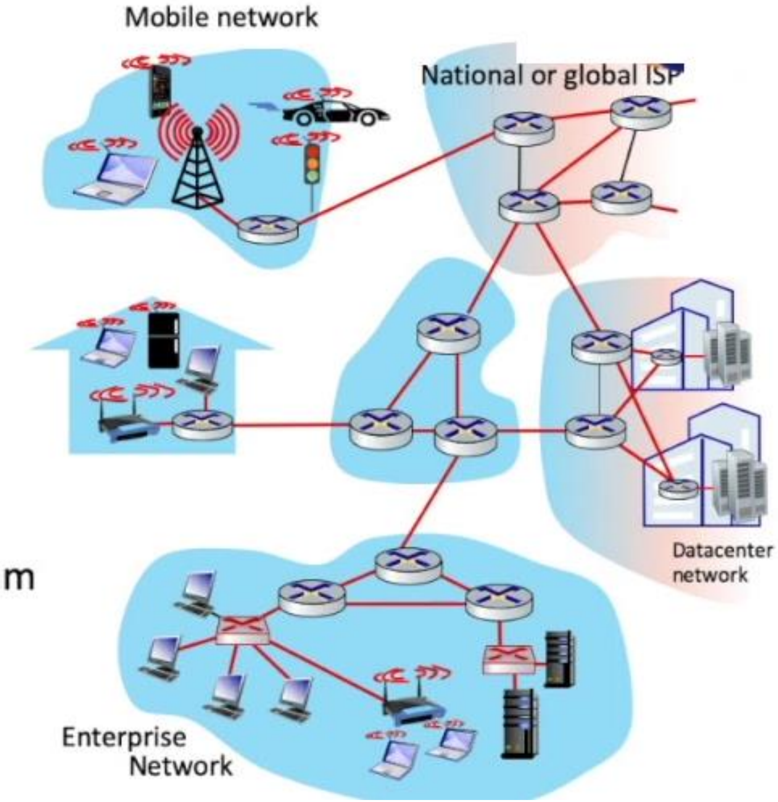
Link Layer

- Review of fundamentals of link layer protocols;
- Ethernet Switches,
- LANs,
- Link Layer Switches,
- VLANs,
- Complete tracking of traversal of a packet over internet between two applications

Introduction

- Hosts and routers: **Nodes**
- Communication channels that connect adjacent nodes along communication path: **Links**
  - Wired links
  - Wireless links
  - LANs
- Layer-2 packet: **Frame,** encapsulates datagram
- **Data-link layer** has responsibility of transferring datagram from one node to **physically adjacent** node over a link



Introduction

- Datagram transferred by different link protocols over different links
    - Example: 802.11 on the first link
      Ethernet on the following links

- Each link protocol provides different services
    - Example: May or may not provide reliable data transfer over link

Link Layer Services

- **Framing**
  - Encapsulate datagram into frame (adding header and trailer)

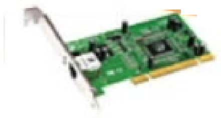
- **Link Access**
  - Channel access if shared medium
  - MAC addresses used in frame headers to identify source, destination
    - Different from IP address!

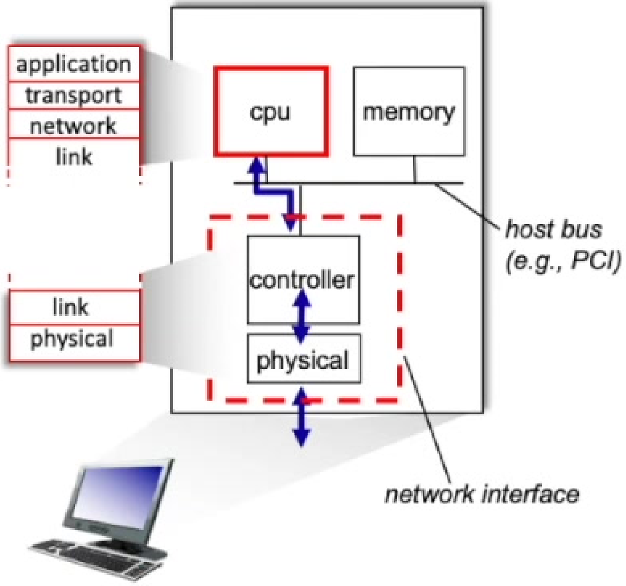
- **Error Detection and Correction (EDC)**

- **Reliable delivery between adjacent nodes**
  - 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?

Implementation

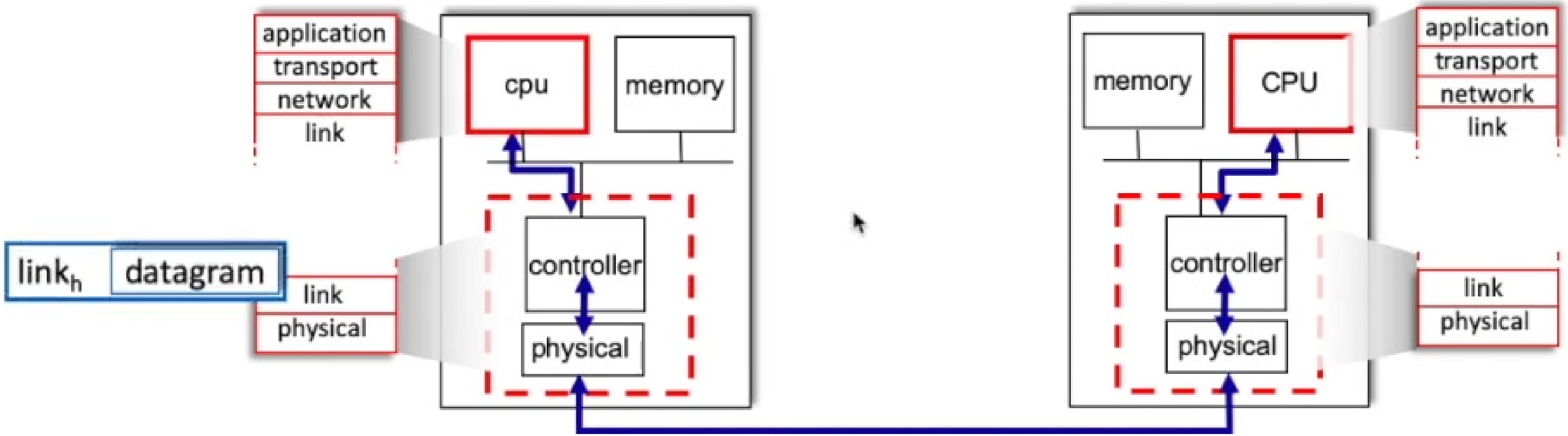
- In each and every host
- Link layer implemented in adaptor (**Network Interface Controller:** NIC) or on a chip
    - implements link, physical layer
        - Ethernet card
        - 802.11 card
        - Ethernet chipset
- Attaches into system buses of host
- Combination of hardware, software, firmware





Adaptors Communicating

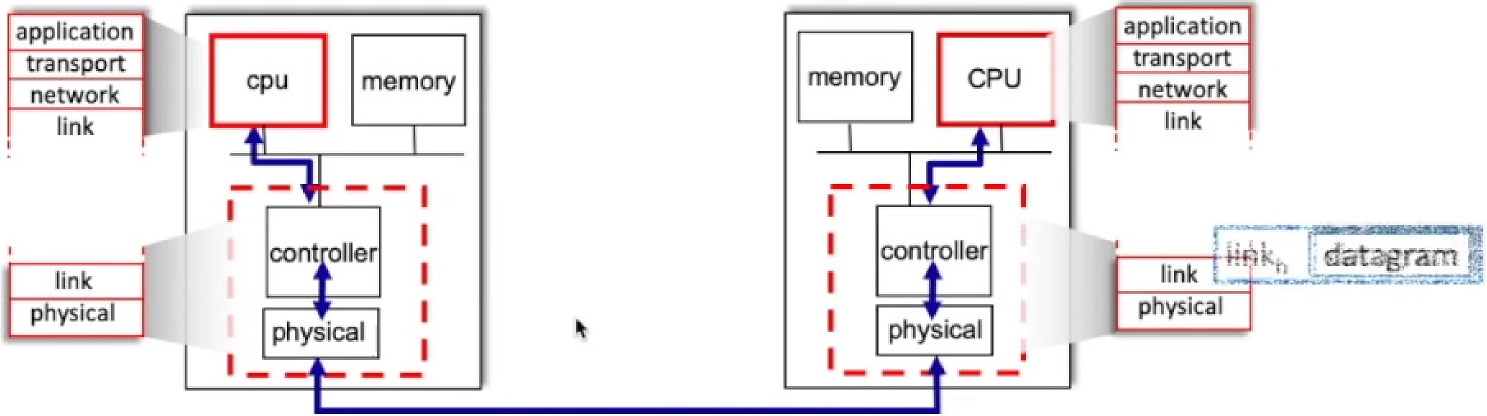
- Sending side
  - Encapsulates datagram in frame
  - Adds error checking bits, reliable data transfer, flow control, etc.



- Receiving side
  - Looks for errors, reliable data transfer, flow control, etc.
  - Extracts datagram, passes to upper layer at receiving side

Adaptors Communicating

- Sending side
  - Encapsulates datagram in frame
  - Adds error checking bits, reliable data transfer, flow control, etc.



- Receiving side
  - Looks for errors, reliable data transfer, flow control, etc.
  - Extracts datagram, passes to upper layer at receiving side

MAC Addresses & 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
  - Example: 1A-2F-BB-76-09-AD

LAN Address

- MAC address allocation administered by IEEE

- Manufacturer buys portion of MAC address space (to assure uniqueness)

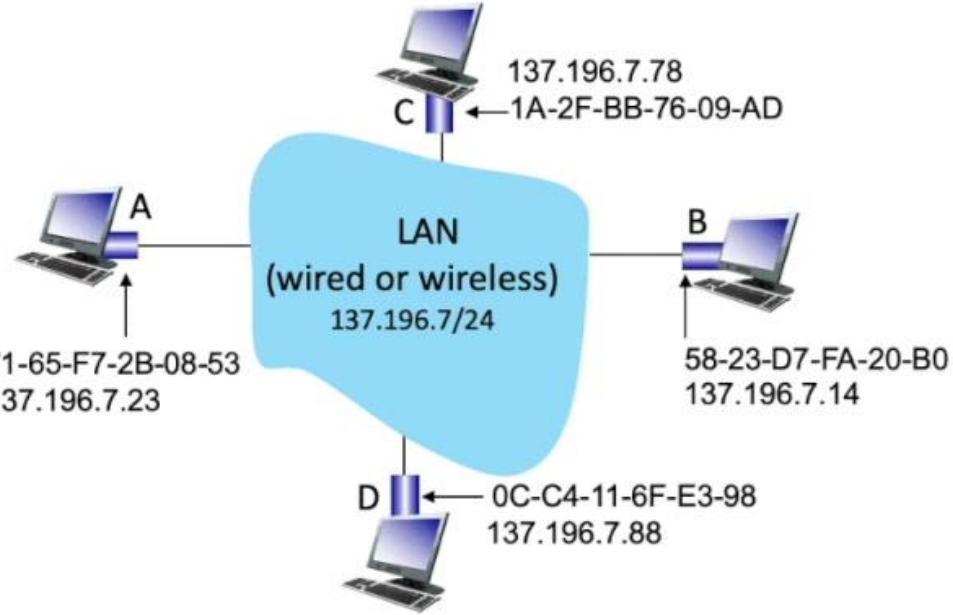
- Analogy
  - MAC address: Like Social Security Number
  - IP address: Like postal address

- 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

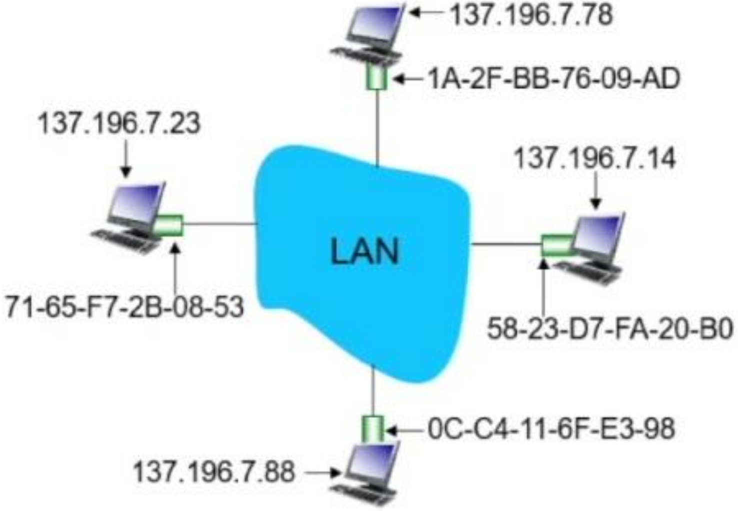
MAC Addresses & ARP

- Each adapter on LAN has unique **LAN** address



ARP: Address Resolution Protocol

Question: How to determine interface MAC address knowing its IP address?



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

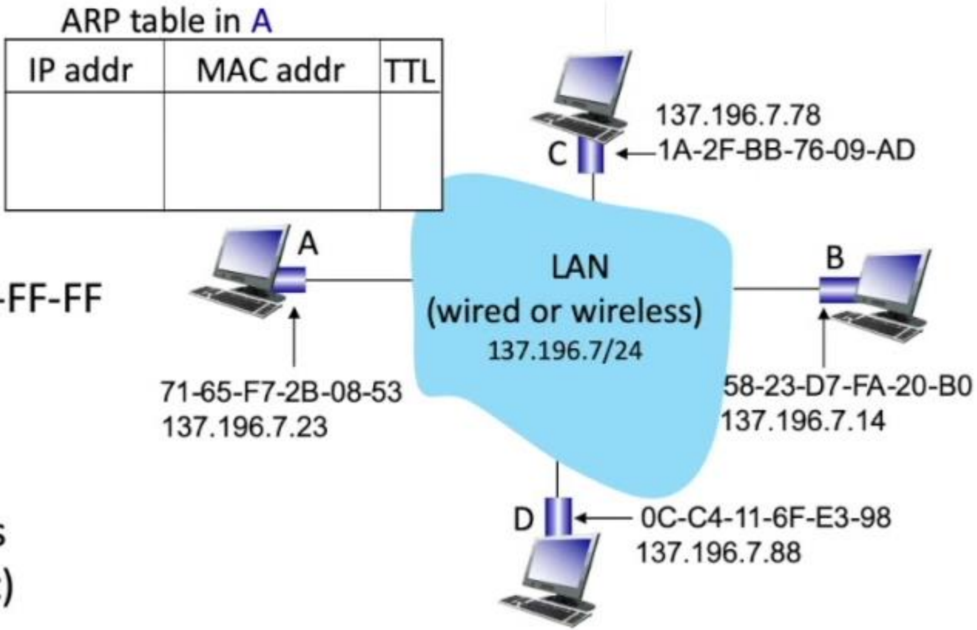
- IP/MAC address mappings for some LAN nodes

  < IP address; MAC address; TTL>

- TTL (Time To Live): Time after which address mapping will be forgotten (typically 20 min)

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 IP address for B
  - Destination MAC address = FF-FF-FF-FF-FF-FF
  - All nodes on LAN receive ARP query
- B receives ARP packet
  - B replies to A with its (B's) MAC address
  - Frame sent to A's MAC address (**unicast**)



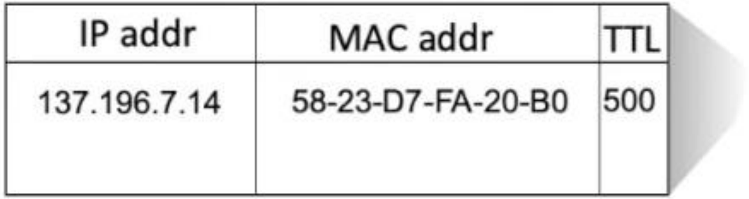
ARP table in A

IP addr	MAC addr	TTL

ARP Protocol: Same LAN

- A caches (saves) IP-to-MAC address pair in its ARP table

ARP table in A



IP addr	MAC addr	TTL
137.196.7.14	58-23-D7-FA-20-B0	500

  - 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**

Ethernet

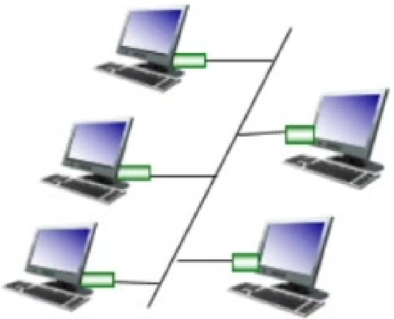
- Dominant wired LAN technology
  - Single chip, multiple speeds (e.g., Broadcom BCM5761)
  - First widely used LAN technology
  - Kept up with speed race: 10 Mbps – 10 Gbps

- **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)

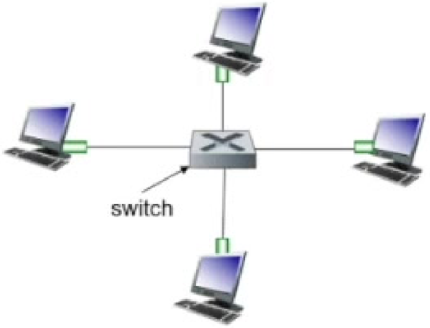
Ethernet: Physical Topology

- **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)

Bus: Coaxial cable

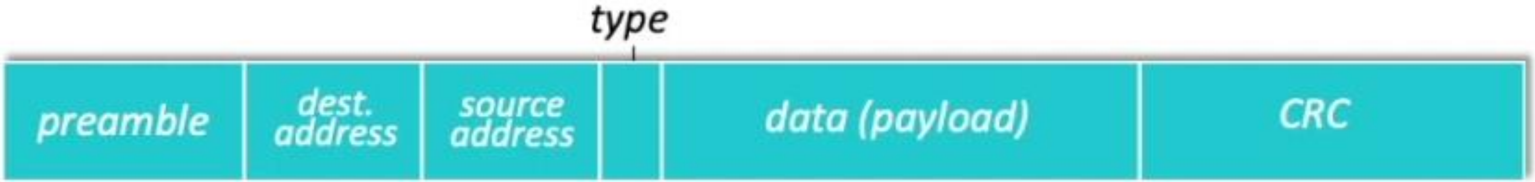


Star



Ethernet Frame Structure

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



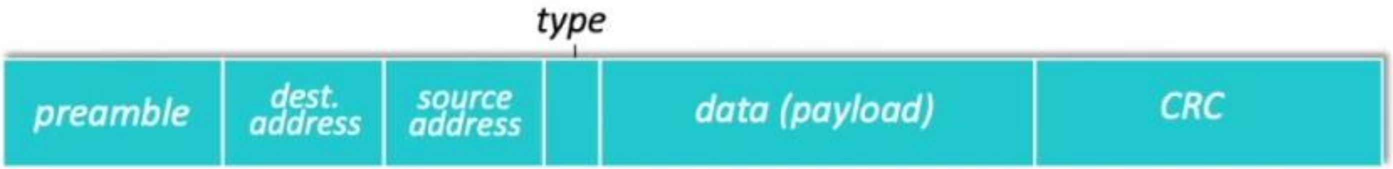
Preamble:

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

Ethernet Frame Structure

- **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



Ethernet: Unreliable, Connectionless

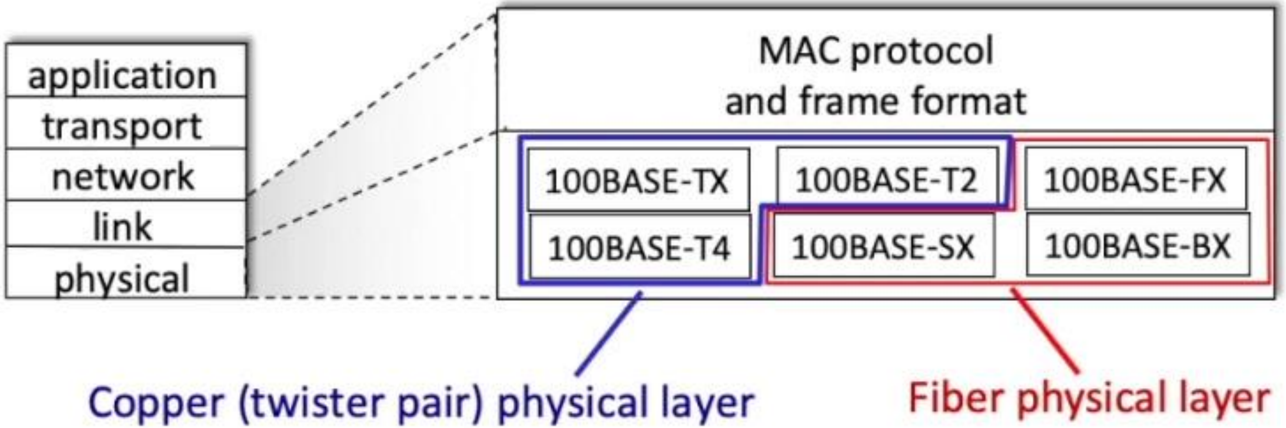
- **Connectionless:** No handshaking between sending and receiving NICs

- **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**

802.3 Ethernet Standards: Link & Physical Layer

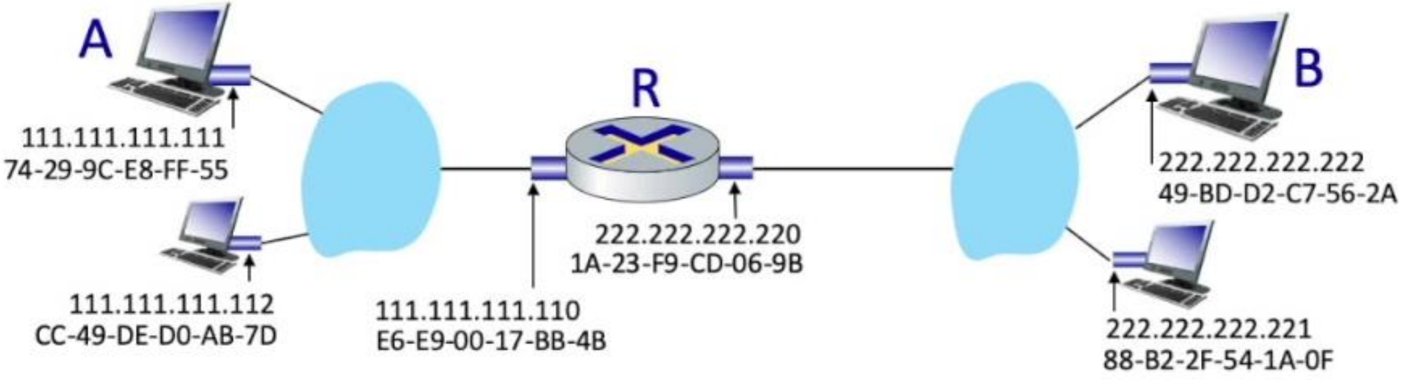
- **Many** different Ethernet standards
  - Common MAC protocol and frame format
  - Different speeds: 2 Mbps, 10 Mbps, 100 Mbps, 1Gbps, 10 Gbps, 40 Gbps
  - Different physical layer media: fiber, cable



Addressing: Routing to Another LAN

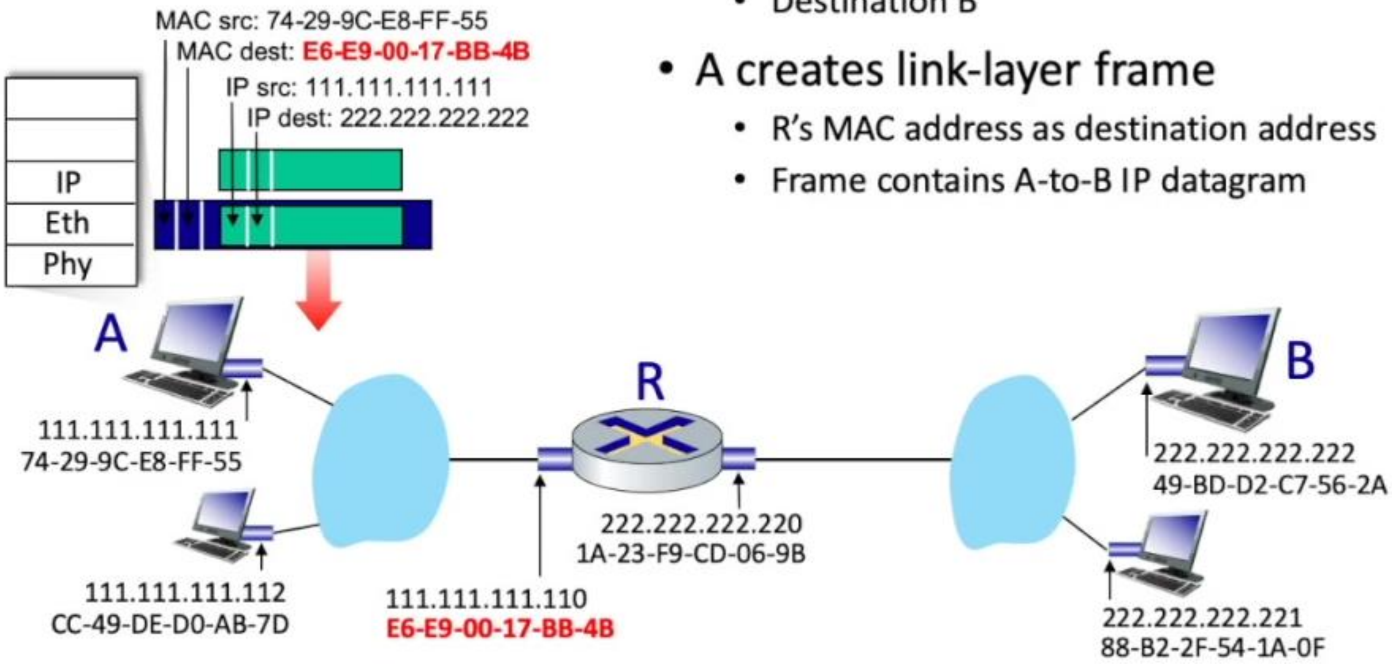
Walkthrough**: Send datagram from A to B via R**

- 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?)



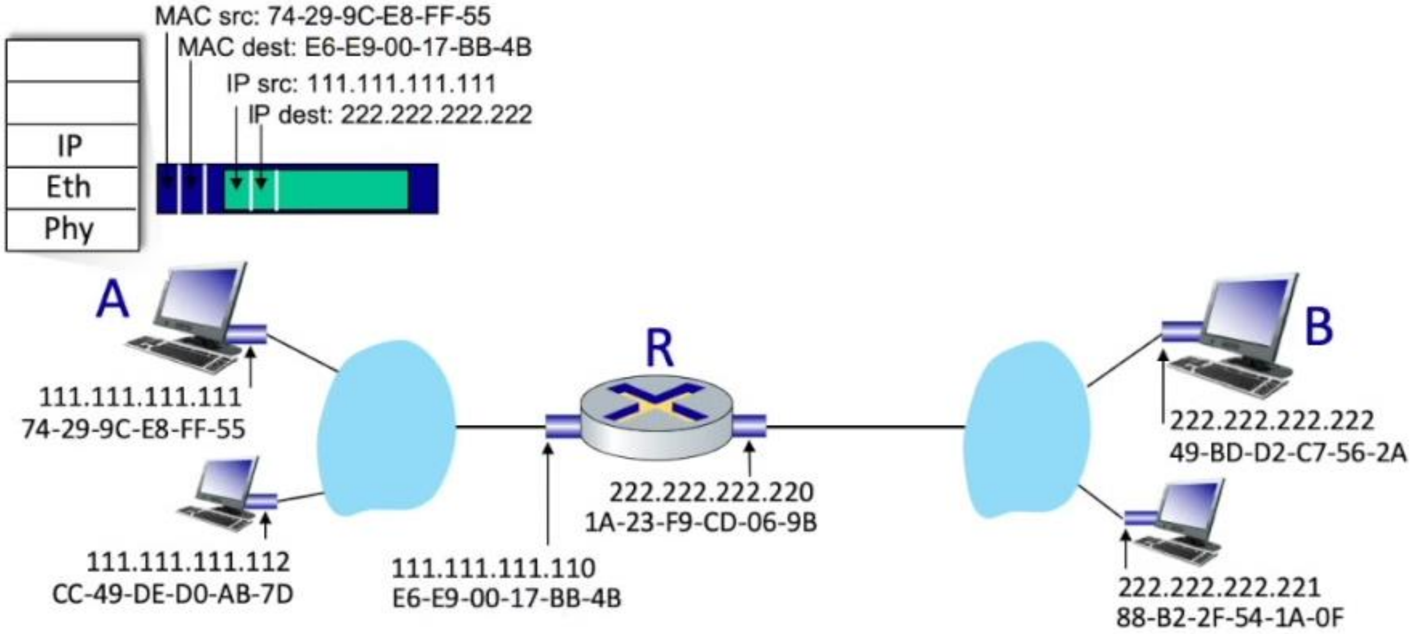
Addressing: Routing to Another LAN

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



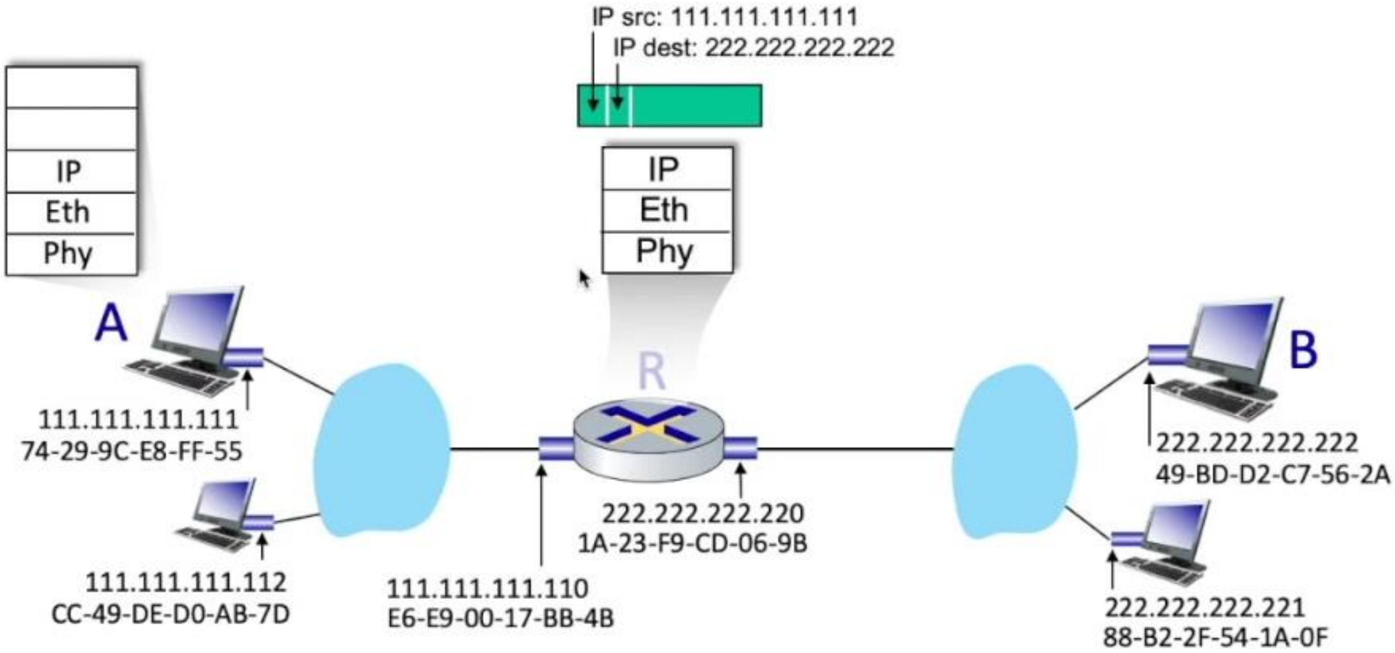
Addressing: Routing to Another LAN

- Frame sent from A to R
- Frame received at R → Datagram removed → Passed up to IP



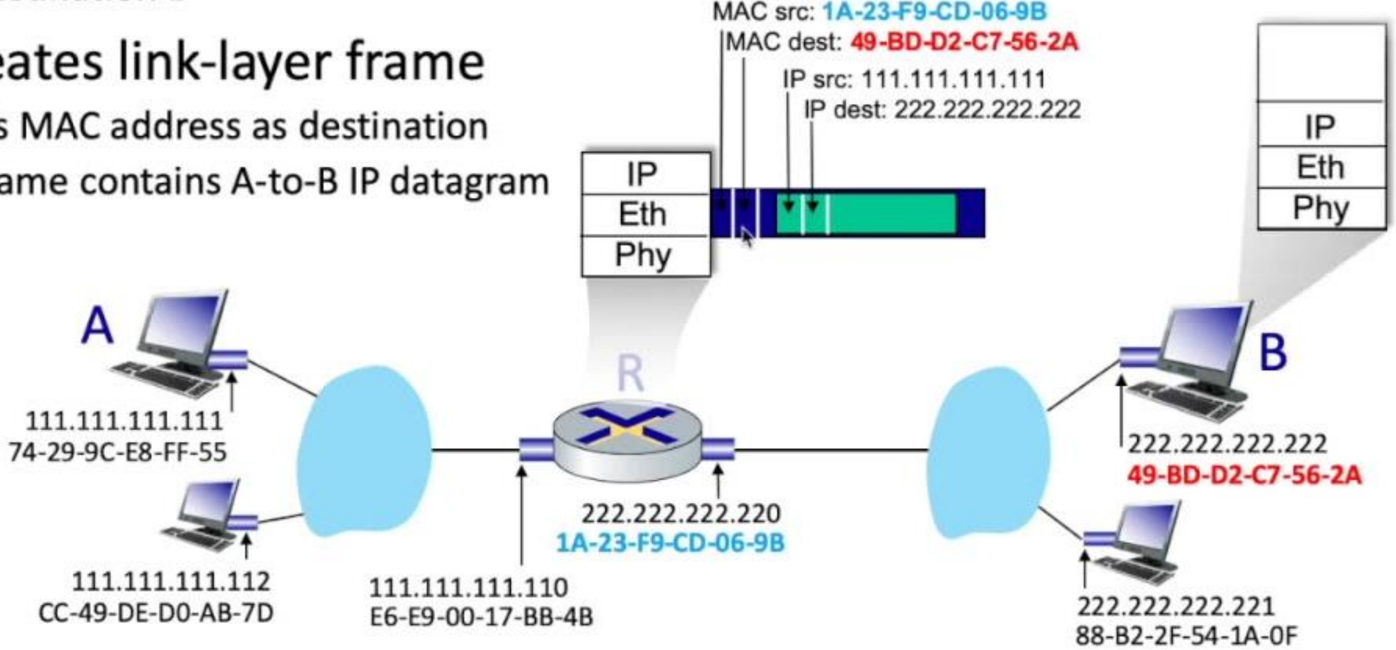
Addressing: Routing to Another LAN

- Frame sent from A to R
- Frame received at R → Datagram removed → Passed up to IP



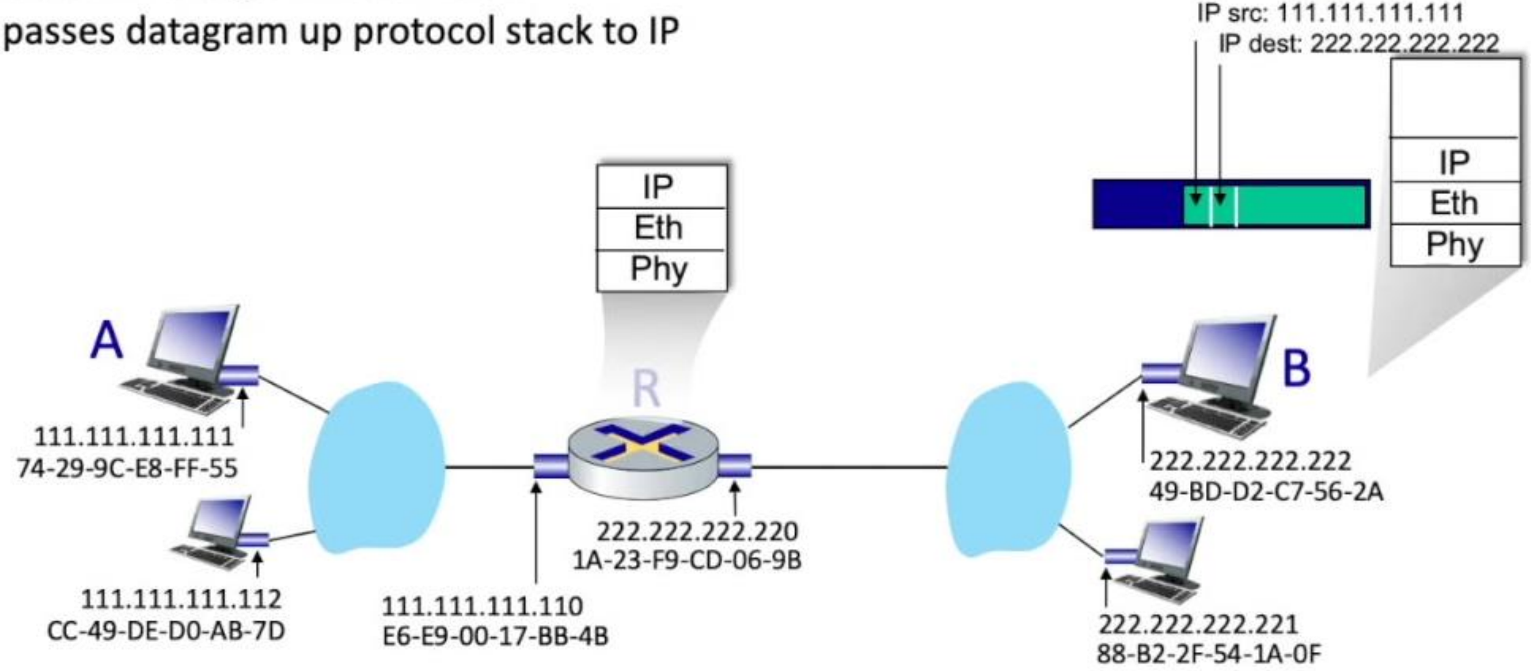
Addressing: Routing to Another LAN

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



Addressing: Routing to Another LAN

- B receives frame
    - Extracts IP datagram destination B
    - B passes datagram up protocol stack to IP



Link Layer Switch

- **Link-layer device: Takes an active role**
  - Store and 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

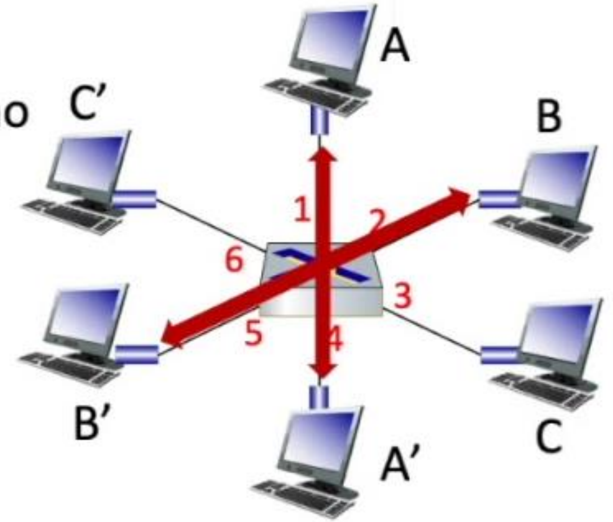
- **Plug-and-play & self-learning**
  - Switches do not need to be configured

Switch: Multiple Simultaneous Transmissions

- Hosts have dedicated, direct connection to switch

- 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)

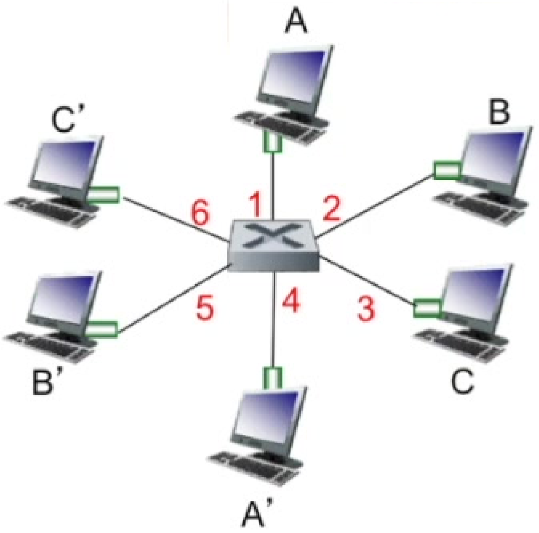
Switch Forwarding Table

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!

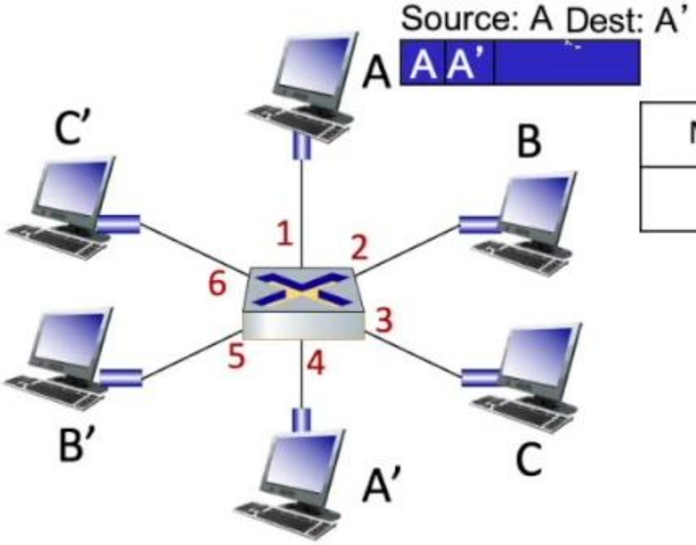
Q: How are entries created, maintained in switch table?
Something like a routing protocol?



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

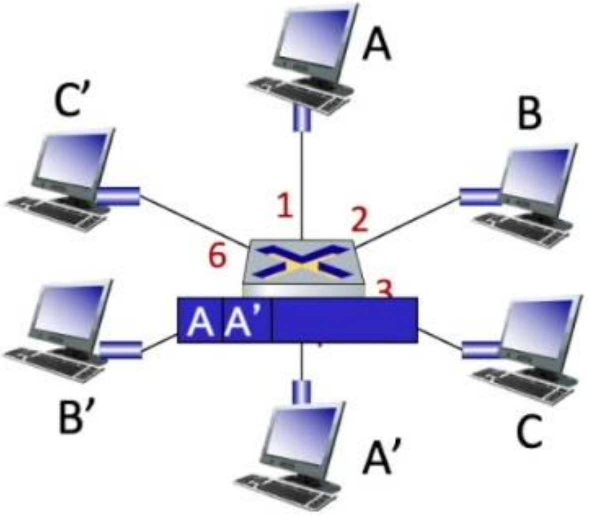


MAC addr	interface	TTL

Switch table (initially empty)

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

Switch table (initially empty)

Switch: Frame Filtering/Forwarding

When frame received at switch

- Record incoming link, MAC address of sending host
- Index switch table using MAC destination address

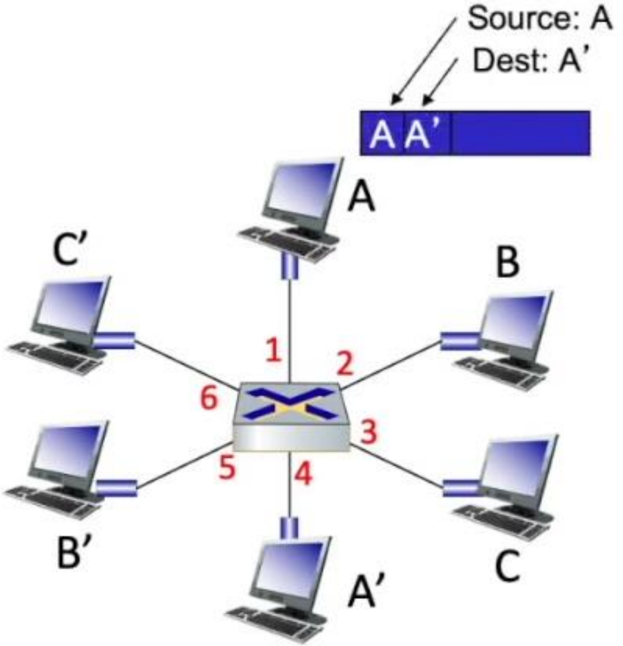
```
If entry found for destination
then {
if destination on segment from which frame arrived
  then drop frame
  else forward frame on interface indicated by entry
 }
 else flood /* forward on all interfaces except arriving interface */
```

Example: Self-Learning & Forwarding

- Frame destination, A’, location unknown: **Flood**
- Destination A location known: **Selectively send on just one link**

MAC addr	interface	TTL

Switch table (initially empty)

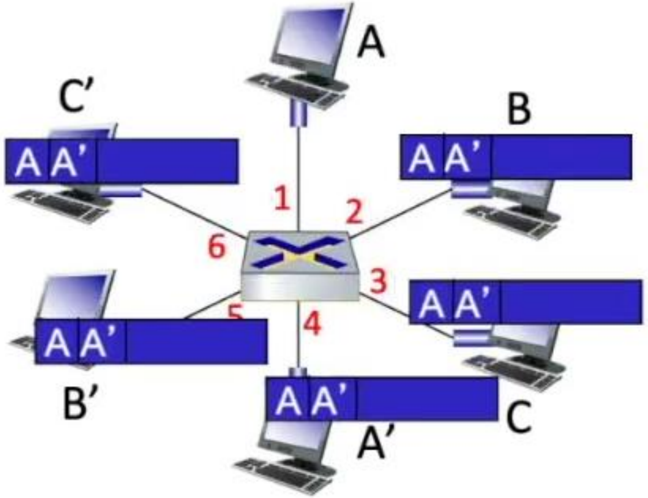


Example: Self-Learning & Forwarding

- Frame destination, A', location unknown: **Flood**
- Destination A location known: **Selectively send on just one link**

MAC addr	interface	TTL
A	1	60

Switch table (initially empty)

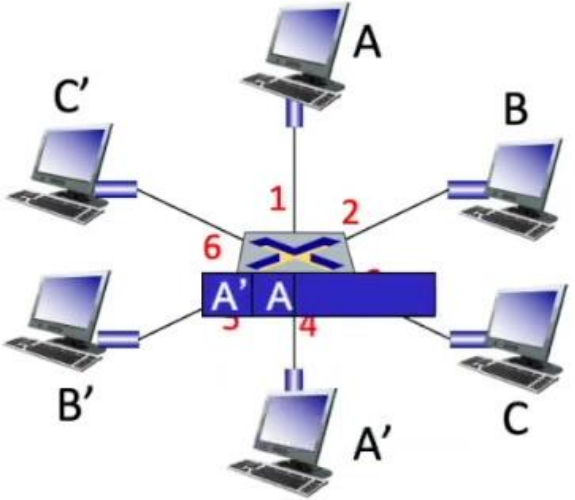


Example: Self-Learning & Forwarding

- Frame destination, A', location unknown: **Flood**
- Destination A location known: **Selectively send on just one link**

MAC addr	interface	TTL
A	1	60

Switch table (initially empty)

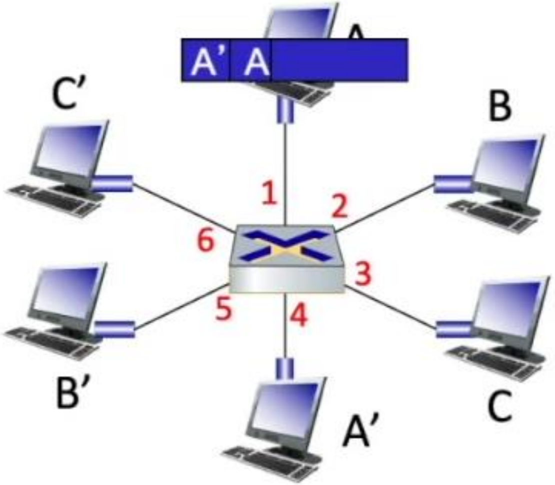


Example: Self-Learning & Forwarding

- Frame destination, A', location unknown: **Flood**
- Destination A location known: **Selectively send on just one link**

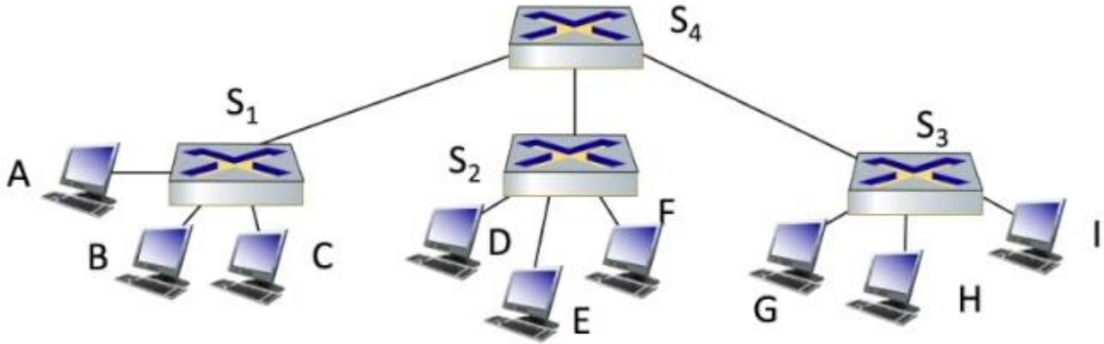
MAC addr	interface	TTL
A	1	60
A'	4	60

Switch table (initially empty)



Interconnecting Switches

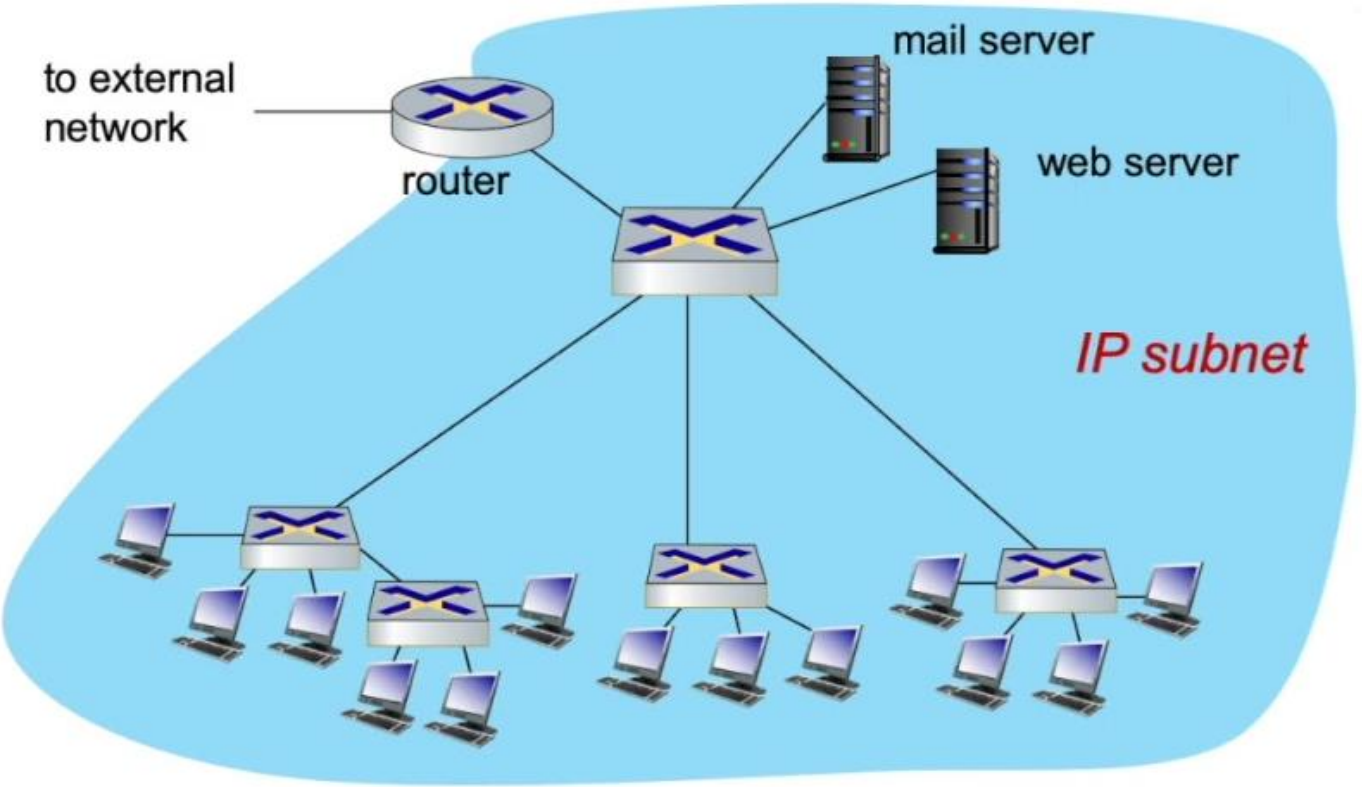
- Self-learning switches can be connected together:



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!

Institutional Network



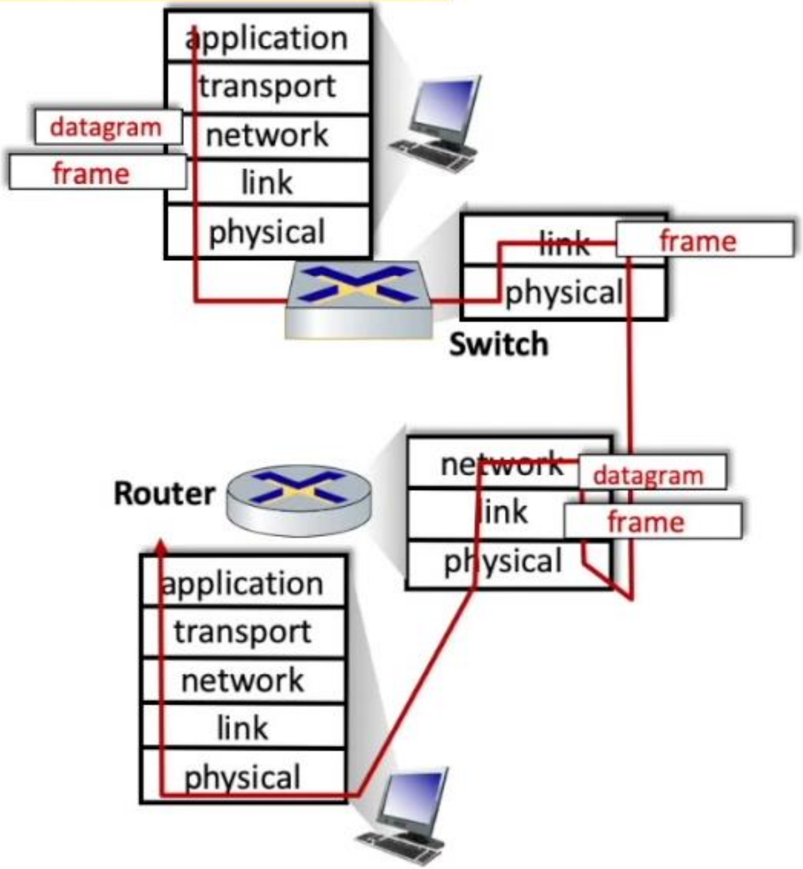
Switches vs. Routers

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

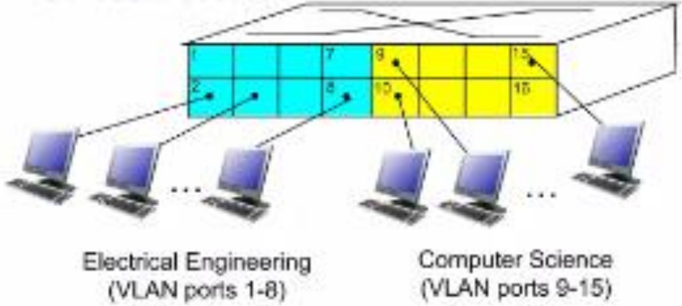


VLANs

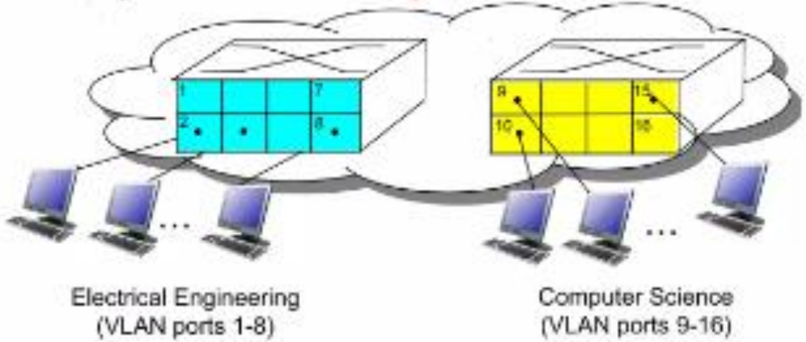
Virtual Local Area Network

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 ……

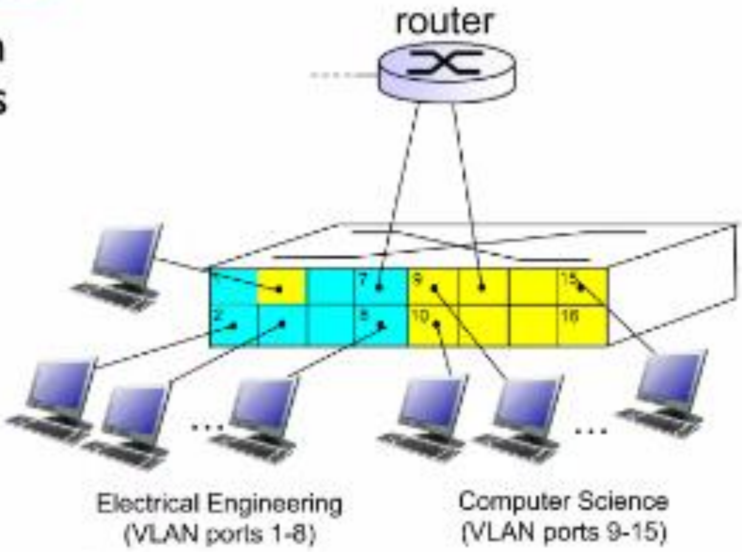


… operates as *multiple* virtual switches

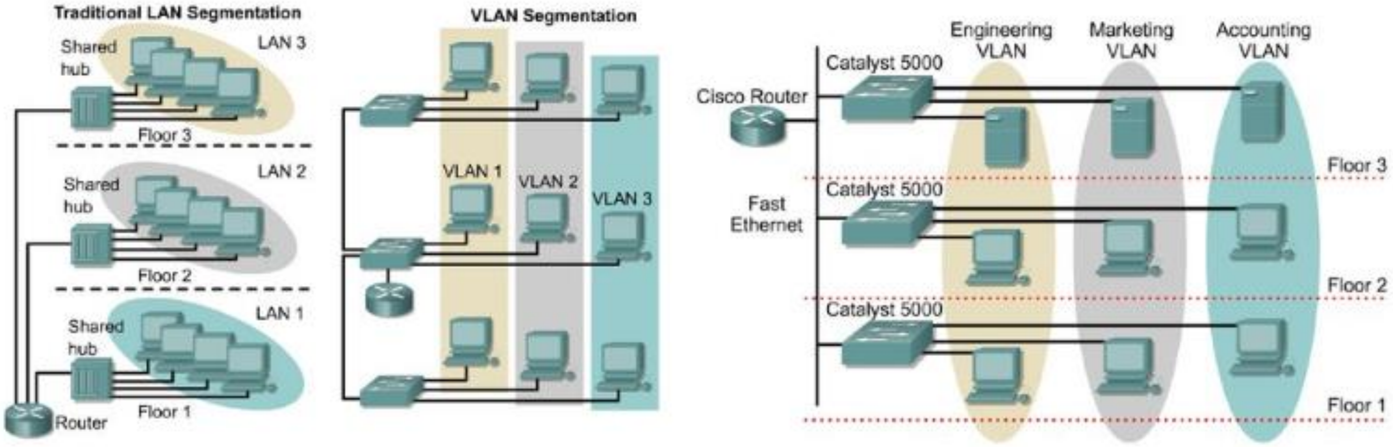


Port-based VLAN

- *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

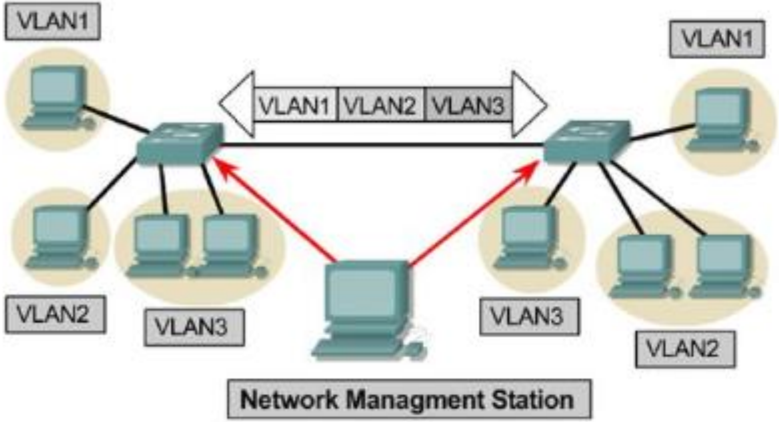


VLAN Introduction



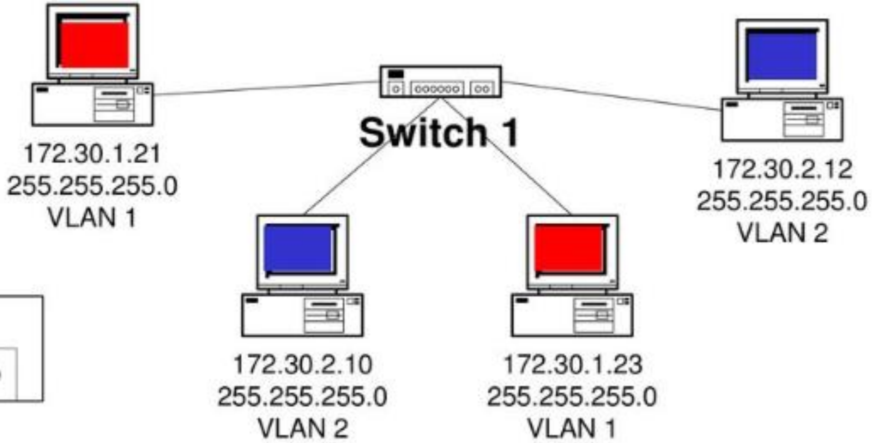
- VLANs provide segmentation based on broadcast domains.
- VLANs logically segment switched networks based on the functions, project teams, or applications of the organization regardless of the physical location or connections to the network.
- All workstations and servers used by a particular workgroup share the same VLAN, regardless of the physical connection or location.

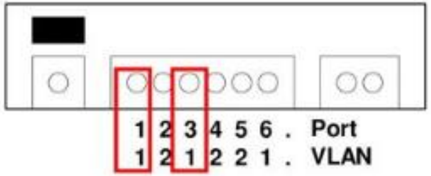
VLAN Operation



- Static membership VLANs are called port-based and port-centric membership VLANs.
- As a device enters the network, it automatically assumes the VLAN membership of the port to which it is attached.
- The default VLAN for every port in the switch is the management VLAN. The management VLAN is always VLAN 1 and may not be deleted.
- All other ports on the switch may be reassigned to alternate VLANs.

VLAN Operation





1	2	3	4	5	6	.	Port
1	2	1	2	2	1	.	VLAN

Two VLANs

- Two Subnets

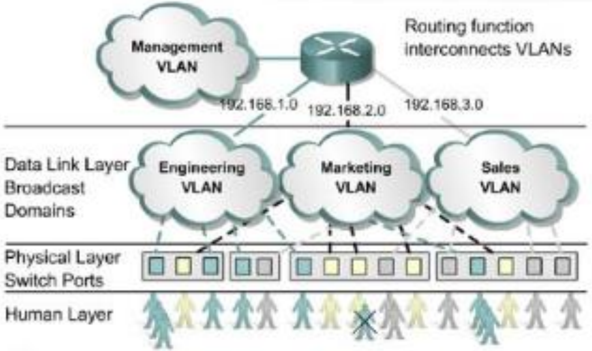
Important notes on VLANs:

1. VLANs are assigned on the switch port. There is no VLAN assignment done on the host.
2. In order for a host to be a part of that VLAN, it must be assigned an IP address that belongs to the proper subnet.

   Remember: VLAN = Subnet
3. Assigning a host to the correct VLAN is a 2-step process:
   1. Connect the host to the correct port on the switch.
   2. Assign to the host the correct IP address depending on the VLAN membership

Benefits of VLANs

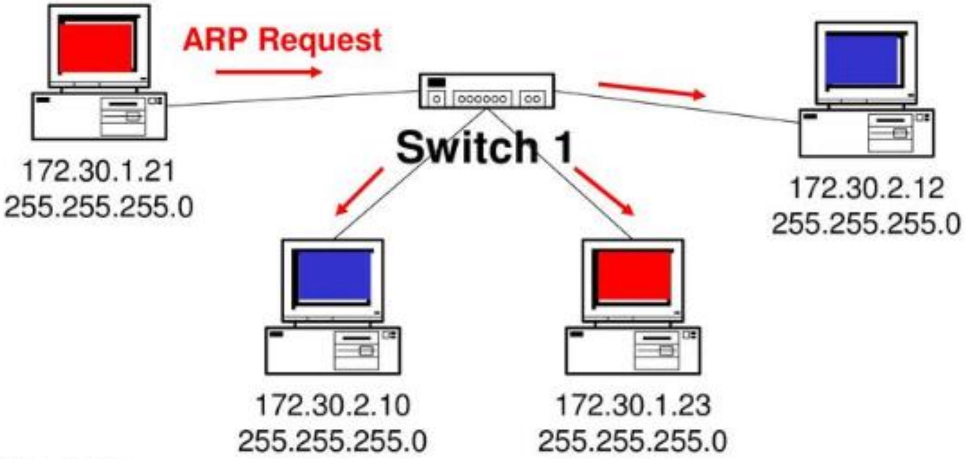
All users attached to the same switch port must be in the same VLAN.



If a hub is connected to a specific VLAN port on a switch, all devices on that hub must belong to the same VLAN.

- The key benefit of VLANs is that they permit the network administrator to organize the LAN logically instead of physically.
- This means that an administrator is able to do all of the following:
  - Easily move workstations on the LAN.
  - Easily add workstations to the LAN.
  - Easily change the LAN configuration.
  - Easily control network traffic.
  - Improve security.

Without VLANs – No Broadcast Control

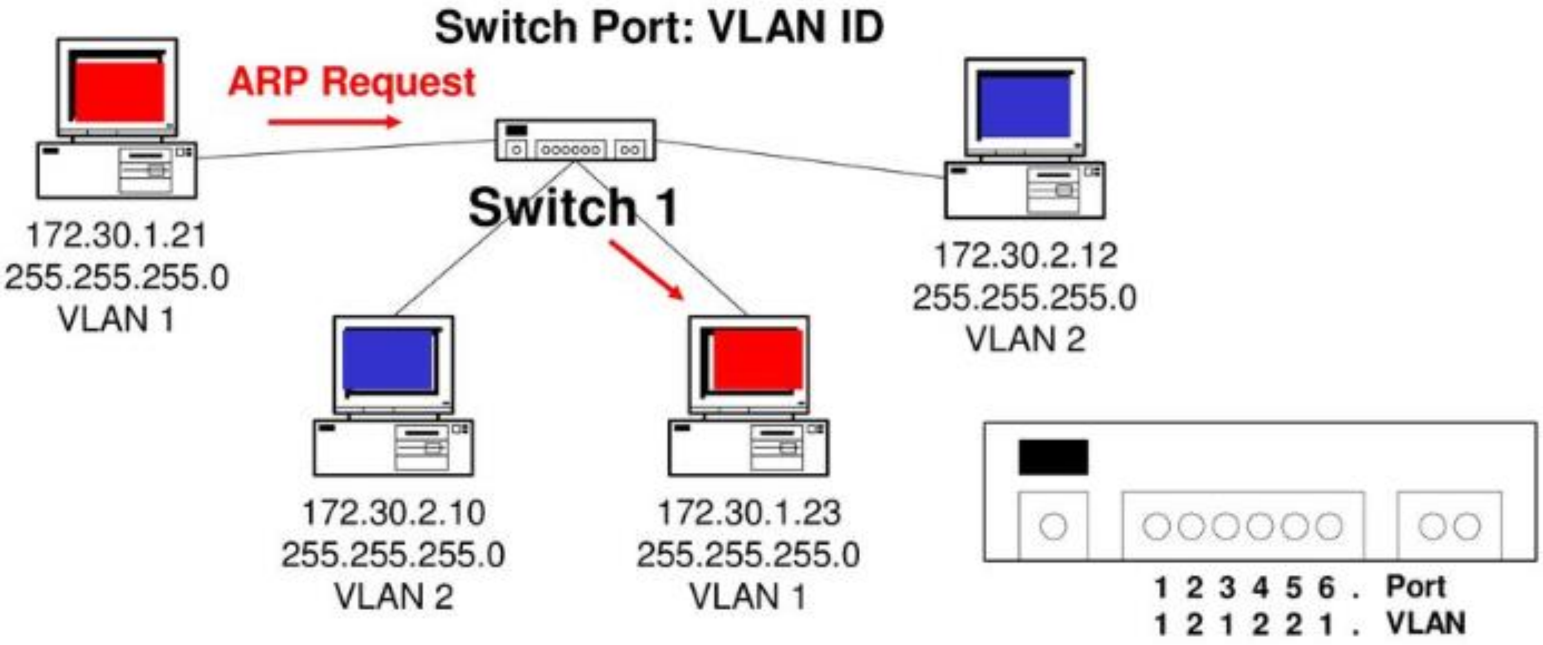


No VLANs

- Same as a single VLAN
- Two Subnets

- Without VLANs, the ARP Request would be seen by all hosts.
- Again, consuming unnecessary network bandwidth and host processing cycles.

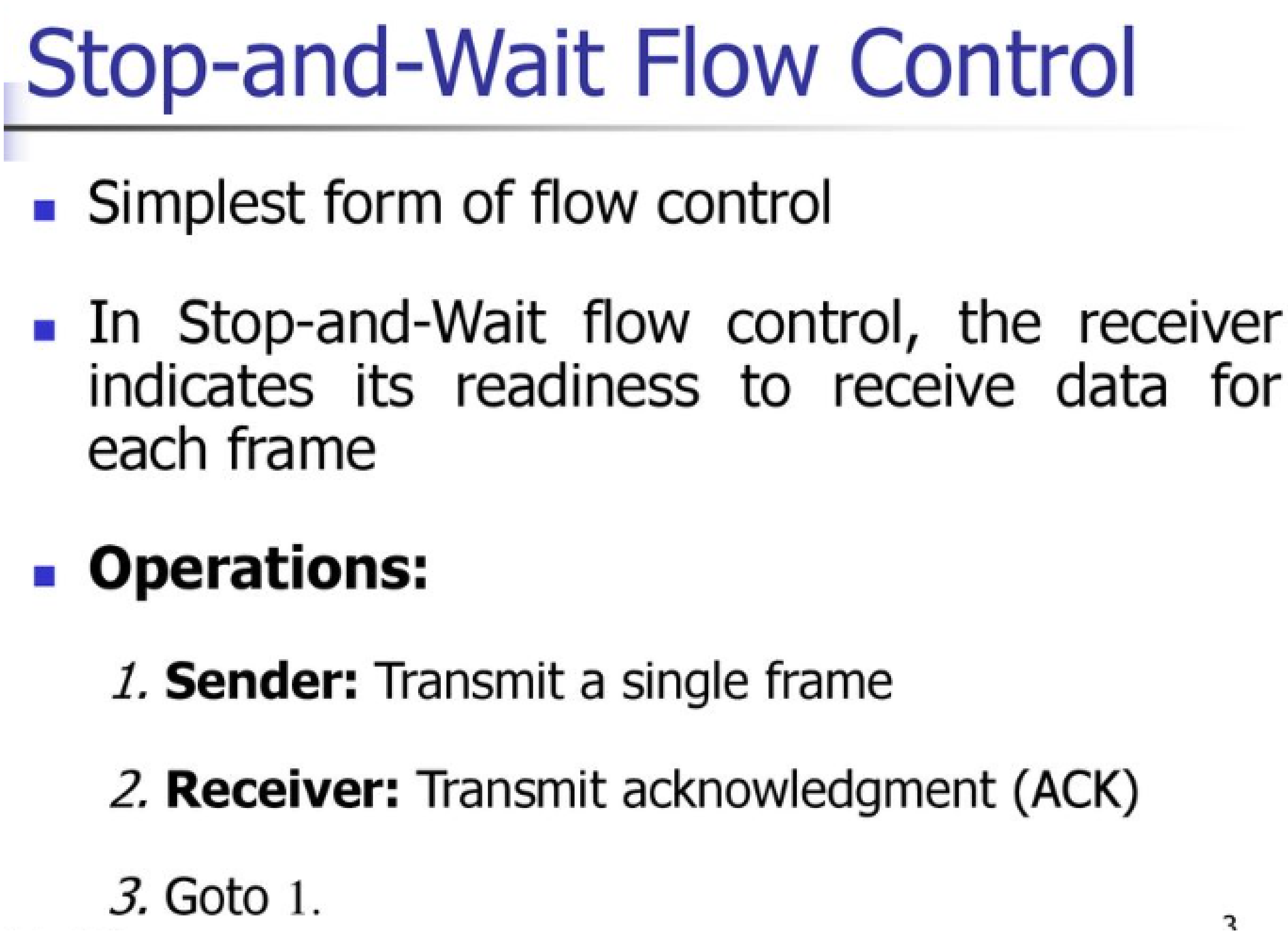
With VLANs – Broadcast Control



1	2	3	4	5	6	.	Port
1	2	1	2	2	1	.	VLAN

Two VLANs

- Two Subnets



Stop-and-Wait Flow Control

- Simplest form of flow control
- In Stop-and-Wait flow control, the receiver indicates its readiness to receive data for each frame
- **Operations:**

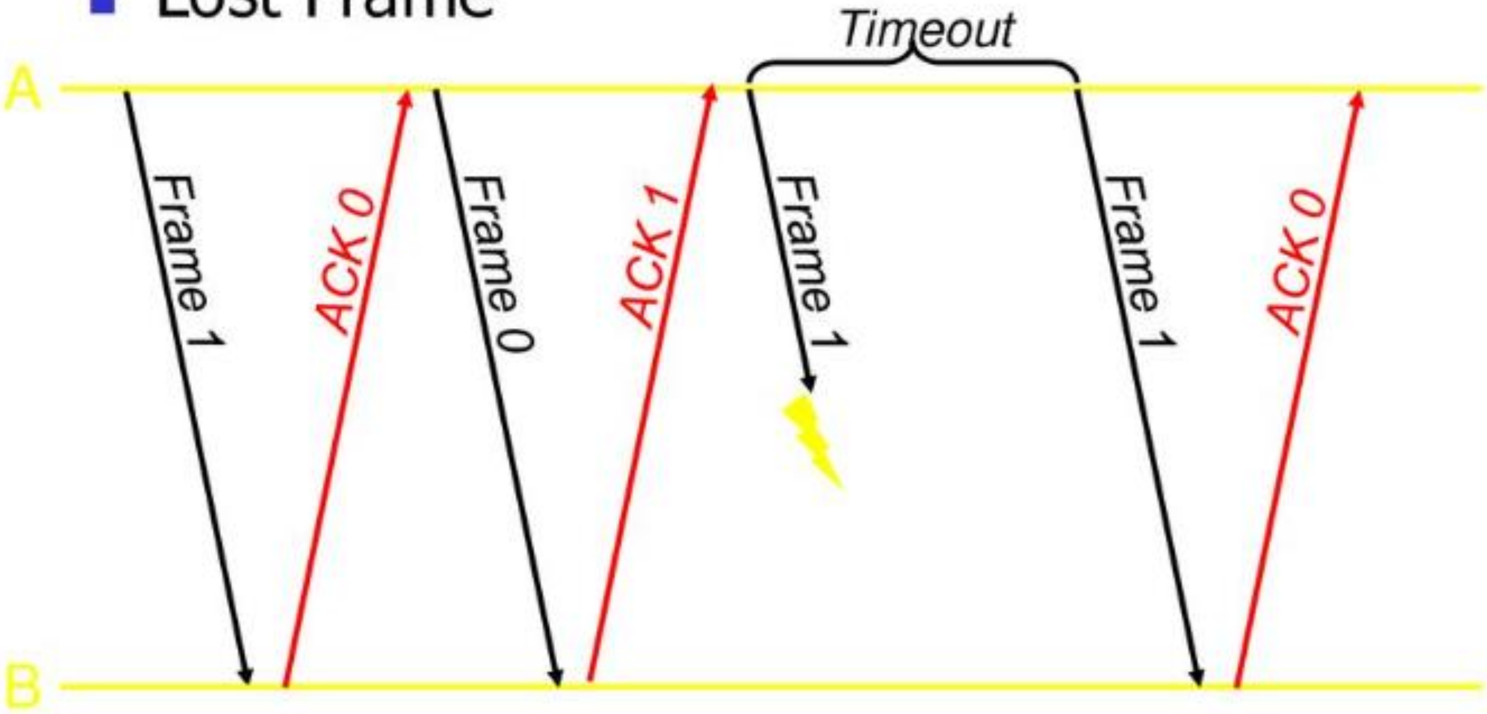
1. **Sender:** Transmit a single frame

2. **Receiver:** Transmit acknowledgment (ACK)

3. Goto 1.

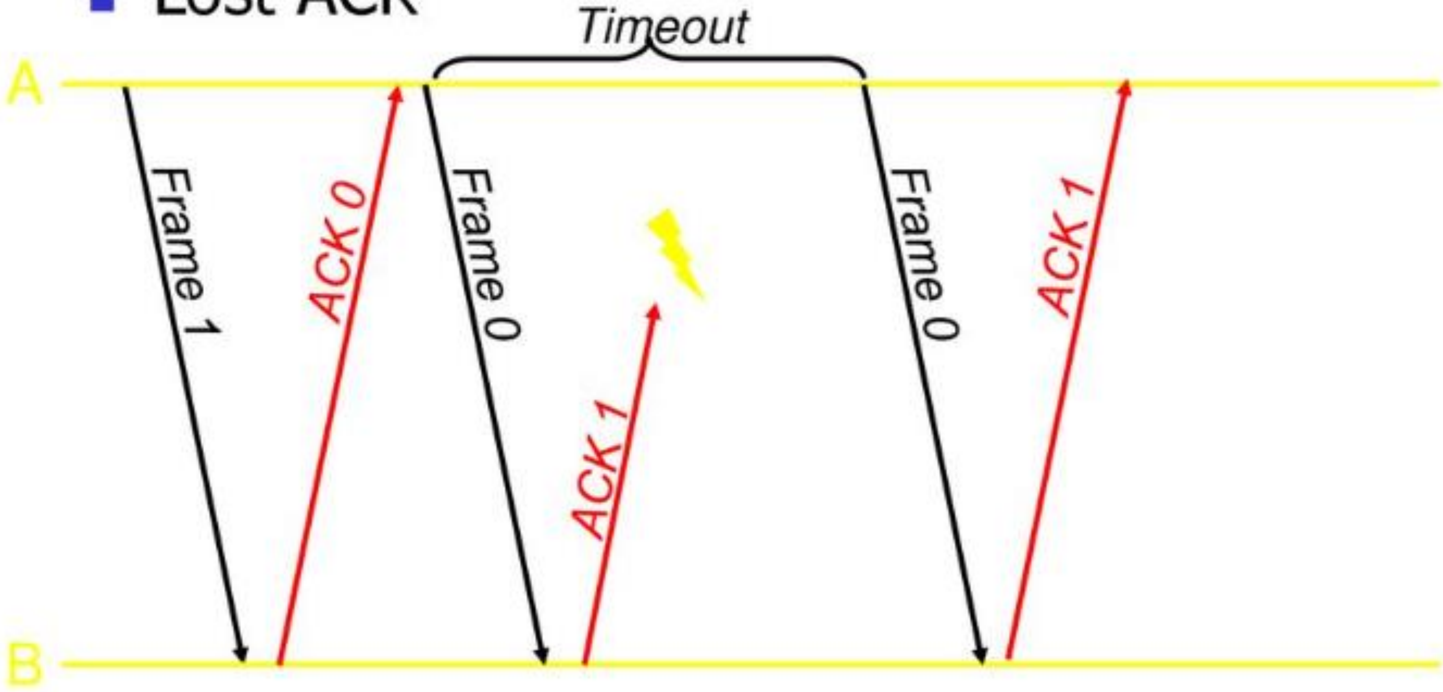
Stop-and-Wait ARQ

- Lost Frame



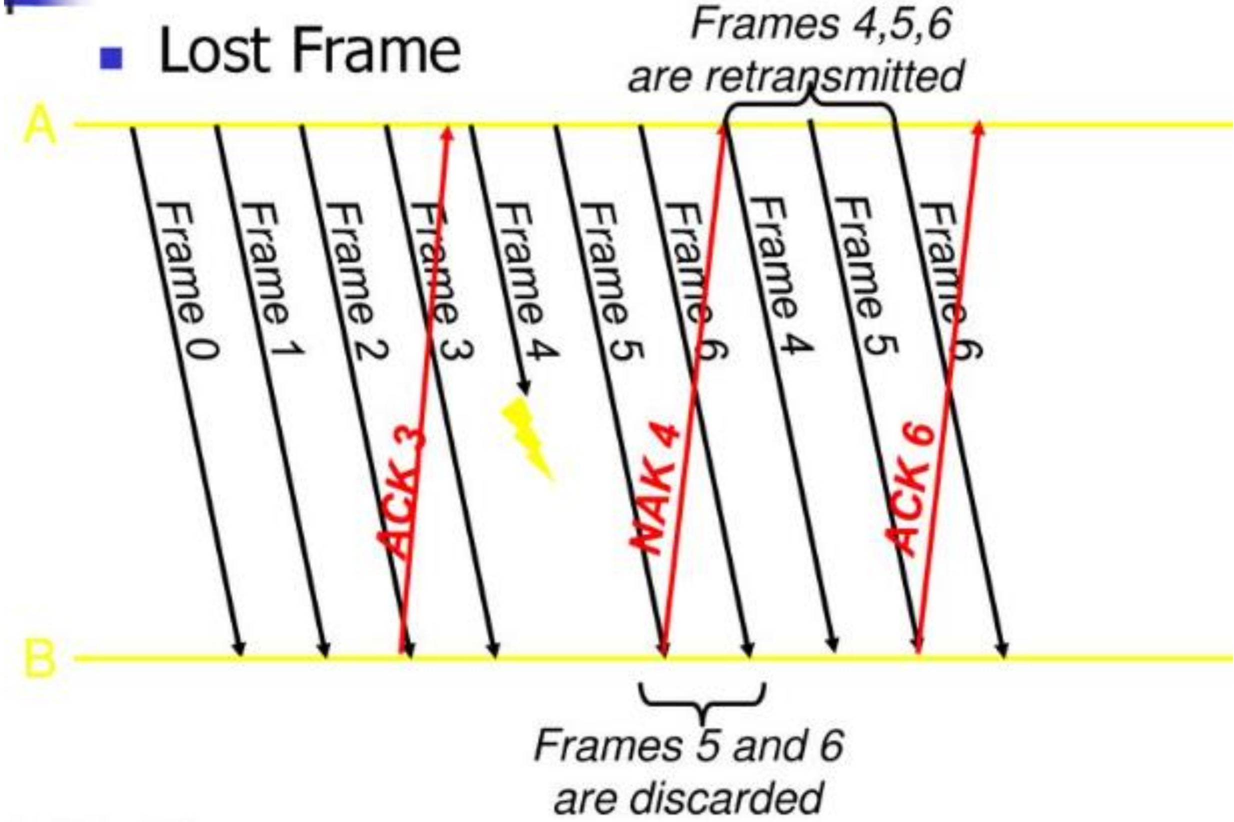
Stop-and-Wait ARQ

- Lost ACK



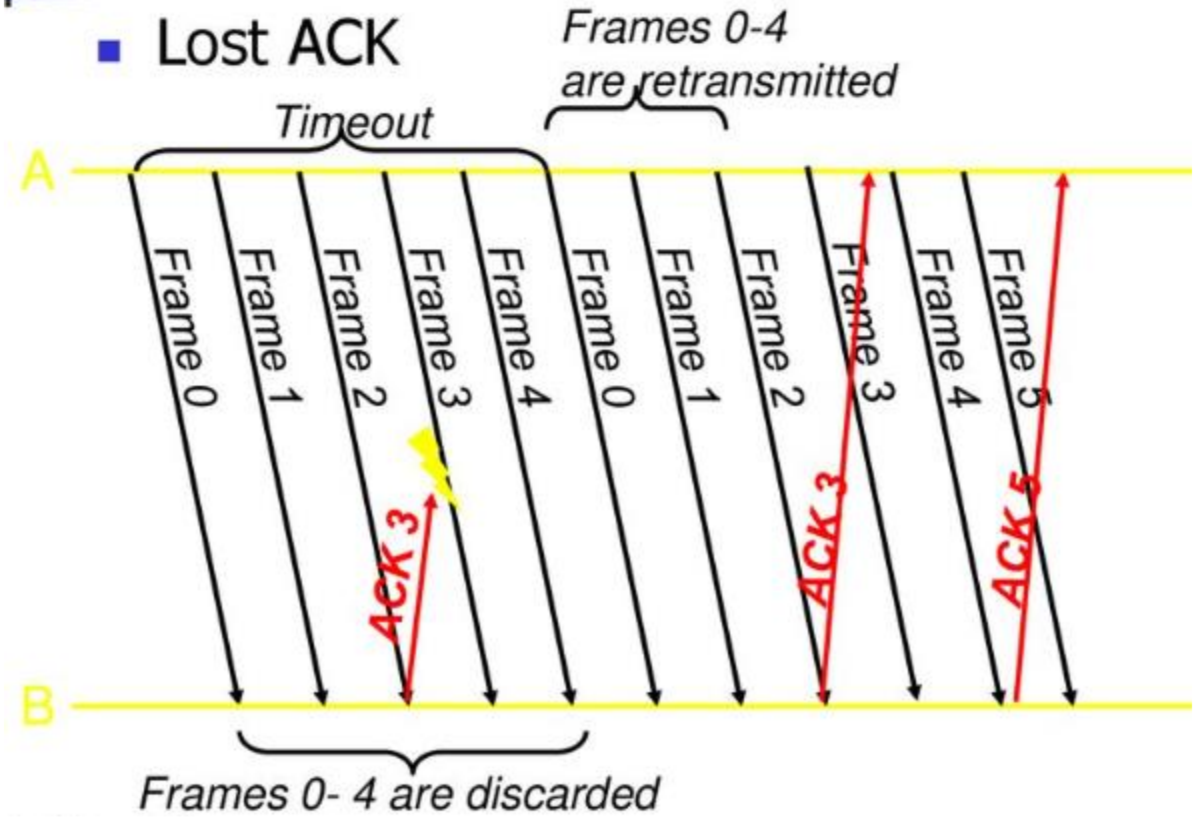
Go-Back-N ARQ

- Lost Frame



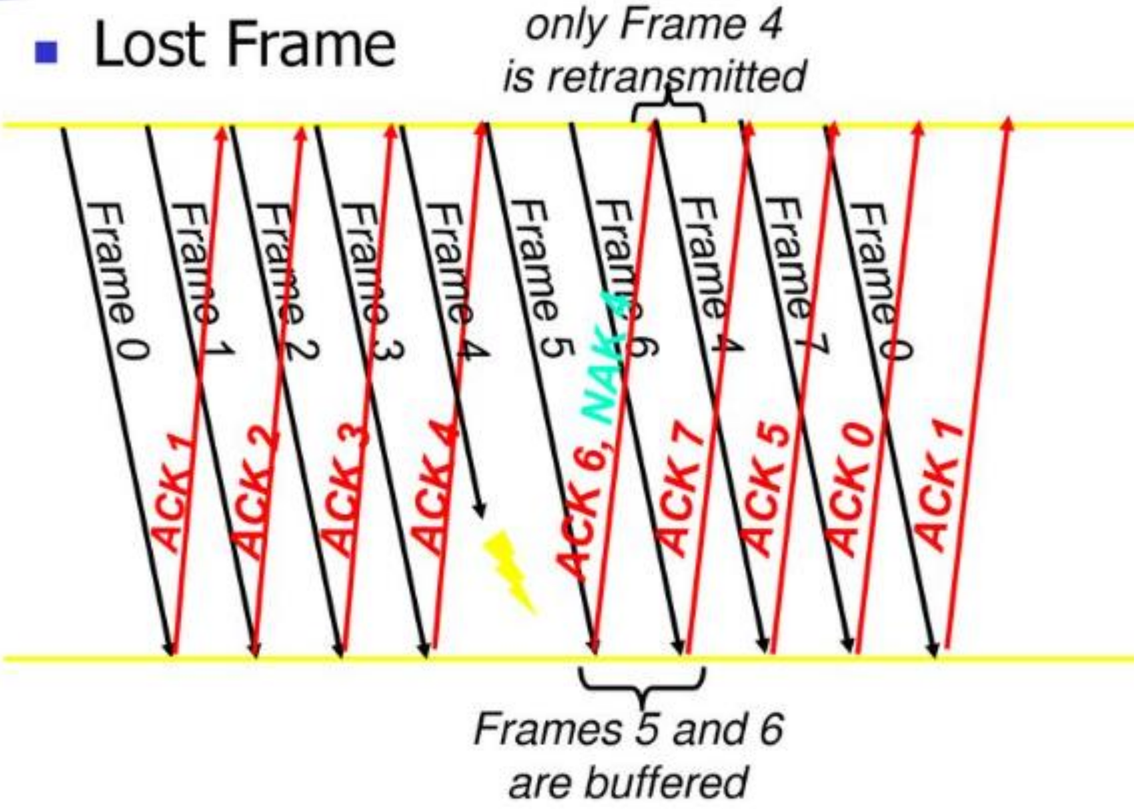
Go-Back-N ARQ

- Lost ACK

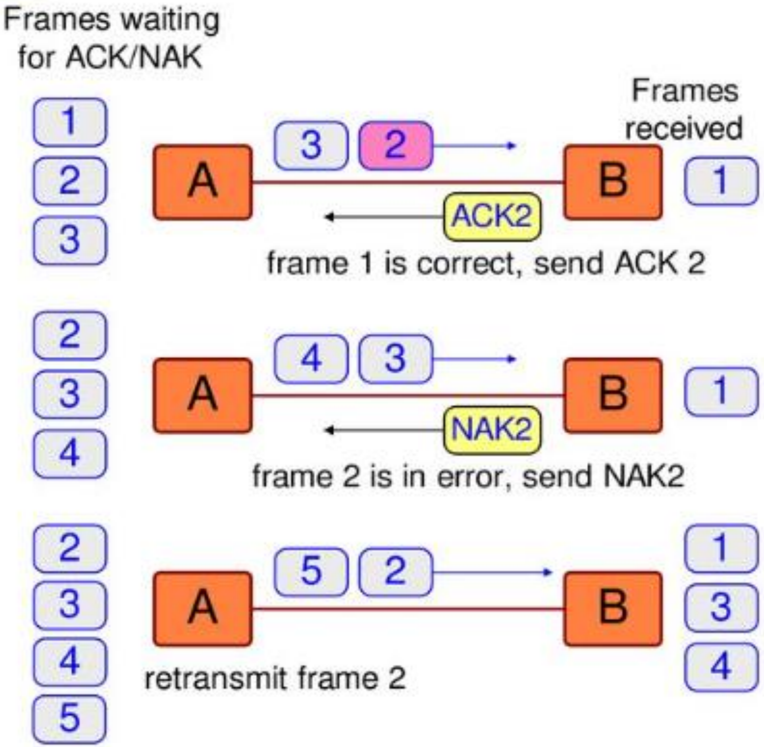


Selective-Repeat ARQ

- Lost Frame



Example of Selective-Repeat ARQ


Frames waiting for ACK/NAK
1
2
3
A
3
2
B
Frames received
1
ACK2
frame 1 is correct, send ACK 2
2
3
4
A
4
3
B
1
NAK2
frame 2 is in error, send NAK2
2
3
4
5
A
5
2
B
1
3
4
retransmit frame 2

Receiver must keep track of `holes' in the sequence of delivered frames

Sender must maintain one timer per outstanding packet