ELEC / COMP 177 – Fall 2011

Computer Networking → Ethernet

Some slides from Kurose and Ross, *Computer Networking*, 5th Edition

Project #2

- **Peer evaluations**

Schedule

- **Homework #5** – Due Thursday, Nov 17th
- **Homework #6** – Presentation on security/ privacy
- **Project #3** – Due Tuesday, Dec 6th

Homework #6

- *Looking for a change of pace…*
- **In-class oral presentations**
  - Pick a single topic related to network privacy or security
    - Attacks? Defenses? Revolutionary new network designs?
  - Read about it and understand it
  - Present topic to *your peers* in this class
  - **4-6 slides, 8-9 minutes talking, 1 minute questions**

Homework #6 Requirements

- **Topic must be approved** by instructor
  - Prevents overlap in topics
  - A quick email is fine
  - Due Tuesday, Nov 22nd
- **Slides must be uploaded to Sakai**
  - I'll assemble them into a single file on my laptop
  - Due Monday, Nov 28th by midnight
  - PowerPoint or PDF please…
- **Present!** – Tuesday, Nov 29th (and Thursday?)

Project #3: Web Proxy

- **Due:** Tuesday, December 6th by 11:59pm
- What is a web proxy?
  - Makes HTTP requests on behalf of a client
- Why proxy?
  - Performance (from caching)
  - Content Filtering and Transformation
    - Block pages? (security)
    - Reformat pages? (for mobile devices)
  - Privacy – harder to link HTTP request to a specific individual

Project #3 – Web Proxy

- Client (web browser) must be modified!
  - IP and port of proxy
  - Capabilities? HTTP/1.0, no pipelining
- Client sends out *slightly* different HTTP request
  - Without proxy:
    - `GET /about HTTP/1.0`
  - With proxy:
    - `GET http://www.google.com/about HTTP/1.0`
    - Now the proxy knows what the destination server is!

Project #3 – Web Proxy

1. Proxy is running on server and listening on a port
2. User enters URL in browser and hits enter
3. Client connects and sends *modified* HTTP request to **proxy**, not to destination server
4. Proxy decodes URL
5. Proxy opens connection to destination server
6. Proxy sends *normal* HTTP request for object
7. Proxy receives response from destination server
8. Proxy forwards full response to client (Headers and data!)
9. Proxy closes connection to destination server and client

Project #3 – Web Proxy

- Tip – Use netcat when debugging to listen on a port and see what your client is sending

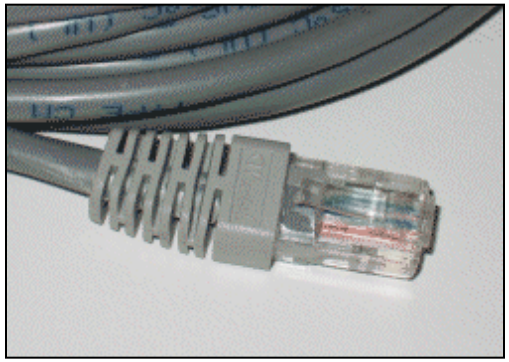
```
jshafer:~> netcat -l -p 4567 -v
listening on [any] 4567 ...
connect to [127.0.0.1] from localhost [127.0.0.1] 49711
GET http://www.opensuse.org/ HTTP/1.0
Host: www.opensuse.org
User-Agent: Mozilla/5.0 (X11; U; Linux x86_64; en-US; rv:
1.9.2.10) Gecko/20100914 SUSE/3.6.10-0.3.1 Firefox/3.6.10
Accept: text/html,application/xhtml+xml,application/
xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Connection: close
Proxy-Connection: close
```

New physical layers

Scaling Ethernet

New Technology Needed

- No more single wire shared by all devices!
  - Too hard to increase to higher speeds
- Point-to-point networking
  - Still use MAC protocol and frame format
  - New network device: Ethernet repeater / hub
  - New physical layer
    - Straight-through cable (device ↔ hub) or crossover cable (device ↔ device)



New Physical Layers

- 100 Mb/s
  - 100Base-T4 (4 pairs copper, 100 meters max)
  - 100Base-TX (2 pairs high-quality copper, 100 meters max)
  - 100Base-FX (2 optical fibers)
  - … and others
- 1000 Mb/s
  - 1000Base-T (4 pairs high-quality copper, 100 meters max)
  - 1000Base-FX (2 optical fibers)
  - … and others
- Different physical layers (and encoding standards)
- Same frame format, error correction, and MAC protocol

Gigabit Ethernet – Same 4 Challenges

- Encoding
  - Encoding formats grow in sophistication as clock rate increases and stresses physical limits of copper/fiber media
  - 5-level Pulse Amplitude Modulation
  - 4-D 8-State Trellis Forward Error Correction Encoding
- Framing – Same format
- Error Detection
  - CRC still used at high frame level
  - Encoding method has reserved illegal symbols that automatically indicate error (noise / corruption) if received
- Media Access Control
  - Point-to-point links remove need for CSMA / CD protocol (but it remains for backwards compatibility)

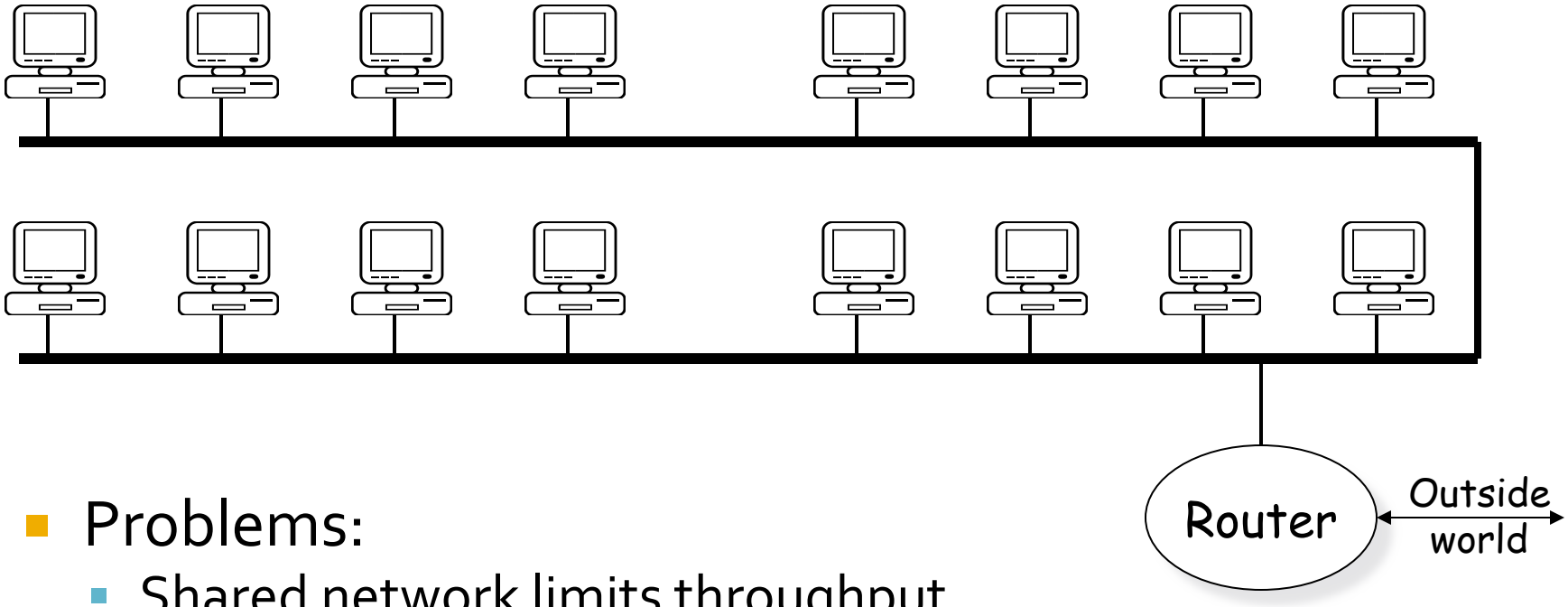
Full Duplex Ethernet

- Simultaneous two-way transmission (send and receive)
- No more collisions or retransmissions! (at least due to Ethernet)
- Only useful over point-to-point links, not shared bus (or hub topology)
  - Design enabled by pervasive deployment of switches

New network topologies

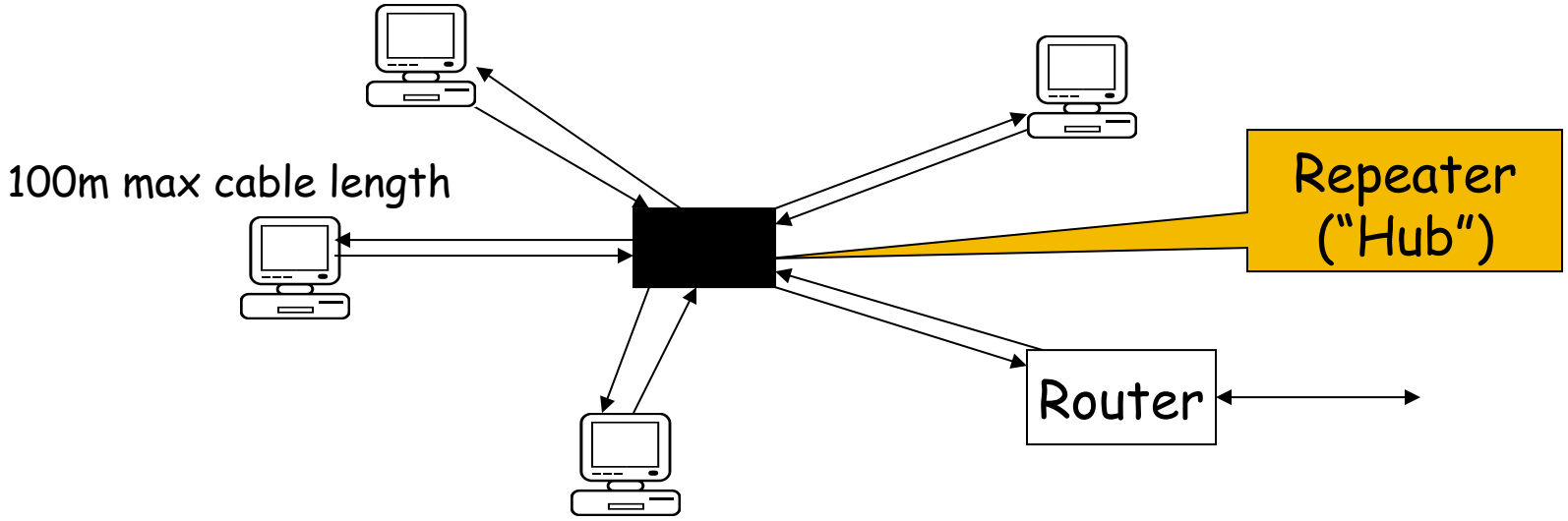
Scaling Ethernet

Traditional Ethernet Network
(10Base-5 or 10Base-2 – Shared Bus Architecture)



- Problems:
  - Shared network limits throughput
  - Frequent collisions reduce efficiency
  - Poor Reliability – Failure at one node can break shared link

Ethernet Star Topology

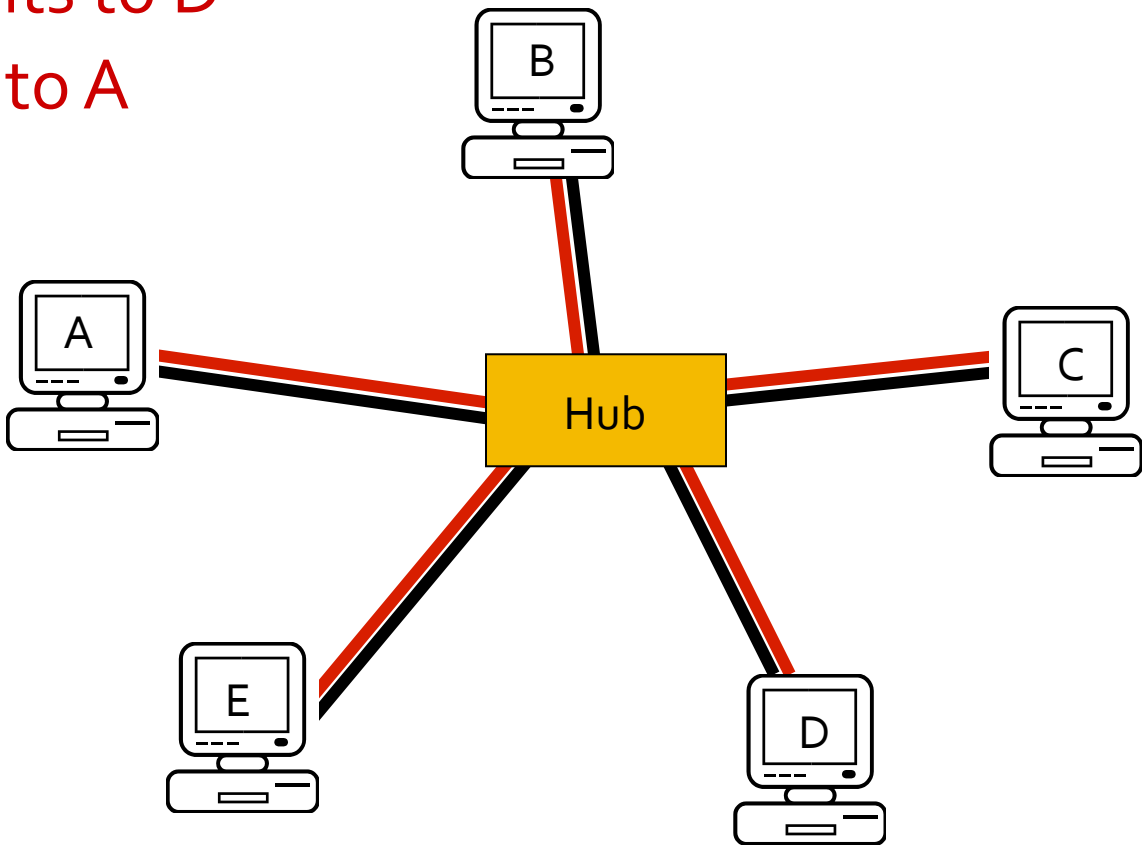


- Direct links instead of shared bus
- MAC protocol still operates as if Ethernet was a single wire
  - Collisions still possible
  - Network still shared
- Increase reliability from wire failure

Ethernet Hub - Operation

A transmits to D

D replies to A



Problems with Ethernet Hub

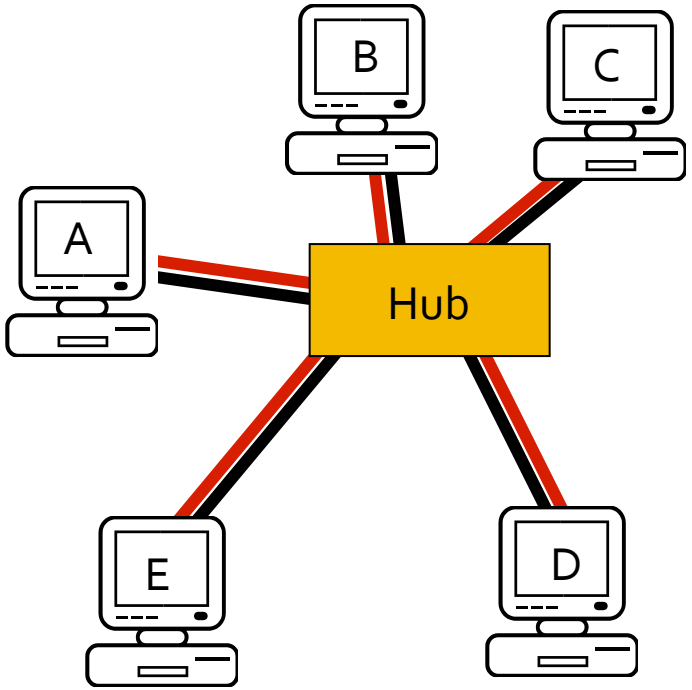
- Security concerns with broadcasting
- Performance
  - Unnecessary broadcasts waste network capacity and cause congestion
  - Communication is serialized – Independent connections between independent devices cannot occur in parallel
- Shared bus architecture limits maximum length of network
  - Due to MAC CSMA algorithm and signal propagation across entire network

Ethernet Switch

- New solution – Bridges
  (aka Ethernet **switches**)
  - Allow multiple hub-based networks to be partitioned and interconnected
  - Reduces collisions
  - Allow parallel communication between independent devices
  - Allow full duplex communication between multiple pairs of devices

Ethernet Hub vs Switch

Ethernet Hub

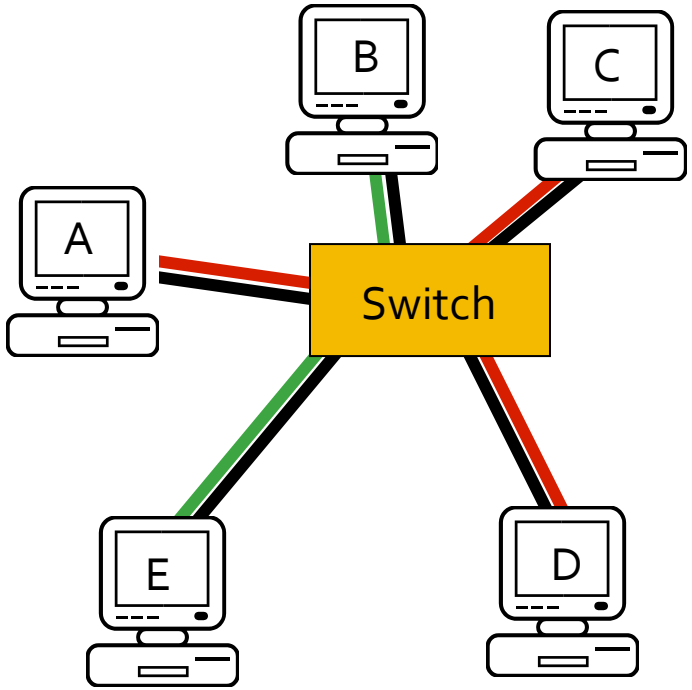


A transmits to D

D replies to A

Ethernet Switch

(assume learning already occurred)



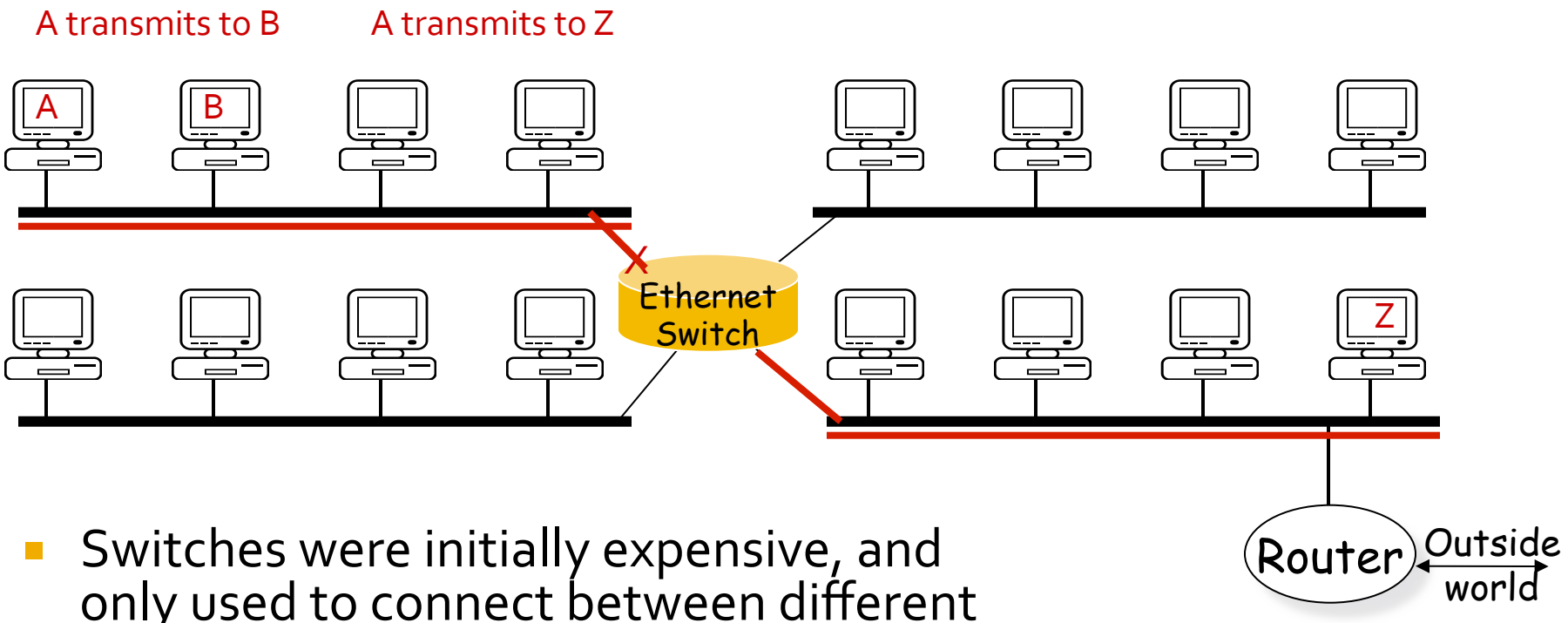
A transmits to D

D replies to A

E transmits to B,
and A to C

Animation from
http://www.datacottage.com/nch/eoperation.htm

Combining Hubs and Switches



- Switches were initially expensive, and only used to connect between different hub-based networks
- As cost decreased, hubs have been removed entirely
  - Gigabit+ networks are always switched
  - No more collisions!

Switch Design

- Internal FIFOs on each port buffer incoming packet
- Forwarding options
  - *Store-and-Forward*
    - Buffer entire packet before sending it to output port
    - Can verify packet CRC
  - *Cut-Through*
    - Buffer only long enough to examine destination address and then immediately stream data through to output port
      - Will fall back to store-and-forward if output port is busy
    - Cannot validate packet – By the time error is detected, it is too late!

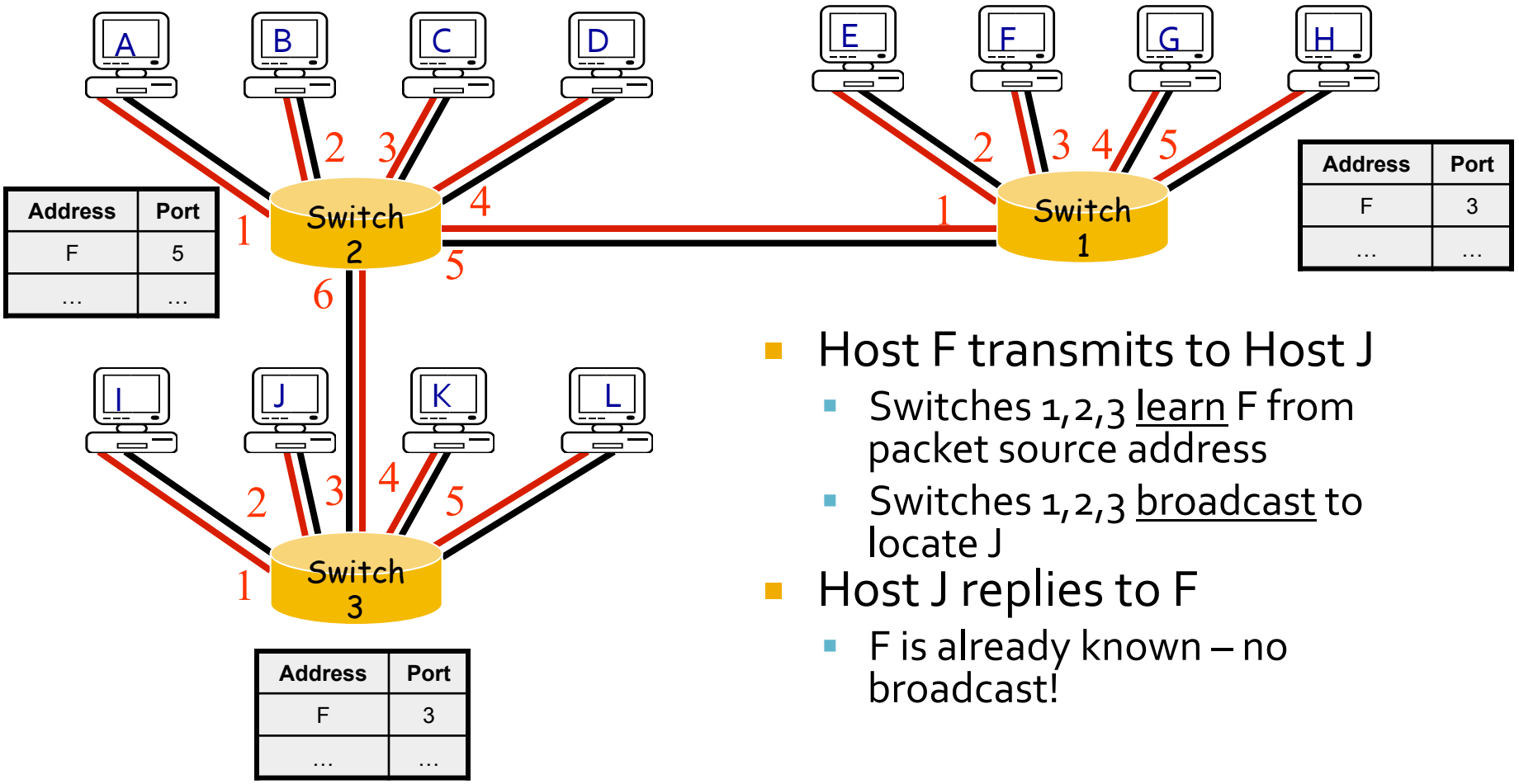
Challenges for Ethernet Switch

- *Forwarding* – Where does the next packet go?
- *Migration* – What if devices move on the network?
- *Congestion* – What if too much traffic is received?
- *Preventing Loops* – How to avoid forwarding packets in a big loop?
- *Configuration* – How to determine speed of every device connected to switch
- *Isolation* – How to isolate devices from each other (i.e. student computers from faculty computers)

Challenge – Forwarding Packets

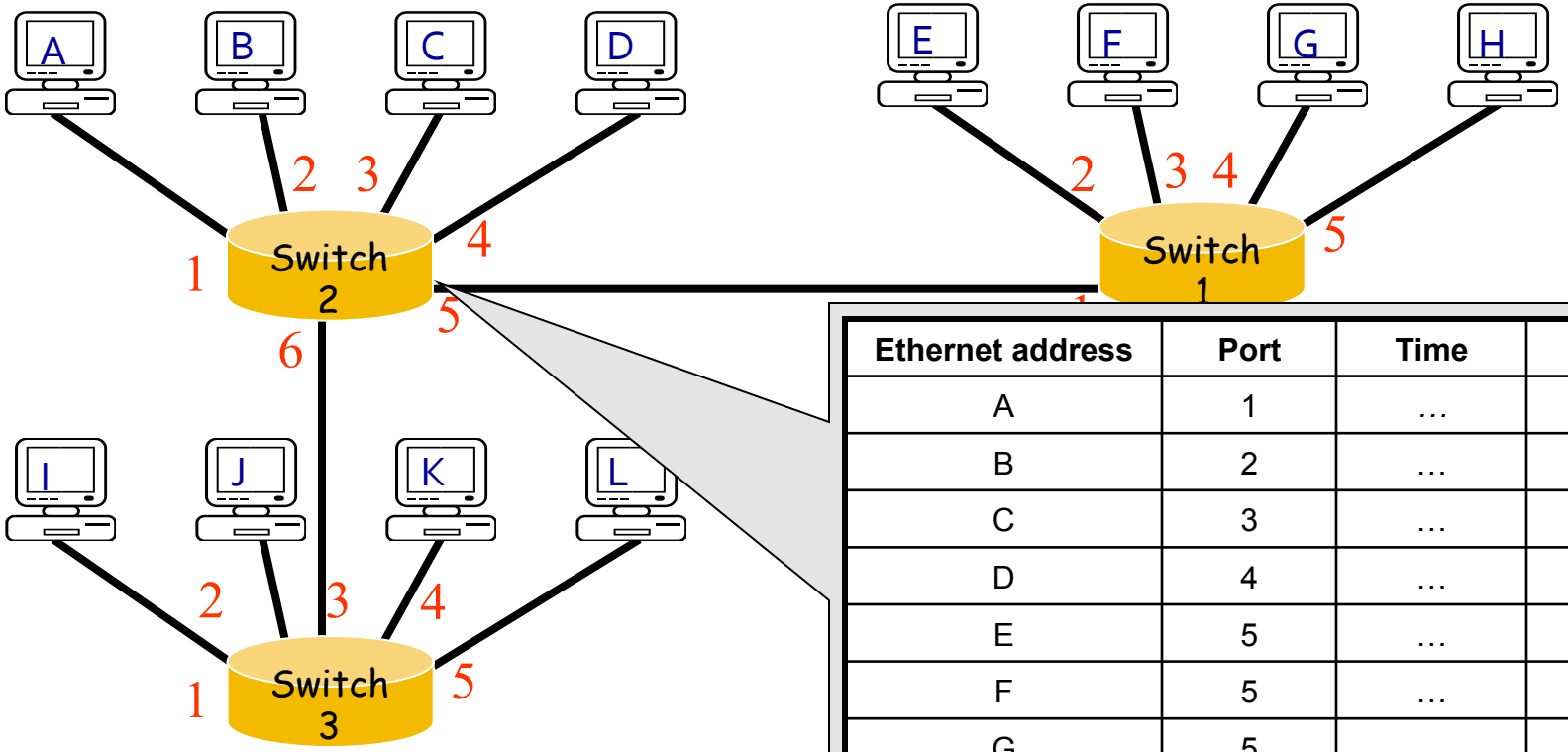
- Basic operation of Ethernet Switch
  - Examines header of each arriving frame
  - Learn that Ethernet SA is accessible from arriving port and update forwarding table
  - Examine Ethernet DA and search *Forwarding Table* on the switch
    - If in table, forward frame to the correct output port(s)
    - If not in table, broadcasts frame to all ports (except the one through which it arrived)

Switches - Learning Addresses



- Host F transmits to Host J
  - Switches 1,2,3 learn F from packet source address
  - Switches 1,2,3 broadcast to locate J
- Host J replies to F
  - F is already known – no broadcast!

Switches – Forwarding Table



Ethernet address	Port	Time	Valid
A	1	…	Yes
B	2	…	Yes
C	3	…	Yes
D	4	…	Yes
E	5	…	Yes
F	5	…	Yes
G	5	…	Yes
H	5	…	Yes
I	6	…	Yes
J	6	…	Yes
…	…	…	…

Forwarding Table Capacity

- At NewEgg in 2008:
  - $300 Netgear gigabit switch – 8000 devices
  - $1800 3Com gigabit switch – 16000 devices
  - $7500 Cisco gigabit switch – 12000 devices (but has 128MB DDR for packet buffering)
- Capacity is not infinite, but 12000+ is a lot of computers on a network without routing
  - Except, perhaps, for a large cluster computer…

Forwarding Table Maintenance

- How to remove stale entries from the table? (e.g. device leaves the network)
  - Entries expire if no communication from device within last epoch
  - 5 minute timer is default on Cisco switches

Forwarding Table Maintenance

- What if the table is full? What entry do we remove to make room for a new one?
  - Round-robin (oldest device)
    - Pros: Simple!
    - Cons: Oldest entry might be very active device
  - Least-Recently Used
    (e.g. device that last transmitted a packet a long time ago)
    - Pros: High effectiveness (device not likely to transmit again soon)
    - Cons: Complicated – Switch must count # of packets per device, and sort/ search the table to determine LRU device
  - None – Don't learn that device until a table entry expires normally. Until then, broadcast any packets destined to it
    - Pros: Simple. Ensures old (but active) devices are not evicted
    - Cons: If new devices is high traffic, entire network will suffer (due to broadcasts) until there is space in forwarding table
    - Used by Cisco switches

Challenge - Migration

- What if a network device (e.g. laptop computer) moves from one port to another? (on same switch)
  - Data is forwarded to wrong port until either:
    - Forwarding table entry expires
    - Device transmits a packet, and switch learns new port
- What if the device moves from one switch to another?
  - Have to wait for entry on old switch to expire (unless device happens to send a packet through that switch)

Challenge - Switch Congestion

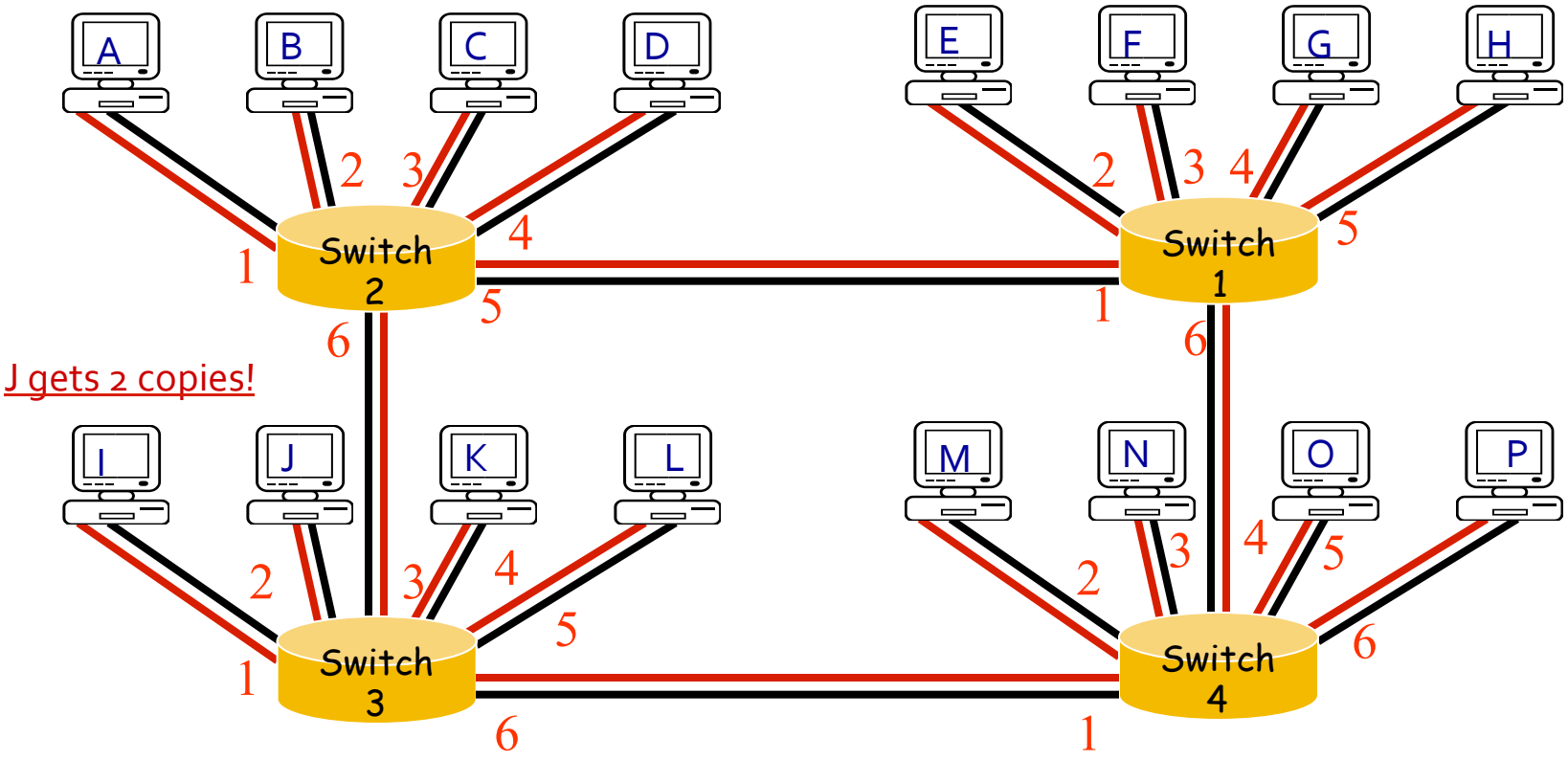
- What happens if the switch is too busy?
  - Example: Traffic from 10 input ports all heading out single output port
- Easiest solution
  - Switch drops traffic as internal buffers overflow
  - Devices don't know and keep transmitting!
    - A higher level protocol such as TCP might eventually notice and throttle back…
- Can we do better?

Ethernet Flow Control

- Required in full-duplex point-to-point operation
- Receiving node (such as a switch) can send pause request to transmitting node if it is congested, and specify time to wait before resuming transmission
- Pause request is normal Ethernet frame with special field values
  - Type: ox 8808 (Control Protocol)
  - Destination MAC: ox 01-80-C2-00-00-01
  - Data: 4 bytes – PAUSE (0x0001) + length of time to sleep (in units of time to transmit 512 bits)
    - Followed by padding to minimum Ethernet frame size
- Flow control packets are never forwarded to other ports on a switch, or to upper protocol layers
  - Purely point-to-point across single wire

Broadcasting on Topology with Redundant Paths

- Host F sends message to Host J



Problems with Loop Topology

- Broadcast Storm
  - Packets are forwarded forever
  - Ethernet has no time-to-live field
- Forwarding Table Oscillation
  - Packets from host are received via multiple ports. Table is constantly updated

Photo from http://www.extremeinstability.com/

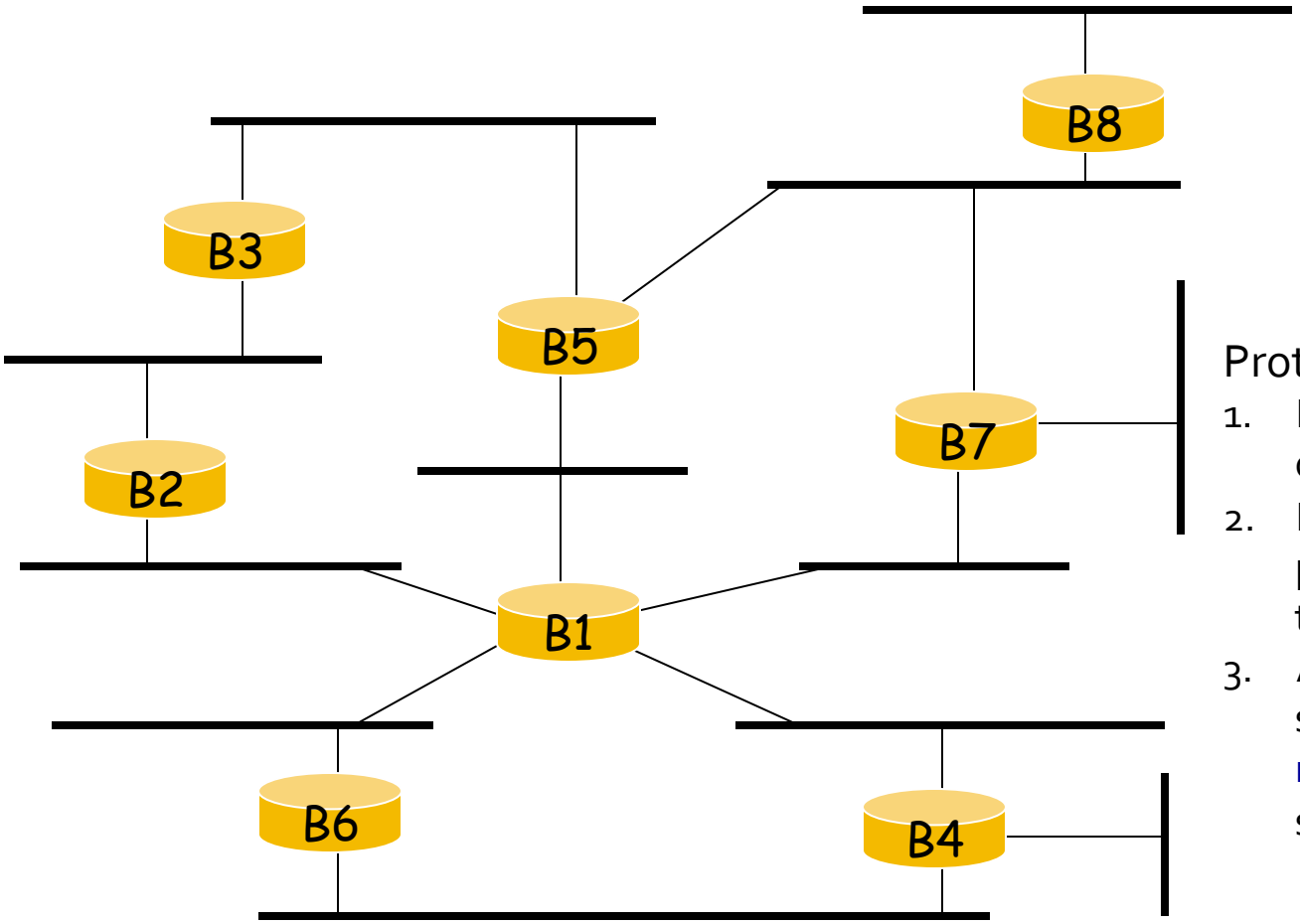
Topology Challenge – Loops!

- Can't we just avoid creating loops?
  - Redundant paths are useful for reliability
  - What if a loop is accidentally created? (Have you *seen* some of these wiring closets?)

Spanning Tree Protocol (IEEE 802.1D)

- Principles
  - Raw network is a mesh / graph
  - Create a tree from this mesh
    - Tree is a subgraph that spans all the vertices (switches) without loops
  - Disable all links not part of the tree - prevents loops!
- Features
  - Decentralized – Switches communicate among themselves via Bridge Protocol Data Units
  - Automatic – No user configuration required
  - Fault tolerant – Spanning tree will adapt if links fail (and can automatically use redundant links that were previously disabled)

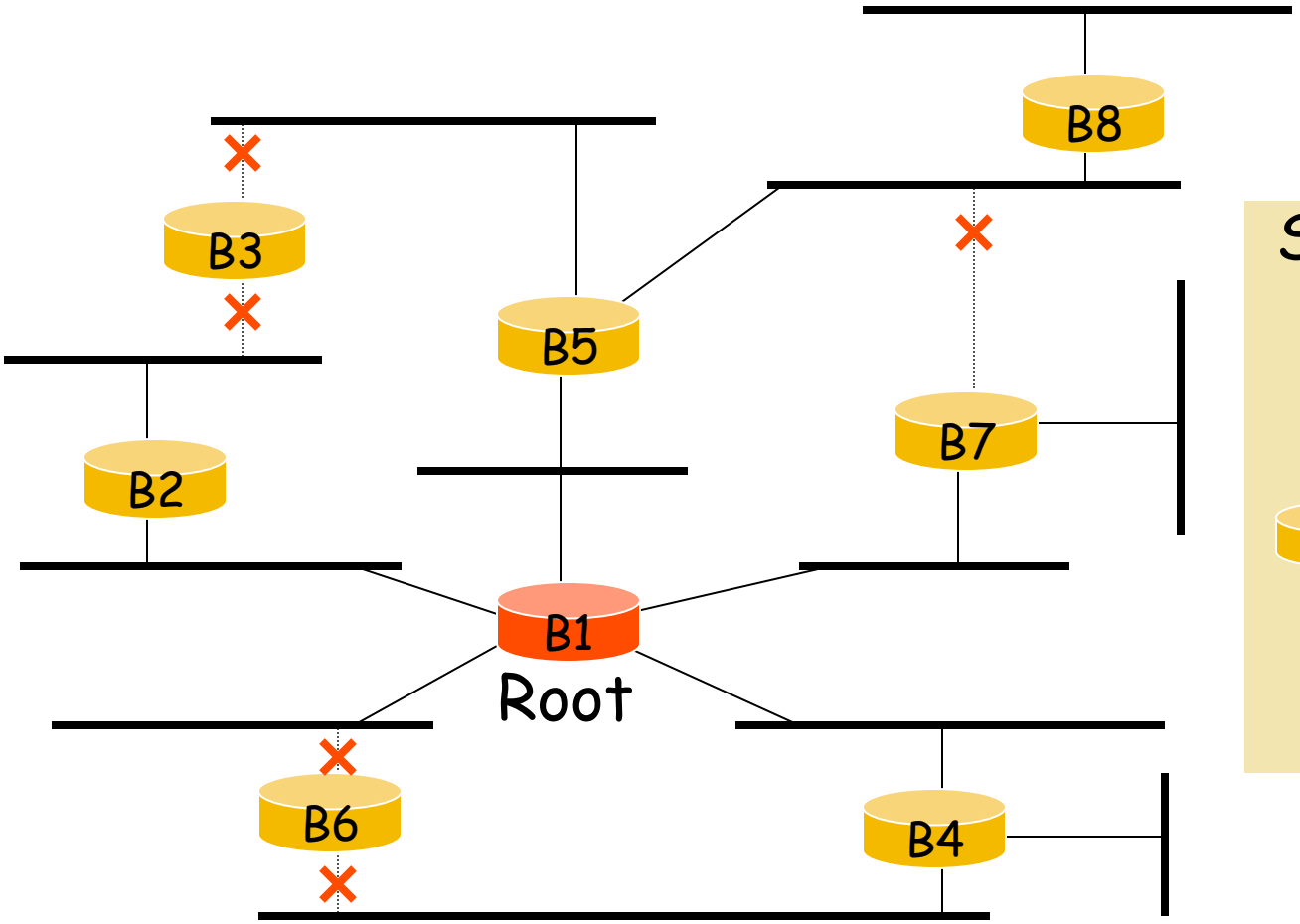
Example Spanning Tree



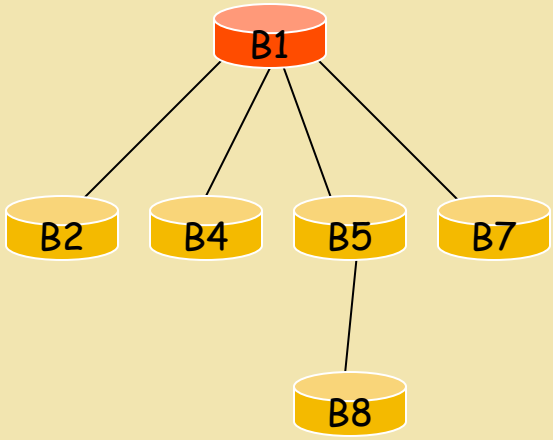
Protocol operation:

1. Pick a root. The root forwards over all its ports.
2. For each segment, pick a designated switch that is closest to the root.
3. All switches on a segment send packets towards the root via the designated switch.

Example Spanning Tree



Spanning Tree:



Spanning Tree Issues

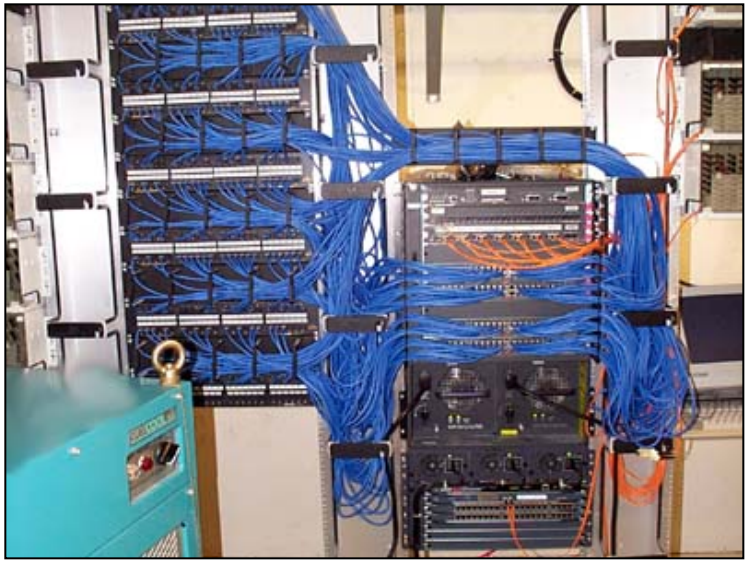
- Spanning Tree is not guaranteed to be a minimum spanning tree
  - Packets might take a longer path than necessary
  - Root switch might not be anywhere near "center" of network
- Solution?
  - Manual tweaking – Administrators can adjust device IDs to force different root

Challenge – Switch Configuration

- Problem – Each port on the switch might connect to a device running at different speed (10, 100, 1000Mbps) or duplex setting
- Do we want to configure each device manually?
  - Of course not.
- Solution: Auto-Negotiation
  - Upon power-up, each network device sends custom signals across link to other end announcing its capabilities
  - Each device listens and picks the highest mutually supported transmission mode
  - Format is backwards compatible down to 10Base-T, half duplex
- Modern switches have internal FIFOs that can buffer data between devices with varying performance capabilities
  - 1Gbps device → 100Mbps device – flow control useful!

Challenge – Device Isolation

- Imagine I have a campus network, and want to isolate a few devices on a "private network". How do I do it?
  - Buy more switches?
  - *Could get expensive…*
  - *Imagine the mess in the wiring closet…*



Challenge – Device Isolation

- Better idea – Make the switch more intelligent and have it provide device isolation
- Virtual LAN (VLAN) technology
  - Virtualizes the network – Each network device on a VLAN communicates as if they were connected to the same physical network, even if they are not
  - Can create a virtual LAN composed of machines from around the world

VLAN Overview

- Controlled by network switch
  - Each port is mapped to a VLAN
  - Forwarding / broadcast is only allowed to other ports on the same VLAN (provides isolation)
  - Spanning Tree Protocol can be run independently over each VLAN
    - Might even have different topology!
- Joining VLAN – How to assign devices?
  - Static – Port is permanently mapped to VLAN
  - Dynamic – Based on MAC address or user authentication (e.g. Cisco CleanAccess)

VLAN Operation

- Standardized format: IEEE 802.1Q
  - TCI stores VLAN ID, frame priority level, and format bits
  - CRC is recalculated