Agenda

- Upcoming deadlines and due dates
- Today’s class
  - Chapter 2
    - Links, frames, Ethernet
  - Chapter 3
    - Switching
      - Datagrams
      - Virtual circuits
      - Source routing

Due Dates and Last Class

- Programming Assignment I (due March 29,11:59PM)
- Term Paper Email (due February 5, 11:59PM)
- Term Paper Proposal (due February 19, 11:59PM)
- **Midterm Examination (March 12)**
- Full term Paper (due May 6, 11:59PM)
- Last Class
  - Performance
    - Bandwidth, Latency, Delay x Bandwidth
    - In-class examples

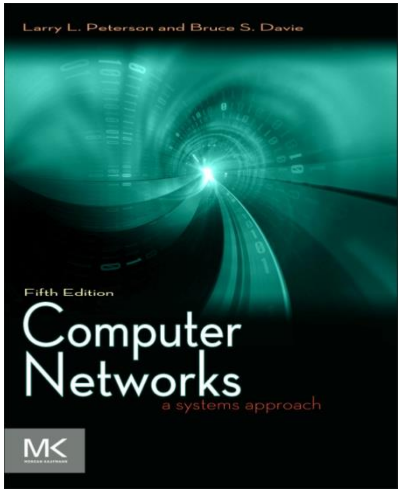
Term Paper: What to and not to do

- Encryption/Steganography for computer networks, successes, failures and the future
  - Which method, how is this related to networks?
- Computer networks in gaming, how they enhance player experience and the emergence of eSports
  - Too general, make it more specific on the issues you want to address (e.g., latency)
- Google and social networks, and how their pervasive collection of personal data can undermine your security.
  - Data leakage in social network, what type of data leakage, etc.
  - Using Social Networks as a Side-Channel
- I want to choose “attacks on browsers and networks’
  - Scripting attacks {Specificity}

Term Paper: What to and not to do

- "End-to-End Routing"
  - Will be covered in class.
- "Security Challenges in Software Defined Networks"
  - May be too broad, can you identify one security challenge and propose solutions, or critique existing countermeasures, etc.
  - A Research Paper on Network Security Issue in Business
  - Too wide, will turn out to be a survey paper

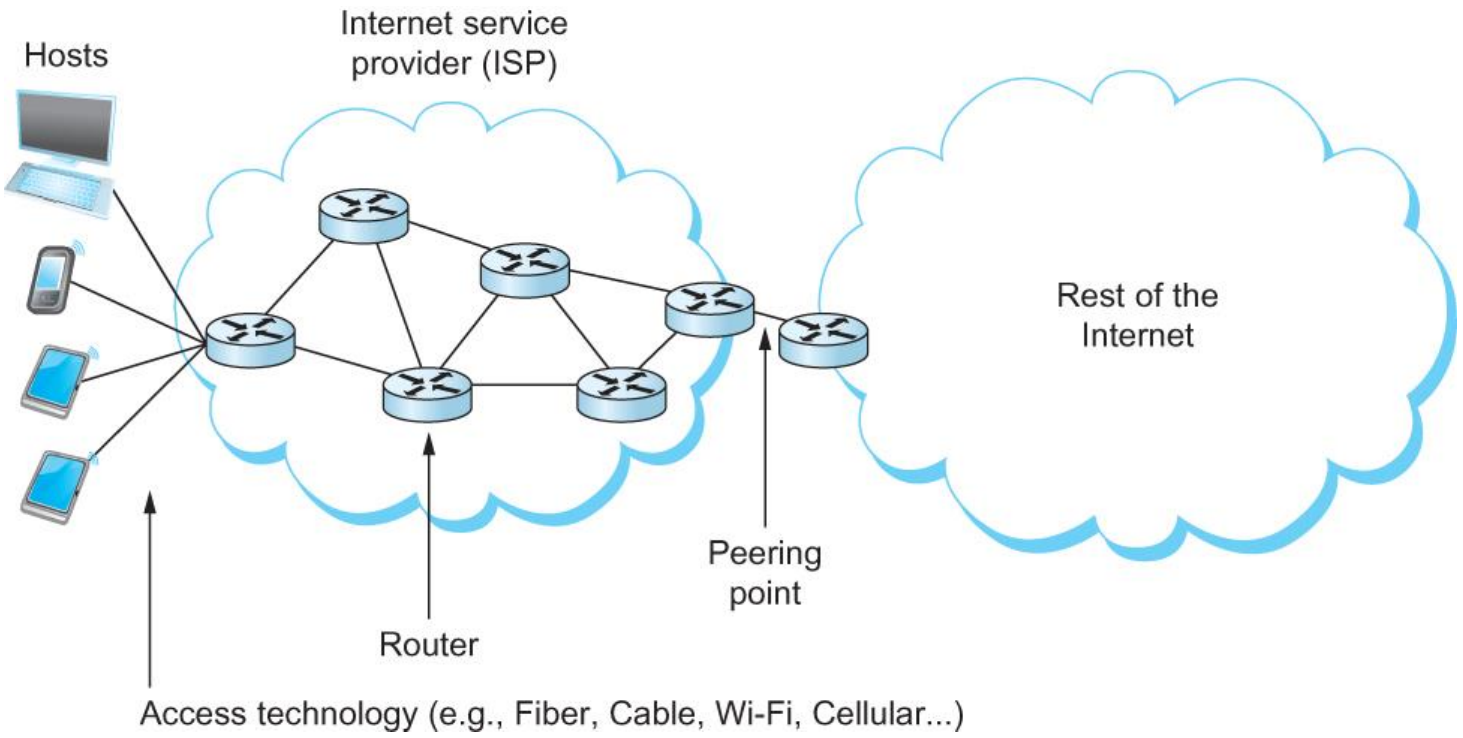
SEARCH AND READ IEEE, ACM, and USENIX PAPERS!!!



Chapter 2

Getting Connected

Perspectives on Connecting

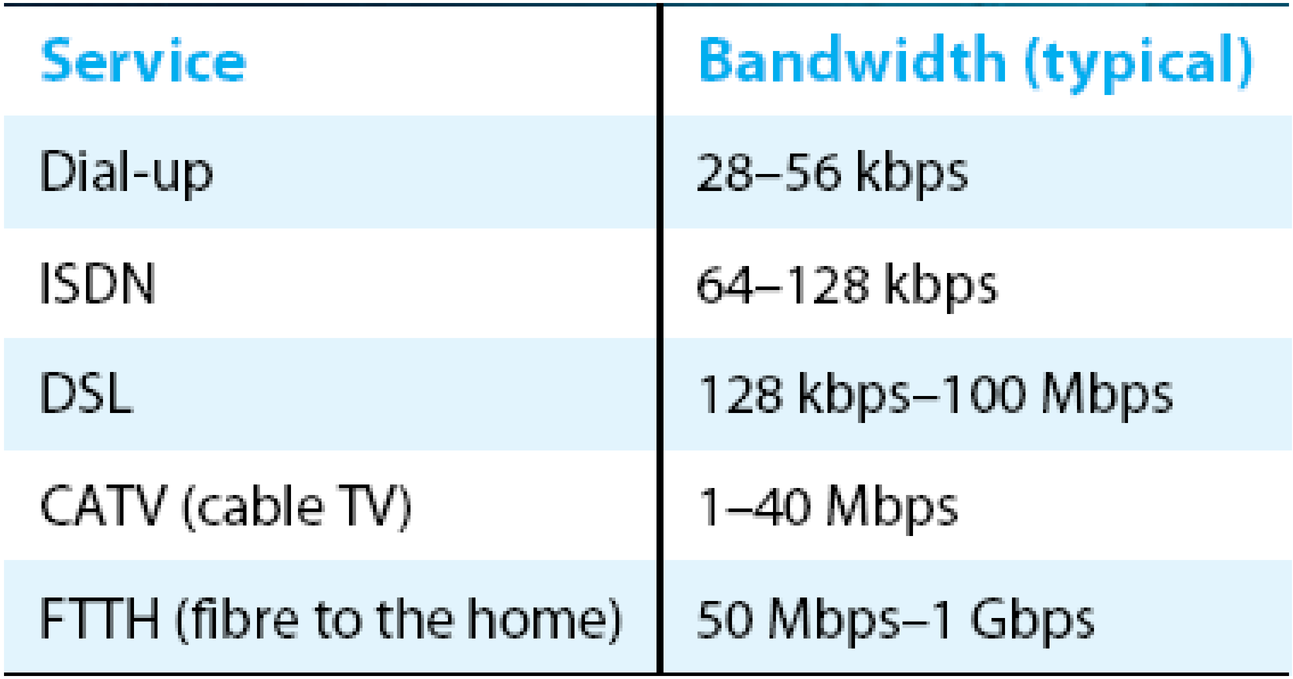

Hosts
Internet service provider (ISP)
Rest of the Internet
Peering point
Router
Access technology (e.g., Fiber, Cable, Wi-Fi, Cellular...)

An end-user’s view of the Internet

Links

- All practical links rely on some sort of electromagnetic radiation propagating through a medium or, in some cases, through free space
- One way to characterize links, then, is by the medium they use
  - Typically copper wire in some form (as in Digital Subscriber Line (DSL) and coaxial cable),
  - Optical fiber (as in both commercial fiber-to-the home services and many long-distance links in the Internet' s backbone), or
  - Air/free space (for wireless links)

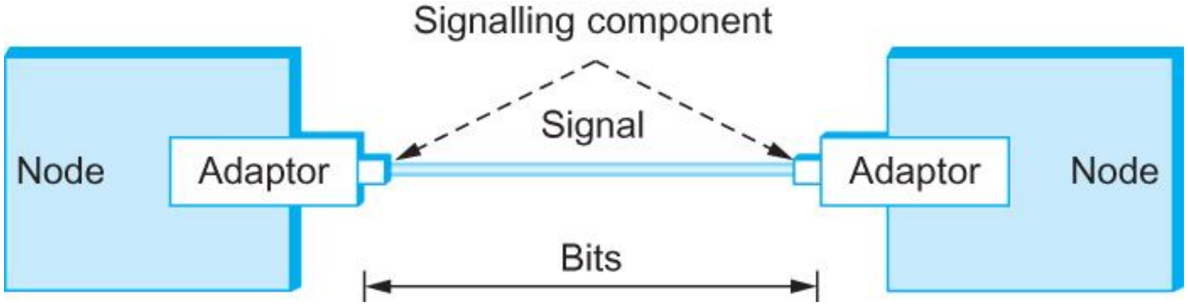
Links



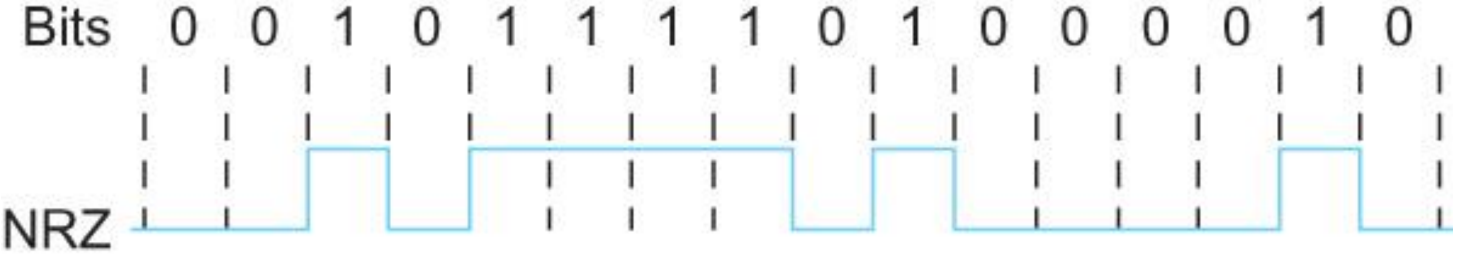
Service	Bandwidth (typical)
Dial-up	28–56 kbps
ISDN	64–128 kbps
DSL	128 kbps–100 Mbps
CATV (cable TV)	1–40 Mbps
FTTH (fibre to the home)	50 Mbps–1 Gbps

Common services available to connect your home

Encoding



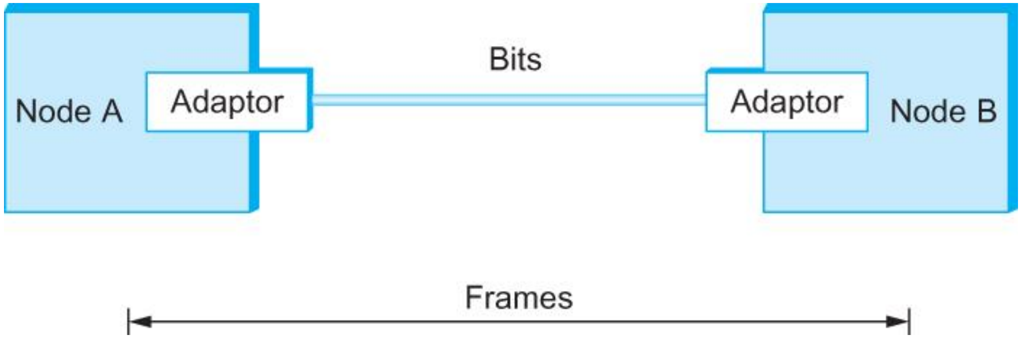
Signals travel between signaling components; bits flow between adaptors



NRZ encoding of a bit stream

Framing

- We are focusing on packet-switched networks, which means that blocks of data (called *frames* at this level), not bit streams, are exchanged between nodes.
- It is the network adaptor that enables the nodes to exchange frames.



Bits flow between adaptors, frames between hosts

Framing

- When node A wishes to transmit a frame to node B, it tells its adaptor to transmit a frame from the node's memory. This results in a sequence of bits being sent over the link.
- The adaptor on node B then collects together the sequence of bits arriving on the link and deposits the corresponding frame in B's memory.
- Recognizing exactly what set of bits constitute a frame—that is, determining where the frame begins and ends—is the central challenge faced by the adaptor

Framing

- Byte-oriented Protocols
  - To view each frame as a collection of bytes (characters) rather than bits
  - BISYNC (Binary Synchronous Communication) Protocol
    - Developed by IBM (late 1960)
  - DDCMP (Digital Data Communication Protocol)
    - Used in DECNet

Ethernet

- Most successful local area networking technology of last 20 years.
- Developed in the mid-1970s by researchers at the Xerox Palo Alto Research Centers (PARC).
- Uses CSMA/CD technology
  - Carrier Sense Multiple Access with Collision Detection.
  - A set of nodes send and receive frames over a shared link.
  - Carrier sense means that all nodes can distinguish between an idle and a busy link.
  - Collision detection means that a node listens as it transmits and can therefore detect when a frame it is transmitting has collided with a frame transmitted by another node.

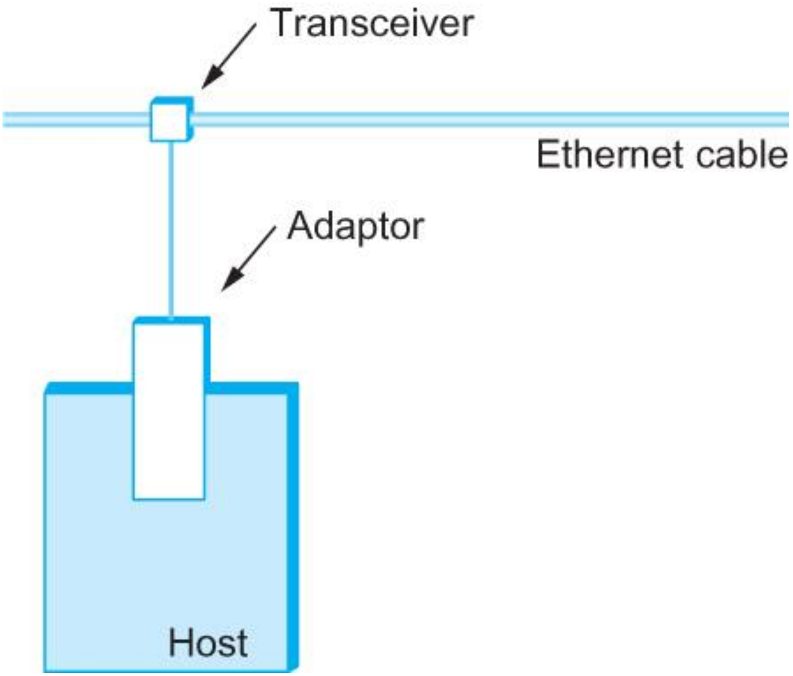
Ethernet

- Uses ALOHA (packet radio network) as the root protocol
  - Developed at the University of Hawaii to support communication across the Hawaiian Islands.
  - For ALOHA the medium was atmosphere, for Ethernet the medium is a coax cable.
- DEC and Intel joined Xerox to define a 10-Mbps Ethernet standard in 1978.
- This standard formed the basis for IEEE standard 802.3
- More recently 802.3 has been extended to include a 100-Mbps version called Fast Ethernet and a 1000-Mbps version called Gigabit Ethernet.

Ethernet

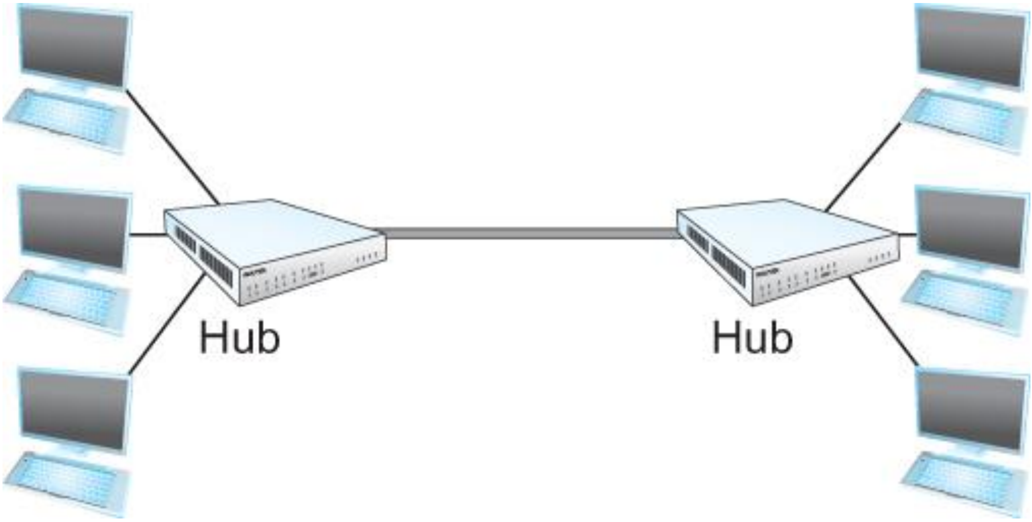
- An Ethernet segment is implemented on a coaxial cable of up to 500 m.
  - This cable is similar to the type used for cable TV except that it typically has an impedance of 50 ohms instead of cable TV's 75 ohms.
- Hosts connect to an Ethernet segment by tapping into it.
- A transceiver (a small device directly attached to the tap) detects when the line is idle and drives signal when the host is transmitting.
- The transceiver also receives incoming signal.
- The transceiver is connected to an Ethernet adaptor which is plugged into the host.
- The protocol is implemented on the adaptor.

Ethernet



Ethernet transceiver and adaptor

Ethernet

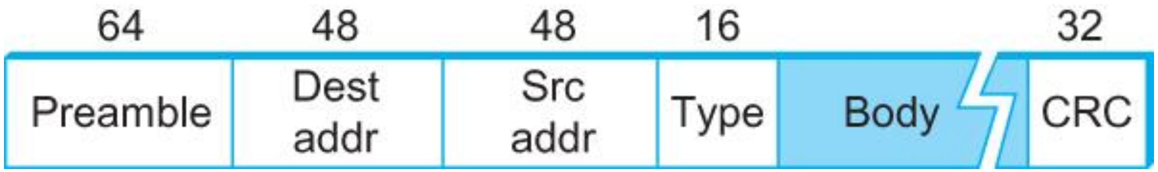


Ethernet Hub

Access Protocol for Ethernet

- The algorithm is commonly called Ethernet’s Media Access Control (MAC).
  - It is implemented in Hardware on the network adaptor.
- Frame format
  - Preamble (64bit): allows the receiver to synchronize with the signal (sequence of alternating 0s and 1s).
  - Host and Destination Address (48bit each).
  - Packet type (16bit): acts as demux key to identify the higher level protocol.
  - Data (up to 1500 bytes)
    - Minimally a frame must contain at least 46 bytes of data.
    - Frame must be long enough to detect collision.
  - CRC (32bit)

Ethernet Frame

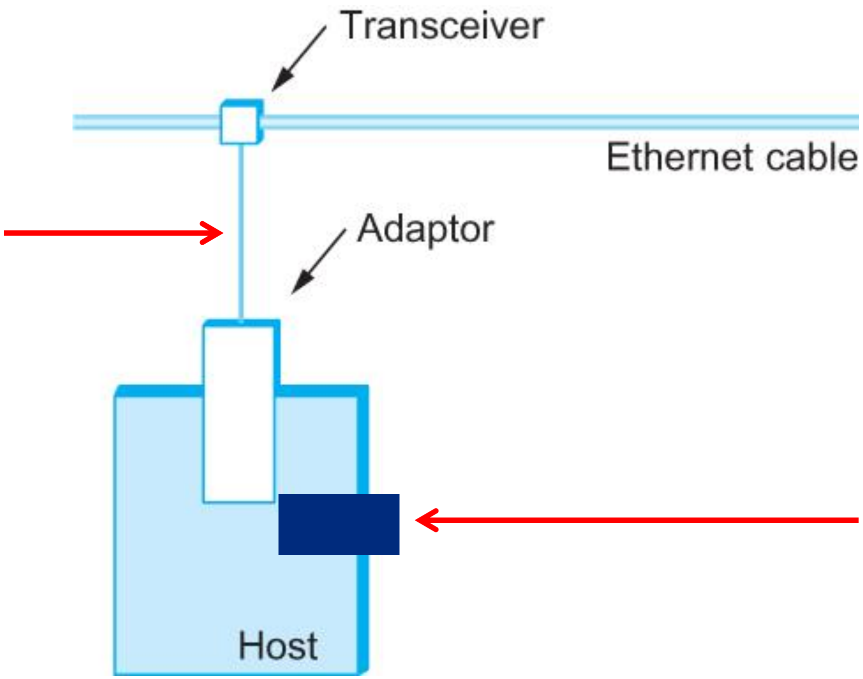


Ethernet Frame Format

Frame versus Packet

FRAME

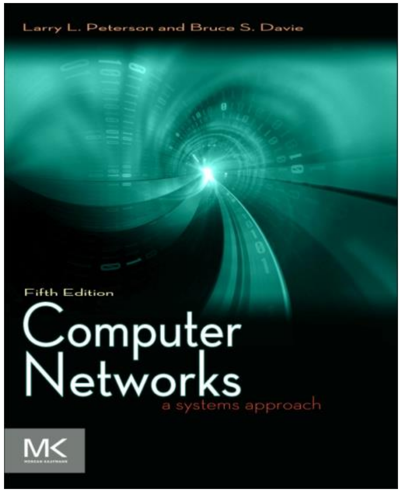
Bit/Bytes



Packet

Message/Stream

Ethernet transceiver and adaptor



Chapter 3

Internetworking

Chapter Outline

- Switching and Bridging
- Basic Internetworking (IP)
- Routing

Switching and Forwarding

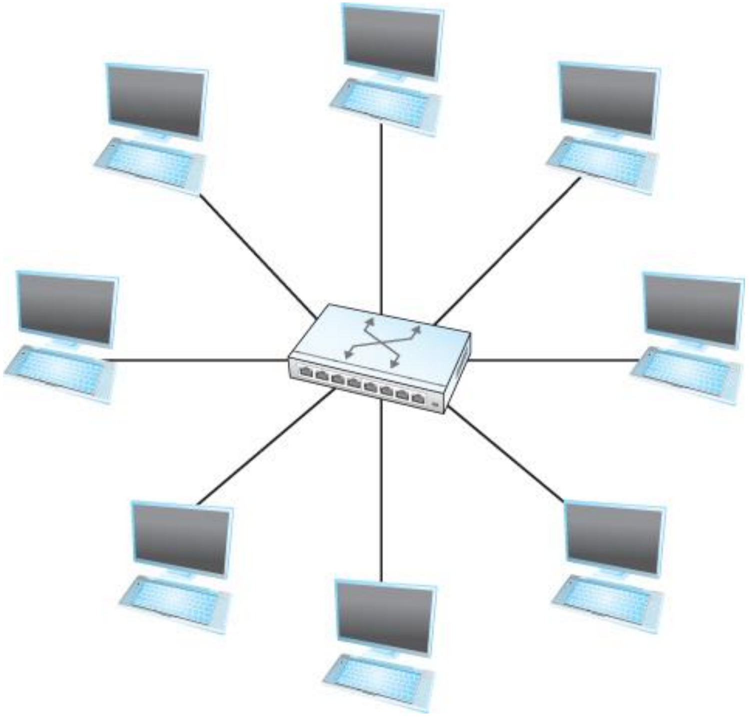
- Store-and-Forward Switches
- Bridges and Extended LANs
- Cell Switching
- Segmentation and Reassembly

Switching and Forwarding

- Switch
  - A mechanism that allows us to interconnect links to form a large network
  - A multi-input, multi-output device which transfers packets from an input to one or more outputs

Switching and Forwarding

Adds the star topology to the point-to-point link, bus (Ethernet), and ring (802.5 and FDDI) topologies



Switching and Forwarding

- A switch is connected to a set of links and for each of these links, runs the appropriate data link protocol to communicate with that node
- A switch’s primary job is to receive incoming packets on one of its links and to transmit them on some other link
  - This function is referred as *switching and forwarding*
  - According to OSI architecture this is the main function of the network layer

Switching and Forwarding

- How does the switch decide which output port to place each packet on?
  - It looks at the header of the packet for an identifier that it uses to make the decision
  - Two common approaches
    - *Datagram or Connectionless approach*
    - *Virtual circuit or Connection-oriented approach*
  - A third approach *source routing* is less common

Switching and Forwarding

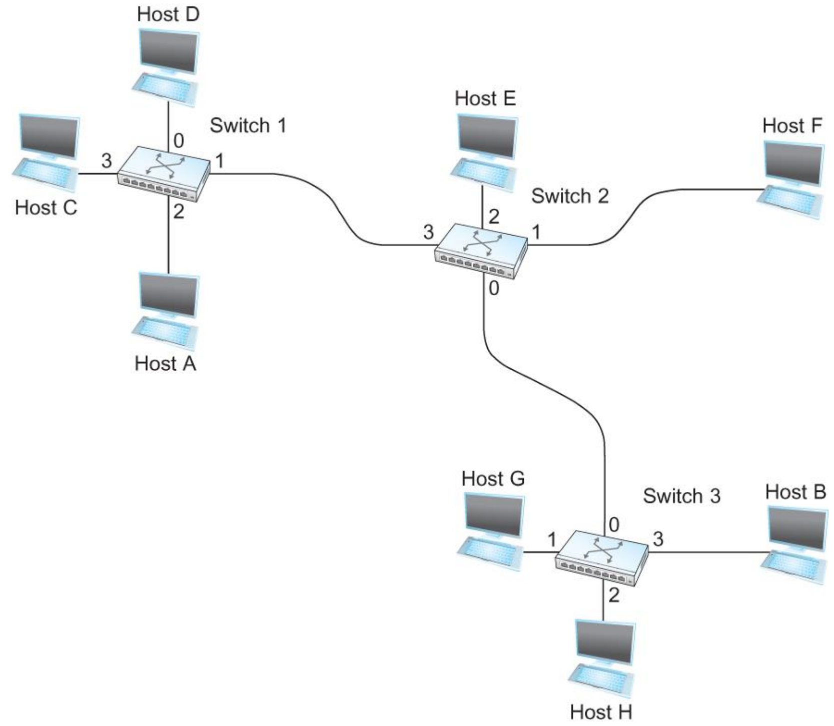
- Assumptions
  - Each host has a globally unique address
  - There is some way to identify the input and output ports of each switch
    - We can use numbers
    - We can use names

Switching and Forwarding

- Datagrams
  - 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

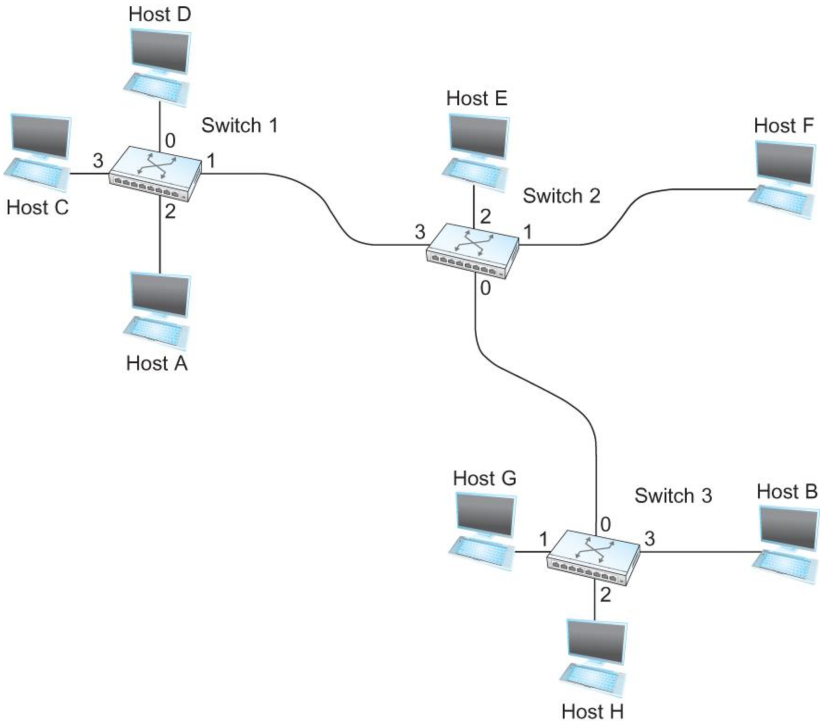
Switching and Forwarding

An example network



- To decide how to forward a packet, a switch consults a *forwarding table* (sometimes called a *routing table*)

Switching and Forwarding



Destination	Port
A	3
B	0
C	3
D	3
E	2
F	1
G	0
H	0

Forwarding Table for Switch 2

Switching and Forwarding

Characteristics of Connectionless (Datagram) Network

- A host can send a packet anywhere at any time, since any packet that turns up at the switch can be immediately forwarded (assuming a correctly populated forwarding table)
- When a host sends a packet, it has no way of knowing if the network is capable of delivering it or if the destination host is even up and running
- Each packet is forwarded independently of previous packets that might have been sent to the same destination.
  - Thus two successive packets from host A to host B may follow completely different paths
- A switch or link failure might not have any serious effect on communication if it is possible to find an alternate route around the failure and update the forwarding table accordingly

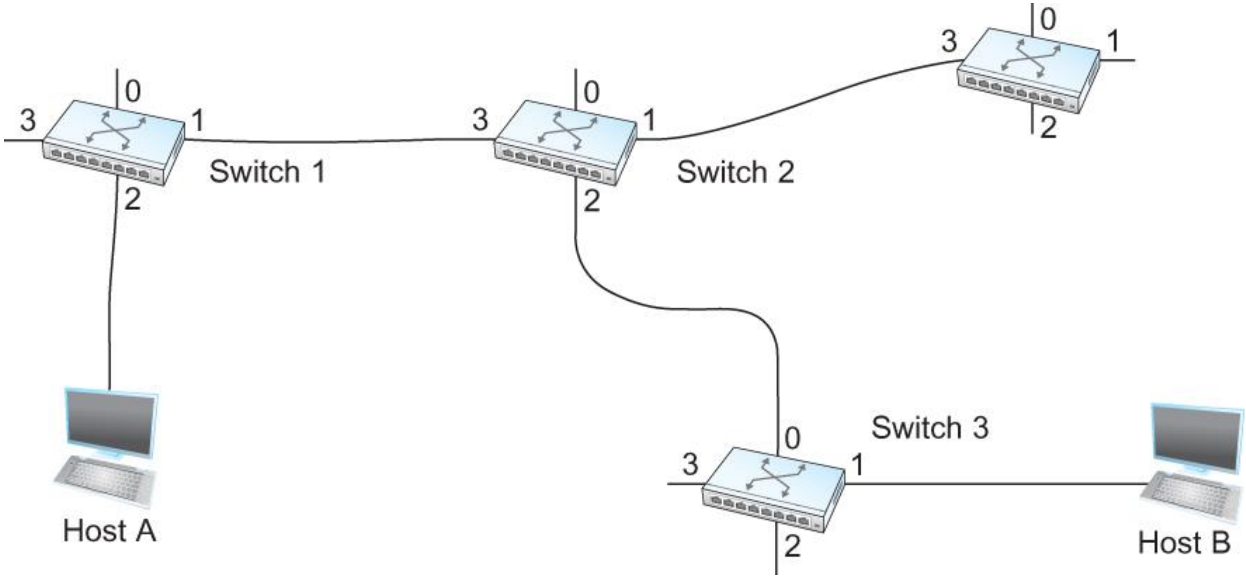
Switching and Forwarding

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

Switching and Forwarding

- Host A wants to send packets to host B



Switching and Forwarding

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

Switching and Forwarding

One entry in the VC table on a single switch contains

  - A virtual circuit identifier (VCI) that uniquely identifies the connection at this switch and that will be 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

- The semantics for one such entry is
  - If a packet arrives on the designated incoming interface and that packet contains the designated VCI value in its header, then the packet should be sent out the specified outgoing interface with the specified outgoing VCI value first having been placed in its header

Switching and Forwarding

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.

Switching and Forwarding

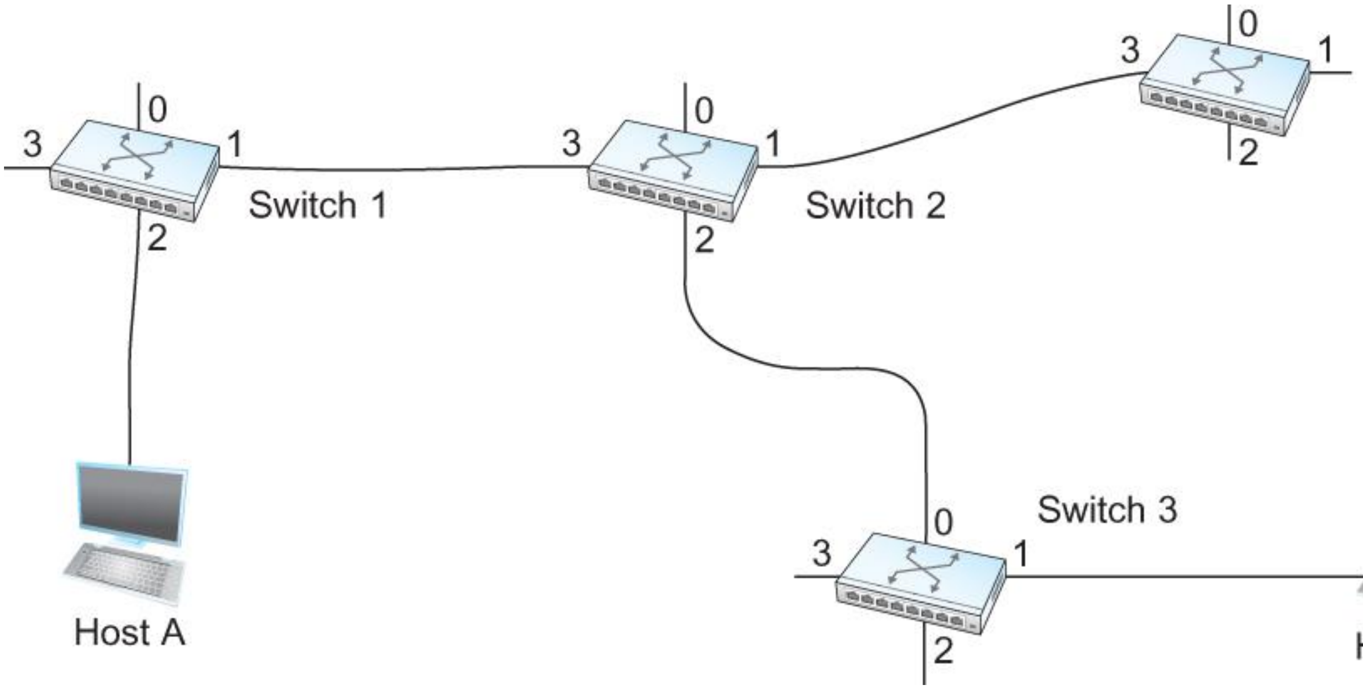
Two broad classes of approach to establishing connection state

- 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 signalling 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

Switching and Forwarding

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



Switching and Forwarding

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

Incoming Interface	Incoming VC	Outgoing Interface	Outgoing VC
2	5	1	11

Switching and Forwarding

Similarly, suppose

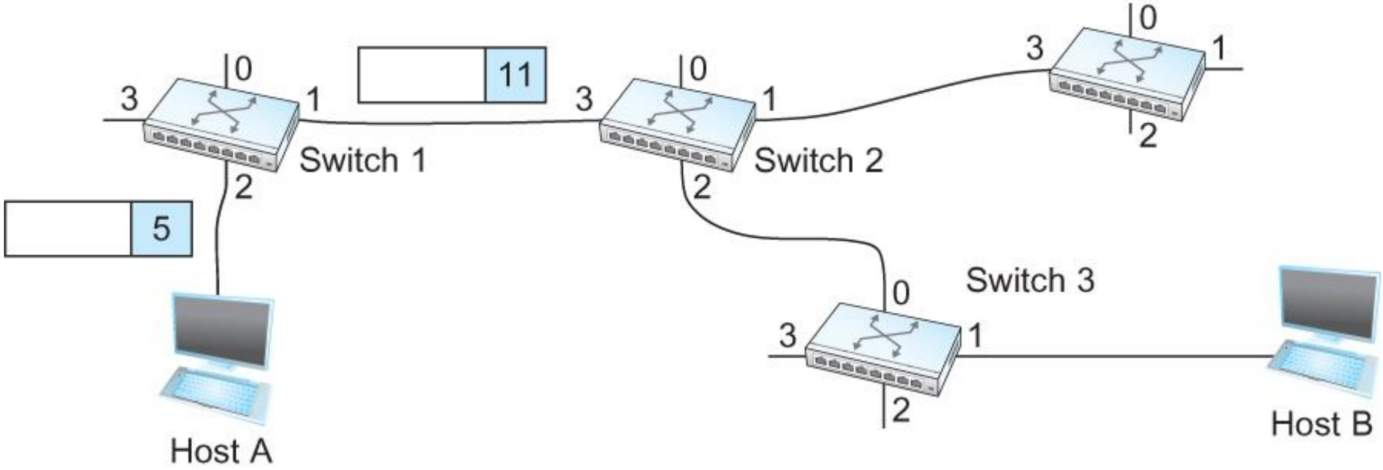
- VCI of 7 is chosen to identify this connection on the link from switch 2 to switch 3
- VCI of 4 is chosen for the link from switch 3 to host B
- Switches 2 and 3 are configured with the following VC table

Incoming Interface	Incoming VC	Outgoing Interface	Outgoing VC
3	11	2	7

Incoming Interface	Incoming VC	Outgoing Interface	Outgoing VC
0	7	1	4

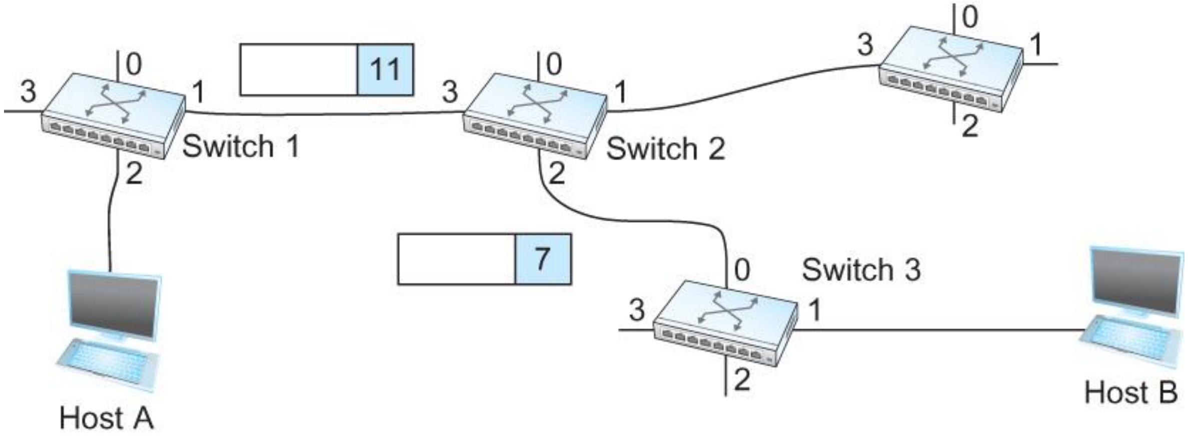
Switching and Forwarding

- 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



Switching and Forwarding

- 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



Switching and Forwarding

- 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 signalling is almost always used, even when setting up “permanent” VCs
  - In case of PVCs, signalling is initiated by the network administrator
  - SVCs are usually set up using signalling by one of the hosts

Switching and Forwarding

- How does the signalling work
  - To start the signalling 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 to figure out how to do that
  - 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

Switching and Forwarding

- How does the signalling work (contd.)
  - When switch 2 receives the setup message, it performs the similar process and it picks the value 11 as the incoming VCI
  - Similarly switch 3 picks 7 as the value for its incoming VCI
    - Each switch can pick any number 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

Switching and Forwarding

- 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

Switching and Forwarding

- 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

Switching and Forwarding

- Characteristics of VC
  - Since host A has to wait for the connection request to reach the far side of the network and return before it can send its first data packet, there is at least one RTT of delay before data is sent
  - While the 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.
    - Thus 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

Switching and Forwarding

- Good Properties of VC
  - By the time the host gets the go-ahead to send data, it knows quite a lot about the network-
    - For example, that there is really 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

Switching and Forwarding

- 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 along the virtual circuit, and this protocol is 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

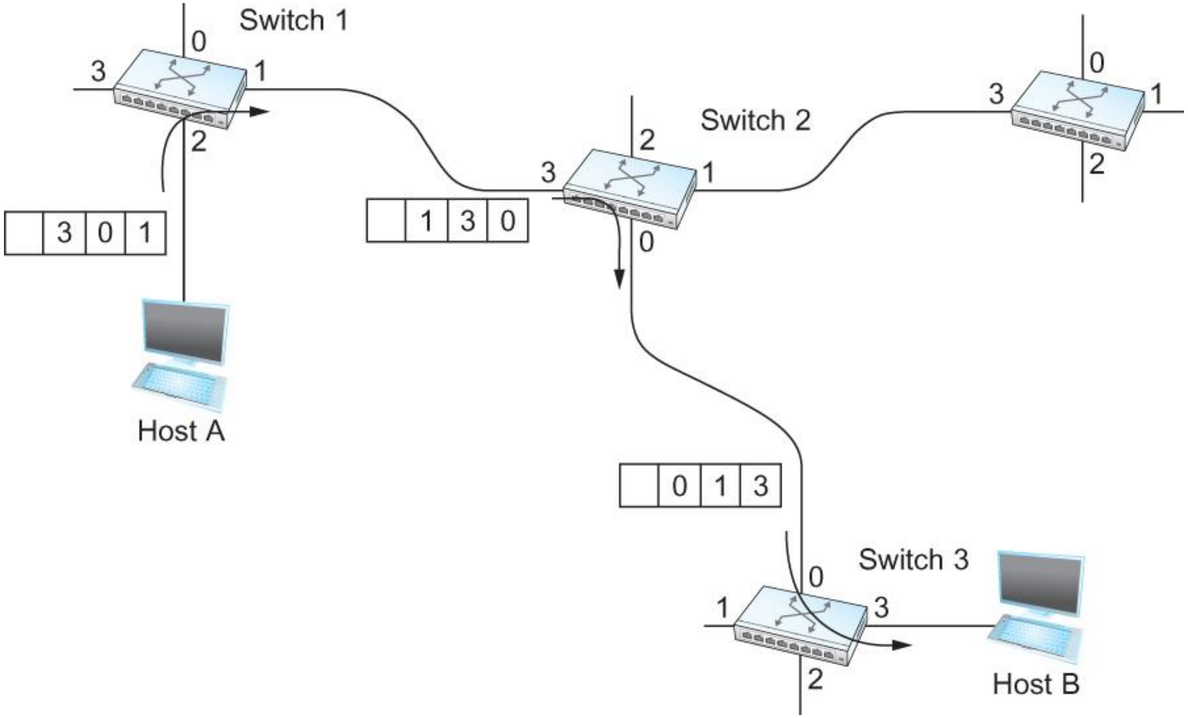
Switching and Forwarding

- 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 VC, 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
      - For example, a percentage of each outgoing link's bandwidth
      - Delay tolerance on each switch
- Most popular examples of VC technologies are Frame Relay and ATM
  - One of the applications of Frame Relay is the construction of VPN

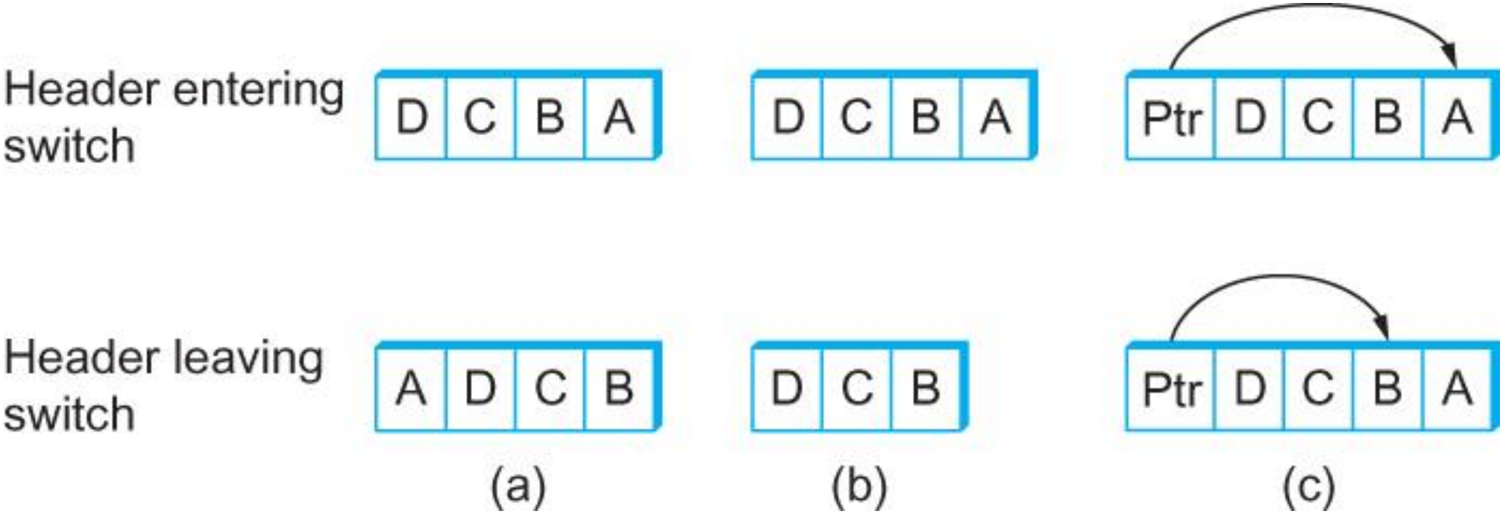
Switching and Forwarding

- Source Routing
  - All the information about network topology that is required to switch a packet across the network is provided by the source host



Switching and Forwarding

- Other approaches in Source Routing

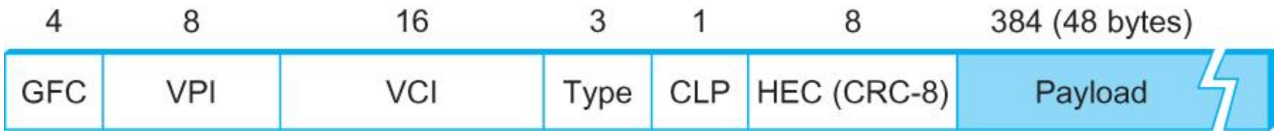


Switching and Forwarding

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

Switching and Forwarding

- ATM
  - User-Network Interface (UNI)
    - Host-to-switch format
    - GFC: Generic Flow Control
    - 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