Networking Test 2

**Switching and Bridging**

Definitions:

* **Circuit Switching:** A general strategy for switching data through a network. It involves establishing a dedicated path (circuit) between the source and destination. In contrast with *packet switching*
  + Most commonly used in telephone networks
  + Physical connection is set up from one end to the other
  + Data is transmitted,
  + Connection is torn down
  + **Connection-Oriented approach**:
    - Or virtual circuit
    - Requires a virtual connection from the source host to the destination host is set up before any data is sent.
    - No one else can use any of the bandwidth during an active session—even if it is idle!
* **Packet Switching:** 
  + Data link layer — single broadcast and collision domain
  + Data is sent across links when space is available
  + Known as a ***connectionless*** (or datagram) approach
  + **Connectionless (datagram) approach:** Just make sure that every packet contains enough information to enable any switch to decide how to get it to its destination.
    - Key Idea:
      * Every packet contains enough information to enable any switch to decide how to get it to destination
      * Every packet contains the complete destination address
* **Asymmetric Routing:**
  + We can have a switch with 100 attached clients at 100Mbps linked through 10Mbps to another switch with 100 clients at 100Mbps
  + Obviously not the best solution because there's contention at the interconnect
  + Bandwidth is not reserved from one end to another
  + COMMON IN PACKET SWITCHING
* **Hub:** Repeats incoming signals out of ***all*** outgoing ports (like a repeater)
  + Backbone hub interconnects LAN segments
  + Extends max distance between nodes
  + But individual segment collision domains become one large collision domain
* **Bridge:**
  + *Link layer device*
    - stores and forwards Ethernet frames
    - examines frame header and ***selectively*** forwards frame based on MAC dest address
    - when frame is to be forwarded on segment, uses CSMA/CD to access segment
  + **transparent**
    - hosts are unaware of presence of bridges
  + **plug-and-play, self-learning**
    - bridges do not need to be configured
* **Flooding:**
  + forward on all but the interface on which the frame arrived
* **Root Bridge (spanning tree protocol)**
  + The one with the lowest number
* **VLAN**
  + Or, Virtual LAN
  + IEEE 802.1Q standard
  + VLANs separate the collision domain as well as the broadcast domain
  + Hosts in each VLAN are in the same Virtual LAN
    - “Color coded”
  + “Trunks” carry multiple VLANs between switches
  + Security
    - Data on a VLAN is separated from other data
    - VLAN can span multiple switches
    - Example: Resnet
  + Flexibility
    - Now, users can connect to the closest switch and be put onto a VLAN with similar systems
  + VLAN tagged frames are carried as standard data link layer (802.3) frames
  + Type field is modified from 0x8000 to 0x8100
  + DST and SRC addresses are preserved
  + LEN/TYPE fields are modified to include the VLAN tag
  + Data field is preserved
  + TAG field adds 22 bytes to the frame
  + **VLAN** **Notes:**
    - 4096 VLANs allowed
    - Most switches only support up to 1024 VLANs
    - Spanning tree should be run on each VLAN

Short Answer:

* **Bridge advantages over hubs**
  + Hubs have Large collision domain
  + Bridges are Self-learning
  + Bridges don’t need to be configured
  + Isolates collision domains resulting in higher total max throughput
  + Limitless number of nodes and geographical coverage
  + Can connect different Ethernet types
  + Transparent (“plug-and-play”): no configuration necessary
* **Bridge self-learning algorithm**
  + A bridge has a ***bridge table***
  + Entry in bridge table:
    - (Node LAN Address, Bridge Interface, Time Stamp)
    - Stale entries in table dropped (TTL can be 60 min)
  + Bridges learn which hosts can be reached through which interfaces
    - When frame received, bridge “learns” location of sender: incoming LAN segment
    - Records sender/location pair in bridge table
  + Forwarding/Filtering
  + ***When bridge receives a frame:***
  + index bridge table using MAC dest address

**if** entry found for destination **{**  
 **if** dest on segment from which frame arrived  
 drop the frame

**else** forward the frame on interface indicated

**}**

**else** flood

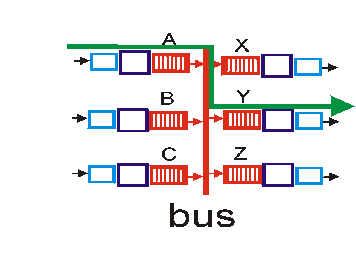
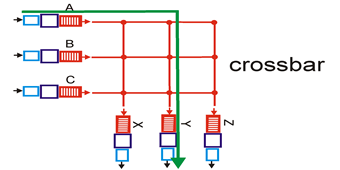
* **Need for Spanning Tree Algorithm**
  + For increased reliability, desirable to have redundant, alternative paths from source to dest
    - But we don’t want the loop problem!
  + Solution: organize bridges in a spanning tree by disabling subset of interfaces
* **Be able to construct the spanning tree by looking at a network diagram**
  + Don’t have to be able to write out all the configuration messages

**Routers**

Definitions

* **Router:** are nodes that interconnect networks
  + Often called *gateways*
  + Works with IP addresses
  + Connects heterogeneous networks based off of different data link protocols
* **Queueing Delay:** 
  + Happens when fabric is slower than input ports combined -> queueing may occur at input queues
  + queueing delay and loss due to input buffer overflow!
* **Types of Switching Fabrics:**
  + Memory
  + Bus
  + Crossbar

Short Answer

* **Types of switching fabrics: Memory, bus, crossbar**
  + Memory:
    - 
    - First generation packet switching devices:
      * packet copied by system’s (single) CPU
      * speed limited by memory bandwidth (2 bus crossings per datagram)
    - Modern packet switching devices
      * Input port processor performs lookup, copy into memory
      * Cisco Catalyst 8500
  + Bus:
    - 
    - Datagram from input port memory to output port memory via a shared bus
    - ***Bus contention***: Switching speed limited by bus bandwidth
    - 1 Gbps bus, Cisco 1900: sufficient speed for access and enterprise routers (not regional or backbone)
  + Crossbar
    - 
    - Matrix of pathways that can be configured to connect any input port to any output port
    - Biggest problem is that they require output port to accept packets from all inputs at once
      * Implying each port has memory bandwidth equal to total switch throughput
    - In reality, more complex designs
* **Routers Compared to Switches**
  + Bridges Pros:
    - Bridge operation is simpler requiring less packet processing
    - Bridge tables are self-learning
  + Bridges Cons:
    - All traffic confined to spanning tree, even when alternative bandwidth is available
    - Bridges do not offer protection from broadcast storms
  + Routing Pros:
    - Arbitrary topologies can be supported, cycling is limited by TTL counters (and good routing protocols)
    - Provide protection against broadcast storms
  + Routing Cons:
    - Require IP address configuration (not plug and play)
    - Require higher packet processing
* **Four Basic Components to Router Architecture:**
  + Input Ports
  + Output ports
  + Switching Fabric
  + Routing processor

**Forwarding**

Definitions

* **Forwarding Table:** 
  + To decide how to forward a packet, a switch consults a forwarding table
  + 
* **VCI (Virtual Circuit Identifier):** 
  + Something that uniquely identifies the connection of the entry in the VC table at this switch, and which will be carried inside the header of the packets that belong to this connection

Short Answers

* Identifying full host-to-host virtual circuit connections
  + Examples in hw
* Constructing datagram forwarding tables
  + Examples in hw

Definitions

* Internetwork
  + An arbitrary collection of networks interconnected to provide some sort of host-host packet delivery service
* IP packet delivery model
  + Connectionless, best-effort
* Maximum Transmission Unit (MTU): Each network has some Unit, MTU
  + Ethernet (1500 bytes), FDDI (4500 bytes)