CSE/EE 461 – 10-18-04

Bridging LANs

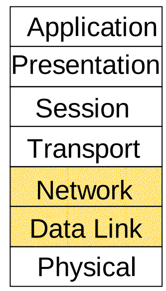
Last Two Times …

- Medium Access Control (MAC) protocols
  - Part of the Link Layer
  - At the heart of Local Area Networks (LANs)
- How do multiple parties share a wire or the air?
  - Random access protocols (CSMA/CD)
  - Contention-free protocols (turn-taking, reservations)
  - Wireless protocols (CSMA/CA and RTS/CTS)

This Time -- SWITCHING

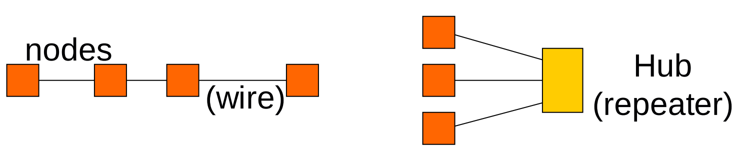
- Focus:
  - What to do when one shared LAN isn't big enough?

- Interconnecting LANs
  - Bridges and LAN switches
  - A preview of the Network layer



Limits of a LAN

- One shared LAN can limit us in terms of:
  - Distance
  - Number of nodes
  - Performance



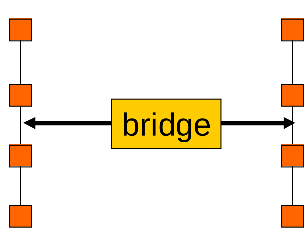
- How do we scale to a larger, faster network?
  - We must be able to interconnect LANs

SWITCHING

- Xferring a packet from one network to another
- Packet switched vs. circuit switched
- Connection vs. Connectionless
- Contention vs. Congestion

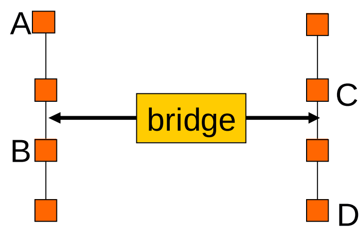
Bridges and Extended LANs

- "Transparently" interconnect LANs with bridge
  - Receive frames from each LAN and forward to the other
  - Each LAN is its own collision domain; bridge isn't a repeater
  - Could have many ports or join to a remote LAN



Backward Learning Algorithm

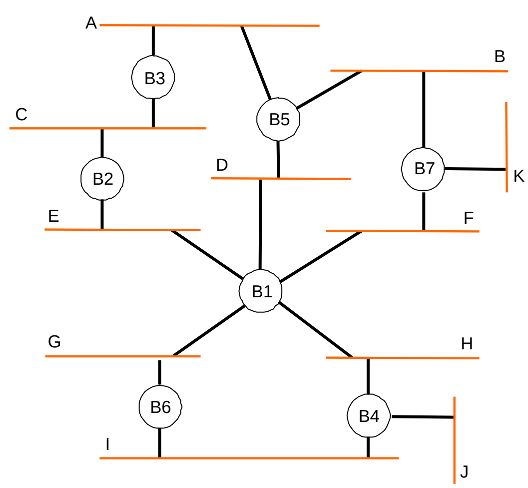
- To optimize overall performance:
  - Shouldn't forward A→B or C→D, should forward A→C and D→B



- How does the bridge know?
  - Learn who is where by observing <u>source</u> addresses and prune
  - Forward using destination address; age for robustness

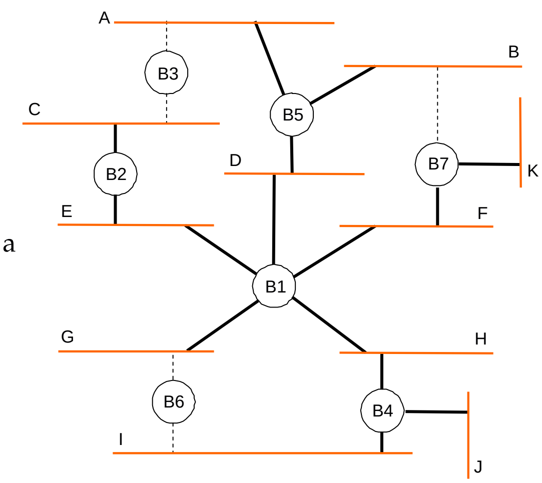
Why stop at one bridge?

- But to avoid loops we must forward only on select bridge ports!
- The Spanning Tree algorithm does this
- It is separate from backward learning



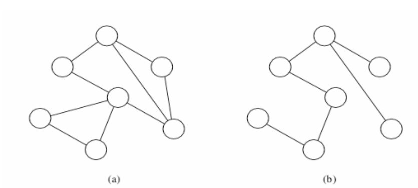
Spanning Tree Example

- Spanning tree uses select bridges so there are no cycles
  - Prune some ports
  - Only one tree
- Q: How do we find a spanning tree?
  - Automatically
  - Think:
    - Rootier
    - Rootiest



Spanning Tree

- Compute ST with *a* bridge as *root* such that
  - Root forwards onto all of its outgoing ports
  - Other bridges forward TO the root if a packet is coming from a bridge further from the root, else they forward away from the root
    - (UP)*(DOWN*)



The Aunt Martha Explanation

- Bridges run an algorithm to determine a spanning tree
- If a bridge is on the path to the root, it forwards messages to the root via the next bridge (up).
- If a bridge is not on the path to the root, it doesn't forward messages to the root.
- If a switch has heard from a given host via a bridge, it forwards to that host via the bridge (down)
- If a bridge is the root, it forwards to all bridges.

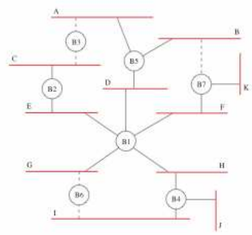
Spanning Tree Algorithm

- Distributed algorithm to compute spanning tree
  - Robust against failures, needs no organization
  - Developed by Radia Perlman at DEC
    - IEEE 802.1 spec

- Outline: Goal is to turn some bridge ports off
  1. Elect a root node of the tree (lowest address)
  2. Grow tree as shortest distances from the root (using lowest address to break distance ties)
     - All done by bridges sending periodic configuration messages over ports for which they are the "best" path
     - Then turn off ports that aren't on "best" paths

Algorithm Overview

- Each bridge has a unique id (e.g., B1, B2, B3)
- Select bridge with smallest id as root
- Select bridge on each LAN that is closest to the root as that LAN's designated bridge (use id to break ties)
- Each bridge forwards frames over each LAN for which it is the designated bridge



Algorithm continued

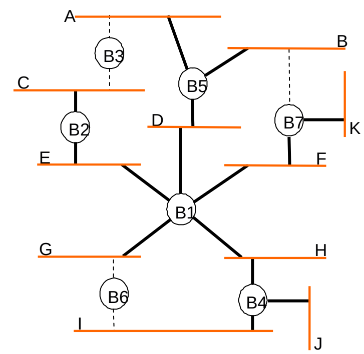
- Bridges exchange configuration messages
  - id for bridge sending the message
  - id for what the sending bridge believes to be root bridge
  - distance (hops) from sending bridge to root bridge
- Each bridge records current best configuration message for each port
- Initially, each bridge believes it is the root
- When learn not root, stop generating configuration message
  - in steady state, only root generates configuration messages

Algorithm More…

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

Algorithm Example

- Message format: (root, dist to root, bridge)
- Sample messages sequences 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, B2) but doesn't as its nowhere "best"
     B2 and b5 are better choices.
  6. B3 receives (B1, 1, B2) and (B1, 1, B5) again … stable
     - Data forwarding is turned off to the LAN A

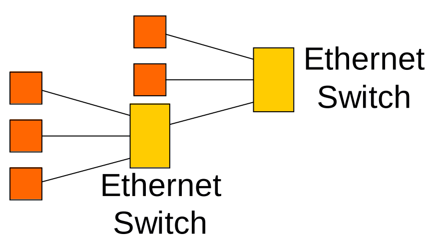


Some other tricky details

- Configuration information is aged
  - If the root fails a new one will be elected
- Reconfiguration is damped
  - Adopt new spanning trees slowly to avoid temporary loops

LAN Switches

- LAN switches are multi-port bridges
  - Modern, high performance form of bridged LANs
  - Looks like a hub, but frames are switched, not shared
  - Every host on a separate port, or can combine switches



Limitations of Bridges/Switches

- LAN switches form an effective small-scale network
  - Plug and play for real!
- Why can't we build a large network using bridges?
  - Little control over forwarding paths
  - Size of bridge forwarding tables grows with number of hosts
  - Broadcast traffic flows freely over whole extended LAN
  - Spanning tree algorithm limits reconfiguration speed
  - Poor solution for connecting LANs of different kinds

Key Concepts

- We can overcome LAN limits by interconnection
  - Bridges and LAN switches
  - But there are limits to this strategy …
- Next Topic: Routing and the Network layer
  - How to grow large and really large networks