# CPE348: Introduction to Computer Networks

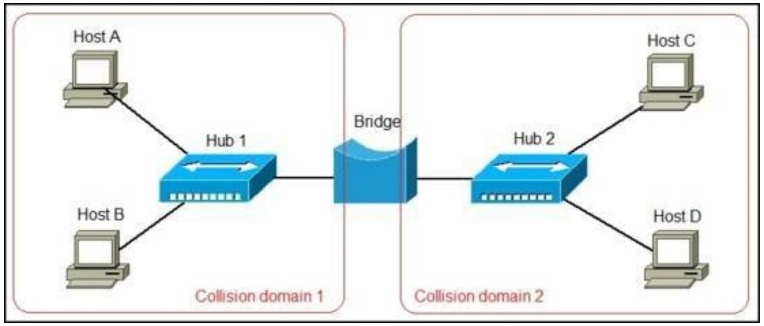
## Lecture #10: Chapter 3.3

Jianqing Liu
Assistant Professor of Electrical and Computer Engineering, University of Alabama in Huntsville

jianqing.liu@uah.edu
http://jianqingliu.net

# Bridges and LAN Switches

- Bridges and LAN Switches
  - Class of switches that is used to forward packets between shared-media LANs
  - Connect a pair of Ethernets
    - **One approach is put a repeater in between them**
      - It might exceed the physical limitation of the Ethernet
      - No more than a total of 2500 m in length is allowed
    - **Alternatively, put a node between the two Ethernets**
      - This node is called a **Bridge**
      - A collection of LANs connected by one or more bridges is usually said to form an **Extended LAN** (split domains)

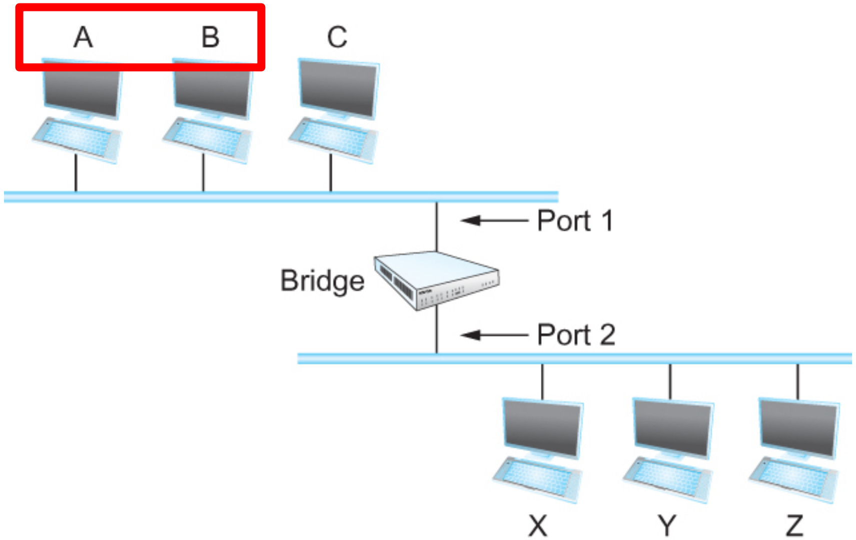


# Bridges and LAN Switches

- Simplest Strategy for Bridges
  - Accept LAN frames on their inputs and forward them out to **all** other outputs
  - Used by early bridges
- Learning Bridges
  - Observe that there is no need to forward all the frames that a bridge receives

# Bridges and LAN Switches

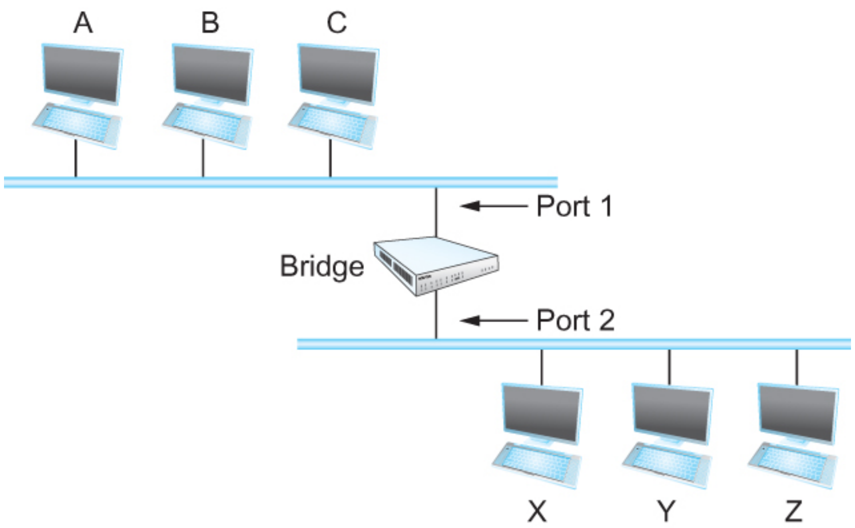
- Consider the following figure
  - When a frame from host A that is addressed to host B arrives on port 1, there is no need for the bridge to forward the frame out over port 2.




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

# Bridges and LAN Switches

- Solution
  - Download a table into the bridge

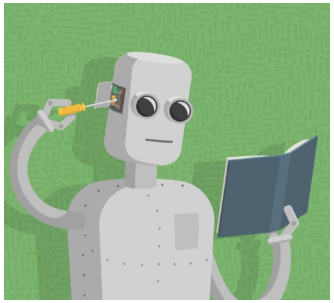



  - Who does the download?
    - Human - Too much work for maintenance

**Bridge Routing Table**

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

# Bridges and LAN Switches

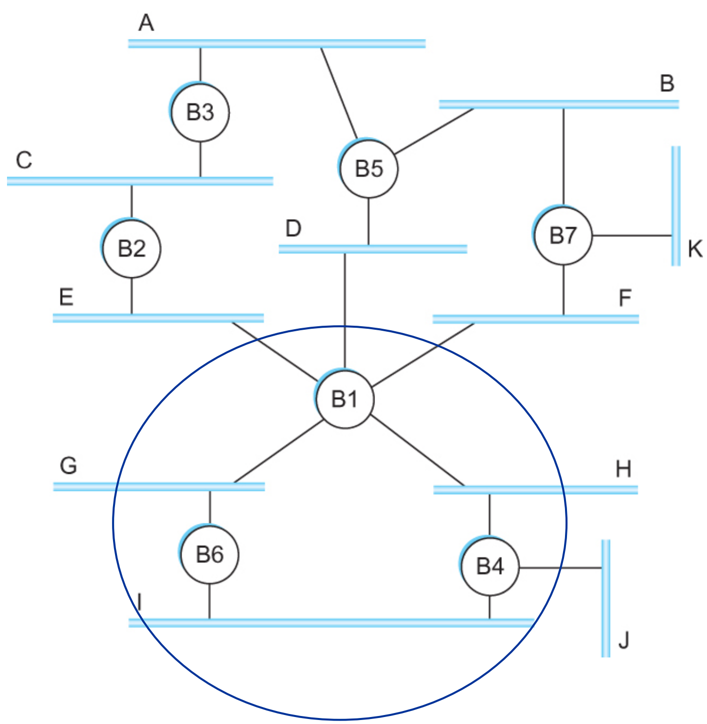


- Can the bridge learn this information by itself?
  - Yes – Learning Bridge
- How?
  - Each bridge inspects the source address in all the frames it receives
  - Record the information at the bridge and build the table
  - When a bridge first boots, this table is empty
  - Entries are added over time
  - A timeout is associated with each entry and he bridge discards the entry after a specified period of time
    - Due to node mobility
- If the bridge receives a frame that is addressed to host not currently in the table
  - Forward the frame out on all other ports

# Bridges and LAN Switches



- Strategy works fine if the extended LAN does not have a loop in it
- Why? Frames potentially loop through the extended LAN forever




- Bridges B1, B4, and B6 form a loop

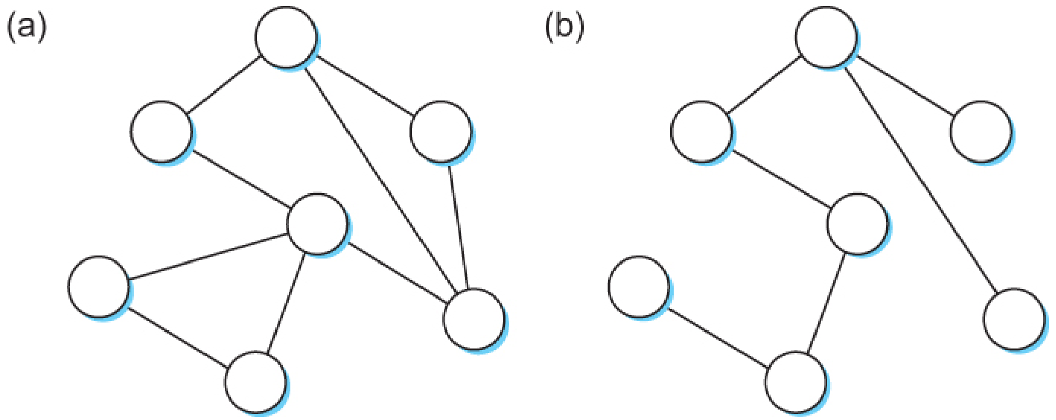
# Bridges and LAN Switches



- How does an extended LAN come to have a loop in it?
  - Loops are built into the network to provide redundancy in case of failures
- Solution
  - Distributed Spanning Tree Algorithm

# 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
  - throw out some of the edges




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

# Spanning Tree Algorithm

- IEEE 802.1 specification for LAN bridges is based on this algorithm
- Each bridge decides the ports over which it is and is not willing to forward frames
  - In a sense, it is by removing ports from the topology

# Spanning Tree Algorithm

- Algorithm is dynamic
  - The bridges are always prepared to reconfigure themselves into a new spanning tree if some bridges fail
- Main idea
  - Each bridge selects the ports over which they will forward the frames

# Spanning Tree Algorithm

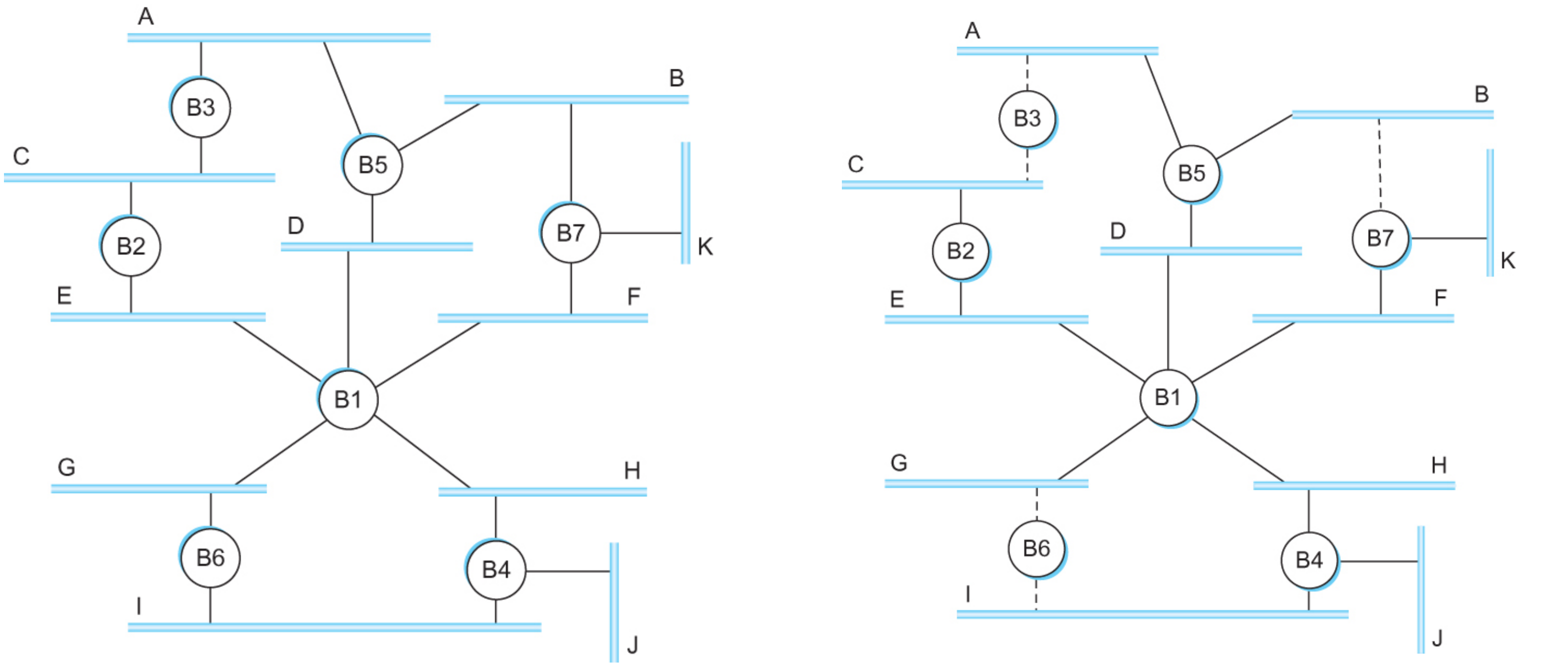
- Spanning Tree Algorithm selects ports as follows:
  - Each bridge has a unique **identifier** (B1, B2, B3,…).
  - Elect the bridge with the smallest id as the **root** of the spanning tree
  - The **root** bridge always forwards frames out over all of its ports
  - Each bridge computes the **shortest path** to the root and notes which of its ports is on this path
  - Finally, all the bridges connected to a given LAN elect a single *designated bridge* that will be responsible for forwarding frames toward the root bridge

# Spanning Tree Algorithm

- Each LAN’s designated bridge is the one that is closest to the root
- If two or more bridges are equally close to the root,
  - Then select bridge with the smallest id

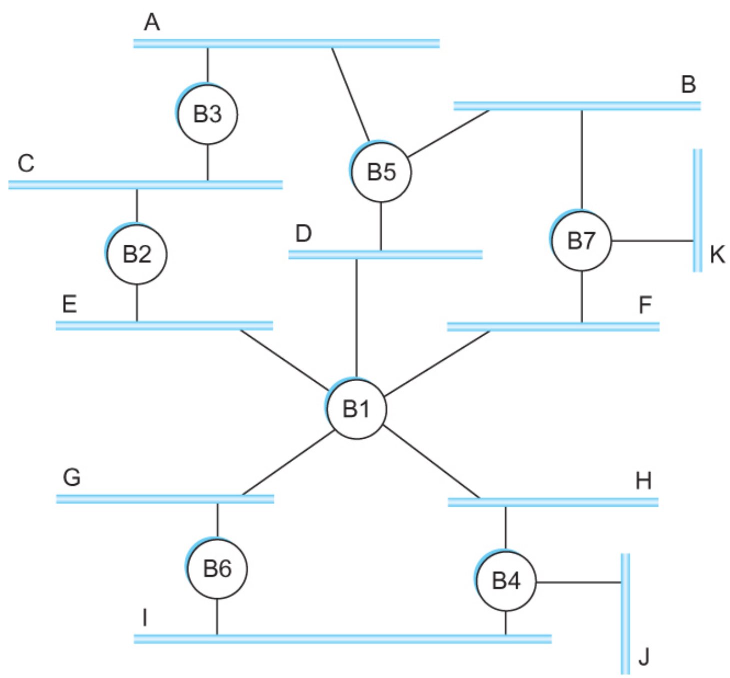
# Spanning Tree Algorithm

- B1 is the root bridge
- B3 and B5 are connected to LAN A, and B5 is the designated bridge
- B5 and B7 are connected to LAN B, and B5 is the designated bridge
- B3 and B2 are connected to LAN C, and B2 is the designated bridge



# Spanning Tree Algorithm

- Consider the network re-boot:




- All bridges would start off by claiming to be the root

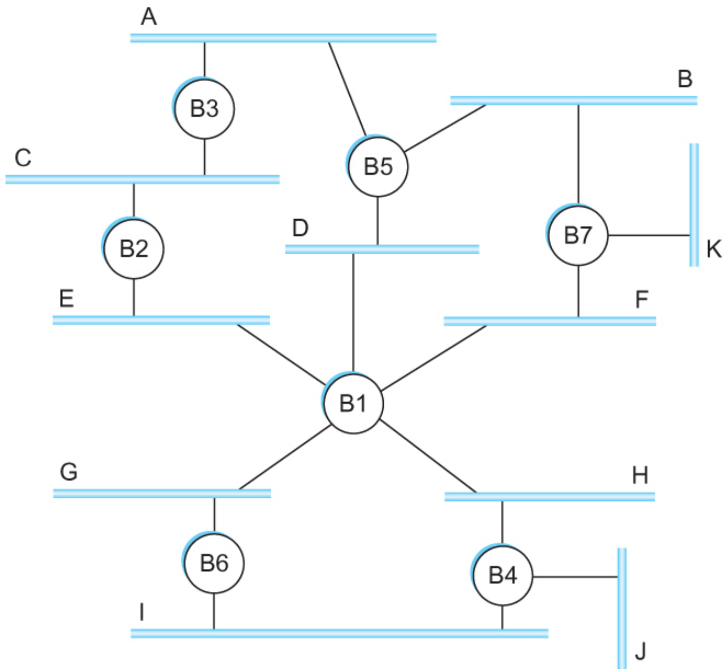
# Spanning Tree Algorithm

- Initially each bridge thinks it is the root, so it sends a configuration message
- Upon receiving a configuration message, the bridge checks to see if the new message is *better* than the current best configuration message
- The new configuration is better than the currently recorded information if
  - It identifies a root with a smaller id or
  - It identifies a root with an equal id but with a shorter distance or
  - The root id and distance are equal, but the sending bridge has a smaller id

# Spanning Tree Algorithm

Example:

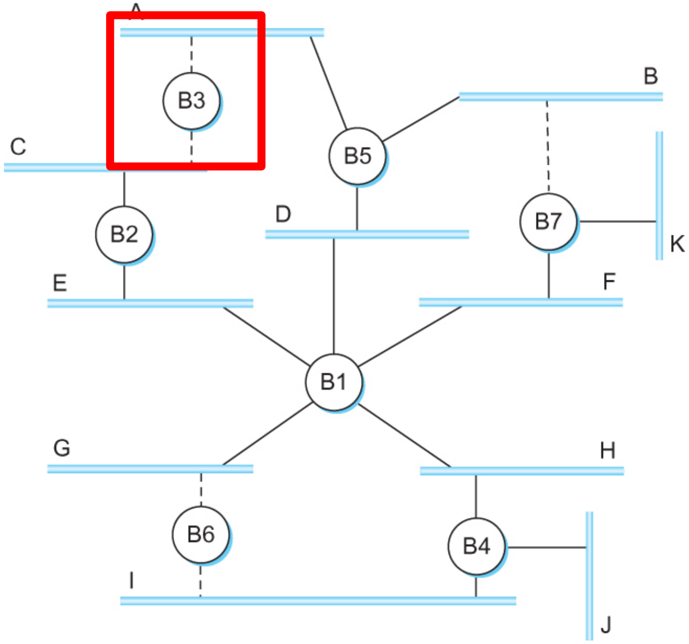
- Denote a configuration message from node X in which it claims to be distance d from the root node Y as (Y, d, X)




- Consider the activity at node B3

# Spanning Tree Algorithm

- B3 receives (B2, 0, B2)
- Since 2 < 3, B3 accepts B2 as root
- B3 adds 1 to the distance advertised by B2 and sends (B2, 1, B3) to B5
- Meanwhile B2 accepts B1 as root because it has the lower id and it sends (B1, 1, B2) toward B3
- B5 accepts B1 as root and sends (B1, 1, B5) to B3
- B3 accepts B1 as root and it notes that both B2 and B5 are closer to the root than it is.
    - Thus B3 stops forwarding messages on both its interfaces
    - This leaves B3 with both ports not selected



# Spanning Tree Algorithm

- Even after the system has stabilized, the root bridge continues to send configuration messages periodically
  - Other bridges continue to forward these messages
- When a bridge fails, the downstream bridges will not receive the configuration messages
- After waiting a specified period of time, they will once again claim to be the root and the algorithm starts again