

Ethernet LAN and Extended LAN

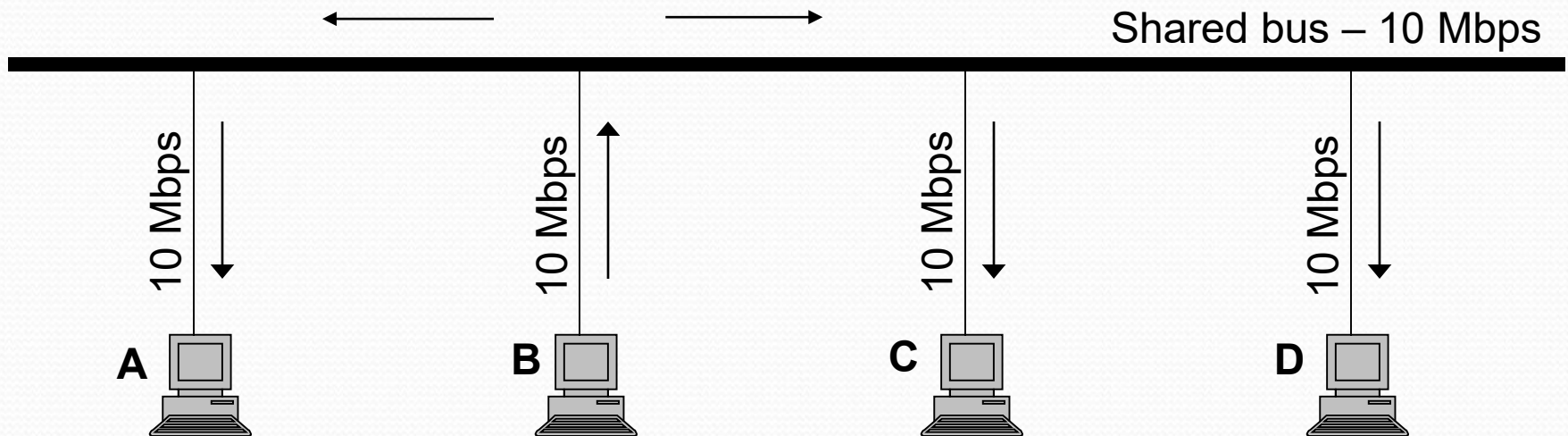
Key Reference:

Peterson and Davie, "Computer Networks:
A Systems Approach", 4th Edition, Morgan
Kaufmann, 2007

Medium Access Control (MAC)

- Broadcast LAN (See Figure in next slide)
- MAC is a sublayer of the data link layer (Layer 2)
- MAC is to control access to multi-access (multiple-access or random access) channels (links, lines)
- When more than one host share a channel we need MAC
- Usually used in LANs
- Ethernet (IEEE 802.3 standard) (popularly used)
- Token ring (802.5, FDDI) (not popular now)
- Wireless (802.11) (popularly used)

Ethernet Broadcast LAN



Ethernet Cabling

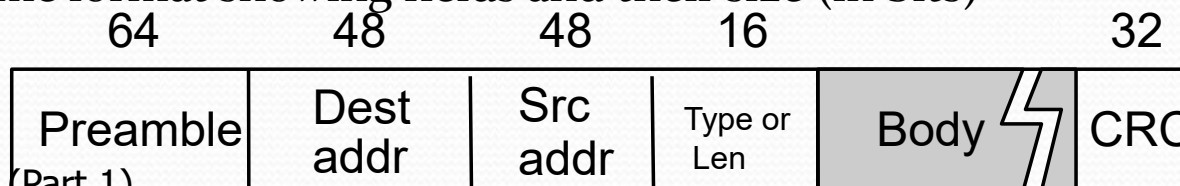
- In traditional Ethernet, hosts are connected to Ethernet cable (shared bus) through adapters
 - Coaxial cable, half-duplex, Up to 10 Mbps,
- Length: Max 2500m
 - There can be at most 5 segments (length up to 500 m) separated by 4 repeaters.
- Later hub and switch based Ethernet evolved
 - Eg: twisted-pair cable based star-connected hub
- See the table in the next slide which lists a few types of Ethernet and features

Different kinds of Ethernet LANs

Name (10Mbps)	cable	Max segment	Nodes/ segment
10Base5	Thick coax	500 m	100
10Base2	Thin coax	200 m	30
10Base-T	Twisted pair	100 m	1024
10Base-F	Fiber optics	2000 m	1024

Ethernet Frame Format

- Manchester encoding is used (Later 4B/5B & others as well)
 - Low-to-high transition to encode a 0 and high-to-low transition to encode a 1
 - 1: transmit high signal followed by low signal
 - 0: transmit low signal followed by high signal
- Preamble
 - 7 bytes (10101010) used for clock synchronization
 - 1 byte (10101011) used to mark the start of frame
- Type (or length)
 - Used as a demultiplexing key. Usually >1500
 - (eg: VLAN frame, ARP frame)
 - Can also be used as a length field (0 to 1500 bytes)
 - Frame format showing fields and their size (in bits)



Ethernet Addresses

- Addresses
 - Unique world-wide, 48-bit unicast address assigned to each adapter
 - example: **08 : c0 : 65 : b1 : 2a : 5d**
 - broadcast: all 1s: **ff: ff: ff: ff: ff: ff**
 - multicast: first bit is 1 (**the rightmost bit of the most significant byte**)
 - First byte is the most significant byte
- Ethernet adapter receives all frames and accepts
 - Frames addressed to it
 - Frames addressed to the broadcast address
 - Frames addressed to a multicast address if instructed
 - All frames if it operates in promiscuous mode (eg: Ethernet switch)

CSMA - CD

- Ethernet uses CSMA-CD technique
- CSMA-CD: carrier sense multiple access – collision detection
- Carrier sense
 - A host senses the link and can distinguish if the link is idle or busy (if there is any signal transmission going on the link or not)
- Collision detect
 - A host listens what it is transmitting and therefore can detect if it collides with any other frame transmitted by some other host

1-Persistent CSMA

- p-persistent CSMA
 - If a host is ready to send a frame, it continuously senses the channel (link). If it is idle, then transmit frame with probability p
- 1-persistent CSMA
 - If a host is ready to send a frame, it continuously senses the channel (link). If it is idle, then transmit frame with probability 1
 - Ethernet uses 1-persistent protocol

Transmit Algorithm

- A node (host) can transmit independent of what other nodes (hosts) are doing.
- uses *exponential backoff algorithm* to dynamically adapt to the number of nodes (hosts) trying to send
 - To estimate the number of active hosts in the event of collisions
- If line (link) is idle...
 - send immediately
 - upper bound message size of 1500 bytes
- If line (link) is busy...
 - wait until idle and transmit immediately

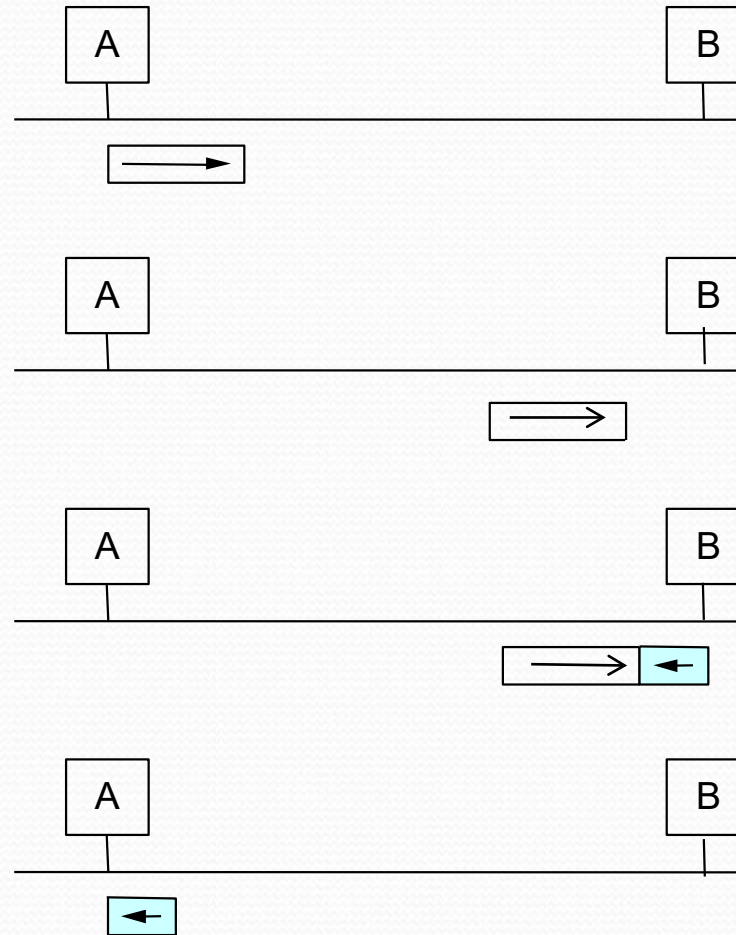
Transmit Algorithm (contd.)

- If collision...
 - Transmit a 32 bit jamming sequence (noise burst) along with 64 bit preamble, then stop transmitting frame (Why jam signal?)
 - delay and try again
 - 1st collision: waits for n slots where n is chosen randomly from the interval $[0,1]$, 1 slot is usually $51.2\mu\text{s}$
 - 2nd consecutive collision: waits for n slots where n is chosen randomly from the interval $[0,3]$
 - i^{th} consecutive collision: waits for n slots where n is chosen randomly from the interval $[0,2^i - 1]$
 - for $i > 10$, the interval used is $[0,2^{10} - 1]$
 - give up after several tries (usually 16)

Minimum frame size

- Minimum size is 64 bytes: 14 bytes header 46 bytes data, and 4 bytes CRC (WHY?) A situation should not arise wherein, the sending host has transmitted the frame, without detecting any collision, but there is actually a collision
- From the Figure (shown in next slide, see reference book Peterson and Davie), it is observed that a host needs to send for “RTT” to detect all possible collisions ($T_f \geq RTT$)
 - Host A finds the link is free and sends a frame
 - Just before the arrival of frame bits, host B finds that the link is free and starts transmitting a frame
 - Collision occurs near host B’s link interface which is detected by Host B; jam signal is sent by host B
 - If host A does not transmit its frame for “RTT”, it cannot detect collision
 - Because it has stopped frame transmission when the jam signal reaches
- Ethernet length is limited to 2500 m and four repeaters; for this case RTT is estimated to be bound by $51.2\mu s = 512$ bits (64 bytes) for 10 Mbps Ethernet
 - $T_f = \text{size}/B$
- [Why jam signal?] If host B terminates its transmission after sending only a very few bits without jam sequence, host A may not receive sufficient energy to detect collision

Need for minimum frame size



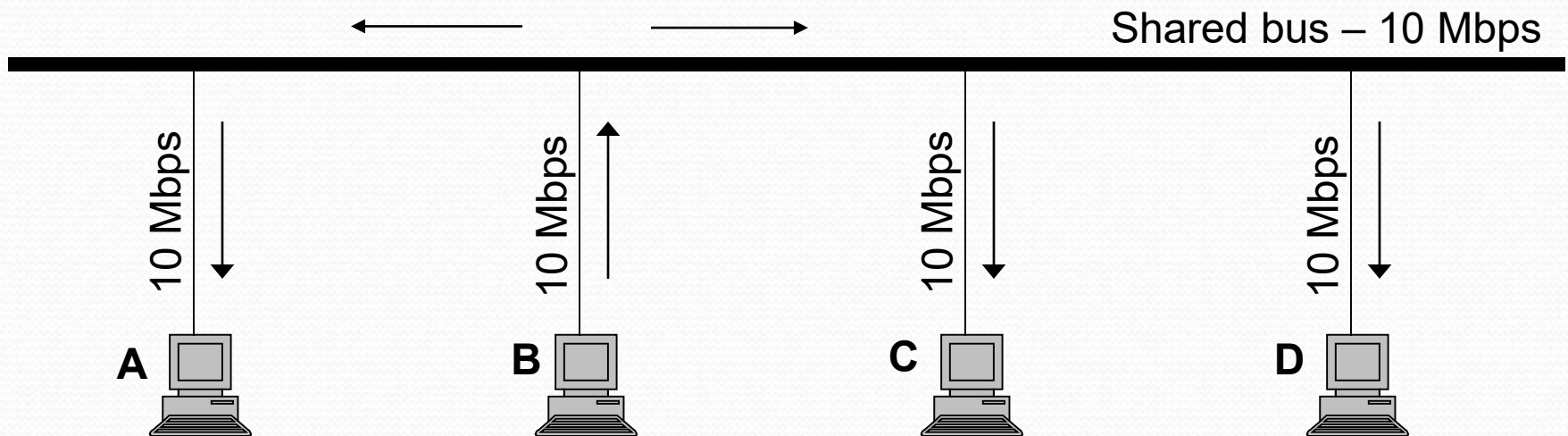
LAN Hubs and Switches

- Shared-Medium Bus (also referred to as a bus)
 - popular from late 1970s to early 1990s
- Shared-Medium Hub (also referred to as a hub)
 - popular from late 1990s
- Switching Hub (also referred to as a switch)
 - popular from early 2000s

Bus-based LAN

- Shared-Medium Bus (or simply a bus)
 - Bus configuration
 - Traditional Ethernet (e.g. 10BASE 5)
 - Single collision domain
 - Set of nodes (hosts) wherein a frame sent by a node can possibly collide with frames sent by any other node
 - 10 Mbps Bus LAN
 - 10Mbps shared by all hosts; total no of bits transmitted by all hosts in one second is at most 10 million
 - All stations (hosts) share the total capacity of the LAN or bus
 - One station transmits, others receive
 - Cable cut disconnects the network

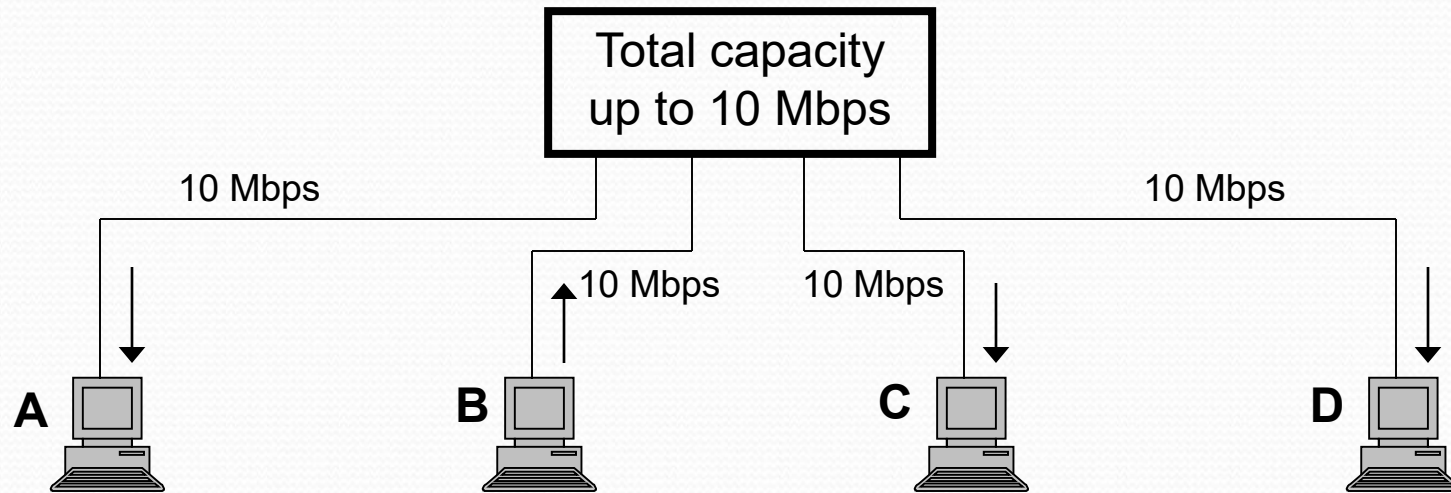
Bus LAN



Hub-based LAN

- Shared-Medium Hub (or simply a hub)
 - Star configuration, e.g. 10BASE-T Ethernet
 - E.g. 802.3u 100 Mbps Fast Ethernet (e.g. 100 BASE T)
 - When a frame is received on a port, the hub copies it to all the other ports
 - Single collision domain, hub transmits jam signal to all when collision occurs
 - All stations share the total capacity of the LAN or hub
 - 10 Mbps Hub LAN: 10Mbps shared by all hosts; total no of bits transmitted by all hosts in one second is at most 10 million
 - One station transmits, others receive
 - Can exploit building wiring practices for cable layout
 - Hub can recognize a malfunctioning station that jams the network and remove it from the network
 - Cable cut does not disconnect the network

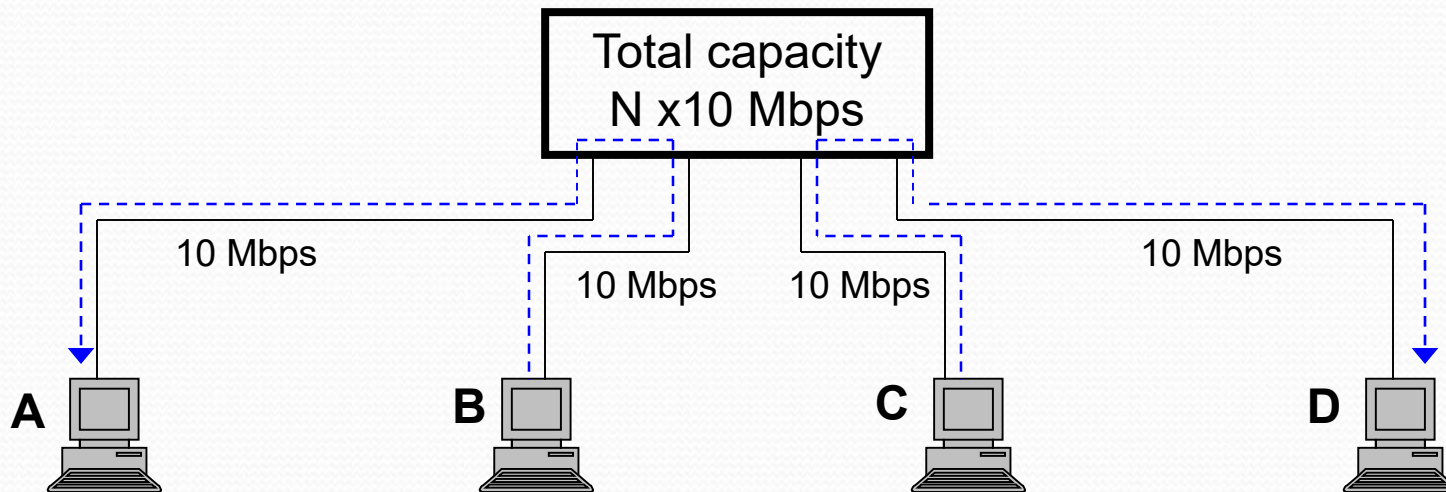
Hub LAN



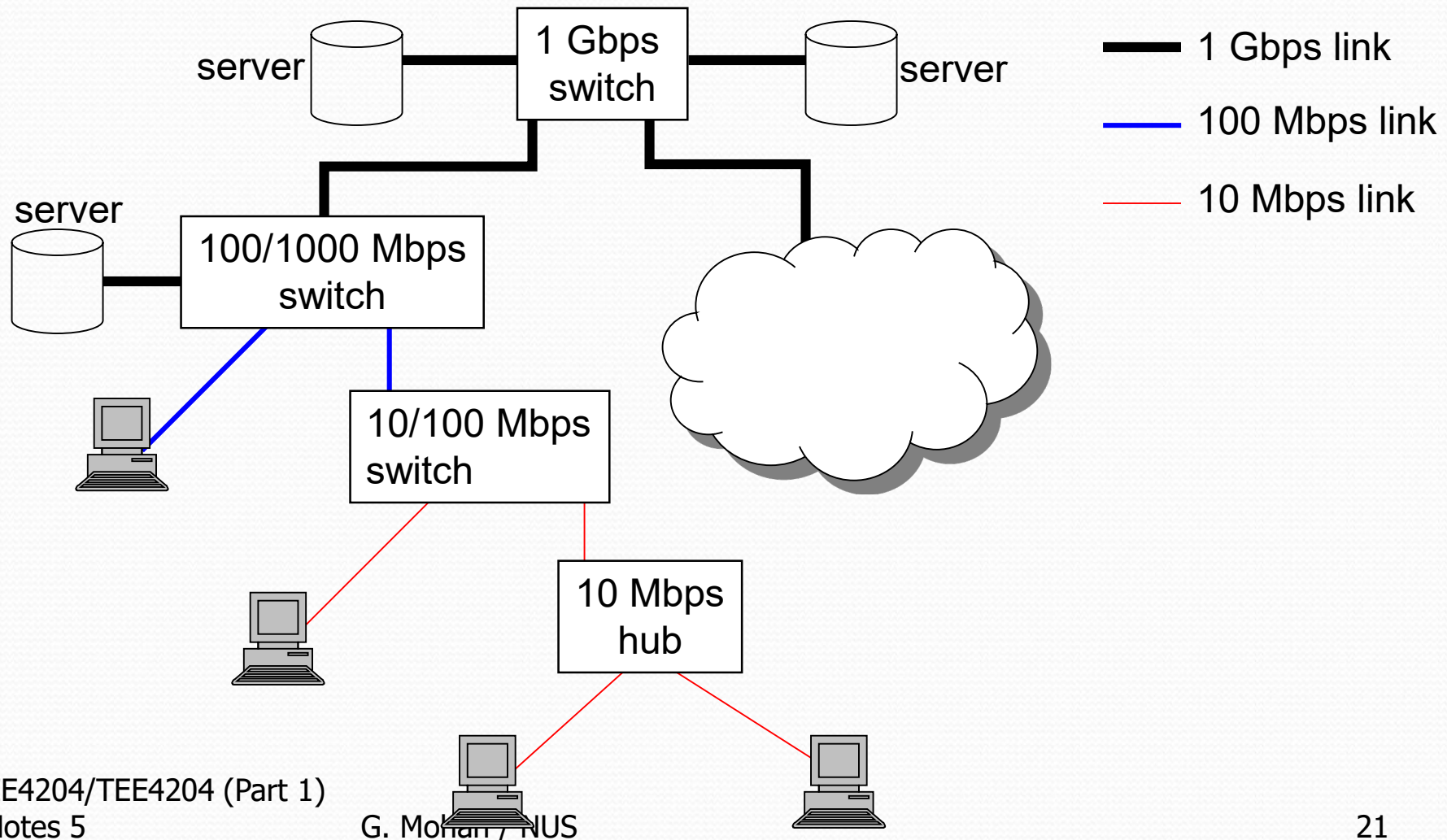
Switch-based LAN

- Switching Hub (or simply a switch)
 - Star configuration
 - E.g. 802.3u 100 Mbps Fast Ethernet (e.g. 100 BASE T)
 - E.g. 802.3z Gigabit Ethernet (e.g. 1000 BASE SX, 1000 BASE LX)
 - SX: short wavelength 0.85 micron, multimode fiber
 - LX: long wavelength 1.3 micron, single mode fiber
 - Store and Forward Packet Switch, use buffer to keep the excess frames
 - No collision between ports, Port is the collision domain, when only one station is connected to a port, there is no collision
 - More than one pair can communicate simultaneously.
 - Switch with N 10-Mbps ports: Total no of bits transmitted by all hosts in one second is at most $N \times 10$ million
 - Cable cut does not disconnect the network
 - Without any change in hardware or software of the attached stations, a bus LAN can be converted into a hub LAN or to a switch LAN.

Switch LAN



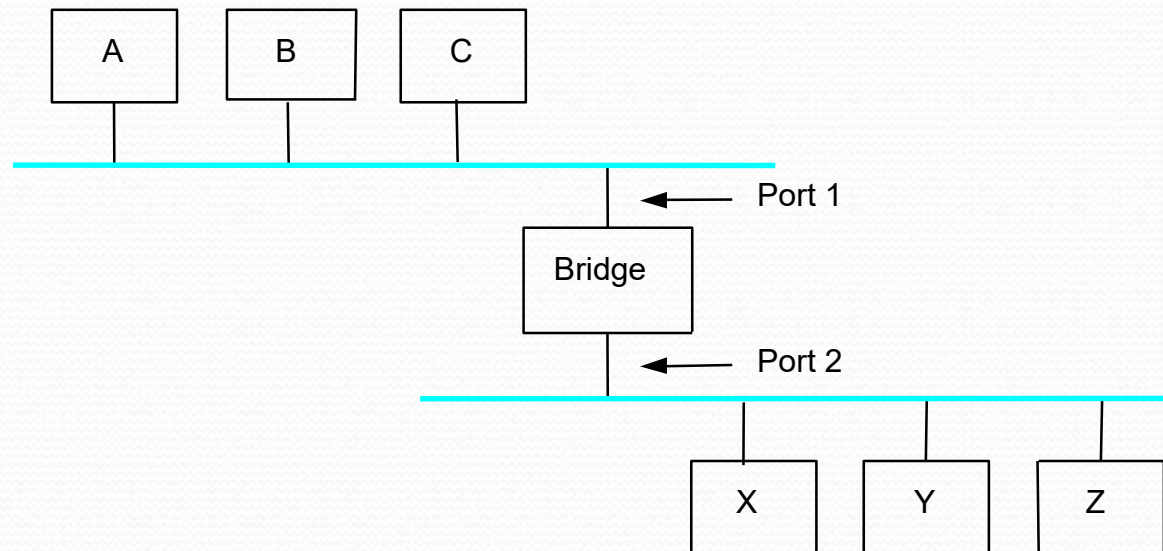
Ethernet Configuration in a Campus Network



Bridges and Extended LANs

- LANs have physical limitations (e.g., 2500m)
- Extended LAN: Interconnection of two or more LANs by one or more bridges
- Note: LAN bridges and switches are similar. In this lecture notes, they can be used interchangeably.
 - Switches (e.g. Ethernet switches) operate at layer 2. Routers (e.g. IP routers) also perform switching function but operate at layer 3 with more intelligent routing techniques
- Source routing bridge
 - Source host attaches complete address to the destination to the frame header ; token ring 802.5 group
- Transparent bridge or Spanning Tree bridge
 - Hosts need (do) not have the knowledge of the presence of bridges; CSMA/CD 802.3 group; WE STUDY TRANSPARENT BRIDGES
- A *bridge (switch)*
 - Operates in promiscuous mode; Multi-input and multi-output switch
 - An Ethernet bridge connecting n number of 10 Mbps Ethernet segments can carry up to 10n Mbps traffic
 - Operates in the data link layer. Uses accept and forward strategy; does not add packet header

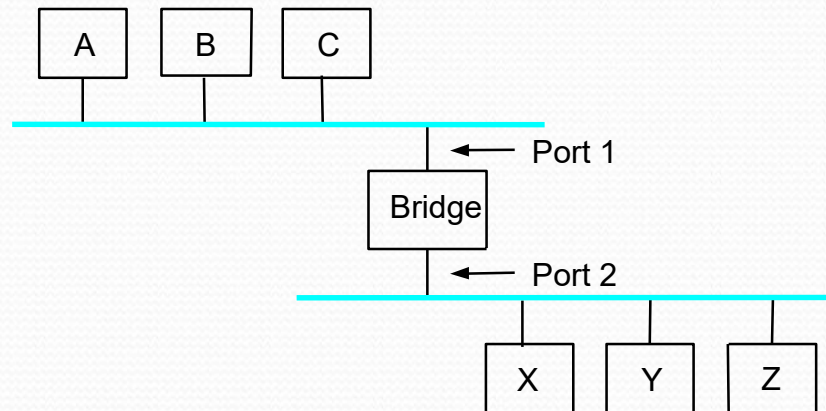
An extended LAN with a bridge



Learning Bridges

- Learn the ports through which a given host can be reached
- Maintain forwarding table

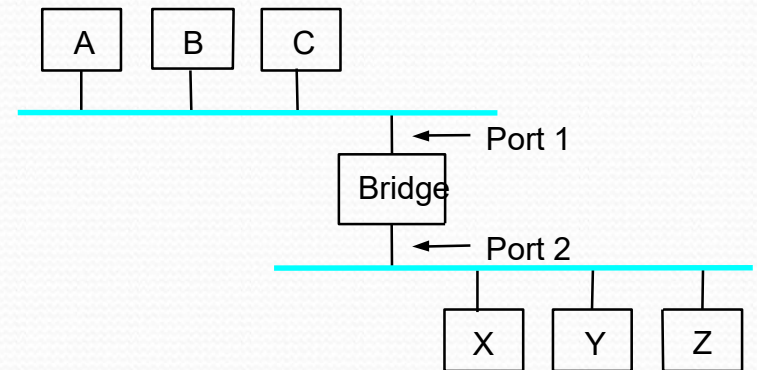
Host	Port
A	1
B	1
C	1
X	2
Y	2
Z	2



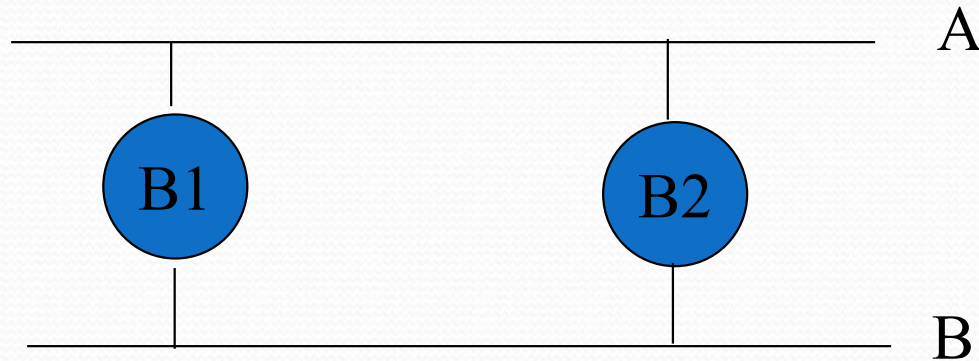
- Learn table entries based on source address
- Table need not be complete; can dynamically change (Why?)
- Always forward broadcast frames

Backward Learning Method

- Initially the forwarding table is empty
- When the bridge sees a frame $\langle A, B \rangle$ with source A and destination B, it learns where A is; i.e. through which port/interface A can be reached. Since the location of B is not known, the frame is forwarded through all the **other** ports; here port 2; as the frame was received from port 1. An entry for A is made in the table.
- When $\langle Y, A \rangle$ is received, it is forwarded to port 1 as the bridge has already learnt A's location. Now the bridge learns Y's location and makes an entry in the forwarding table.
- When $\langle B, Z \rangle$ is received, the port associated with B is learnt. Frame is forwarded to all the **other** ports; in this case, through port 2
- When $\langle C, B \rangle$ is received, the port associated with C is learnt the bridge does not forward it to port 2 as it already knows that B is on port 1.



Loops - Problem

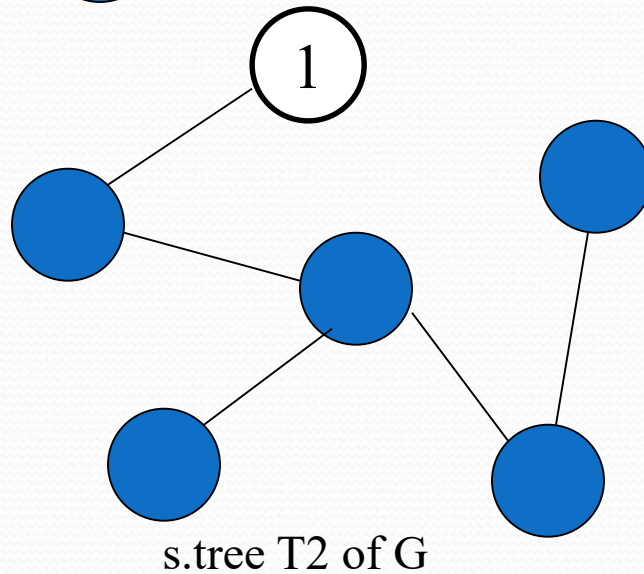
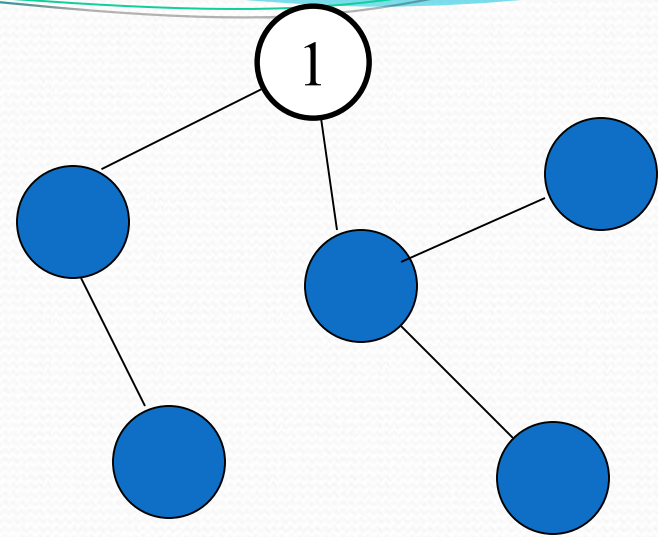
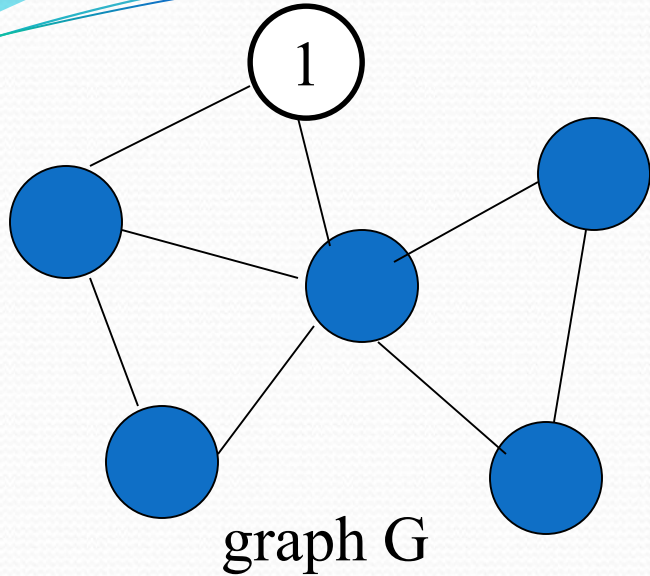


- Loops can exist to increase reliability but it may result in a situation where frames loop forever
- When frame F with unknown destination arrives at LAN B, B1 forwards it to LAN A generating frame F₁, B2 forwards to LAN B generating frame F₂. B1 on seeing F₂ will forward it to LAN A generating F₃. Similarly B2 on seeing F₁ will forward it to LAN B generating F₄. This continues forever.

Spanning tree Bridges

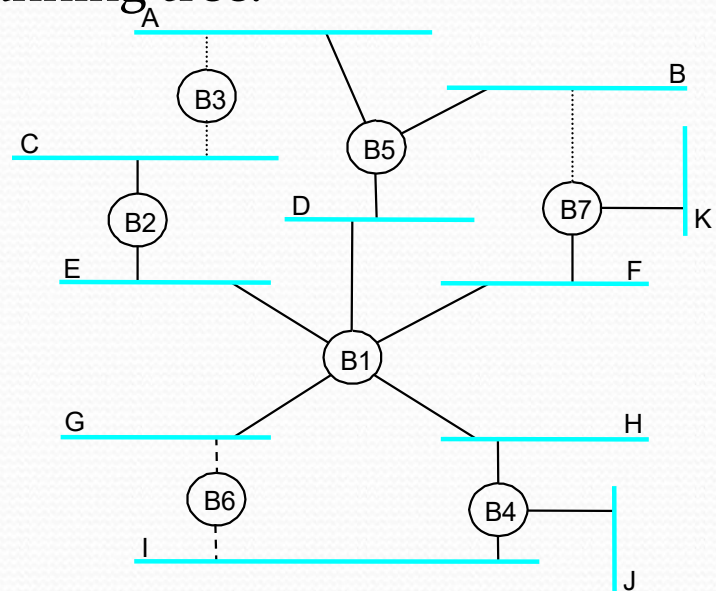
- To avoid loops, generate a spanning tree topology over the actual topology
 - Graph: A set of nodes and edges
 - Spanning tree of a graph: subgraph with all nodes and a subset of edges; no loops; unique path from the root to any node; unique path between any two nodes (eg: trees T1 and T2 in the next slide, root: node 1)
 - Shortest path spanning tree: formed by shortest paths from the root to every other node (eg.: tree T1 in the next slide; root: node 1)
- The spanning tree spans all the LANs; but some bridges (or ports) may be (logically) removed to avoid loops
- There is a unique path between any two LANs
- Using a distributed spanning tree algorithm all bridges agree on the spanning tree
 - select which bridges on which ports actively forward
 - developed by Radia Perlman
 - now IEEE 802.1 D specification

SPANNING TREE : EXAMPLE



Spanning Tree Algorithm Overview

- Each bridge has unique id (e.g., B1, B2, B3) (*See figure - Ref book by Peterson and Davie*)
 - Select bridge with smallest id as root
 - Create a tree of shortest paths from every bridge to the root
 - Select bridge on each LAN closest to root as designated bridge (use id to break ties)
 - Forward frames following the spanning tree.
-
- Each bridge forwards frames over each LAN for which it is the designated bridge



Algorithm Details

- Bridges exchange configuration messages: (Y, d, X)
 - Id (X) for bridge sending the message
 - id (Y) for what bridge X believes to be root bridge
 - distance (hops) (d) from sending bridge to root bridge
- Each bridge records current best configuration message for each port
- Initially, each bridge believes it is the root
 - Send $(X, 0, X)$

Algorithm Detail (contd.)

- When learn not root, stop generating config messages
 - in steady state, only root generates configuration messages
- When learn not designated bridge, stop forwarding config messages
 - in steady state, only designated bridges forward config messages
- Root continues to periodically send config messages
- If any bridge does not receive config message after a period of time, it starts generating config messages claiming to be the root

Spanning Tree Algorithm: An illustration

- B₃ receives (B₂, 0, B₂) on LAN C
 - B₃: accepts B₂ as root since $2 < 3$; sends (B₂, 1, B₃) to B₅ on LAN A
- B₂ receives (B₁, 0, B₁) on LAN E
 - B₂ accepts B₁ as root; sends (B₁, 1, B₂) to B₃ on LAN C
- B₅ receives (B₁, 0, B₁) on LAN D
 - B₅ accepts B₁ as root; sends (B₁, 1, B₅) to B₃ on LAN A
- B₃ receives (B₁, 1, B₂) from B₂ on LAN C
 - B₃ accepts B₁ as root;
 - stops forwarding to LAN C as B₂ is closer to B₁ than itself
- B₃ receives (B₁, 1, B₅) from B₅ on LAN A.
 - B₃ accepts B₁ as root; stops forwarding to LAN A as B₅ is closer to B₁ than itself

