

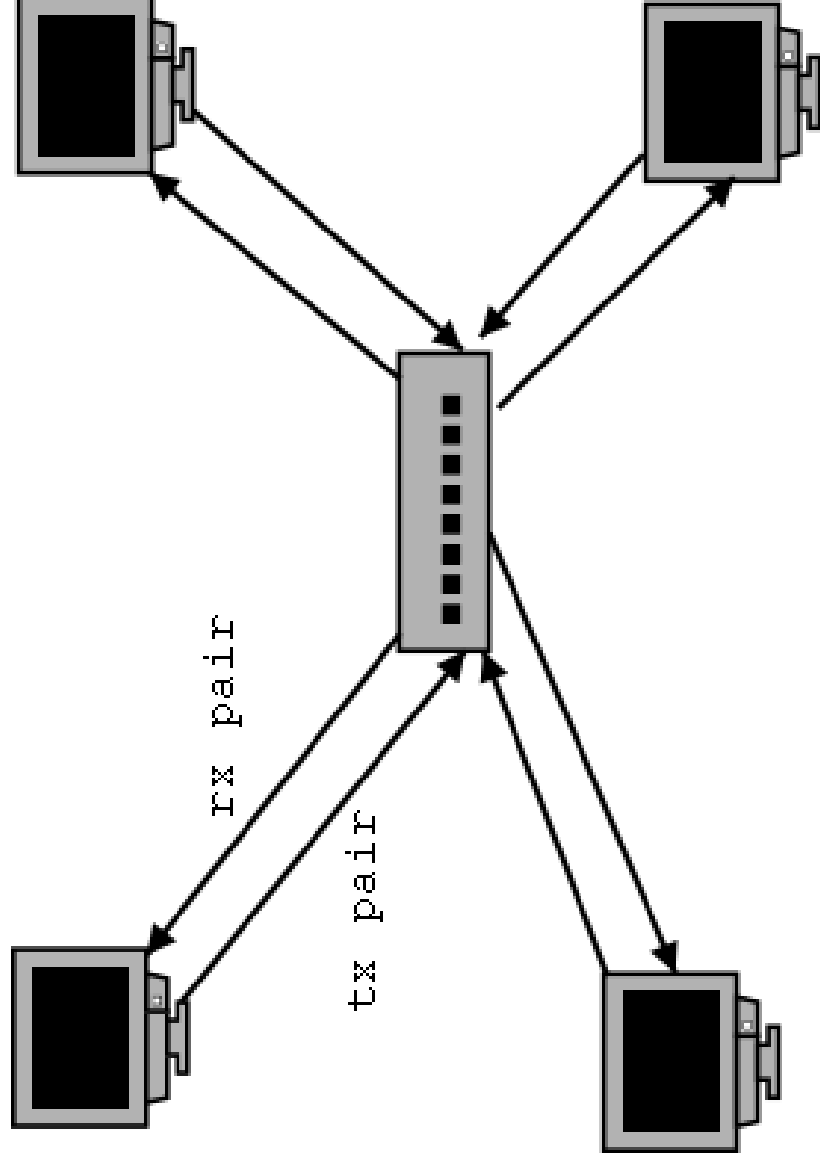
# Interconnection Technologies

## *Bridging III*

# Full duplex LAN connections

- No more *shared* LAN media/access
- Dedicated media between station and switch
- Allows simultaneous transmit and receive
- Connection becomes a point-to-point link
- Removes need for a MAC
  - Ethernet/token ring relegated to frame formats
  - No need for CS, MA, CD nor a token

# Full duplex illustrated



# Link Aggregation

- Combines parallel links to use as one
- Increases connection capacity
- Provides link redundancy
- Enables relief from *forklift* upgrades
- Often used for switch-to-switch connection
- May be used for station to switch connection
- Specified in IEEE 802.3ad
  - Yes, standard only exists for Ethernet

# Introduction to LAN multicast

- Ethernet supported multicast from the start
  - Half of 48-bit addresses are multicast addresses
  - Token ring has *functional addresses* – \*sigh\*
- Assigned to groups of apps/processes
- Allows greater communication without an incremental increase in LAN load
- Part of the IEEE 802.1p standard
- Independent of layer 3 protocol (IP multicast)

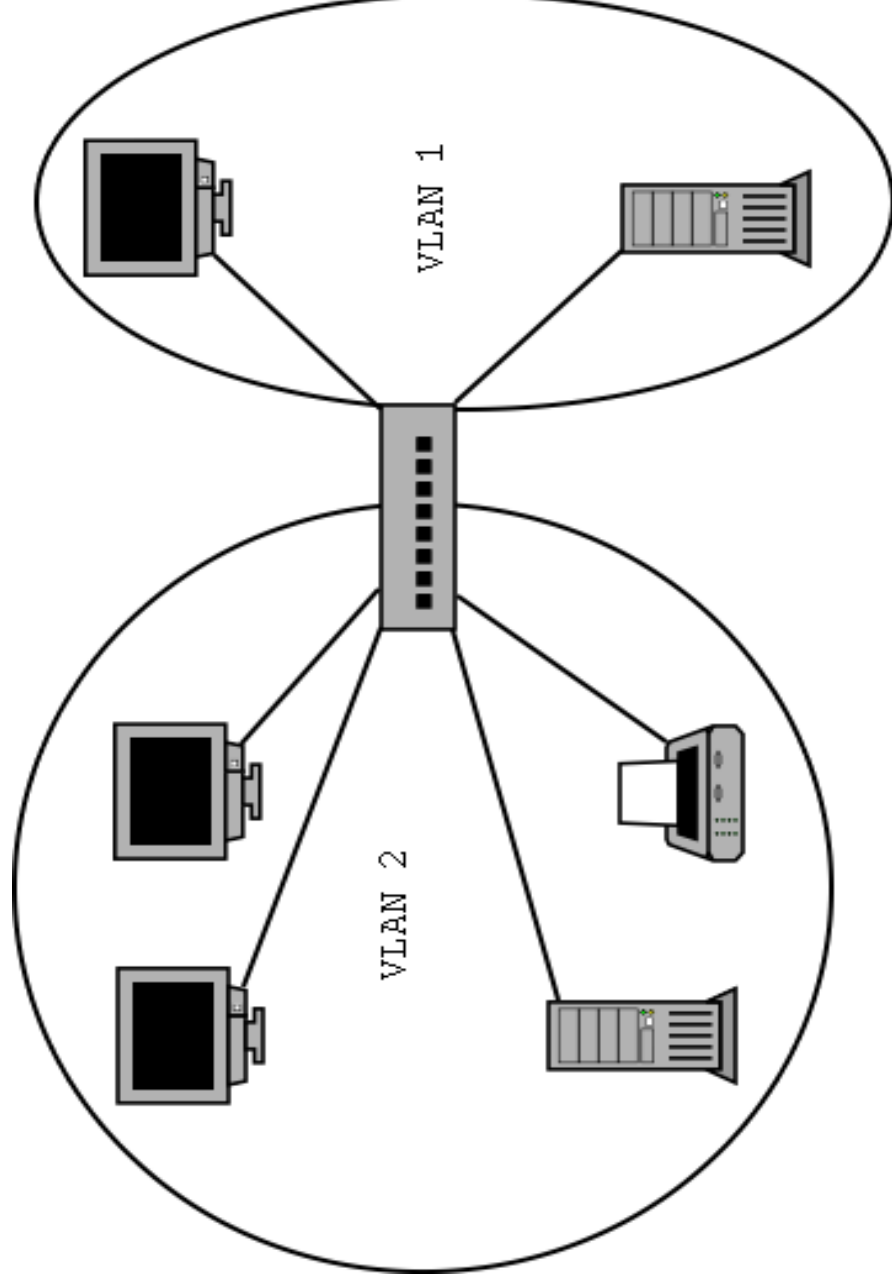
# Multicast registration and pruning

- Flooding multicast traffic is wasteful
- Need a mechanism to prune spanning tree
- Receiver sends a multicast group registration
- Switches propagate registration info
- Switches prune links that have no active multicast receivers
- We'll talk more when we get to IP multicast

# Virtual LANs

- Allows separation of physical LAN from a logical LAN
- Allows the configuration of *virtual* switches
- Each VLAN isolated from each other
- Need a layer 3 protocol to cross VLAN boundaries
- A VLAN can be configured by physical port, protocol, address, etc.

# VLANs illustrated



# VLAN frame tags

- Used to determine which VLAN a frame is on
- May be implicit or explicit
  - Implicit – parse frame base decision on contents
  - Explicit – examine a VLAN id field within frame
- End stations can be VLAN–aware
- VLANs can span multiple switches
- Often used to group hosts into a particular IP subnet for management/security reasons

# IEEE 802.1Q

- Standard for VLAN-aware switches
- Extends IEEE 802.1D standard
  - Address table must know associated VLAN
  - Includes support for frame tagging
  - Priority standard included
  - Other network management functions
- *Spanning forest* is being worked on

# IEEE 802.1p

- LAN priority tagging mechanism
- Goes hand-in-hand with 802.1Q
- Provides 8 levels of priority
  - 7 is the highest – network management
  - 0 is the default – best effort
  - 1 is the lowest – background traffic
- Not widely used
- We'll talk about class/quality of service later

# LAN switch management

- In-band
  - TELNET, SNMP, web browser
- Out-of-band
  - Modem (RS-232)
- Port mirroring
- Software/firmware
- Configuration

# Simple Network Management Protocol (SNMP)

- Manager/agent
- Management information base (MIB)
- Network protocol
- Agent MIB values can be queried, maybe set
- Objects arranged in MIB according to OID
- *SNMP traps* can be generated automatically
- *SNMPv1* most popular, but very insecure

# Remote monitoring (RMON)

- An extensive MIB for gathering switch info
  - Ethernet statistics group
  - HostTopN group
  - Matrix group
  - Packet capture group
- Internal switch RMON probes often limited
- Additional external hardware can be used to get additional functionality

# Protocol analysis

- Many LAN frame capture and analysis exist
- When you have an idea of what you're looking for
- This is the *network microscope* tool
- Privacy issues may crop up
- Very good for understanding how networks *really* work

# Protocol analysis illustrated

The screenshot displays the Wireshark interface with a packet list, packet details, and packet bytes pane. The packet list shows a sequence of ARP and TCP packets. The details pane for the selected packet (No. 2) shows Ethernet II, ARP, and IP headers. The bytes pane shows the raw data of the packet in hexadecimal and ASCII.

No.	Time	Source	Destination	Length	Protocol	Info
1	0.000000	00:e0:1e:f7:bc:40	00:20:af:d0:42:72	60	ARP	Who has 140.192.9.1? Tell 140.192.9.120
2	0.000115	00:20:af:d0:42:72	00:e0:1e:f7:bc:40	60	ARP	140.192.9.1 is at 00:20:af:d0:42:72
3	10838.452477	64:32:40:136	140.192.9.1	60	TCP	sunrpc > sunrpc [FIN, SYN] Seq=1597357078 Ack=1032676069 Min=1028 Len=0
4	10838.453270	00:20:af:d0:42:72	ff:ff:ff:ff:ff:ff	60	ARP	Who has 140.192.9.120? Tell 140.192.9.1
5	10838.454733	00:e0:1e:f7:bc:40	00:20:af:d0:42:72	60	ARP	140.192.9.120 is at 00:e0:1e:f7:bc:40
6	10838.454834	140.192.9.1	64:32:40:136	60	TCP	sunrpc > sunrpc [SYN, ACK] Seq=6153560455 Ack=1597357079 Min=32696 Len=0
7	10838.549441	64:32:40:136	140.192.9.1	60	TCP	sunrpc > sunrpc [RST] Seq=1597357079 Ack=0 Min=0 Len=0
8	10838.658676	64:32:40:136	140.192.9.1	74	TCP	4111 > sunrpc [SYN] Seq=26645237 Ack=0 Min=32120 Len=0
9	10838.658784	140.192.9.1	64:32:40:136	74	TCP	sunrpc > 4111 [SYN, ACK] Seq=611875944 Ack=26645237 Min=32120 Len=0
10	10838.759732	64:32:40:136	140.192.9.1	74	TCP	704 > sunrpc [SYN] Seq=254701368 Ack=0 Min=32120 Len=0
11	10838.759846	140.192.9.1	64:32:40:136	74	TCP	sunrpc > 704 [SYN, ACK] Seq=618011412 Ack=254701368 Min=32120 Len=0
12	10838.859180	64:32:40:136	140.192.9.1	110	PORTMAP	V2 DUMP Call XID 0x230733ec
13	10838.859404	140.192.9.1	64:32:40:136	66	TCP	sunrpc > 704 [ACK] Seq=618011413 Ack=254701413 Min=32120 Len=0
14	10838.863491	64:32:40:136	140.192.9.1	66	TCP	704 > sunrpc [ACK] Seq=254701369 Ack=618011413 Min=32120 Len=0
15	10838.863593	140.192.9.1	64:32:40:136	66	TCP	sunrpc > 704 [ACK] Seq=618011413 Ack=254701413 Min=32120 Len=0
16	10838.863593	140.192.9.1	64:32:40:136	260	BOOTP	IP, NIMD, BOOTP, VTH, 0x207222...

**Packet Details:**

- Ethernet II
  - Destination: 00:e0:1e:f7:bc:40 (rtt-cst-bs-3-f0-0, netequip.depaul.edu)
  - Source: 00:20:af:d0:42:72 (3Com\_d0:42:72)
  - Type: ARP (0x0806)
- Trailer: 03780978097809780978097809780978...
- Address Resolution Protocol (reply)
  - Hardware type: Ethernet (0x0001)
  - Protocol type: IP (0x0800)
  - Hardware size: 6
  - Protocol size: 4
  - Opcode: reply (0x0002)
  - Sender hardware address: 00:20:af:d0:42:72
  - Sender protocol address: 140.192.9.1
  - Target hardware address: 00:e0:1e:f7:bc:40
  - Target protocol address: 140.192.9.120

**Packet Bytes:**

```

0000  00 e0 1e f7 bc 40 00 20 af d0 42 72 08 06 00 01  .a.+tg.  _BBr....
0010  08 00 06 04 00 02 00 20 af d0 42 72 8c c0 09 01  .....  _BBr..A..
0020  00 e0 1e f7 bc 40 8c c0 09 78 09 78 09 78 09 78  .a.+tg.A .x.x.x.x
0030  09 78 09 78 09 78 09 78 09 78 09 78 09 78 09 78  .x.x.x.x .x.x
  
```

# Final thoughts

- Bridges/switches are simple
- You generally don't have to spend much time with them once they are initially setup
- With standard Ethernet they're commodity products
- In TCP/IP networks the really interesting stuff mostly happens at layer 3 and layer 4