# Virtual Circuit Switching

Chapter 3

Virtual Circuit Switching

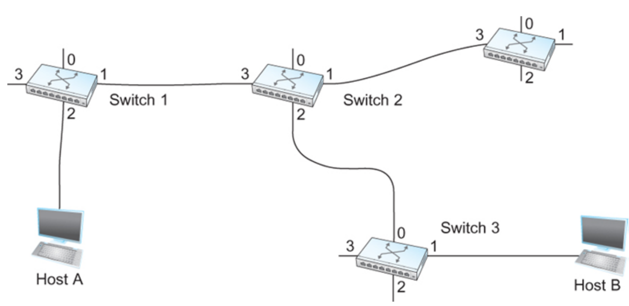
- Widely used technique for packet switching
- Uses the concept of *virtual circuit* (VC)
- Also called a connection-oriented model
- First set up a virtual connection from the source host to the destination host and then send the data

MK MORGAN KAUFMANN 1

# Virtual Circuit Switching

Chapter 3

- Host A wants to send packets to host B




MK MORGAN KAUFMANN 2

## Virtual Circuit Switching

Chapter 3

Two-stage process

- Connection setup
- Data Transfer

- Connection setup
  - Establish "connection state" in each of the switches between the source and destination hosts
  - The connection state for a single connection consists of an entry in the "VC table" in each switch through which the connection passes



## Virtual Circuit Switching

Chapter 3

One entry in the VC table on a single switch contains

- A virtual circuit identifier (VCI)
  - Uniquely identifies the connection at this switch for the link
  - Carried inside the header of the packets that belong to this connection
- An incoming interface on which packets for this VC arrive at the switch
- An outgoing interface in which packets for this VC leave the switch
- A potentially different VCI that will be used for outgoing packets for the same virtual circuit

- Incoming interface(port) and packet VCI indicate which outgoing interface(port) and VCI the switch needs to use in the packet

# Virtual Circuit Switching



Note:

- The combination of the VCI of the packets as they are received at the switch and the interface on which they are received uniquely identifies the virtual connection
- There may be many virtual connections established in the switch at one time
- Incoming and outgoing VCI values are not generally the same
  - VCI is not a globally significant identifier for the connection;
  - rather it has significance only on a given link ←
- Whenever a new connection is created, we need to assign a new VCI for that connection on each link that the connection will traverse
  - We also need to ensure that the chosen VCI on a given link is not currently in use on that link by some existing connection.



# Virtual Circuit Switching



Two broad classes of approach to establishing a virtual circuit

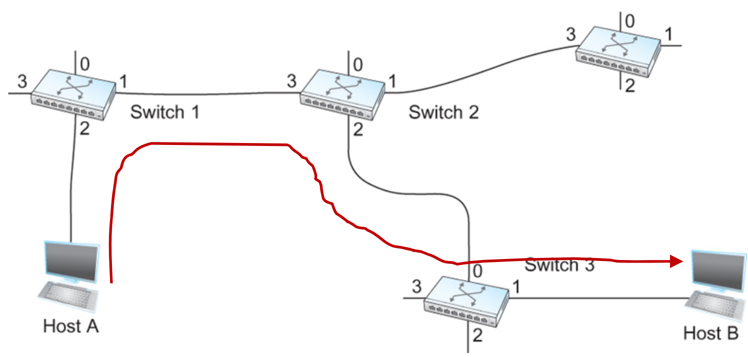
- Network Administrator will configure the state
  - The virtual circuit is permanent (PVC)
  - The network administrator can delete this
  - Can be thought of as a long-lived or administratively configured VC
- A host can send messages into the network to cause the state to be established
  - This is referred as signaling and the resulting virtual circuit is said to be switched (SVC)
  - A host may set up and delete such a VC dynamically without the involvement of a network administrator

# Virtual Circuit Switching

Chapter 3

Let's assume that a network administrator wants to manually create a new virtual connection from host A to host B

- First the administrator identifies a path through the network from A to B




return



# Virtual Circuit Switching

Chapter 3

The administrator then picks a VCI value that is currently *unused* on each link for the connection

- For our example,
  - Suppose the VCI value 5 is chosen for the link from host A to switch 1
  - 11 is chosen for the link from switch 1 to switch 2
  - So the switch 1 will have an entry in the VC table

Switch 1 VC table

| Incoming Interface | Incoming VCI | Outgoing Interface | Outgoing VCI |
|---|---|---|---|
| 2 | 5 | 1 | 11 |

Slide 7 Return

# Virtual Circuit Switching

Chapter 3

Similarly, suppose

- VCI of 7 is chosen to identify this connection on the link from switch 2 to switch 3
- VCI of 4 is chosen for the link from switch 3 to host B
- Switches 2 and 3 are configured with the following VC table

Switch 2 VC table

| Incoming Interface | Incoming VCI | Outgoing Interface | Outgoing VCI |
|---|---|---|---|
| 3 | 11 | 2 | 7 |

Switch 3 VC table

| Incoming Interface | Incoming VCI | Outgoing Interface | Outgoing VCI |
|---|---|---|---|
| 0 | 7 | 1 | 4 |

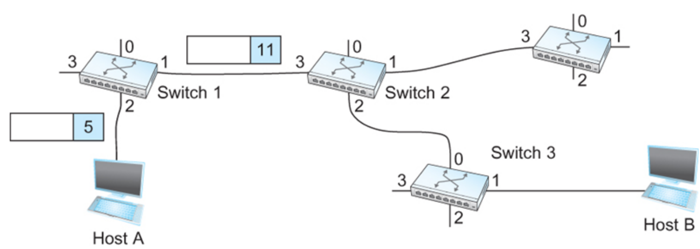
Slide 7 return



# Virtual Circuit Switching

Chapter 3

- For any packet that A wants to send to B, A puts the VCI value 5 in the header of the packet and sends it to switch 1
- Switch 1 receives any such packet on interface 2, and it uses the combination of the interface and the VCI in the packet header to find the appropriate VC table entry.
- The table entry on switch 1 tells the switch to forward the packet out of interface 1 and to put the VCI value 11 in the header

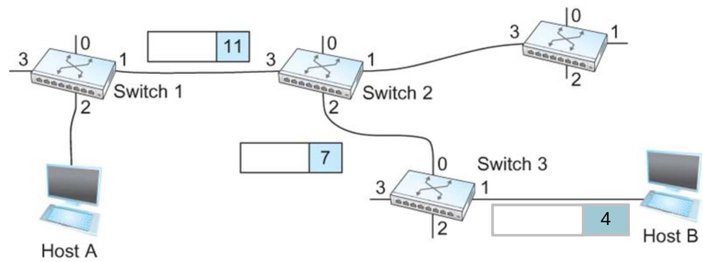



Slide 9

# Virtual Circuit Switching

Chapter 3

- Packet will arrive at switch 2 on interface 3 bearing VCI 11
- Switch 2 looks up interface 3 and VCI 11 in its VC table and sends the packet on to switch 3 after updating the VCI value appropriately
- This process continues until it arrives at host B with the VCI value of 4 in the packet
- To host B, this identifies the packet as having come from host A




Slide 9



# Virtual Circuit Switching

Chapter 3

- In real networks of reasonable size, the burden of configuring VC tables correctly in a large number of switches would quickly become excessive
  - Thus, some sort of signaling is almost always used, even when setting up “permanent” VCs
  - In case of PVCs, signaling is initiated by the network administrator
  - Switched Virtual Circuits (SVCs) are usually set up using signaling by one of the hosts

## Virtual Circuit Switching

Chapter 3

- To start the signaling process, host A sends a setup message into the network (i.e. to switch 1)
  - The setup message contains (among other things) the complete destination address of B.
  - The setup message needs to get all the way to B to create the necessary connection state in every switch along the way
  - It is like sending a datagram to B where every switch knows which output to send the setup message so that it eventually reaches B
  - Assume that every switch knows the topology of the network
- When switch 1 receives the connection request, in addition to sending it on to switch 2, it creates a new entry in its VC table for this new connection
  - The entry is exactly the same shown in the previous table
  - Switch 1 picks the value 5 for this connection (Note no outgoing VCI)

| Incoming Interface | Incoming VCI | Outgoing Interface | Outgoing VCI |
|---|---|---|---|
| 2 | 5 | 1 | |

13

## Virtual Circuit Switching

Chapter 3

- How does the signaling work (contd.)
  - When switch 2 receives the setup message, it performs the similar process and it picks the value 11 as the incoming VCI

| Incoming Interface | Incoming VCI | Outgoing Interface | Outgoing VCI |
|---|---|---|---|
| 3 | 11 | 2 | |

  - Similarly switch 3 picks 7 as the value for its incoming VCI (slide 9)

| Incoming Interface | Incoming VCI | Outgoing Interface | Outgoing VCI |
|---|---|---|---|
| 0 | 7 | 1 | |

  - Each switch can pick any VCI it likes, as long as that number is not currently in use for some other connection on that port of that switch
  - Finally the setup message arrives at host B.
  - Assuming that B is healthy and willing to accept a connection from host A, it allocates an incoming VCI value, in this case 4.
    - This VCI value can be used by B to identify all packets coming from A

14

## Virtual Circuit Switching

Chapter 3

- Now to complete the connection, everyone needs to be told what their downstream neighbor is using as the VCI for this connection
  - Host B sends an acknowledgement of the connection setup to switch 3 and includes in that message the VCI value that it chose (4)
  - Switch 3 completes the VC table entry for this connection and sends the acknowledgement on to switch 2 specifying the VCI of 7
  - Switch 2 completes the VC table entry for this connection and sends acknowledgement on to switch 1 specifying the VCI of 11
  - Finally switch 1 passes the acknowledgement on to host A telling it to use the VCI value of 5 for this connection

MORGAN KAUFMANN 15

## Virtual Circuit Switching

Chapter 3

- When host A no longer wants to send data to host B, it tears down the connection by sending a teardown message to switch 1
- The switch 1 removes the relevant entry from its table and forwards the message on to the other switches in the path which similarly delete the appropriate table entries
- At this point, if host A were to send a packet with a VCI of 5 to switch 1, it would be dropped as if the connection had never existed

MORGAN KAUFMANN 16

# Virtual Circuit Switching



- Characteristics of a Virtual Circuit
  - Connection request requires at least one RTT of delay, so there is a delay before data is sent.
  - Connection request contains the full address for host B (which might be quite large, being a global identifier on the network),
  - Each data packet contains only a small identifier, which is only unique on one link.
    - The per-packet overhead caused by the header is reduced relative to the datagram model
  - If a switch or a link in a connection fails, the connection is broken and a new one will need to be established.
    - Also the old one needs to be torn down to free up table storage space in the switches
  - The issue of how a switch decides which link to forward the connection request on has similarities with the function of a routing algorithm



# Virtual Circuit Switching



- Good Properties of Virtual Circuits
  - By the time the host gets the go-ahead to send data, it knows quite a lot about the network-
    - that there is a route to the receiver and
    - that the receiver is willing to receive data
  - It is also possible to allocate resources to the virtual circuit at the time it is established

## Virtual Circuit Switching

Chapter 3

- For example, an X.25 network – a packet-switched network that uses the connection-oriented model – employs the following three-part strategy
  - Buffers are allocated to each virtual circuit when the circuit is initialized
  - The sliding window protocol is run between each pair of nodes
    - augmented with the flow control to keep the sending node from overrunning the buffers allocated at the receiving node
  - The circuit is rejected by a given node if not enough buffers are available at that node when the connection request message is processed

MK MORGAN KAUFMANN 19

## Virtual Circuit Switching

Chapter 3

- Comparison with the Datagram Model
  - Datagram network has no connection establishment phase and each switch processes each packet independently
  - Each arriving packet competes with all other packets for buffer space
  - If there are no buffers, the incoming packet must be dropped
- In Virtual Circuits, we could imagine providing each circuit with a different quality of service (QoS)
  - The network gives the user some kind of performance related guarantee
    - Switches set aside the resources they need to meet this guarantee
      - A percentage of each outgoing link's bandwidth
      - Delay tolerance on each switch
- Most popular VC technologies are X.25, Frame Relay and ATM
- Virtual circuit technologies have lost popularity
- Replaced by the internet connectionless model

MK MORGAN KAUFMANN 20

## Asynchronous Transfer Mode



- ATM (Asynchronous Transfer Mode)
  - Connection-oriented packet-switched network
  - Packets are called cells (53 bytes total)
    - 5 byte header + 48 byte payload
  - Fixed length packets are easier to switch in hardware
    - Simpler to design
    - Enables parallelism



## Asynchronous Transfer Mode



- ATM Cell Structure
  - User-Network Interface (UNI) (Host-to-Switch format)
    - GFC: Generic Flow Control
    - VPI: Virtual Path Identifier
    - VCI: Virtual Circuit Identifier
    - Type: management, congestion control
    - CLP: Cell Loss Priority
    - HEC: Header Error Check (CRC-8)

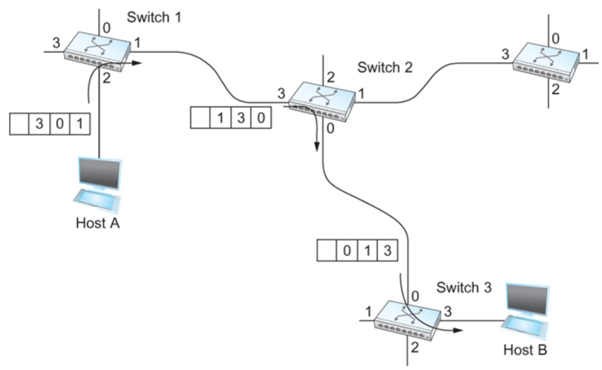
| 4 | 8 | 16 | 3 | 1 | 8 | 384 (48 bytes) |
|---|---|---|---|---|---|---|
| GFC | VPI | VCI | Type | CLP | HEC (CRC-8) | Payload |

  - Network-Network Interface (NNI) (Switch-to-Switch format)
    - GFC becomes part of VPI field

## Source Routing

Chapter 3

- Source Routing
  - All the information about network topology that is required to switch a packet across the network is provided by the source host
  - Header contains next port to use by receiving switch

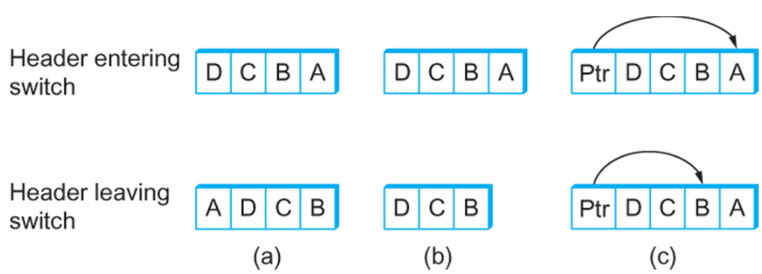



MK MORGAN KAUFMANN 23

## Source Routing

Chapter 3

- Three other approaches in Source Routing




(a) – header is rotated
(b) – header is stripped
(c) – pointer is used

MK MORGAN KAUFMANN 24

## Bridges and LAN Switches

Chapter 3

- Bridges and LAN Switches
  - Class of switches that is used to forward packets between shared-media LANs such as Ethernets
    - Known as LAN switches
    - Referred to as Bridges
  - Suppose you have a pair of Ethernets that you want to interconnect
    - **One approach is put a repeater in between them**
      - It might exceed the physical limitation of the Ethernet
      - No more than four repeaters between any pair of hosts
      - No more than a total of 2500 m in length is allowed
    - **Alternatively, put a node between the two Ethernets**
      - This node is called a **Bridge**
      - The bridge forwards frames from one Ethernet to the other
      - A collection of LANs connected by one or more bridges is usually said to form an **Extended LAN**

MK MORGAN KAUFMANN 25

## Bridges and LAN Switches

Chapter 3

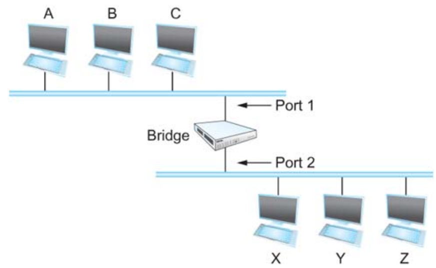
- Simplest Strategy for Bridges
  - Accept LAN frames on their inputs and forward them out to all other outputs
  - Used by early bridges
- Learning Bridges
  - Observe that there is no need to forward all the frames that a bridge receives

MK MORGAN KAUFMANN 26

## Bridges and LAN Switches



- Consider the following figure
  - When a frame from host A that is addressed to host B arrives on port 1, there is no need for the bridge to forward the frame out over port 2.



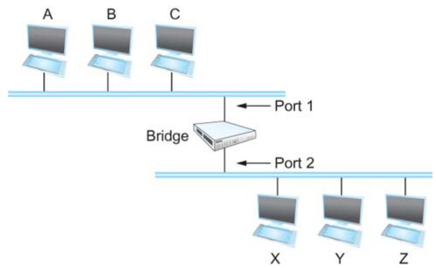

  - How does a bridge come to learn on which port the various hosts reside?



## Bridges and LAN Switches

Chapter 3

- Solution
  - Download a table into the bridge




Bridge Routing Table

| Host | Port |
|---|---|
| A | 1 |
| B | 1 |
| C | 1 |
| X | 2 |
| Y | 2 |
| Z | 2 |

  - Who does the download?
    - Human - Too much work for maintenance

## Bridges and LAN Switches

Chapter 3

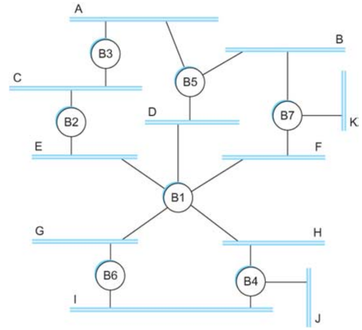
- Can the bridge learn this information by itself?
  - Yes – Learning Bridge
- How?
  - Each bridge inspects the source address in all the frames it receives
  - Record the information at the bridge and build the table
  - When a bridge first boots, this table is empty
  - Entries are added over time
  - A timeout is associated with each entry
  - The bridge discards the entry after a specified period of time
    - To protect against cases where a host moves from one network to another
- If the bridge receives a frame that is addressed to host not currently in the table
  - Forward the frame out on all other ports

29

## Bridges and LAN Switches

Chapter 3

- Strategy works fine if the extended LAN does not have a loop in it
- Why? Frames potentially loop through the extended LAN forever




  - Bridges B1, B4, and B6 form a loop

30

## Bridges and LAN Switches

Chapter 3

- How does an extended LAN come to have a loop in it?
  - Network is managed by more than one administrator
    - For example, it spans multiple departments in an organization
    - It is possible that no single person knows the entire configuration of the network
    - A bridge that closes a loop might be added without anyone knowing
  - Loops are built into the network to provide redundancy in case of failures
- Solution
  - Distributed Spanning Tree Algorithm

31