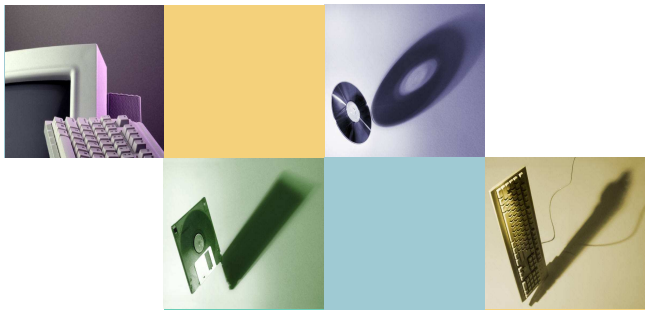
CSE123A discussion session

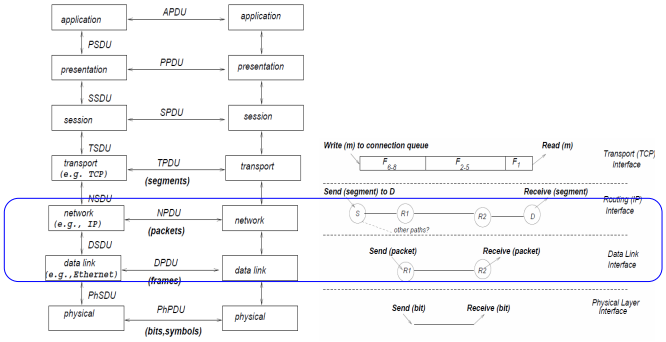


2007/03/02 Ryo Sugihara

Topics

- Review
  - Data Link Layer (7): Bridges
    - Interconnecting LANs
    - Bridge implementation
    - Bridges vs. Routers
  - Network Layer (1): Introduction
    - IP address
      - Terminology
      - Classful addressing
      - Classless addressing: CIDR

Where are we now?



Interconnecting LANs

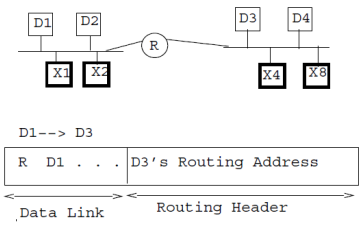
- Why?
  - Limited distance (2.5km)
  - Limited number of stations
- Requirements
  - "Transparency"
    - End stations cannot tell whether they are on a single LAN or bridged LAN
  - Bandwidth
    - We want to keep bandwidth unaffected
- Several ways to interconnect...

Repeater?

- "Repeater"
  - Works at Physical layer (Layer 1)
  - Just amplify the signal
    - Less frequent errors
- Problems
  - Does not remove the limitations
    - Limitations on distance and # of stations
    - Ethernet allows only 4 repeaters

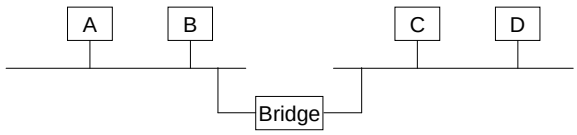
Router?

- "Router"
  - (details later)
  - Works at Network layer (Layer 3)
    - i.e. Reads (up to) layer-3 header
- Problem
  - Not transparent
    - Need to specify the destination in routing header
      - e.g.) D1 → (R) → D3



Bridge!

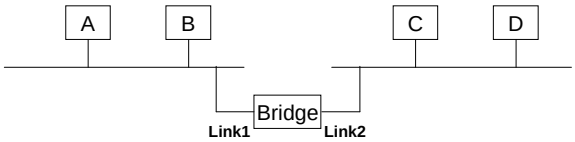
- "Bridge"
  - Works at Data Link layer (layer 2)
  - Read every frame ("promiscuous mode") and forward it to other LANs



- Good for small-scale interconnection
  - Transparent
  - Efficient bandwidth usage
    - By using "learning bridge"

Learning bridge

- Idea: Avoid wasting bandwidth
  - Bridge don't need to forward a frame sent from A to B

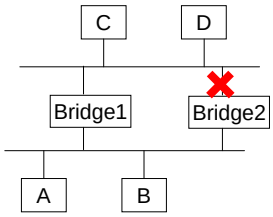


- How?
  - By learning which station is connected to which interface of the bridge
    - From the source address of all frames
    - Find forwarding interface by looking up the table

Station	Link
A	1
B	1
C	2
D	2

Limitations of bridge

- No cycles
  - Why?



- Spanning tree algorithm
  - Disable some ports to eliminate cycles

Toward large-scale interconnections

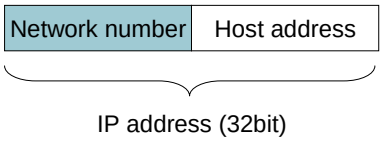
- Bridges are **bad** for
  - Heterogeneous links (e.g. Ethernet & TokenRing)
    - Incompatible address
    - Incompatible Max packet size
    - Incompatible bandwidth
  - Large-scale networks
    - Flat address (i.e. not hierarchical)
      - Table size gets very big: Need one entry for each station
        - (We want to use "group of stations" in each table entry)
    - Spanning tree is not efficient
      - Allows only one path for each destination
- Need for hierarchical addressing scheme
  - Realized in higher layer: "Network layer"

Topics

- Review
  - Data Link Layer (7): Bridges
    - Interconnecting LANs
    - Bridge implementation
    - Bridges vs. Routers
  - Network Layer (1): Introduction
    - IP address
      - Terminology
      - Classful addressing
      - Classless addressing: CIDR

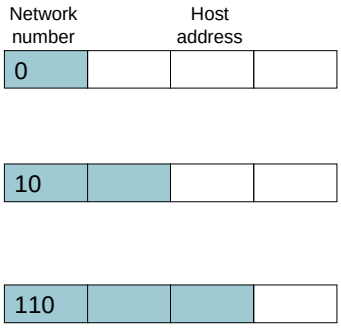
IP address

- IP address
  - 32bit
    - cf.) MAC address: 48bit
  - Globally unique
    - except “private address”
  - **Hierarchical addressing**
    - “Close addresses are close on the network”
- Terminology
  - “Network number”
    - (“Network address”)
    - (“Network prefix”)
  - “Host address”
  - “IP address”
    - Network number + Host address



Classful addressing

- Class A
  - 8bit network number
  - up to 2^24 = 16777216 hosts
  - 0.0.0.0/8 – 127.0.0.0/8
- Class B
  - 16bit network number
  - up to 2^16 = 65536 hosts
  - 128.0.0.0/16 – 191.0.0.0/16
- Class C
  - 24bit network number
  - up to 2^8 = 256 hosts
  - 192.0.0.0/24 – 223.0.0.0/24



- Notation
  - Network number = 10000001 00000001
    - (“1000000100000001*”)
    - “128.1.0.0/16”
    - “128.1.0.0, subnet mask = 255.255.0.0”

Problems of Classful addressing

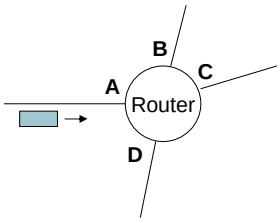
- Background: Lots of small networks
  - Rise of PCs & Ethernet
  - Need for many network numbers
- Problems:
  - Depletion of Class B address
    - Class C is too small, so give Class B address
    - But Class B is too big: a lot of waste
    - Solution: CIDR scheme (next)
  - Depletion of entire address space
    - Due to waste
    - Solution: IPv6, private address with NAT
      - See the lecture note for details

Classless addressing

- Idea: More flexible addressing
  - For more efficient use of address space
- CIDR (Classless Inter-Domain Routing) scheme
  - "Subnetting"
    - Divide one Class B address to multiple ranges
      - e.g.) 128.1.0.0/16 → 128.1.0.0/17, 128.1.128.0/17
  - "Supernetting"
    - Combine multiple (consecutive) Class C addresses to one range
      - e.g.) 192.0.0.0/24, 192.0.1.0/24 → 192.0.0.0/23

Router at work

- Forwarding algorithm
  - For each incoming packet,
    - See the destination IP address
    - Lookup the routing table,
      - Find the <u>longest matching prefix</u> P
      - Get the associated link L
    - Forward the packet to link L
      - If no prefix matches, forward on "default route"
- Example:
  - 5bit address
  - Packets incoming from link A
    - dest = "01100"
    - dest = "11010"
    - dest = "10110"



<u>Routing table</u>

Prefix	Link
001*	A
0*	B
01*	C
11*	C
110*	D
default	B

BACKUP