CSE 3231 / CSE 5231
Computer Networks

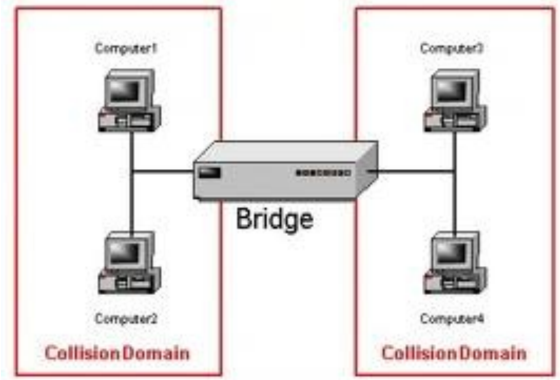
Chapter 4
Medium Access Sub-Layer
part 3

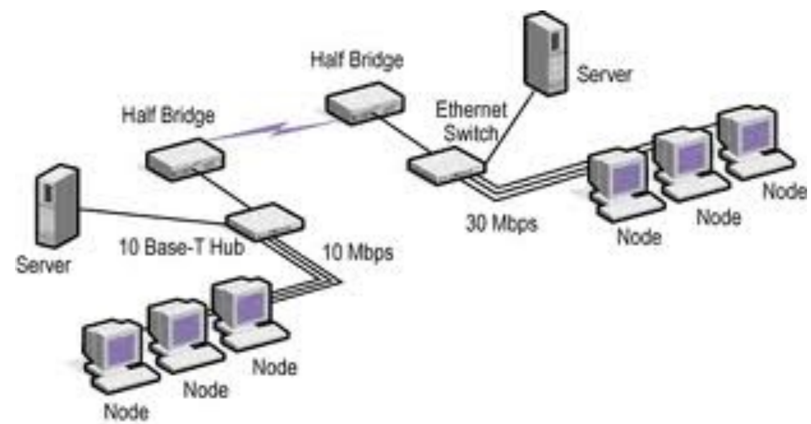
William Allen, PhD

Fall 2021

Bridges

- Bridges provide a separation between collision domains
- Frames transmitted on one side of the bridge will only cross the bridge if their destination is a node in the remote domain
- There are algorithms that allow bridges to "*learn*" when & how to forward frames across domains

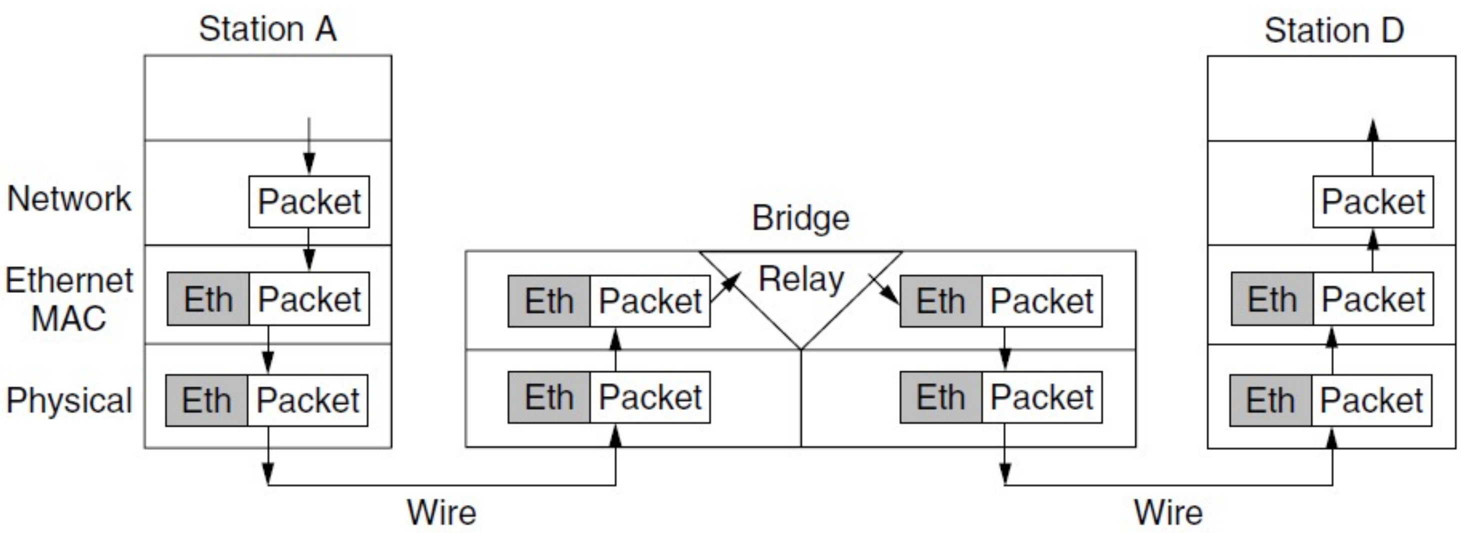




Bridges

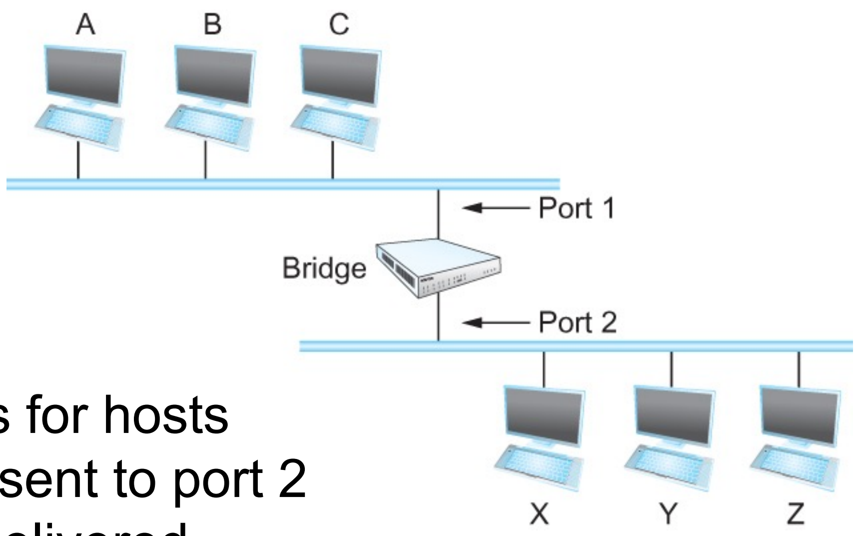
Bridges only use the Ethernet addresses

- they do not look at the Network (IP) address
- they do not change the frame's Ethernet header or addresses



Bridges and LAN Switches

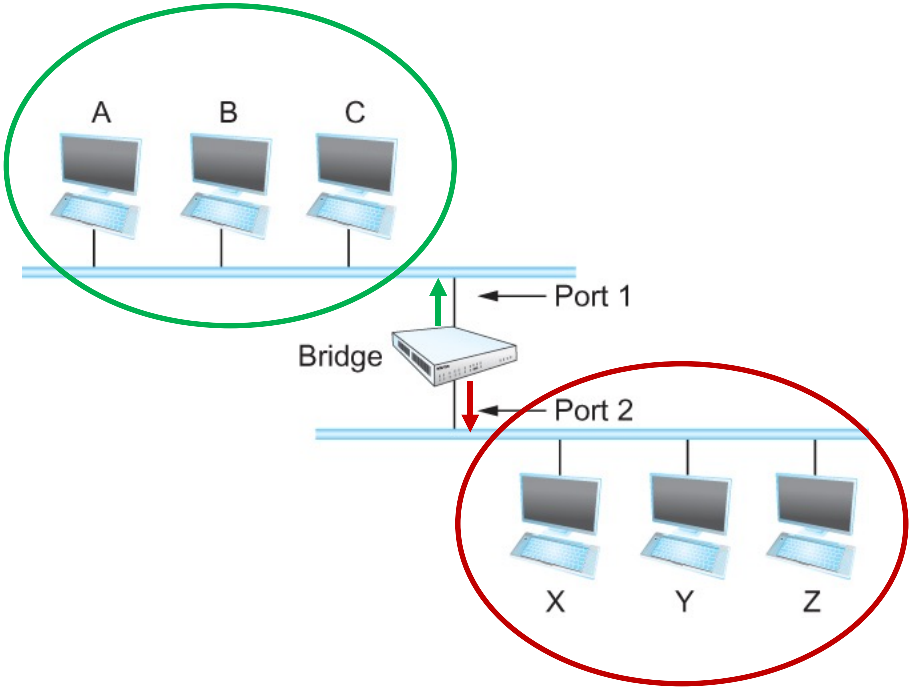
- Consider the following figure
  - When a frame from host A that is addressed to host B or host C arrives on port 1, the bridge will not send the frame out over port 2



However, frames for hosts X, Y or Z will be sent to port 2 so they can be delivered

  - How does a bridge *learn* on which port the various hosts reside?

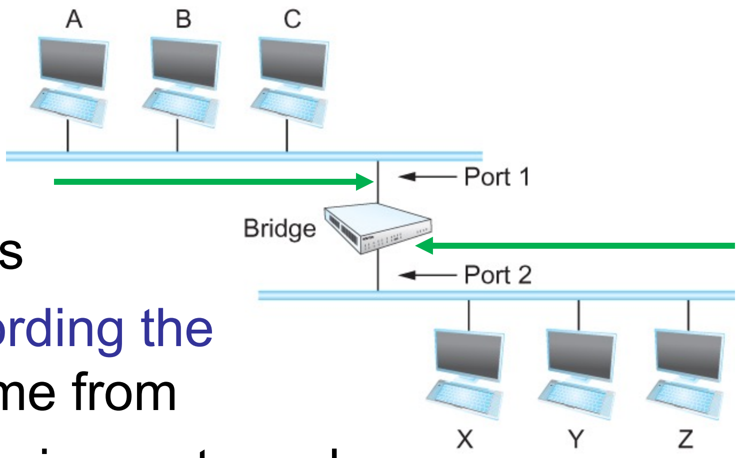
Bridges and LAN Switches



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

- A table shows which side of the bridge hosts are located on.
- Can this be managed by a human administrator?
  - yes, but it is impractical if there are rapid changes in the topology of LANs and/or if there are many devices on the LANs

Bridges and LAN Switches



Learning the Forwarding Tables

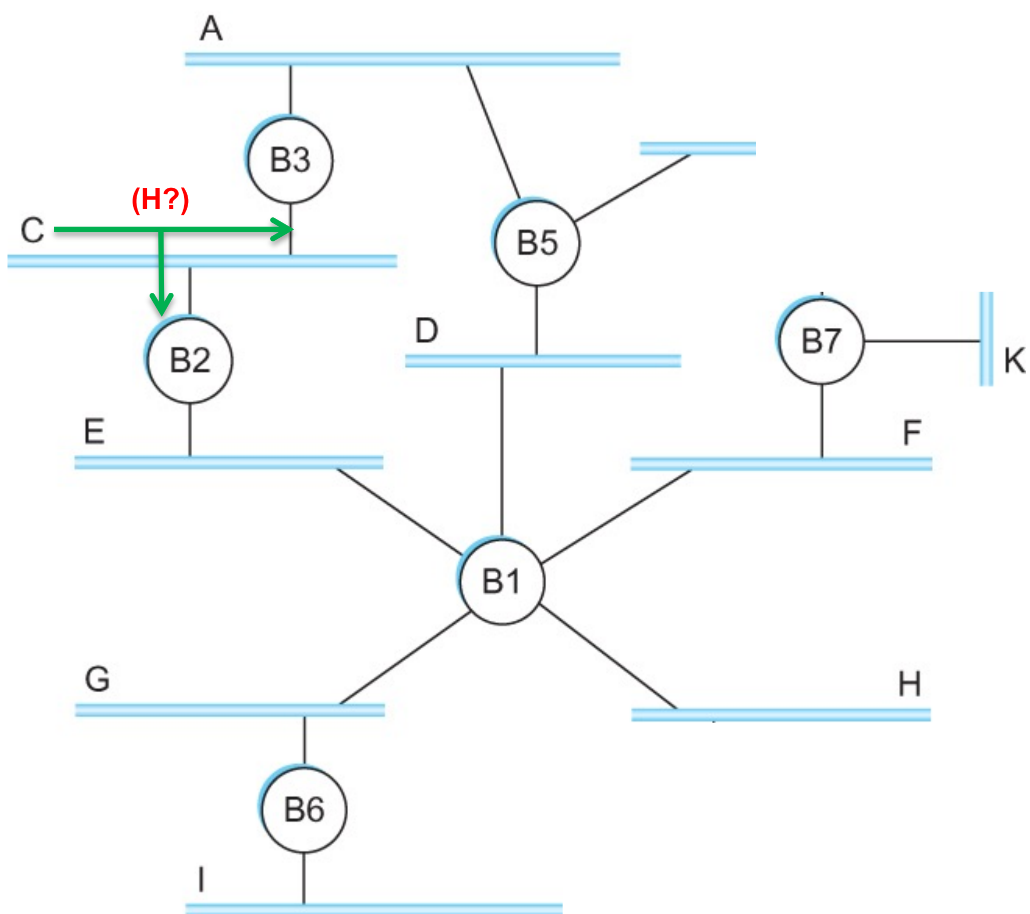
- Each bridge inspects the *source address* in all the frames it receives
- The bridge builds the table by recording the addresses and which port they came from
- When a bridge first boots, this table is empty and entries are added over time
- A timeout is associated with each entry and the bridge discards them when the timeout is reached
  - This handles the situation in which a host is moved from one network to another

If the bridge receives a frame that is addressed to a host *not currently* in the table

- It forwards the frame out on **all** of the other ports

Example

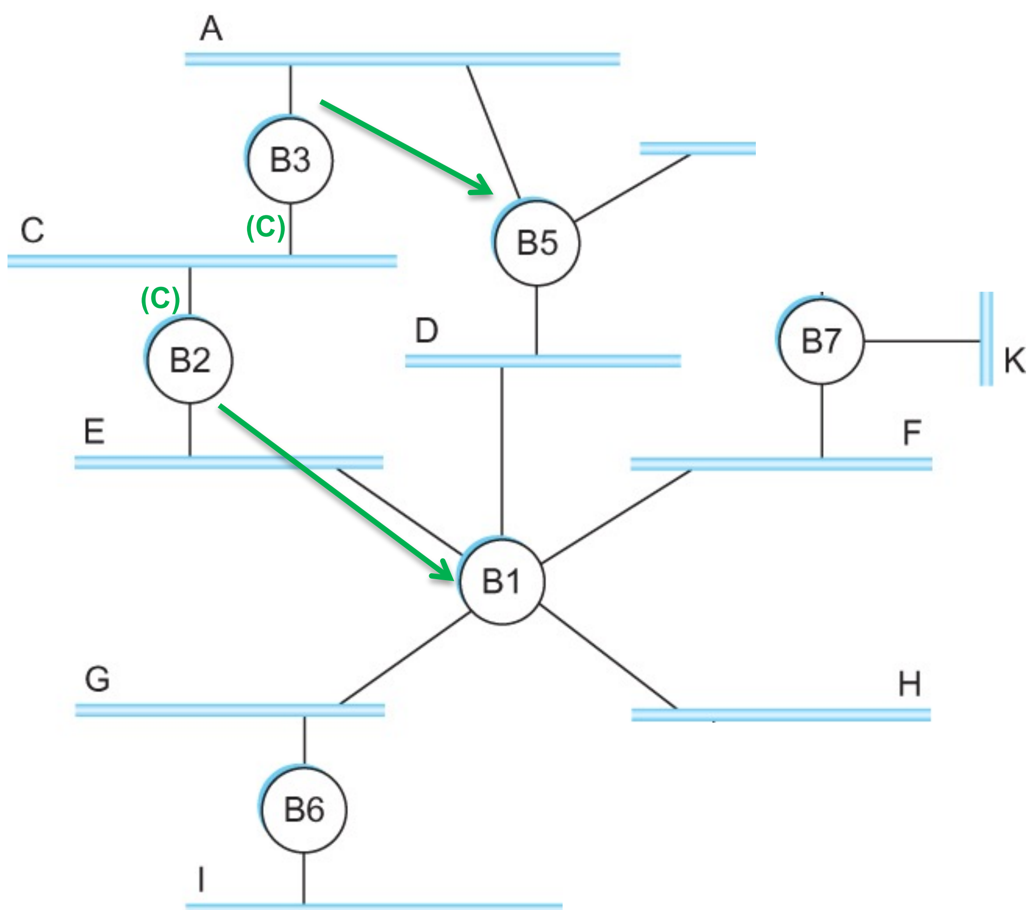
C sends to H



C sends the frame out on the LAN Both B2 and B3 see the frame

Example

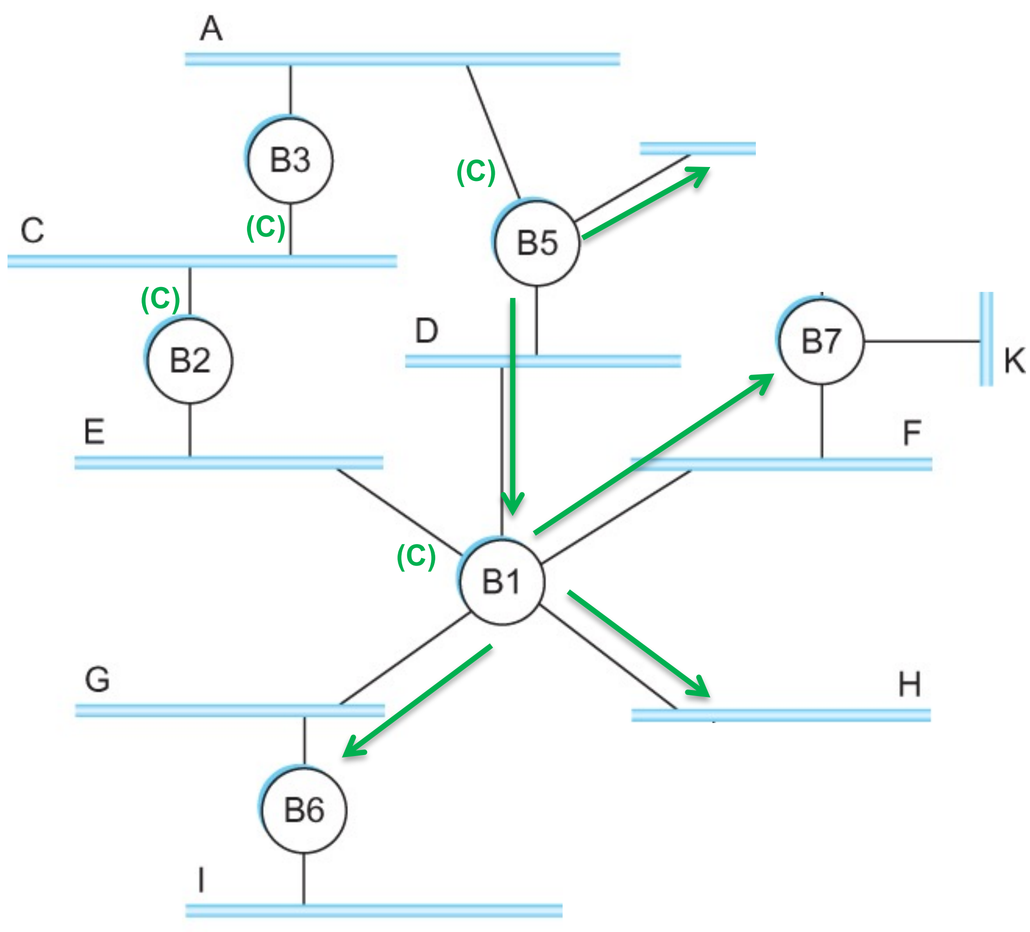
C sends to H



B2 and B3 send the frame out their other port and the next set of bridges receive the frame

Example

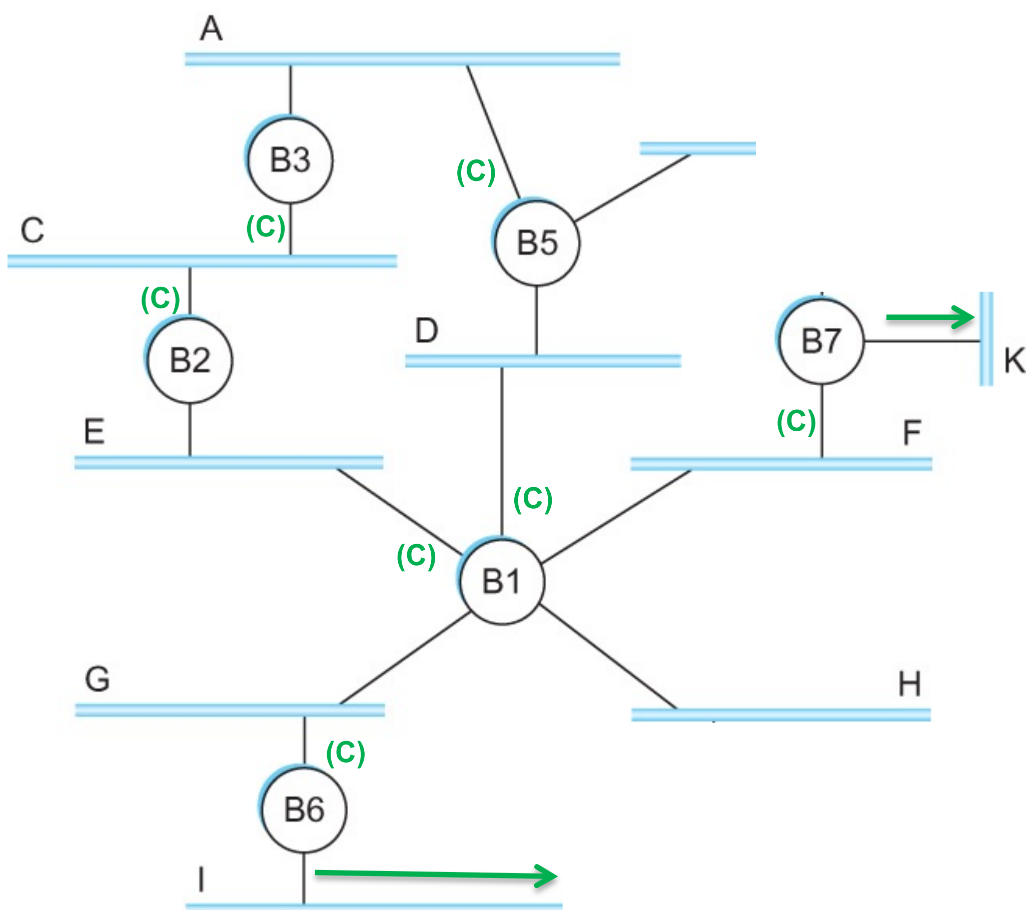
C sends to H



B1 and B5
send the frame
out their other port
and the next set
of bridges also
receive the frame

Example

C sends to H

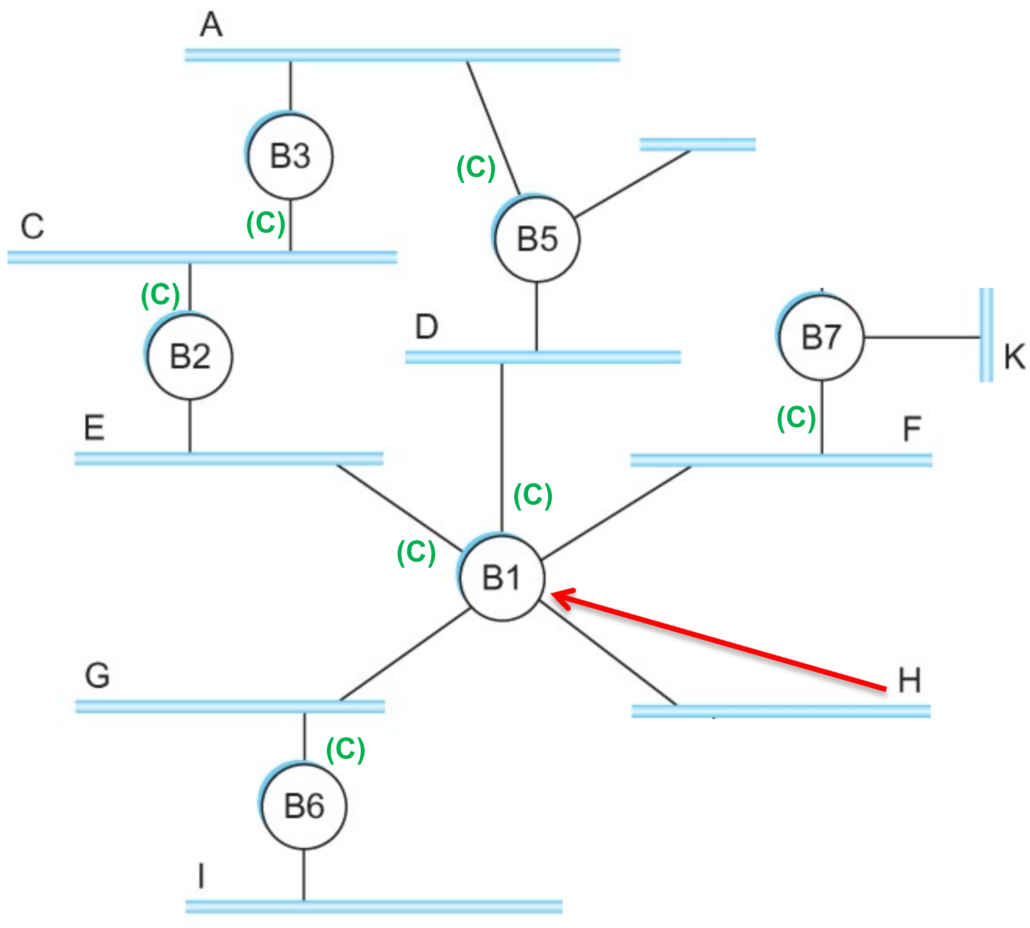


Now, all of the bridges “know” how to get a frame to C

B6 and B7 send the frame out their other port to LANs I and K

Example

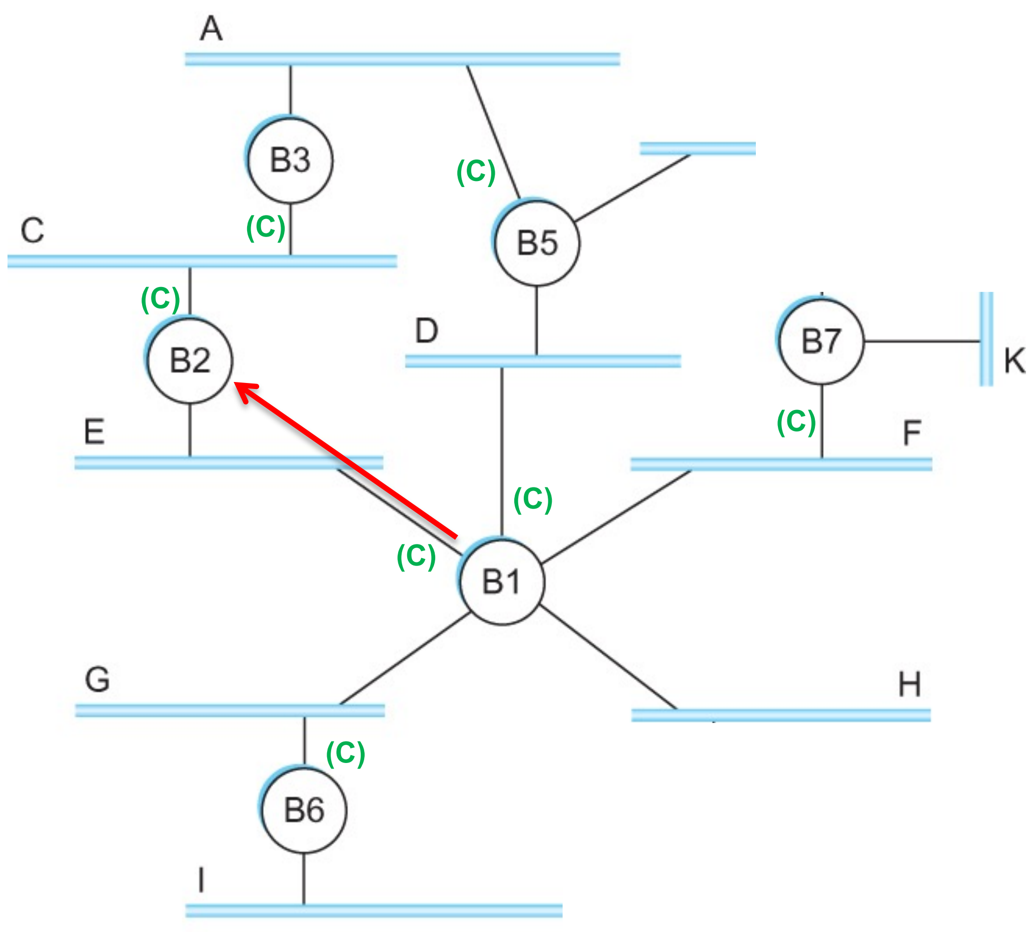
H sends to C



B1 is the local bridge for H

Example

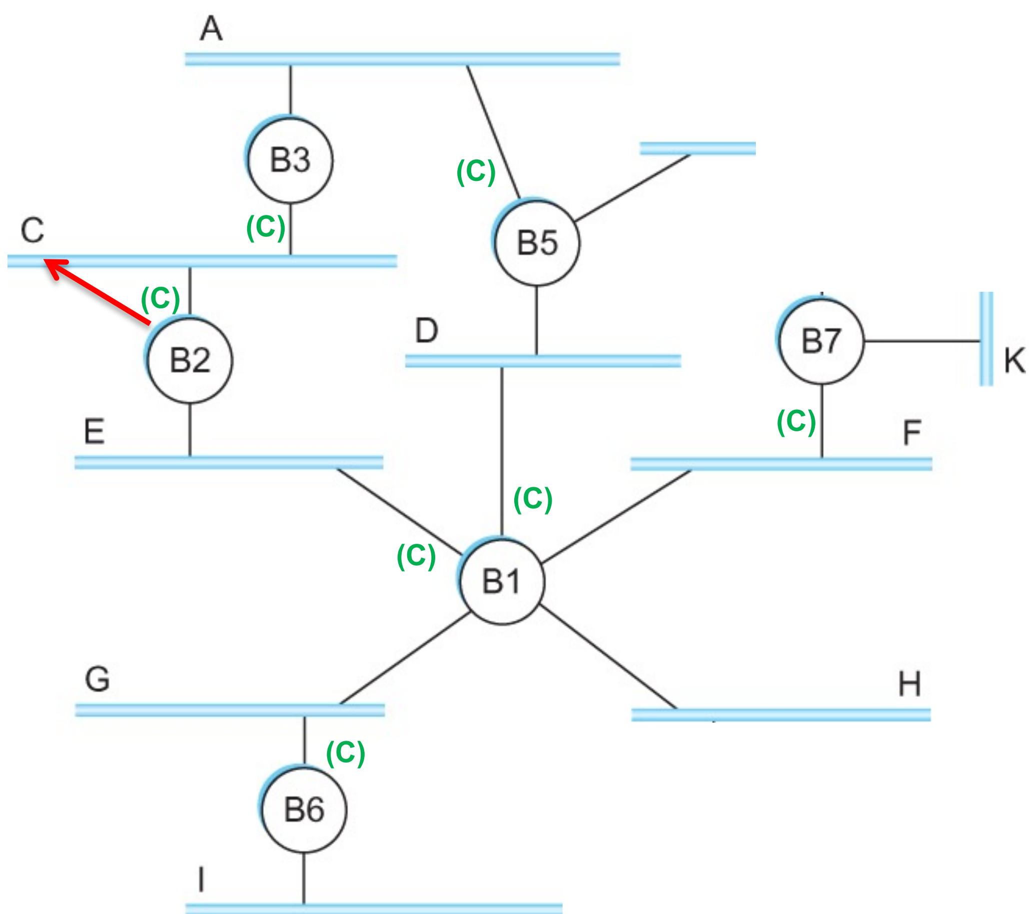
H sends to C



B1 “knows” that both B2 and B5 can get to C, but B2 has the lower ID number.

Example

H sends to C

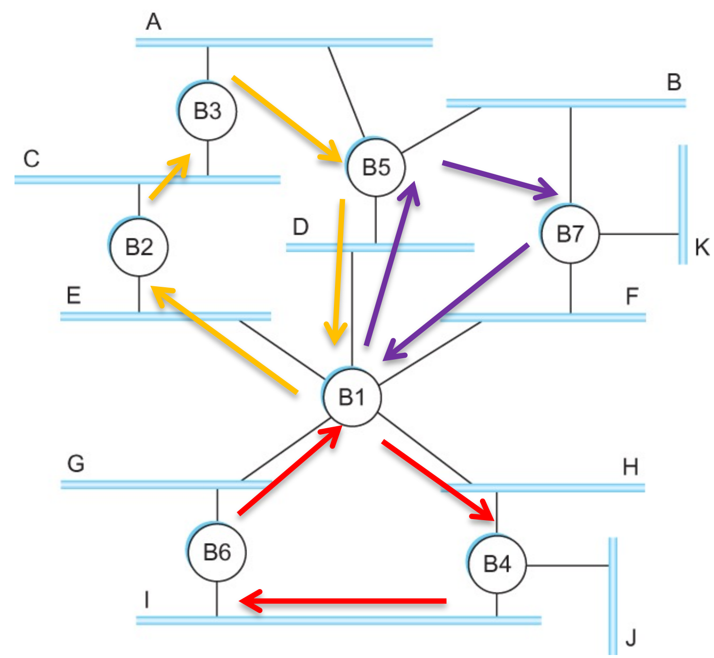


B2 sends the message to C

Bridges and LAN Switches

Learning the Forwarding Tables

- This strategy works fine if the extended LAN does not have a loop in it
- Why?
  - Frames could potentially loop through the extended LAN forever
  - There are several loops in this example:
    - bridges B1, B4, and B6
    - bridges B1, B2, B3, B5
    - bridges B1, B5, B7

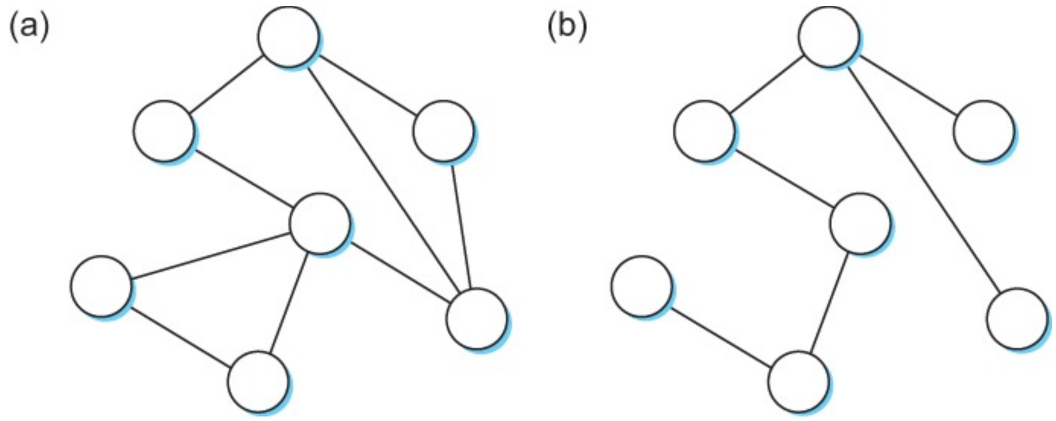


Bridges and LAN Switches

- There are several ways to deal with loops:
  - timeouts: if the frame hasn’t been delivered by a certain time, it is dropped
  - hop counts: after passing through a certain number of bridges/switched, it is dropped
  - only following a path that doesn’t have loops
    - a spanning tree is a path that passes by all of the nodes once, but does not include a loop

Spanning Tree Algorithm

- Think of the extended LAN as being represented by a graph that possibly has loops (cycles)
- A *spanning tree* is a sub-graph of this graph that covers all the vertices but contains no cycles (i.e., not *cyclic*)
  - Spanning tree keeps all the vertices of the original graph but throws out some of the edges



Example of (a) a cyclic graph; (b) a corresponding spanning tree.

Spanning Tree Algorithm

- Idea developed by Radia Perlman at Digital
  - This *protocol* can be used by a set of bridges to find a spanning tree for a particular extended LAN
  - IEEE 802.1 specification for LAN bridges is based on this algorithm
  - Each bridge will decide the ports over which it is and is not willing to forward frames
    - In a sense, it is *removing ports* from the network topology so that the extended LAN is reduced to an acyclic tree
    - It is even possible that an entire bridge will not participate in forwarding frames for a specific destination

Spanning Tree Algorithm

- Algorithm is dynamic
  - The bridges are always prepared to reconfigure themselves into a new spanning tree if some bridges fail or if new bridges or links are added
- Main idea
  - Each bridge selects the ports over which they will forward the frames for a given destination
  - The term "port" refers to a network interface on the bridge, there can be two or more per bridge

Spanning Tree Algorithm

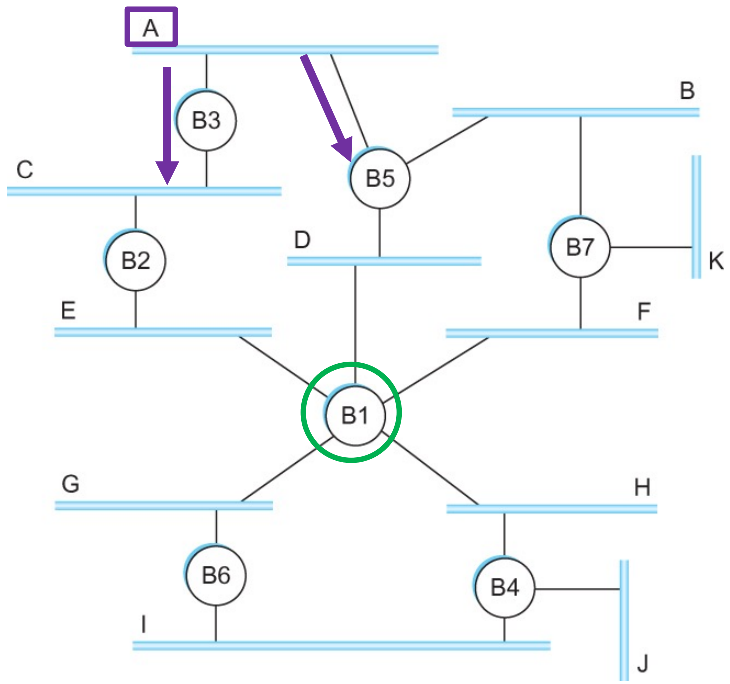
- Algorithm selects ports as follows:
  - Each bridge has a unique identifier
    - for our example, we use B1, B2, B3,…and so on.
  - The bridge with the *smallest ID* in the LAN is the **root** of the spanning tree
  - To start, the root bridge forwards frames out over all of its ports so its neighbors “know” where it is
  - Each bridge computes the *shortest path* to the root and notes which of its ports is on this path
    - This port is selected as the bridge’s preferred path to the root
  - All the bridges connected to a given LAN select a single *designated bridge* that will forward frames toward the root bridge

Spanning Tree Algorithm

- Each LAN's designated bridge is the one that is closest to the root
  - 'distance' is the number of bridges to the root
  - If two or more bridges are equally close to the root, then the bridge with the smallest id is selected
- Each bridge is connected to more than one LAN
  - It participates in the selection of a designated bridge for each LAN it is connected to
  - Each bridge decides if it is the designated bridge relative to each of its ports and forwards frames over those ports for which it is the designated bridge

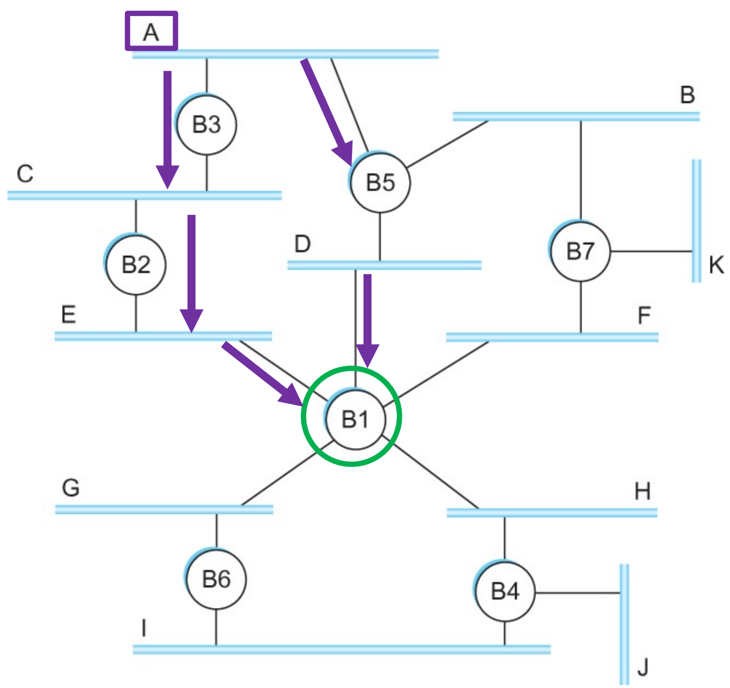
Spanning Tree Algorithm

- We need to build a tree for LAN A where B1 is the root bridge
- LAN A connects to B3 and B5, which is the designated bridge?



Spanning Tree Algorithm

- We need to build a tree for LAN A where B1 is the root bridge
- LAN A connects to B3 and B5, which is the designated bridge?

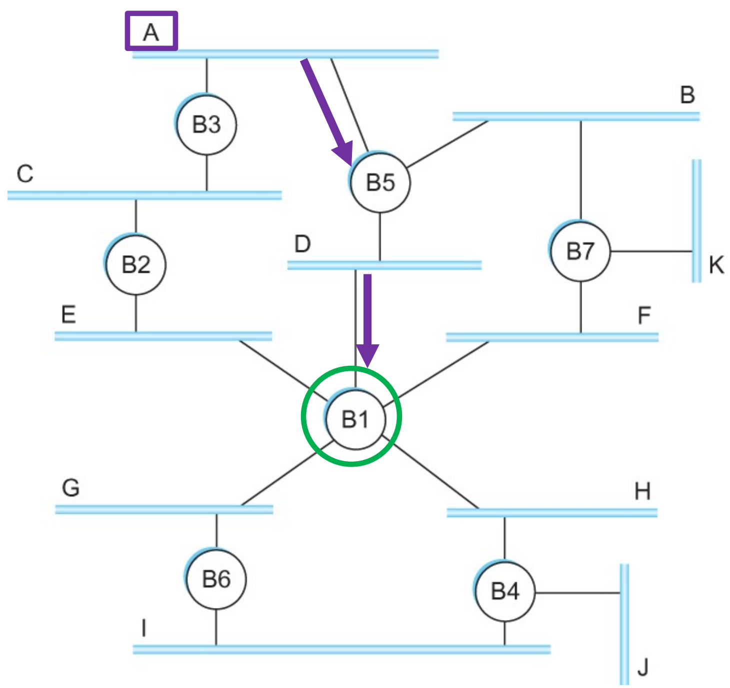


Which path is shorter:
(i.e., passes through fewer bridges to get from A to B1)?

- **A to B5 to D to B1?**
- **A to B3 to C to B2 to E to B1?**

Spanning Tree Algorithm

- We need to build a tree for LAN A where B1 is the root bridge
- LAN A connects to B3 and B5, which is the designated bridge?

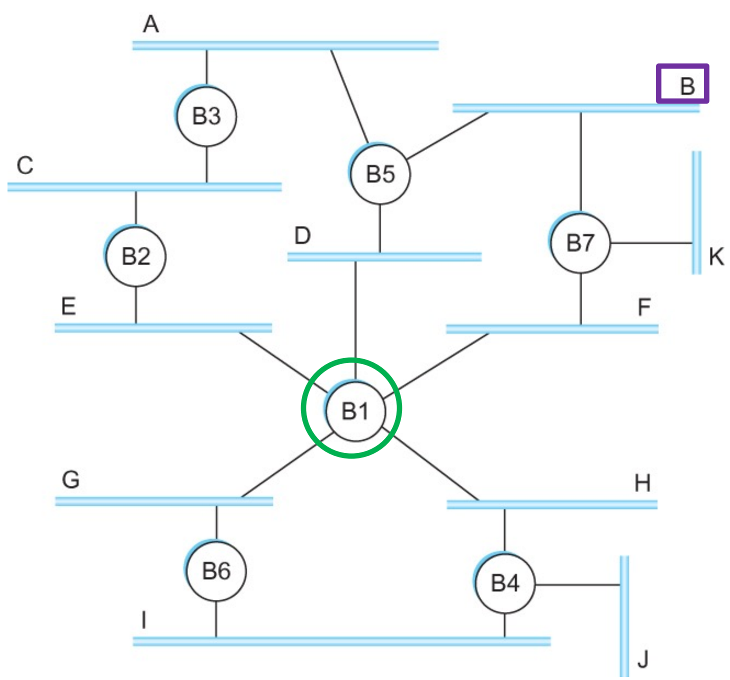


The path though B5 is shorter -- therefore, B5 is the designated bridge for LAN A.

- **A to B5 to D to B1?**
- ~~**A to B3 to C to B2 to E to B1?**~~

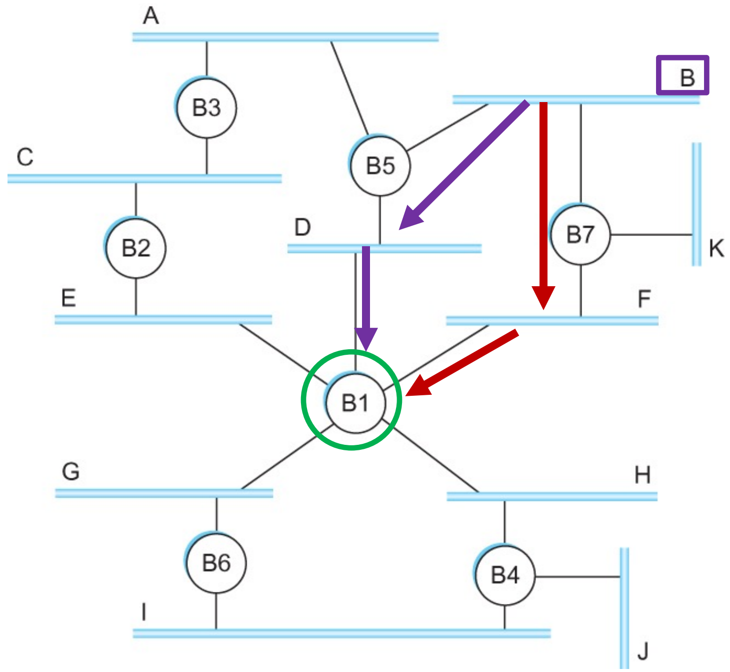
Spanning Tree Algorithm

- We need to build a tree for LAN B where B1 is the root bridge
- LAN B connects to B5 and B7, which is the designated bridge?



Spanning Tree Algorithm

- We need to build a tree for LAN B where B1 is the root bridge
- LAN B connects to B5 and B7, which is the designated bridge?

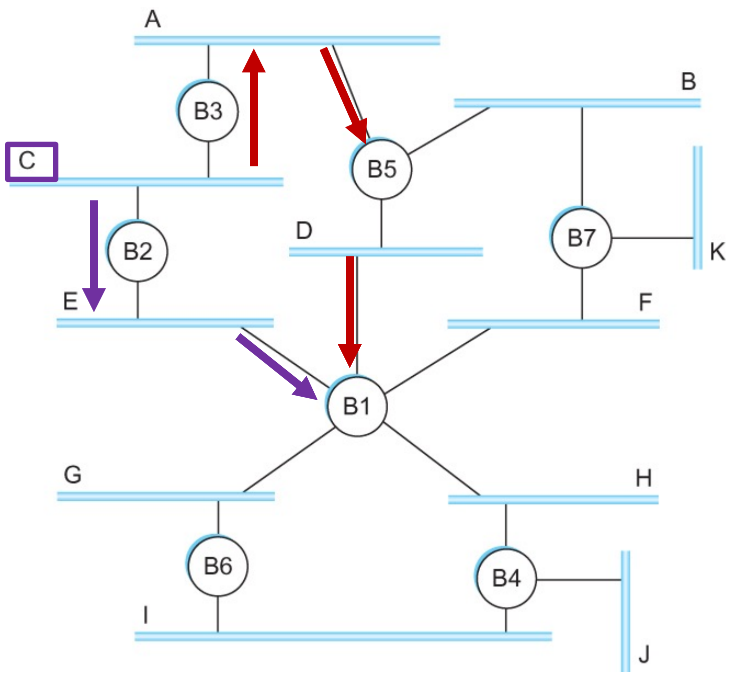


Both paths have equal length. Chose the path through the bridge with lower ID number.

B5 < B7

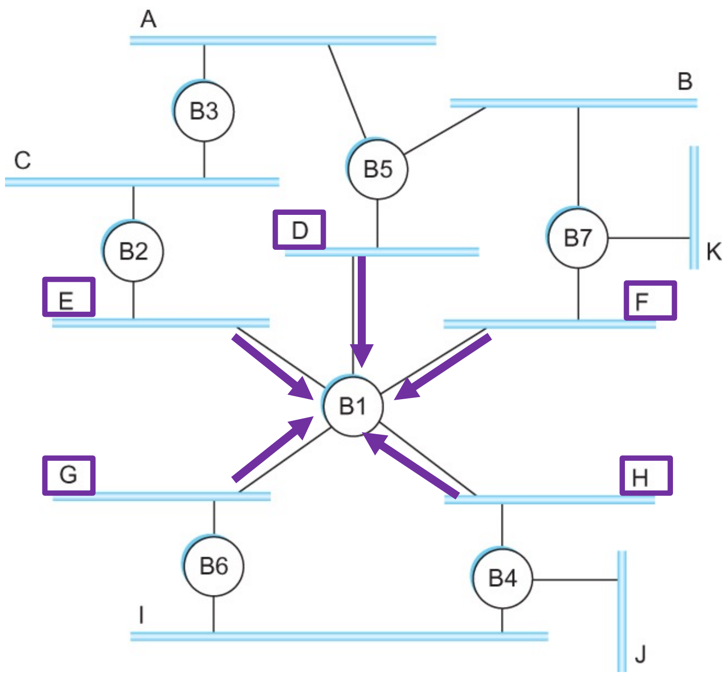
Spanning Tree Algorithm

- C to B2 to E to B1?
- ~~C to B3 to A to B5 to D to B1?~~



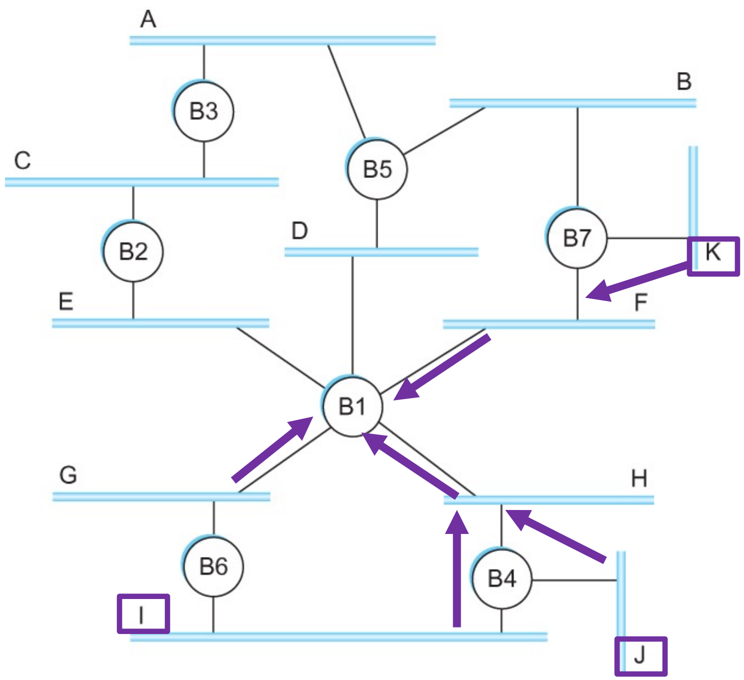
LAN A	**B5**
LAN B	**B5**
LAN C	**B2**
LAN D	
LAN E	
LAN F	
LAN G	
LAN H	
LAN I	
LAN J	
LAN K	

Spanning Tree Algorithm



LAN A	**B5**
LAN B	**B5**
LAN C	**B2**
LAN D	**B1**
LAN E	**B1**
LAN F	**B1**
LAN G	**B1**
LAN H	**B1**
LAN I	
LAN J	
LAN K	

Spanning Tree Algorithm

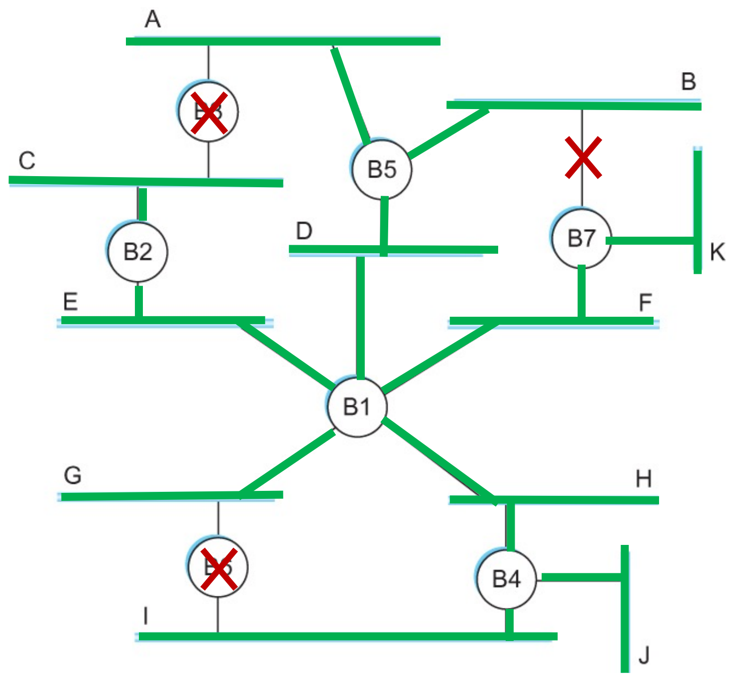


LAN I to B4 because B4 < B6

LAN A	B5
LAN B	B5
LAN C	B2
LAN D	B1
LAN E	B1
LAN F	B1
LAN G	B1
LAN H	B1
LAN I	**B4**
LAN J	**B4**
LAN K	**B7**

Spanning Tree Algorithm

All LANs are connected through bridges
but, **there are no cycles** (loops)

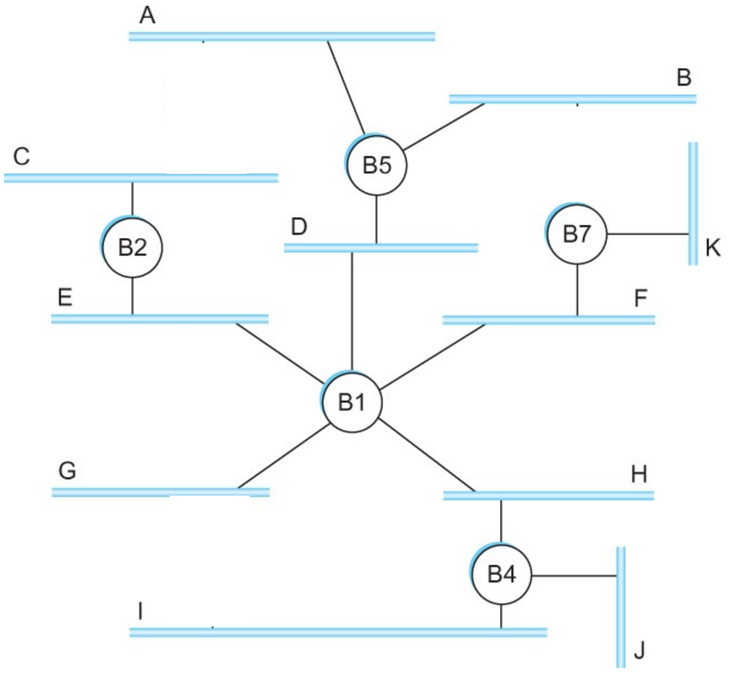


B3 and B6 are not designated bridges,
the link LAN B to B7 is also not used

LAN A	B5
LAN B	B5
LAN C	B2
LAN D	B1
LAN E	B1
LAN F	B1
LAN G	B1
LAN H	B1
LAN I	B4
LAN J	B4
LAN K	B7

Shortest Path

- Traffic from one LAN to another will only be forwarded through the sending LAN's own designated bridge
- Therefore, it cannot return a frame to the LAN that sent it
- This prevents bridges from forwarding frames in an infinite loop

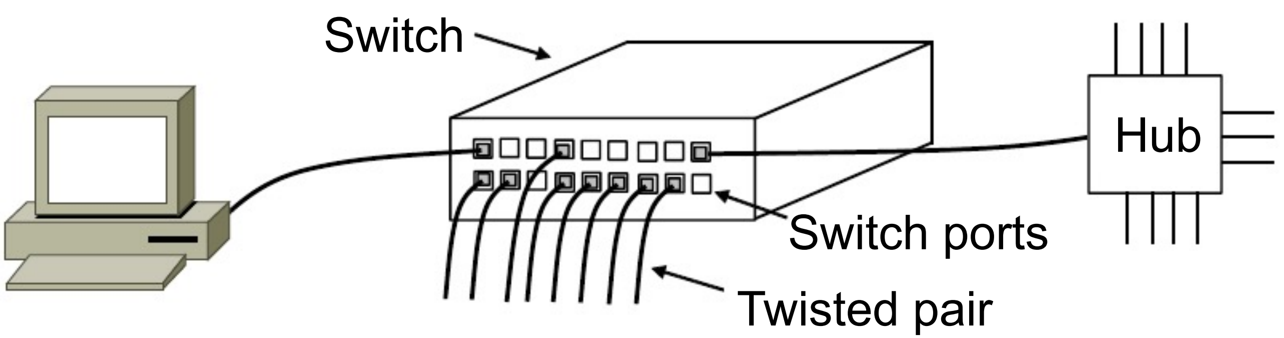


Switched Ethernet

Hubs were replaced with electronic switches

Switches enable the isolation of pairs of nodes by connecting them directly inside the switch

Frames are *no longer* passed to all nodes in the LAN, which reduces the occurrence of collisions and improves security and privacy



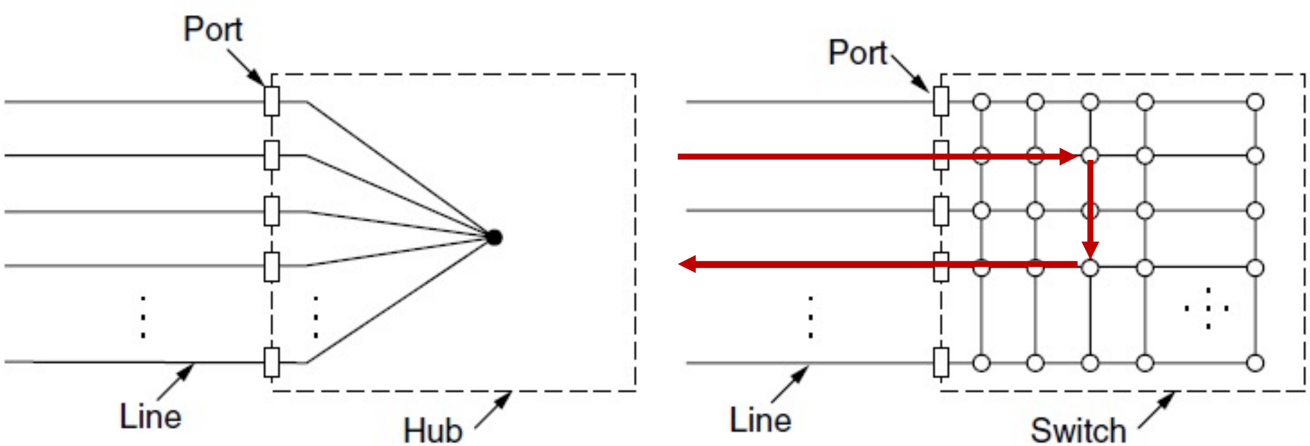
Switched Ethernet

Hubs internally connected all nodes together, allowing collisions when multiple nodes transmit

– CSMA-CD is needed to reduce collisions

Switches isolate each port to a separate domain

– The switch monitors traffic to manage connections
– Pairs of nodes can be connected directly

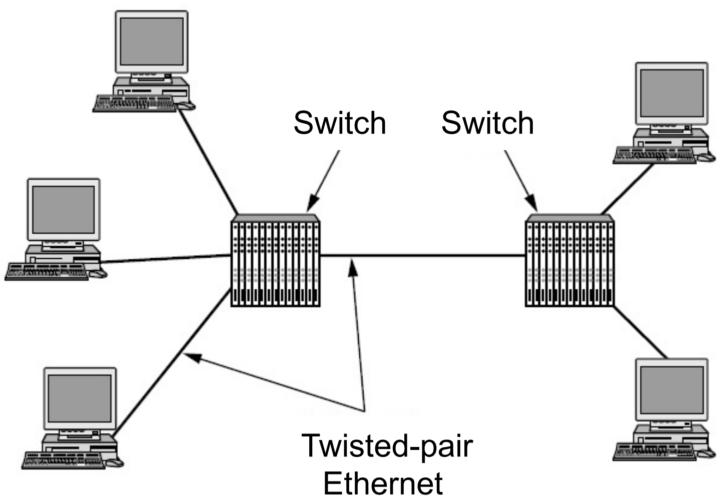


Switched Ethernet

Switched Ethernet can support Gigabit data rates and provides full-duplex connections between any two nodes in the same LAN

Switches removed or changed many of the restrictions on classic Ethernet

- collisions can still occur between the same nodes
- twisted-pair cable length limit is 100m
- higher speeds require special types of cable
- options for larger frame sizes (up to 9000 bytes)



Fast(er) Ethernet

Fast Ethernet extended Ethernet to 100 Mbps

Name	Cable	Max. segment	Advantages
100Base-T4	Twisted pair	100 m	Uses category 3 UTP
100Base-TX	Twisted pair	100 m	Full duplex at 100 Mbps (Cat 5 UTP)
100Base-FX	Fiber optics	2000 m	Full duplex at 100 Mbps; long runs

Gigabit Ethernet runs over fiber or twisted pair

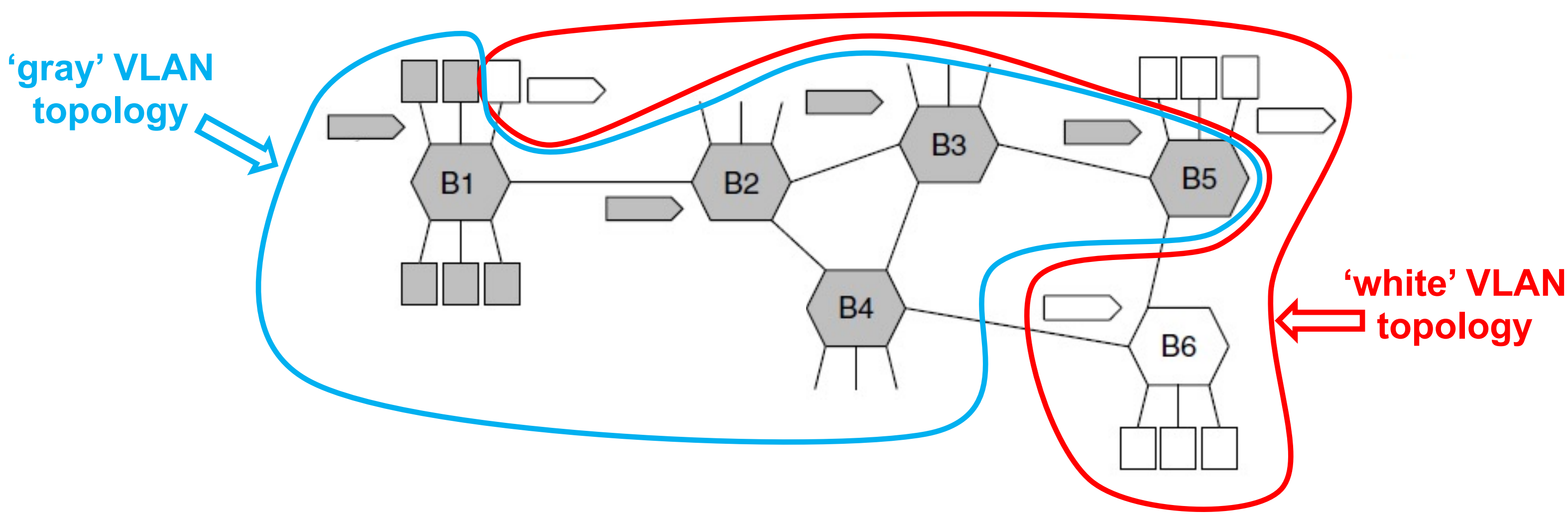
Name	Cable	Max. segment	Advantages
1000Base-SX	Fiber optics	550 m	Multimode fiber (50, 62.5 microns)
1000Base-LX	Fiber optics	5000 m	Single (10 µ) or multimode (50, 62.5 µ)
1000Base-CX	2 Pairs of STP	25 m	Shielded twisted pair
1000Base-T	4 Pairs of UTP	100 m	Standard category 5 UTP

10 Gigabit Ethernet is increasingly used in data centers and for connecting LANs

Virtual LANs

A VLAN (Virtual LAN) splits one physical LAN into multiple *logical* LANs to provide better isolation and to simplify management tasks

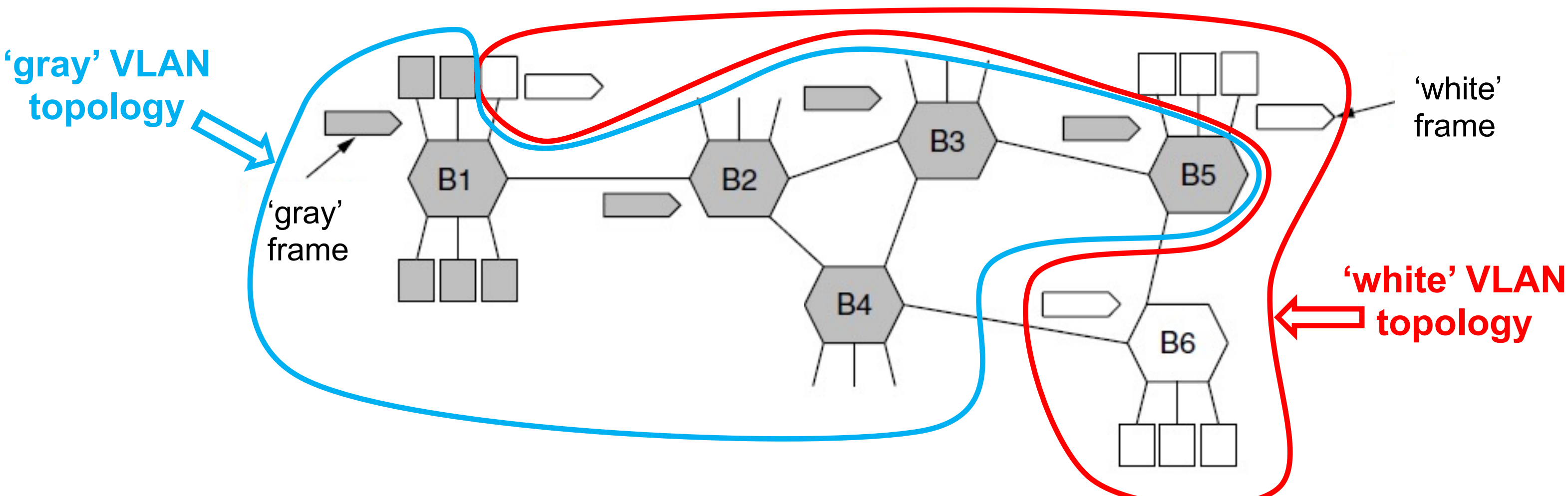
- Frames are assigned to a specific VLAN
- Bridges maintain different forwarding tables for each VLAN to isolate frames from other VLANs



Virtual LANs – IEEE 802.1Q

Bridges need to know which frames belong to each VLAN to send them out the correct port

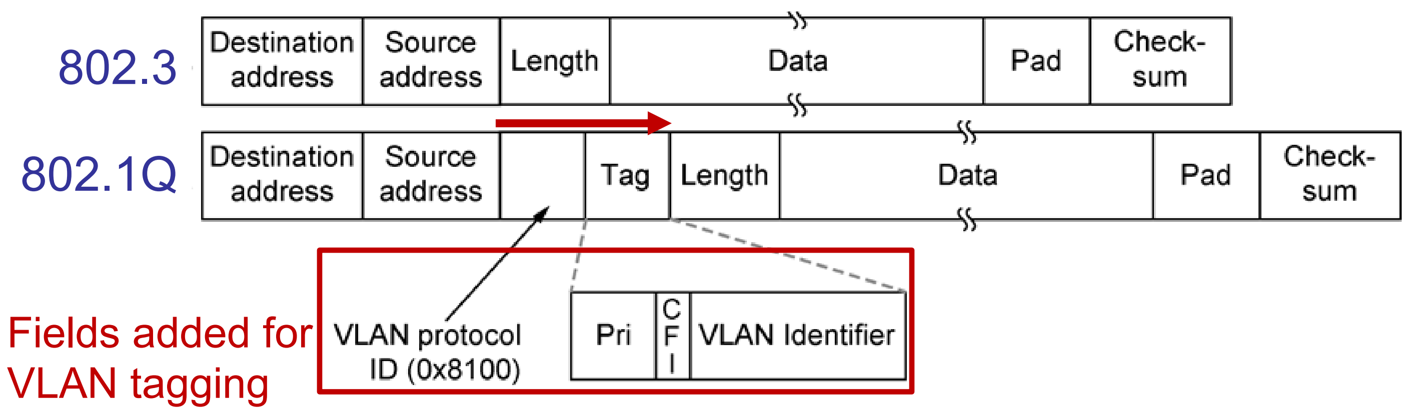
- IEEE 802.1Q describes how frames are *tagged* with their "color" and how they are processed in bridges
  - "color" tags can be added to un-tagged frames
  - however, bridge hardware and software must support 802.1Q



Virtual LANs (3) – IEEE 802.1Q

The VLAN protocol and Tag fields are added to 802.1Q frames so they can join a VLAN

- The value `0x8100` is > the maximum length for an Ethernet frame, so it indicates this is a VLAN frame
- The bridge then reads the Tag field to determine which VLAN “color” this frame belongs to and forwards it to nodes on the VLAN given that “color”



Ethernet Over Optical Fiber

- In networks where buildings are larger or farther apart, Ethernet LANs can use fiber optic cables to avoid interference and loss
  - Fiber optic cables can carry high-speed Ethernet with no loss of bandwidth
  - Also used to connect LANs to an ISP