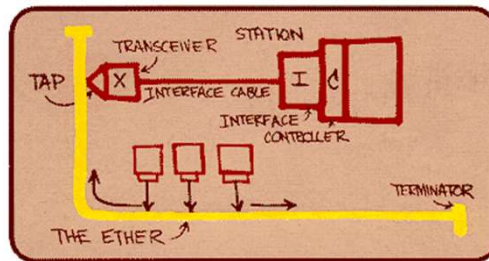
Link Layer and LANs 6-14

14

Ethernet

“dominant” wired LAN technology:

- single chip, multiple speeds (e.g., Broadcom BCM5761)
- first widely used LAN technology
- simpler, cheap
- kept up with speed race: 10 Mbps – 10 Gbps



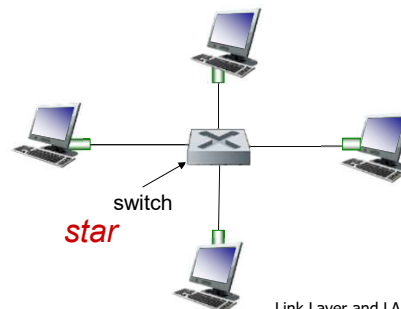
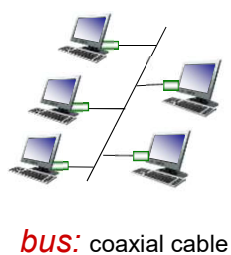
Metcalfe's Ethernet sketch

Link Layer and LANs 6-15

15

Ethernet: physical topology

- **bus**: popular through mid 90s
 - all nodes in same collision domain (can collide with each other)
- **star**: prevails today
 - active **switch** in center
 - each “spoke” runs a (separate) Ethernet protocol (nodes do not collide with each other)



Link Layer and LANs 6-16

16

Ethernet frame structure

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



preamble:

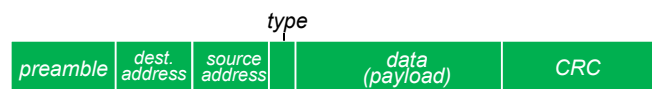
- 7 bytes with pattern 10101010 followed by one byte with pattern 10101011
- used to synchronize receiver, sender clock rates

Link Layer and LANs 6-17

17

Ethernet frame structure (more)

- **addresses:** 6 byte source, destination MAC addresses
 - if adapter receives frame with matching destination address, or with broadcast address (e.g. ARP packet), it passes data in frame to network layer protocol
 - otherwise, adapter discards frame
- **type:** indicates higher layer protocol (mostly IP but others possible, e.g., Novell IPX, AppleTalk)
- **CRC:** cyclic redundancy check at receiver
 - error detected: frame is dropped



Link Layer and LANs 6-18

18

Ethernet: unreliable, connectionless

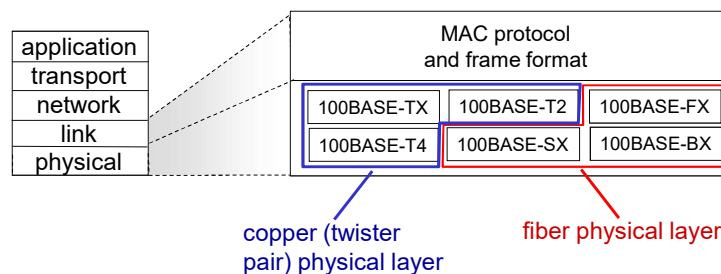
- **connectionless**: no handshaking between sending and receiving NICs
- **unreliable**: receiving NIC doesn't send acks or nacks to sending NIC
 - data in dropped frames recovered only if initial sender uses higher layer rdt (e.g., TCP), otherwise dropped data lost
- Ethernet's MAC protocol: unslotted **CSMA/CD with binary backoff**

Link Layer and LANs 6-19

19

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, 1 Gbps, 10 Gbps, 40 Gbps
 - different physical layer media: fiber, cable



Link Layer and LANs 6-20

20

Ethernet switch

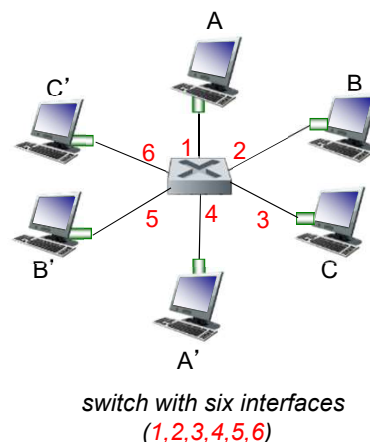
- **link-layer device: takes an *active* role**
 - store, forward Ethernet frames
 - examine incoming frame's MAC address, **selectively** forward frame to one-or-more outgoing links when frame is to be forwarded on segment, uses CSMA/CD to access segment
- ***transparent***
 - hosts are unaware of presence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

Link Layer and LANs 6-21

21

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



Link Layer and LANs 6-22

22

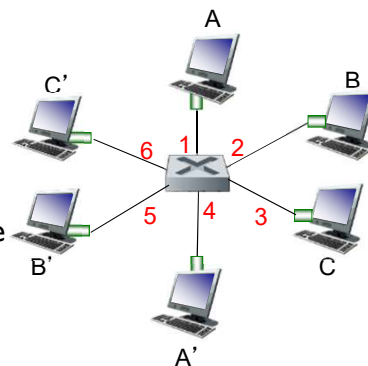
Switch forwarding table

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?



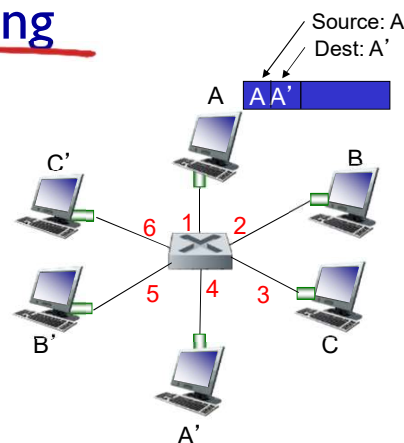
switch with six interfaces
(1,2,3,4,5,6)

Link Layer and LANs 6-23

23

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



MAC addr	interface	TTL
A	1	60

Switch table
(initially empty)

Link Layer and LANs 6-24

24

Switch: frame filtering/forwarding

when frame received at switch:

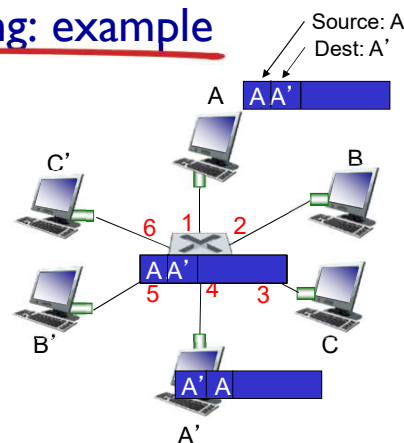
1. record incoming link, MAC address of sending host
2. index switch table using MAC destination address
3. if entry found for destination
 - then {
 - if destination on segment from which frame arrived
 - then drop frame
 - else forward frame on interface indicated by entry
 - }
 - else flood /* forward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

Link Layer and LANs 6-25

25

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

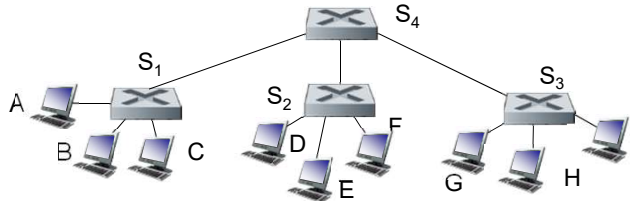
switch table
(initially empty)

Link Layer and LANs 6-26

26

Interconnecting switches

self-learning 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 ?

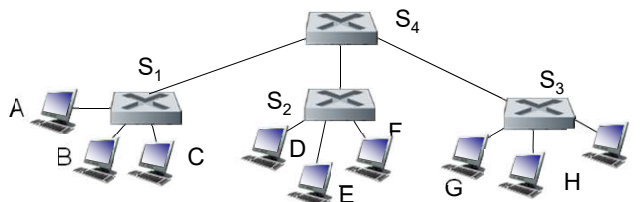
- **A:** self learning! (works exactly the same as in single-switch case!)

Link Layer and LANs 6-27

27

Self-learning multi-switch example

Suppose C sends frame to I, I responds to C

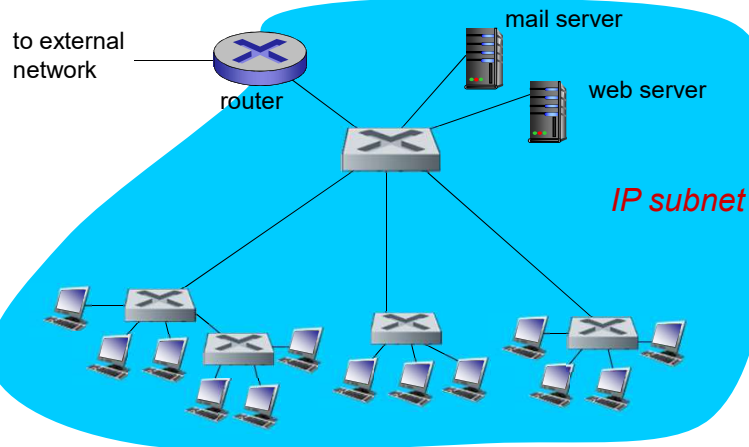


- **Q:** show switch tables and packet forwarding in S_1, S_2, S_3, S_4

Link Layer and LANs 6-28

28

Institutional network



Link Layer and LANs 6-29

29

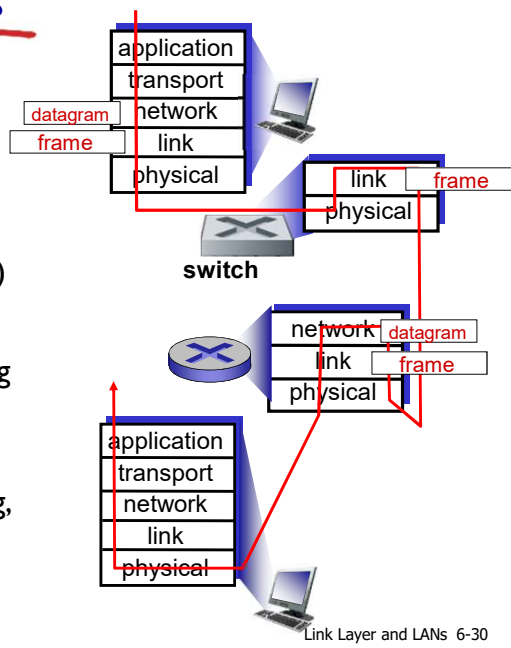
Switches vs. routers

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



Link Layer and LANs 6-30

30