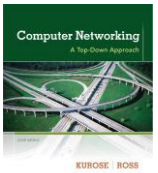
# Chapter 5
# Link Layer




*Computer Networking: A Top Down Approach*
*6th edition*
*Jim Kurose, Keith Ross*
*Addison-Wesley*
*March 2012*

A note on the use of these ppt 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



*The course notes are adapted for Bucknell's CSCI 363*
*Xiannong Meng*
*Spring 2016*



# Link layer, LANs: outline

5.1 introduction, services
5.2 error detection, correction
5.3 multiple access protocols
5.4 LANs
- addressing, ARP
- Ethernet
- switches
- VLANS

5.5 link virtualization: MPLS
5.6 data center networking
5.7 a day in the life of a web request



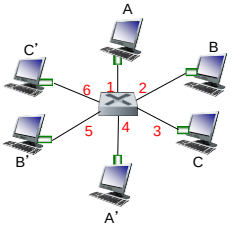
# Ethernet switch

- 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



# 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)*



# Ethernet Switch Example

Avaya ERS 2550T-PWR 50-port network switch

http://en.wikipedia.org/wiki/File:2550T-PWR-Front.jpg

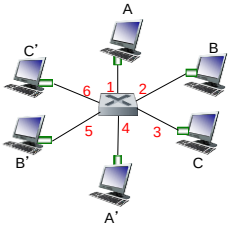


# Switch forwarding table

*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!*

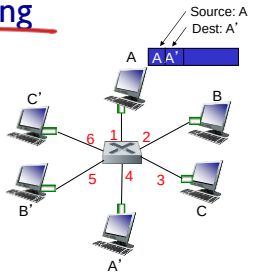
*Q: how are entries created, maintained in switch table?*
- *something like a routing protocol?*




*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




| MAC addr | interface | TTL |
|---|---|---|
| *A* | *1* | *60* |

*Switch table (initially empty)*



## Switch: frame filtering/forwarding

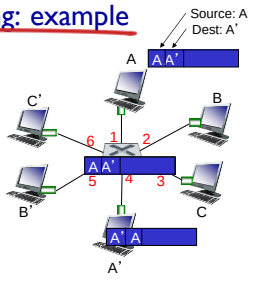
when frame received at switch:

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



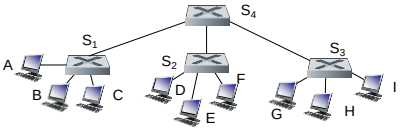

| MAC addr | interface | TTL |
|---|---|---|
| *A* | *1* | *60* |
| *A'* | *4* | *60* |

*switch table (initially empty)*



## Interconnecting switches

- switches can be connected together



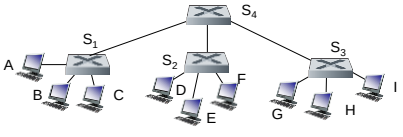

Q: sending from A to G - how does $S_1$ know to forward frame destined to F via $S_4$ and $S_3$?

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



## Self-learning multi-switch example

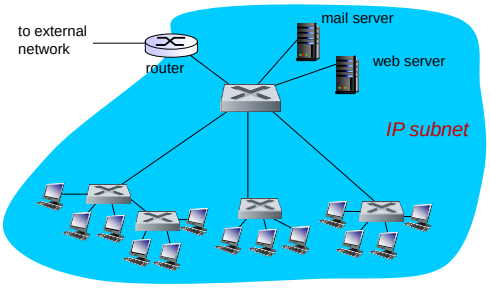
Suppose C sends frame to I, I responds to C




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



## Institutional network



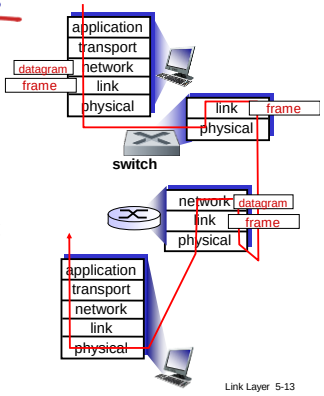
## Switches vs. routers

both are store-and-forward:

- *routers:* network-layer devices (examine network-layer headers)
- *switches:* link-layer devices (examine link-layer headers)

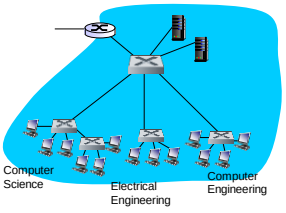
both have forwarding tables:

- *routers:* compute tables using routing algorithms, IP addresses
- *switches:* learn forwarding table using flooding, learning, MAC addresses






## 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: issues to address

- Drawbacks of a switch-based LAN
  - Lack of traffic isolation: traffic from different logical groups may have to be on the same network
  - Inefficient use of switches: each group may want a switch, not all ports are used (similar to static IP)
  - Difficult to manage users: if a user moves between groups (or a computer moves from LAN to LAN), it is hard to switch network connection

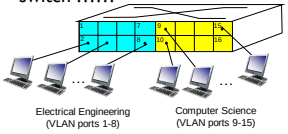


## VLANs

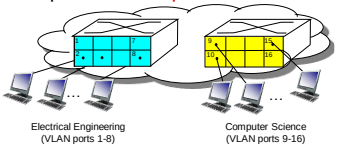
***Virtual Local Area Network***

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 ……



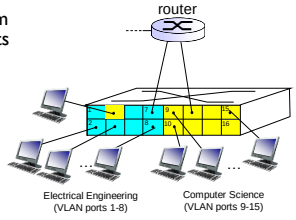

… operates as *multiple* virtual switches





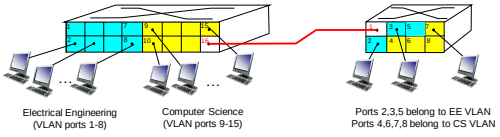

## Port-based VLAN

- *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



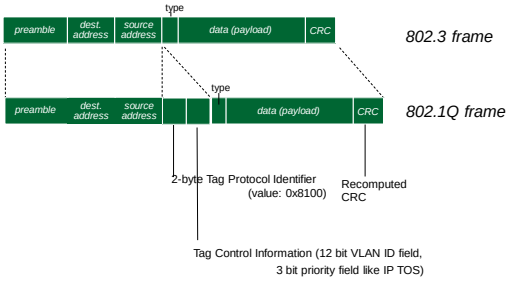



## VLANS spanning multiple switches



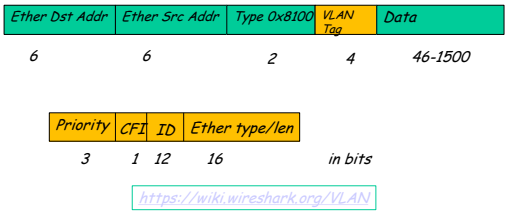

- *trunk port:* carries frames between VLANS defined over multiple physical switches
  - frames forwarded within VLAN between switches can't be vanilla 802.1 frames (must carry VLAN ID info)
  - 802.1q protocol adds/removed additional header fields for frames forwarded between trunk ports

# 802.1Q VLAN frame format






# VLAN frame explained




https://wiki.wireshark.org/VLAN



# Another view

https://en.wikipedia.org/wiki/EtherType

By Bill Stafford - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=24629079



# Link layer, LANs: outline

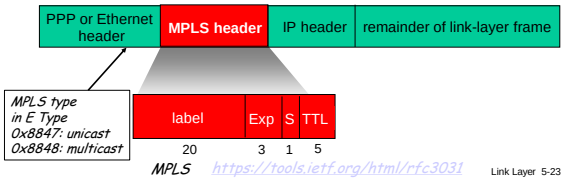
5.1 introduction, services
5.2 error detection, correction
5.3 multiple access protocols
5.4 LANs
- addressing, ARP
- Ethernet
- switches
- VLANS

5.5 link virtualization: MPLS
5.6 data center networking
5.7 a day in the life of a web request



# Multiprotocol label switching (MPLS)

- initial goal: high-speed IP forwarding using fixed length label (instead of IP address)
  - fast lookup using fixed length identifier (rather than shortest prefix matching)
  - borrowing ideas from Virtual Circuit (VC) approach
  - but IP datagram still keeps IP address!



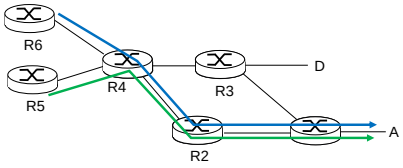

https://tools.ietf.org/html/rfc3031



# MPLS capable routers

- a.k.a. label-switched router
- forward packets to outgoing interface based only on label value (*don't inspect IP address*)
  - MPLS forwarding table distinct from IP forwarding tables
- *flexibility:* MPLS forwarding decisions can *differ* from those of IP
  - use destination *and* source addresses to route flows to same destination differently (traffic engineering)
  - re-route flows quickly if link fails: pre-computed backup paths (useful for VoIP)

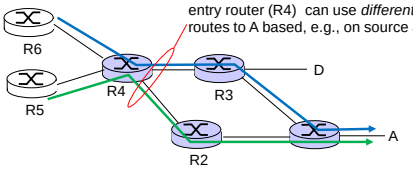
# MPLS versus IP paths




- *IP routing: path to destination determined by destination address alone*




# MPLS versus IP paths



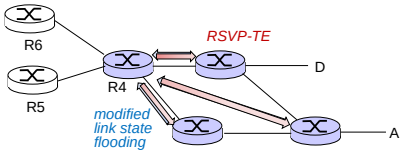

- *IP routing: path to destination determined by destination address alone*
- *MPLS routing:* path to destination can be based on source *and* dest. address
  - *fast reroute:* precompute backup routes in case of link failure





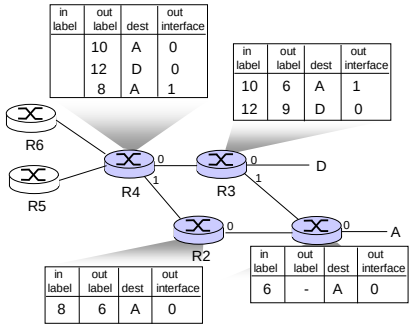
# MPLS signaling

- modify OSPF, IS-IS link-state flooding protocols to carry info used by MPLS routing,
  - e.g., link bandwidth, amount of "reserved" link bandwidth
- *entry MPLS router uses RSVP-TE signaling protocol to set up MPLS forwarding at downstream routers*






# MPLS forwarding tables




| in label | out label | dest | out interface |
|---|---|---|---|
| | 10 | A | 0 |
| | 12 | D | 0 |
| | 8 | A | 1 |

| in label | out label | dest | out interface |
|---|---|---|---|
| 10 | 6 | A | 1 |
| 12 | 9 | D | 0 |

| in label | out label | dest | out interface |
|---|---|---|---|
| 8 | 6 | A | 0 |

| in label | out label | dest | out interface |
|---|---|---|---|
| 6 | - | A | 0 |