Lecture 7: Bridging & Switching

CSE 123: Computer Networks

Stefan Savage

Notice:
Project 1 is up

Last time

- How do multiple hosts share a single channel?

- Medium Access Control (MAC) protocols
  - Channel partitioning (FDMA,TDMA,CDMA)
  - Contention-based protocols (CSMA/CD)

Lecture 7 Overview

- Moving beyond one wire
  - Link technologies have limits on physical distance
  - Also frequently on number of hosts connected
- Methods to interconnect LANs
  - Repeaters and bridges
  - Switching

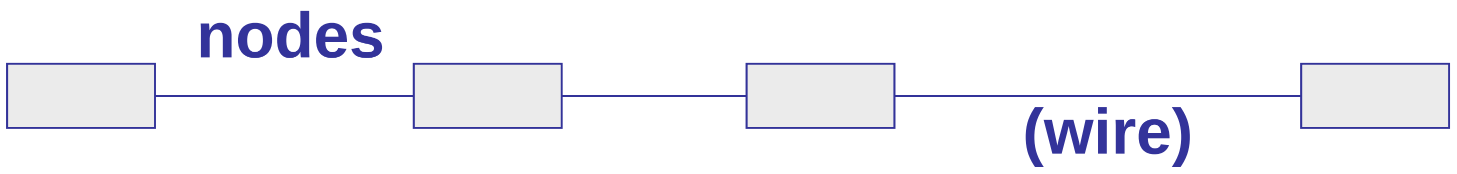
Today

- What if one wire isn’t enough?

- Interconnecting different LANs
  - Hubs/Repeaters: bit-for-bit rebroadcast
  - Bridges: selective rebroadcast
  - Switches: multi-port selective rebroadcast

Limits of a LAN

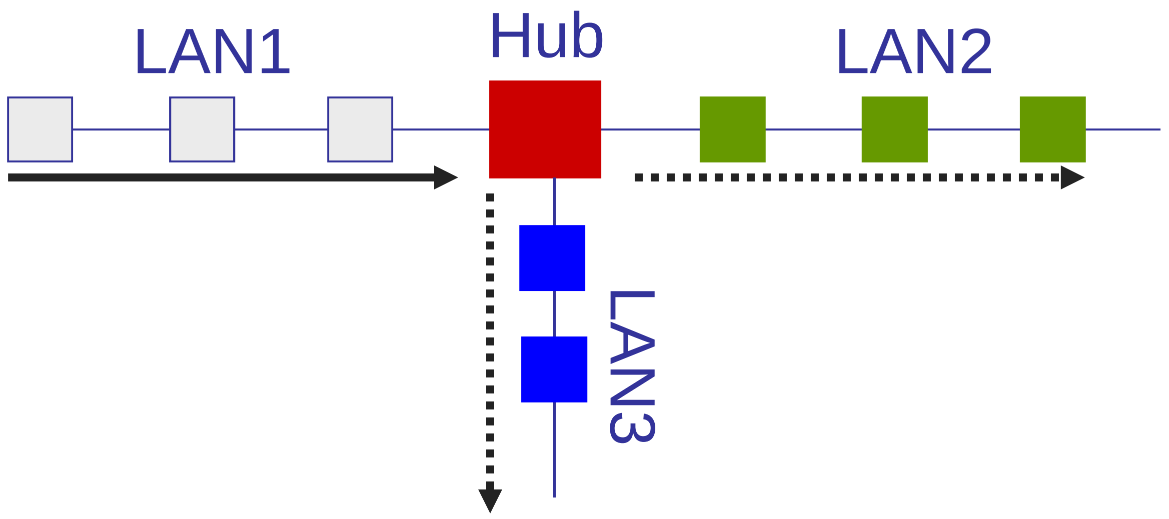
- One shared LAN can limit us in terms of:
  - Distance
    - » Max Ethernet segment is 2500m
  - Number of nodes
    - » Max nodes for Ethernet is 1024
  - Performance



- What to do?

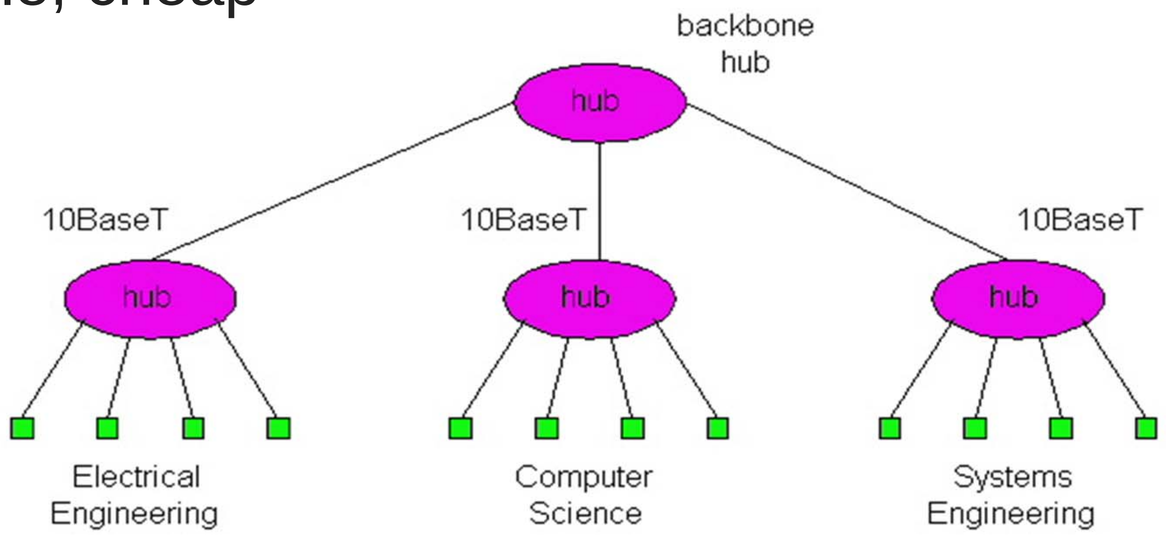
Hubs/Repeaters

- Physical layer device
  - One "port" for each LAN
  - Repeat received *bits* on one port out *all* other ports



Hub Advantages

- Hubs can be arranged into hierarchies
  - Ethernet: up to four hubs between any pair of nodes
- Most of LAN continues to operate if “leaf” hub dies
- Simple, cheap



Still One Big LAN (i.e., a Bus)

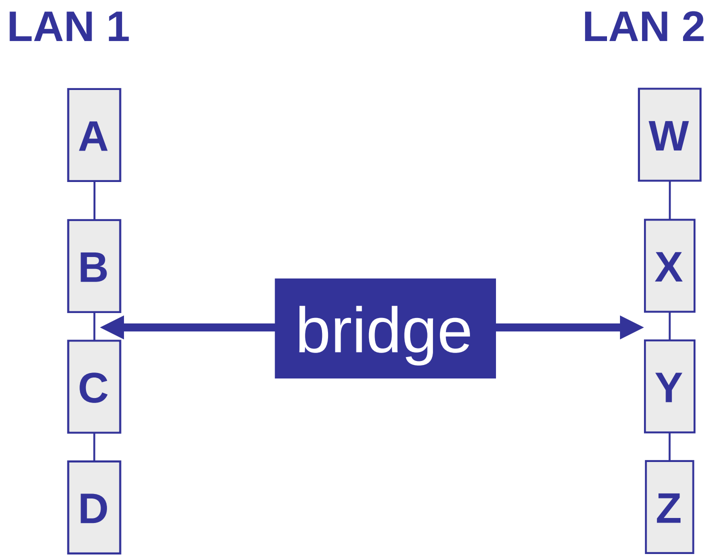
- Single collision domain
  - All hosts compete for access to same logical medium
  - No improvement in max throughput
  - Average throughput < as # of nodes increases
  - Why?
- Still limited in distance and number of hosts
  - Collision detection requirements
  - Synchronization requirements
- Requires performance homogeneity
  - Can’t connect 10 Mbps and 100 Mbps networks

Bridges to the rescue

- Store and forward device
  - Data-link layer device
  - Buffers entire packet and *then* rebroadcasts it on other ports
- Creates *separate* collision domains
  - Uses CSMA/CD for access to each LAN (acts like a host)
  - Can accommodate different speed interfaces (issues?)
  - Improves throughput (why?)
- Can significantly improve performance
  - Not all frames go everywhere. (Why did they with a hub?)

Key value: Selective Forwarding

- Only rebroadcast a frame to the LAN where its destination resides
  - If *A* sends packet to *X*, then bridge must forward frame
  - If *A* sends packet to *B*, then bridge shouldn't



Forwarding Tables

- Need to know “destination” of frame
  - Destination address in frame header (48bit in Ethernet)
- Need know which destinations are on which LANs
  - One approach: statically configured by hand
    - Table, mapping address to output port (i.e. LAN)
  - But we’d prefer something automatic and dynamic…
- Simple algorithm:

```
Receive frame f on port q
Lookup f.dest for output port
If f.dest found
then if output port is q then drop /* already delivered */
        else forward f on output port;
else flood f;
/* forward on all ports but the one where frame arrived*/
```

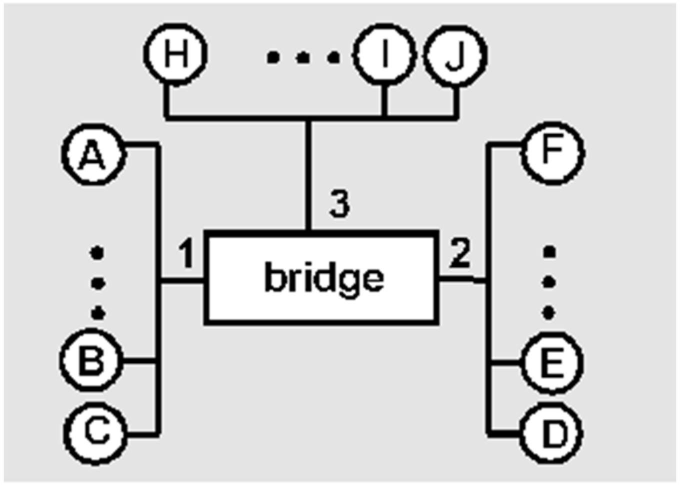
Learning Bridges

- Eliminate manual configuration by learning which addresses are on which LANs
- Basic approach
  - If a frame arrives on a port, then associate its **source** address with that port
  - As each host transmits, the table becomes accurate
- What if a node moves?
- Table aging
  - Associate a timestamp with each table entry
  - Refresh timestamp for each new packet with same source
  - If entry gets too stale, remove it

Host	Port
A	1
B	1
C	1
D	1
W	2
X	2
Y	3
Z	2

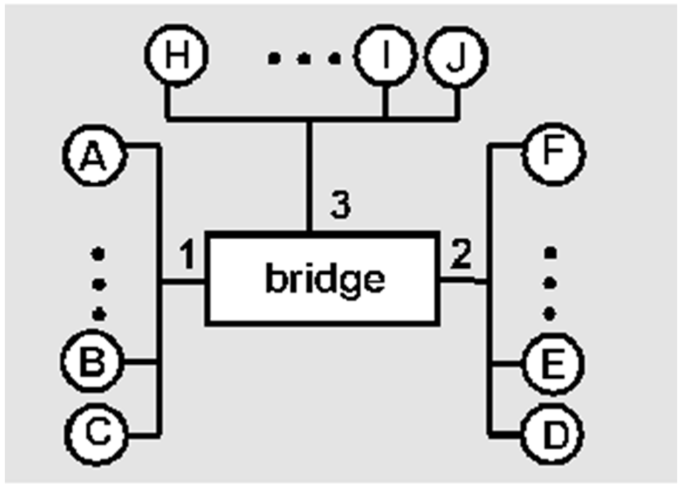
Learning Example

Suppose *C* sends frame to *D* and *D* replies back with frame to *C*



- C sends frame, bridge has no info about D, so floods to both LANs
  - bridge notes that C is on port 1
  - frame ignored on upper LAN
  - frame received by D

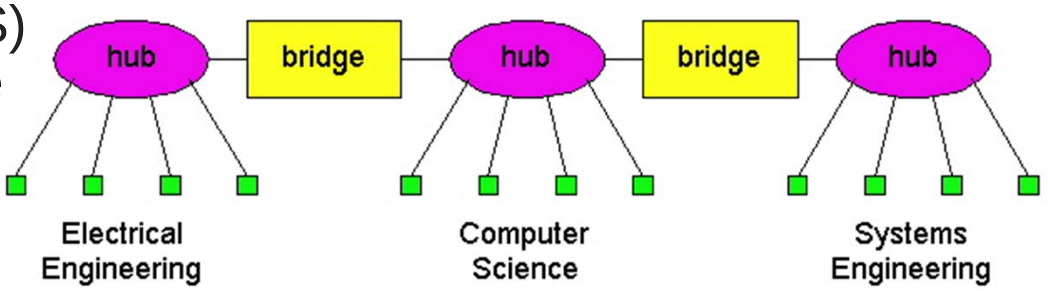
Learning Example



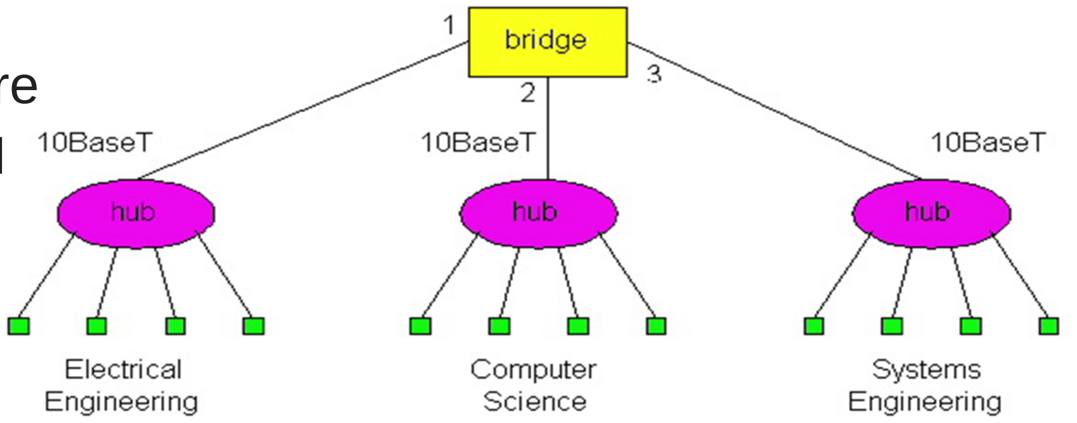
- *D* generates reply to *C*, sends
  - bridge sees frame from *D*
  - bridge notes that *D* is on port 2
  - bridge knows C on port 1, so **selectively** forwards frame via port 1

Network Topology

- Linear organization
  - Inter-bridge hubs (e.g. CS) are single points of failure
  - Unnecessary transit (e.g. EE<->SE must traverse CS)

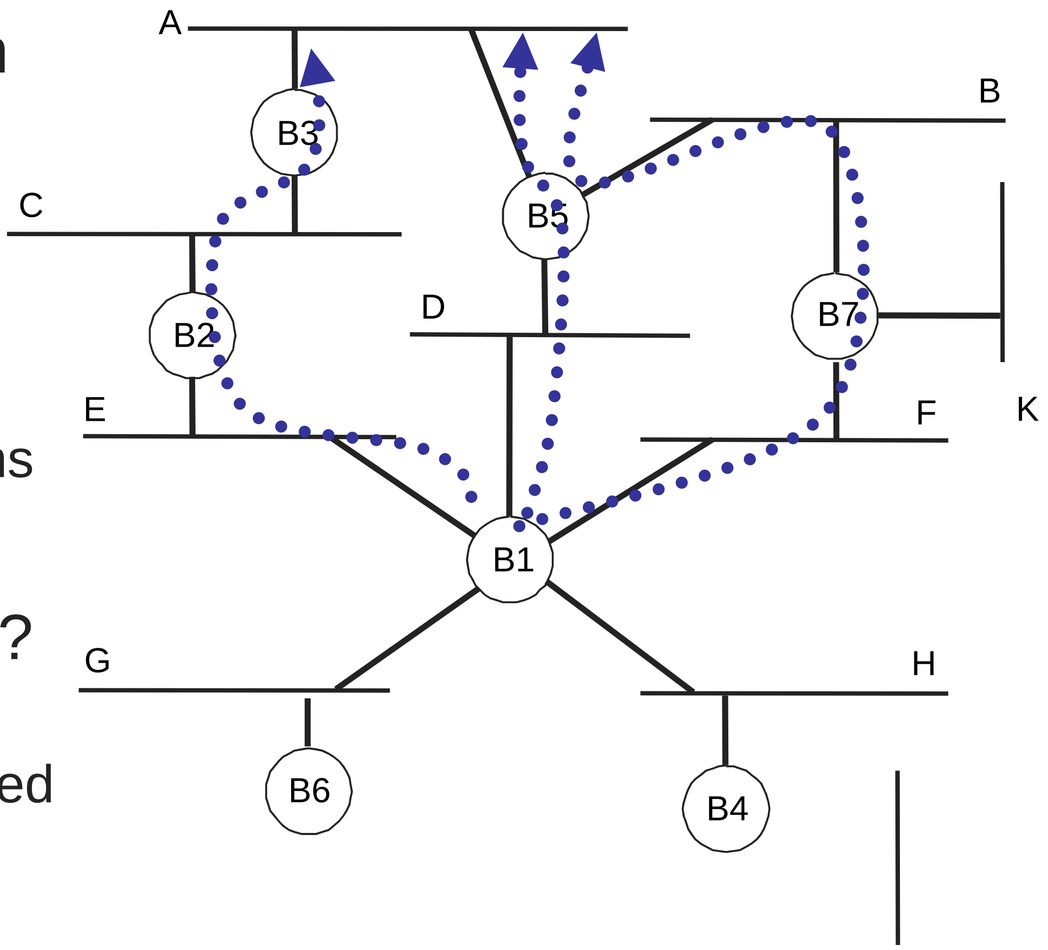


- Backbone/tree
  - Can survive LAN failure
  - Manages all inter-LAN communication
  - Requires more ports



An Issue: Cycles

- Learning works well in tree topologies
- But trees are fragile
  - Net admins like redundant/backup paths
- How to handle Cycles?
  - Where should *B1* forward packets destined for LAN *A*?

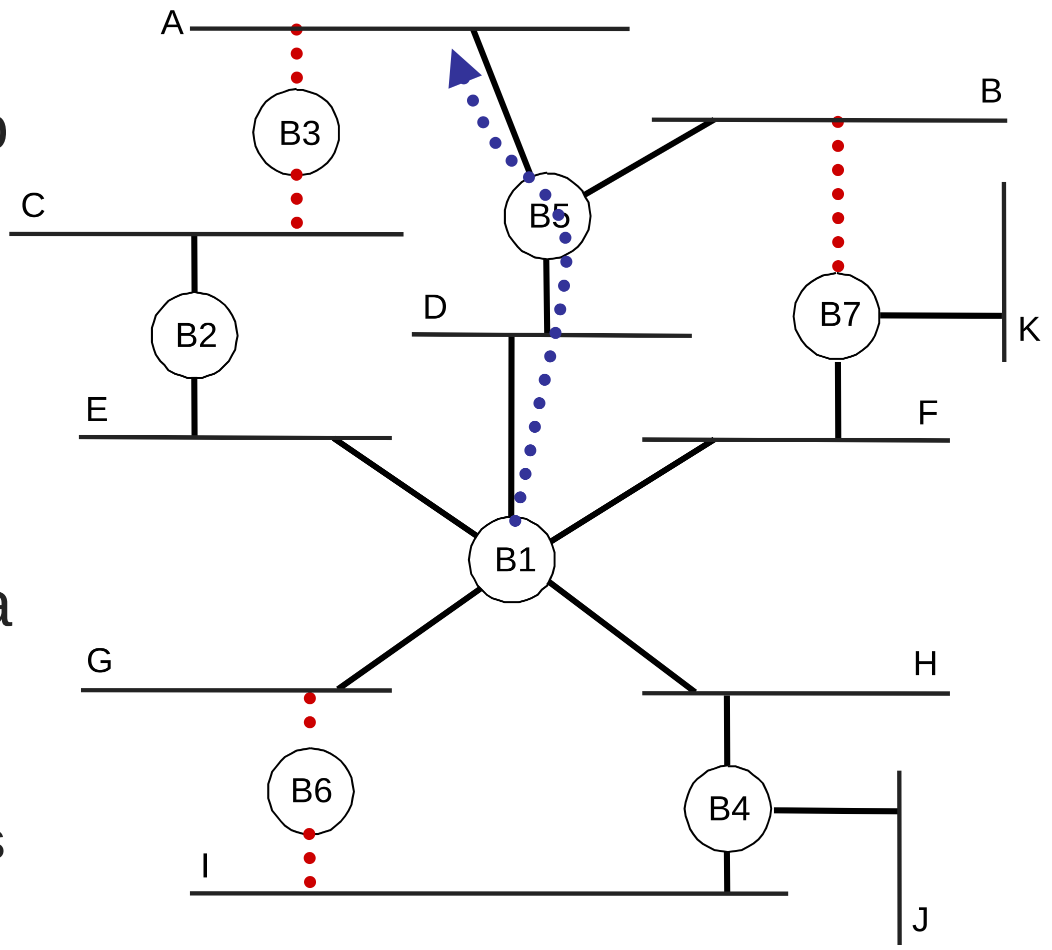


Potential solutions

- Don’t allow redundant links (no loops allowed)
  - E.g., physical tree… downside?
- Distributed routing protocol (SPF)
  [future lecture]
- Create a temporary “virtual tree” on the physical topology
  - Spanning Tree algorithm

Spanning Tree

- Spanning tree uses *subset* of bridges so there are no cycles
  - Prune some ports
  - Only **one** tree
- Q: How do we find a spanning tree?
  - Automatically!
  - Elect root, find paths



Spanning Tree Algorithm: 10,000 foot view

- Elect a root node of the tree (lowest address)

- Grow tree as shortest distances from the root (use lowest address to break distance ties)
  - All bridges send periodic configuration messages over ports for which they are the "best" path
  - Then **turn off** ports that aren't on the "best" paths

Spanning Tree Algorithm: Details

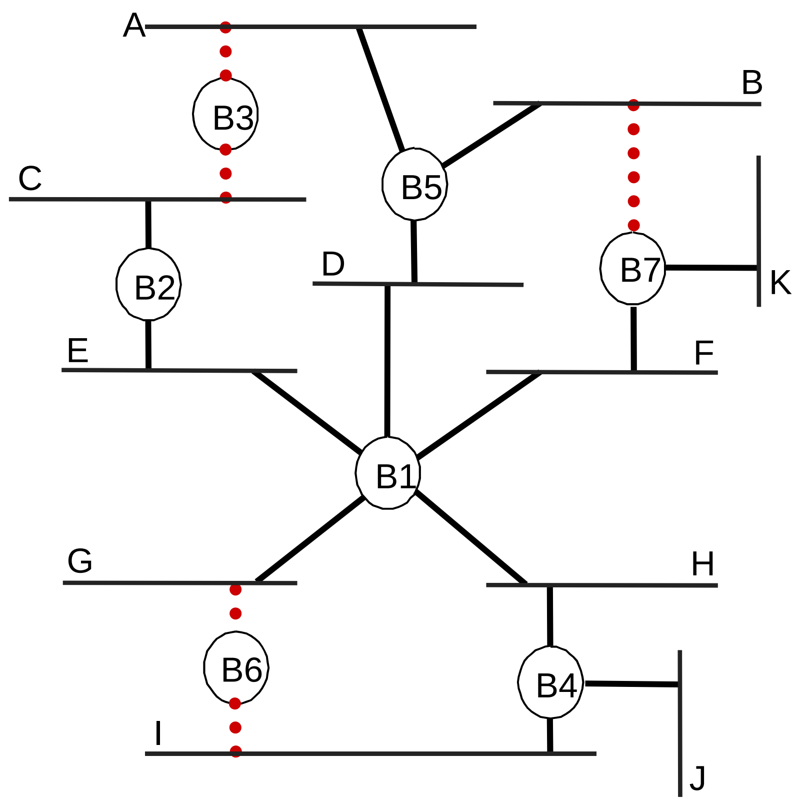
- Each bridge sends periodic configuration messages
  - (RootID, Distance to Root, BridgeID)
    - » "I am BridgeID, the Root is named RootID, I'm X hops away"
  - All nodes think they are the root initially
- Each bridge updates route/Root upon receipt
  - Smaller root address is better, then shorter distance
  - To break ties, bridge with smaller address is better
  - Record best config heard via each port
- Rebroadcast new config only to ports we're "best"
  - Don't bother sending config to LANs with better options
  - Add 1 to distance, send new configs where still "best"
  - Only forward to ports to route or where we're best

Spanning Tree Example

- Sample messages to and from B3:

  1. B3 sends (B3, 0, B3) to B2 and B5
  2. B3 receives (B2, 0, B2) and (B5, 0, B5) and accepts B2 as root
  3. B3 sends (B2, 1, B3) to B5
  4. B3 receives (B1, 1, B2) and (B1, 1, B5) and accepts B1 as root
  5. B3 wants to send (B1, 2, B3) but doesn't as its nowhere "best"
  6. B3 receives (B1, 1, B2) and (B1, 1, B5) again and again…

Data forwarding is turned off for LAN A

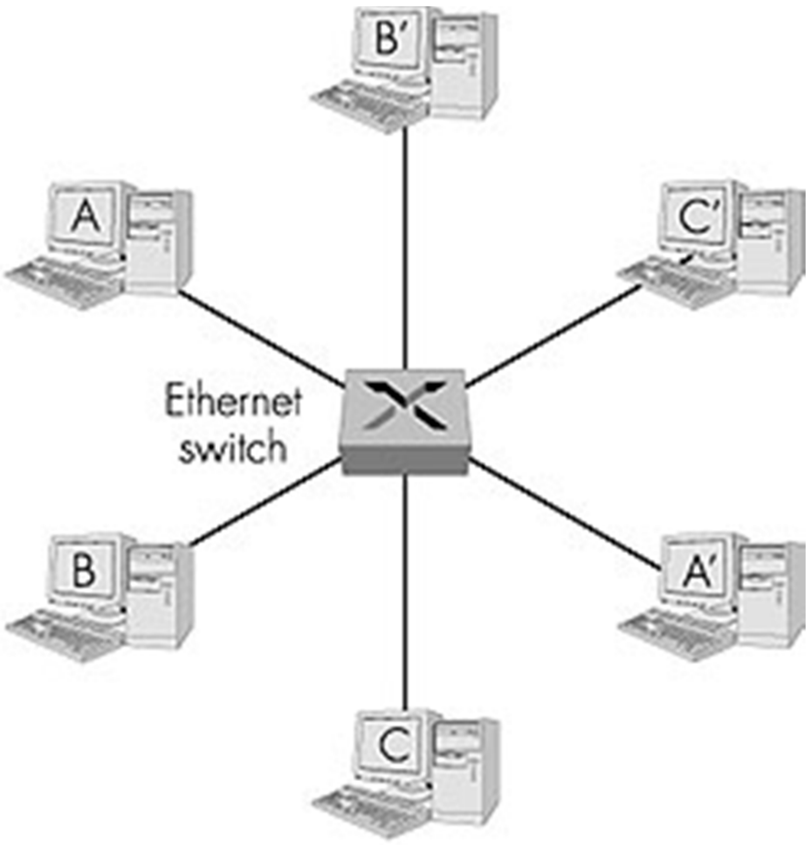


Important Details

- What if root bridge fails?
  - Age configuration info
    - If not refreshed for MaxAge seconds then delete root and recalculate spanning tree
    - If config message is received with more recent age, then recalculate spanning tree
  - Applies to all bridges (not just root)
- Temporary loops
  - When topology changes, takes a bit for new configuration messages to spread through the system
  - Don't start forwarding packets immediately -> wait some time for convergence
- Broadcast packets?
  - Sent on every active port

Switched Ethernet

- Hosts directly connected to a bridge
  - learning + spanning tree protocol
- Bridge supports parallel forwarding
  - A-to-B and A'-to-B' simultaneously
  - Generally full duplex as well (A->B and B->A in parallel)
- Switch backplane capacity varies
  - Ideally, nonblocking
  - I.e., can run at full line rate on all ports
- No longer any shared bus
  - Each link is its own collision domain
  - Collision detection largely irrelevant



Some switching details

- *Cut through* switching optimization
  - Only buffer packet header (for output port lookup)
    - » then forward remaining bits directly
  - Reduced latency, but may forward bad packets
  - Extremely common in real Ethernet switches

- Backpressure flow control
  - Input port=1Gbps, output port = 100Mbps
  - Buffer can only absorb temporary bursts
  - Send JAM signal on input port when buffer gets too full

Modern extension: VLANs

- Scaling problem with switches
  - All LANs in same **broadcast** domain
    - Recall: broadcast packets sent everywhere
  - As # hosts grows, broadcast traffic becomes an issue
- Virtual LANs (VLANs) created to address this issue
  - Each port optionally configured with a VLAN ID
    - Configured statically
  - Inbound packets “tagged” with this ID
  - All switches will only forward on ports that are part of the same VLAN
- Create independent broadcast domains within a single tree

Summary: Layer 2 Forwarding

- Create spanning tree across LANs
  - Learn which ports to use to reach which addresses
- Benefits
  - Higher link bandwidth (point-to-point links)
  - Higher aggregate throughput (parallel communication)
  - Improved fault tolerance (redundant paths)
- Limitations
  - Requires homogeneous link layer (e.g. all Ethernet)
  - Can’t control forwarding topology
- What if we want to connect different link layers?

For Next Time

- Read 3.2 in P&D
- Reminder: Project #1 has been assigned