Virtual Circuit vs. Datagram

- **Virtual Circuit**
  - Need to wait a full RTT for connection setup before sending data
  - Connection request contains full address; subsequent data packets contain only circuit identifier
  - A link or switch fails? tough, need to re-establish path
  - Connection setup provides opportunity to reserve resources

- **Datagram:**
  - No connection setup – can start communication immediately
  - No way to tell if a path exists
  - Packets are routed independently, it is possible to route around failures
  - Every packet must carry full destination information

- **Source Routing**
So what does a switch look like anyways?

- Can be built from a general-purpose computer
- Usually specialized high-performance switches towards the core of the Internet (will look at switch design later)

Learning Bridges

- Idea: if a bridge sees a frame sent by node X coming on port y, it learns that the way to X is through port y
  - What if it sees a frame destined to X appear on y?
  - What if it sees a frame for X appear on a port z?
  - What if it sees a frame it does not know?
- Dynamically constructs a routing table
- Any problems with this scheme?

Bridges/LAN Switches

- Suppose we want to build an ethernet network bigger than 1024 hosts or using more than 3 populated segments in a row
- Can we use a LAN switch?
  - Also known as bridges
- A bridge/LAN switch that operates on link-layer frames (as opposed to network layer packets)
- Idea of bridge is to transparently extend the LAN
  - Is this another way of saying repeater?
Aside – Network Caches

- The “routing table” is a cache where information is dynamically kept
- Caches are used to optimize protocol performance
- Entries in the cache are tagged with a Time to Live (TTL)
  - Periodically, the TTL value on all entries is reduced by 1; an entry expires if its TTL reaches 0
  - Each time a frame is seen from a destination, its entry TTL is set to some maximum value
- Cache has a fixed size (to conserve space and make indexing more efficient)
  - What happens when you see a frame from a new node and the table is full?

Problem – Loops

- Problem – the network can contain loops
  - Learning can go wrong
  - Frames can get stuck indefinitely in the network
- What can be done?
  - Rely on administrator to statically eliminate loops?

Implementation

- All bridges initially assume they are root
- They send out configuration messages on all ports
  - Config. message: (bridge id, distance, root)
- When bridge receives a config message with a better config:
  - It updates its interface information
  - Adds 1 to distance and sends new config. message on all other ports
- A configuration “A” is better than “B” if
  - It identifies a root with a smaller id
  - The root has an equal id, but shorter distance
  - Root and distance are equal but bridge id is smaller
- Bridge only forwards messages on interfaces if the configuration lists them as designated bridge
Example

- B3 receives (2, 0, 2) from 2; accepts 2 as root (2 < 3)
- B3 sends (3, 1, 2) to 5
- B5 receives (1, 0, 1) from 1 and sends (5, 1, 1) to 3
- B3 accepts root as 1, realizes B2 and B5 are closer, stops forwarding on both interfaces

Discussion

- Easy to administer, but optimal?
- Bridge ids have to be unique
- What happens if a bridge or link fails?
  - Heartbeat messages/TTL
- What if two LAN segments are different speed? (e.g., 10Mbit ethernet and 100Mbit ethernet)
- What if two segments use different Link protocols (e.g., FDDI and Ethernet)?
- What to do with broadcast and multicast packets?
  - Can bridge learn multicast groups?
- Transparency (an extended LAN appearing as a regular LAN) is dangerous – why?
- Why not build the Internet out of bridges/switched LANs?

Limitations of Bridges

- Routing is suboptimal – costly for large networks
- No mechanism to impose hierarchy
- Broadcast floods the whole network
- No support for heterogeneity

Discussion – Virtual Circuit Switching?

- Why? Seems counter to the Internet Architecture papers perspective?
- Phone companies (Telecoms) powerful players, they wanted to play a role in data transport
- ATM (Asynchronous Transfer Mode) was the results
  - VC (to better support voice)
  - Fixed packet size (cell switching) for simpler faster router design
  - Small cell size (to better support voice and provide more pipelined use of network)
  - Because cell size is small, IP doesn’t fit in it
    - Add another layer (Application Adaptation Layer) to provide bigger frame size
    - Heavily used for WANs, but not popular for LANs
Virtual LANs

- Logically configure parts of the extended LAN to form a virtual LAN
- Packets sent on one VLAN segment will be visible to all segments of the VLAN (but nowhere else)
- When a packet arrives at a bridge port with a host id as sender, the bridge inserts the VLAN id associated with that port
- Advantage: Enhances scalability – why?
- Disadvantage: Requires administrator intervention

Asynchronous Transfer Mode (ATM)

- Virtual Circuit cell-switched technology
- Packets are fixed size and small
- Why fixed size?
  - Cell switching more efficient: simpler; allows parallelism
  - Conducive to Quality of Service guarantees
- What size?
  - If large, low utilization for small messages
  - If small, high header-to-payload ratio (overhead) and more cells to process
- Why small?
  - Share the link at a finer grain; can preempt for a high-priority packet
  - Store and forward more efficient with small packets; close to cut through
  - Don’t have to delay too much to fill a cell with voice samples

Cell Format

- User-Network Interface (UNI) – host to switch format
- Generic Flow Control (GFC): Generic Flow Control, for arbitrating access for shared medium
- VPI (Virtual Path Identifier) and VCI (Virtual Circuit Identifier), used to identify the virtual connection
- Type: management, congestion control, AAL5 (later)
- CLP: Cell Loss Priority; if 1, ok to drop on congestion
- HEC: Header Error Check (CRC-8) among switches
- Replace GFC with 4 more bits of VPI

What is Ahead

- Internet Protocol (IP) – the solution to the limitations of bridges
- Before then
  - ATM – important “cell switched” technology. Not directly compatible with IP
  - Example of VC switching

SUNY-BINGHAMTON – CS528 FALL ’07 LEC. #9
**Segmentation and Reassembly**

- For Data networks, upper protocols deal with variable size/larger size packets – think sockets
  - Need a segmentation and reassembly (SAR) capability
- Implemented in the ATM Adaptation Layer (AAL)
  - Different AALs defined for different types of traffic
  - AALs also implement the services exported to the application
  - AAL 1 and 2 (fixed rate); AAL 3/4 (packet oriented); AAL 5 (improvement on AAL 3/4)

---

**AAL 3/4**

- Convergence Sublayer Protocol Data Unit (CS-PDU)
- CPI – Common Part Indicator (version)
- BTAG/ETAG – beginning and ending tags
- BSize: hint on the amount of buffering to allocate (why not actual size?)
- PAD: user data padded to multiple of 3 bytes
- Length: size of the full PDU
- How to translate this PDU to ATM cells?

---

**Conversion to Cells – Add AAL Header**

- Type used to indicate whether the cell is:
  - BOM: Beginning of Message
  - COM: Continuation of Message
  - EOM: End of Message
  - SSM: Single Segment Message
- SEQ: 4 bit sequence number to protect against lost/misordered cells (what if 16 consecutive cells are lost?)
- MID: Message ID is used to multiplex link among multiple messages
- Length: Length of the full PDU
AAL 5

- PAD so that the trailer always falls at the end of ATM cell
- Length is the size of the Data
- CRC-32 detects missing or misordered cells
- Cell Format:
  - Very simple: use a single end of PDU bit in ATM header Type field
  - Full cell payload (48 bytes) used for data
- What do we lose/gain vs. AAL 3/4?
- AAL5 has taken over as the de facto standard

ATM for LANs

- Attractive technology before switched LANs arrived
  - High speed
  - Switched technology – can run long distances
- Difficult to make it behave like a LAN – consider broadcasts needed to resolve addresses

LANE – LAN Emulation

- Provides a service interface that makes ATM circuit look Ethernet or token ring
- Main functions of LANE
  - Resolve MAC addresses to ATM address
  - Support broadcasts
- Problem how to emulation a shared medium using a virtual circuit network
  - How is this different from switched LANs?
LANE

- Simple “centralized” protocol
- Clients register with a centralized server and use it for translation (using a well known address)
- A Broadcast server implements broadcasts
- To communicate with a host with unknown address
  - Sent a packet to LES and first packet to BUS
  - LES responds with ATM address, BUS broadcasts the packet
  - VC can now be established, further packets through VC

Internetworking

- We know how to build directly connected networks
- Switches have the potential of scaling to large networks
- Bridges (link-level switching) provided valuable lessons
  - Not scalable (consider spanning tree, or addressing scheme)
  - Not heterogeneous (same MAC address family; compatible payloads, etc.)
- Scalability: must scale indefinitely
  - Addressing and Routing
  - Multicast and broadcast?
- Heterogeneity:
  - Users on different networks must be able to talk
  - Might need to cross several other networks on the way

Internet Protocol (IP)

- Runs on all hosts (remember the “hour-glass” shape of the Internet Protocol Suite)
- Provides isolation from the networking technology
- Must provide one service model that is common to all possible underlying technologies – what should we use?
  - Connectionless best-effort delivery

IP Packet Format

- Version: 4 for IPv4
- Hlen: number of 32-bit words in the header
- TOS: Type of Service; can be used for QoS
- Length: Number of bytes in the datagram
- Ident/Flags/Offset: used for fragmentation and reassembly
- TTL: Time to Live (maximum hop count allowed; why?)
- Protocol: key to identify higher level protocol (e.g., TCP)
- Checksum: applied to header (why not CRC?)
- Source and destination addresses (what addresses?)
- Options
- Payload
- Where does the “link-layer” header fit?
Fragmentation and Reassembly – Why?

- Why is this needed?
- Each network has some Maximum Transmission Unit (e.g., ethernet 1500 bytes; PPP 532 bytes)
  - Restrict IP payload to the smallest MTU; or
  - Use fragmentation and reassembly if necessary

- Strategy:
  - Fragment when necessary (MTU < Datagram)
  - Fragments are self-contained IP packets
  - Try to avoid fragmentation at the source
  - Refragmentation is possible
  - Delay reassembly until destination
  - What if a fragment is lost?

Fragmentation and Reassembly – How?

- Ident is the packet sequence number
- M-bit (flags field) is 1 in all but the last fragment
- How would refragmentation be implemented?
- “path MTU discovery” to minimize fragmentation