University of Southern California

Viterbi School of Engineering

# EE450

# Computer Networks

## Switching Technologies

Shahin Nazarian

Spring 2013

# Switched (or Switching) Technologies

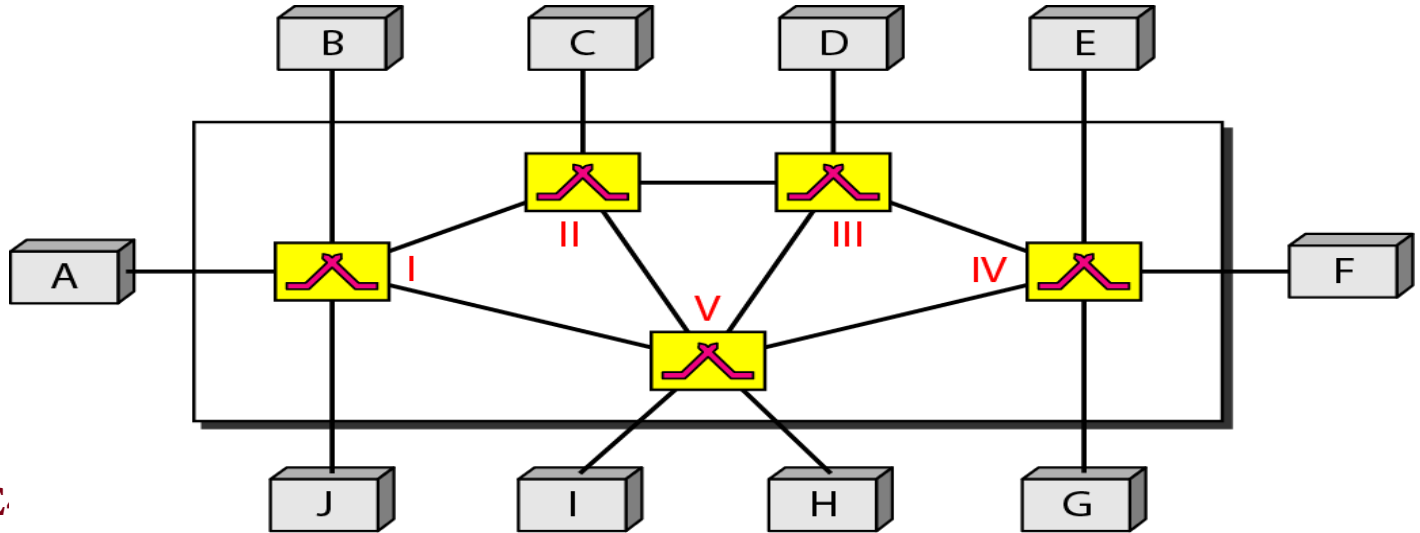
- Two main switching technologies are:
  - Circuit-switched (used in PSTN)
  - Packet-switched (used in Internet)
    - connectionless (or datagram networks) e.g. IP
    - connection-oriented (aka virtual circuits) e.g., ATM (Asynchronous Transfer Mode,) and FR (Frame Relay)
  - Note: don't confuse connectionless UDP and connection-oriented TCP (in layer 4) with connectionless or connection-oriented packet switching. Packet switching is related to layer 3 and to the IP protocol. TCP and UDP both run over IP

# Switching Technologies (Cont.)

- None of those technologies typically use a fully-connected topology, e.g. a fully-connected MAN is not practical
- They are partially connected, but with redundant paths, so multiple routes exist between two end systems so that there are alternatives if one goes wrong
- Typical implementation:
  - Circuit-switched
  - Virtual circuit
  - Connectionless packet-switched

# Switched Network

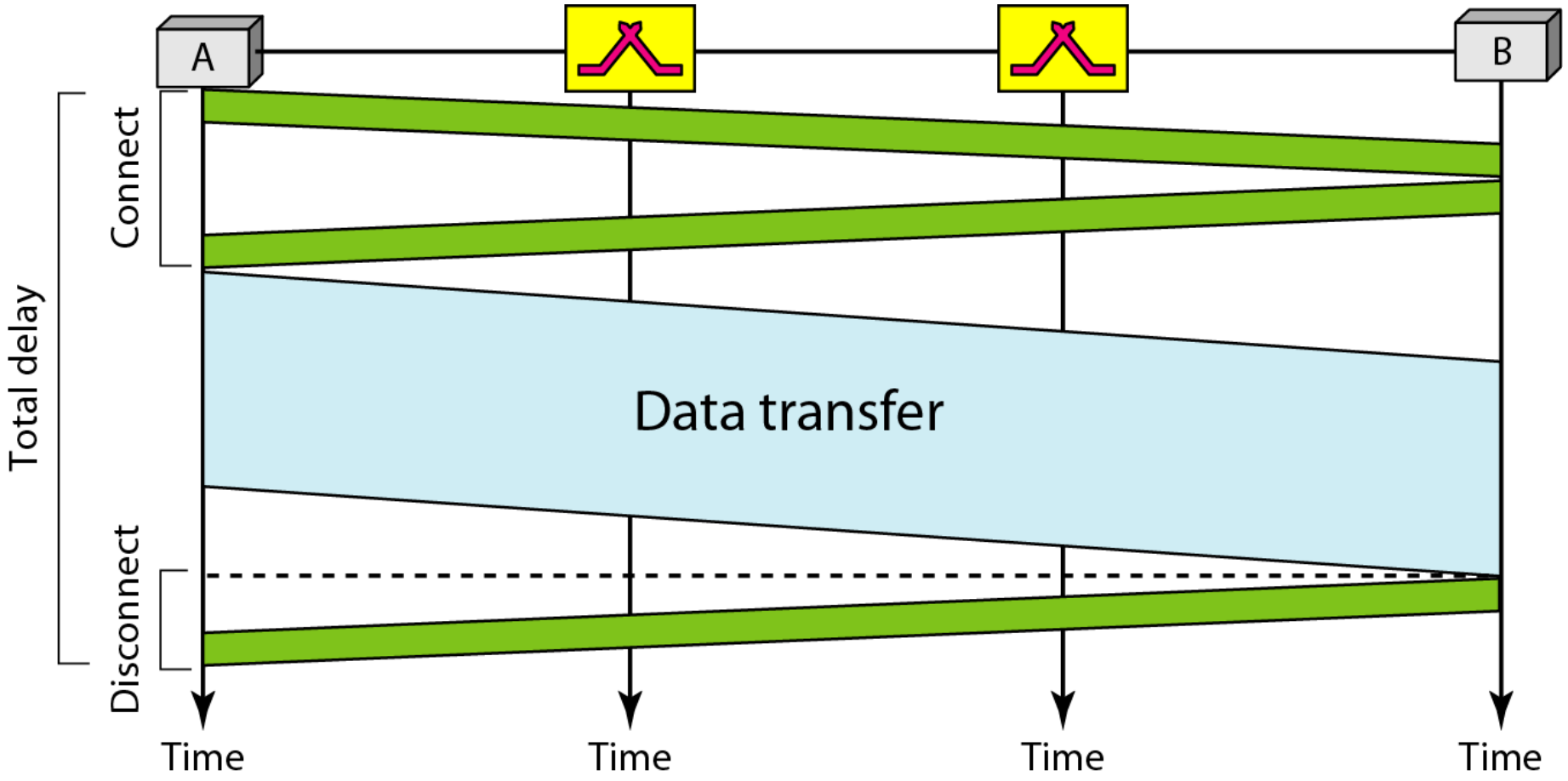
- A switched network is a network that traffic goes from one switch to another (called switches in PSTN, but in Internet they are called routers)
- Suppose J wants to send a message to F. The message won't be received by any other end system than F
- Trunks are not dedicated they are wide and shared using FDM, TDM, or WDM [Wavelength Division Multiplexing]
- The only lines, dedicated are the end lines that are directly connected to the end system



# Circuit-Switched Network

- Go back to our previous discussions on how for a phone call, a dedicated circuit (per demand for the duration of the call, ) is established. Note that only a portion of the trunks in the path of the dedicated circuit is used (e.g., either one frequency portion in FDM, or a time slot in TDM)
- Once the caller ends the phone call, all resources that had been dedicated become free
- 64kbps is dedicated to each phone call (Question, how many calls can be carried through a 2Gbps shared trunk?) Traffic comes in the form of stream (such as voice, and video) or burst i.e., data
- Circuit-switched network is optimized for stream traffic (and real time applications) & not good for data

# Circuit-Switched – Delay Components



# Circuit-Switched Total Delay

- In circuit-switched network as soon as the call is accepted and the circuit is dedicated
- The total delay in the circuit-switched network includes the **setup time** (aka **connect time** aka **establish time)** to make the dedicated path between two end points and that involves making routing decisions. The delay also includes the **disconnect** (**release aka teardown**) **time**, but disconnect time is negligible, e.g., erasing the information from the control memory of the switches which should not take that long
- There is no **store-and-forward** in circuit switching, except during the call setup. However packet switching works based on store-and-forward

# Circuit-Switched Total Delay – Example

- Suppose all the links in a circuit-switched network have a bit rate of 56Kbps. Also it takes 900msec and 200msec to establish and disconnect an end-to-end circuit respectively, before a sender host can transmit data
- How long does it take to send a file of 1MB from host A to host B? The propagation delay between A and B is 0.045 seconds
  - Transmission delay = $8 \times 10^6$ bits/56000bps =142.86 seconds
  - Total delay = 900ms + 142.86s + 0.045s + 200ms =144.005s

# Connectionless Packet-Switched

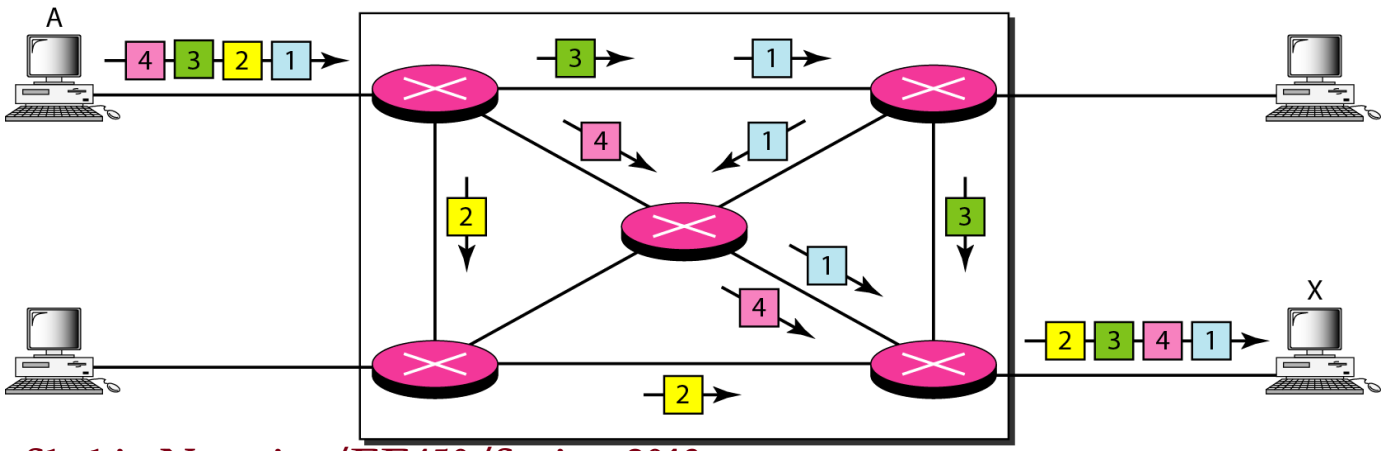
- In TCP and UDP (layer 4) the notions of connection-oriented and connectionless imply the handshaking process existence for TCP (and no handshaking process for UDP,) however the notion of connectionless packet switching (layer 3,) means no setup connection
- In circuit switching addressing is done only during call setup, and once the circuit is set up, we don't need any addressing, because the resources are going to be dedicated
- However in connectionless packet switching networking, everything is shared and there is no dedicated resources, so the message is broken into packets and each packet needs to be addressed to let the router know where to send the packet. That is the reason we put the address of the source and destination in the header of the packet
- Note: the length of the packets is not constant, but what is certain is that each one of the packets is going to have a header

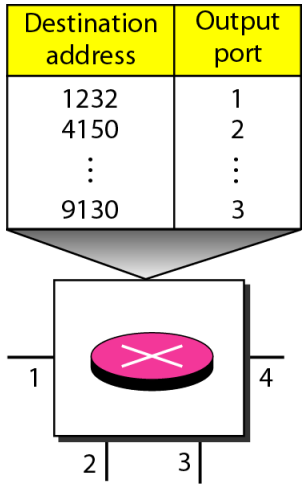
# Connectionless Packet-Switched (Cont.)

- Question: Message is divided into packets. Which approach is better? Dividing the message into shorter packets, i.e., more packets and headers or into a smaller number of packets which means larger packets but less number of header?
- Depending on the application both would work
- Assume a message is divided into n packets. The network will look at each packet as an independent unit, and does not know and doesn't need to know that the n packets belong to the same message. The network processes each packet independently from others
- In packet switching every packet is stored-and-forwarded

# Connectionless Packet-Switched – A Best Effort Service

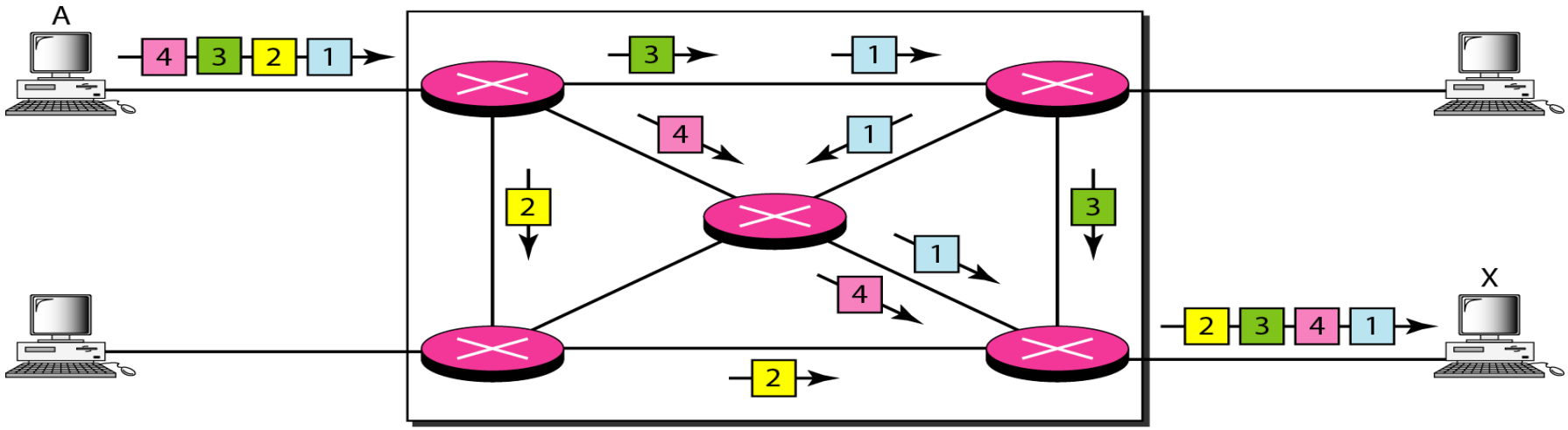
- The concept of store-and-forward is used for every packet
- Packets may not take the same route because they are processed independently from each other. Also they may be received by the destination out of order. One reason is that some of the nodes may be congested or busier, so the packets wait in the queue for a longer time
- It is the receiver's responsibility to put the packets back in order and recover the missing packets
- Connectionless is a best-effort service, meaning that packet delivery isn't guaranteed!



# Connectionless Packet-Switched (Cont.)

- Here A wants to send a message to X, and it divides it into 4 packets. In reality the packets are not numbered
- Each node has an incoming buffer to store packets before forwarding them. The first node receives and stores it. The node looks at the packet header and decides where to forward it. Same goes for other nodes
- If the packet is erroneous the node will not send the packet, it drops it and it doesn't inform the host that it dropped it. It is the host's responsibility to figure out if the packet was dropped on the way

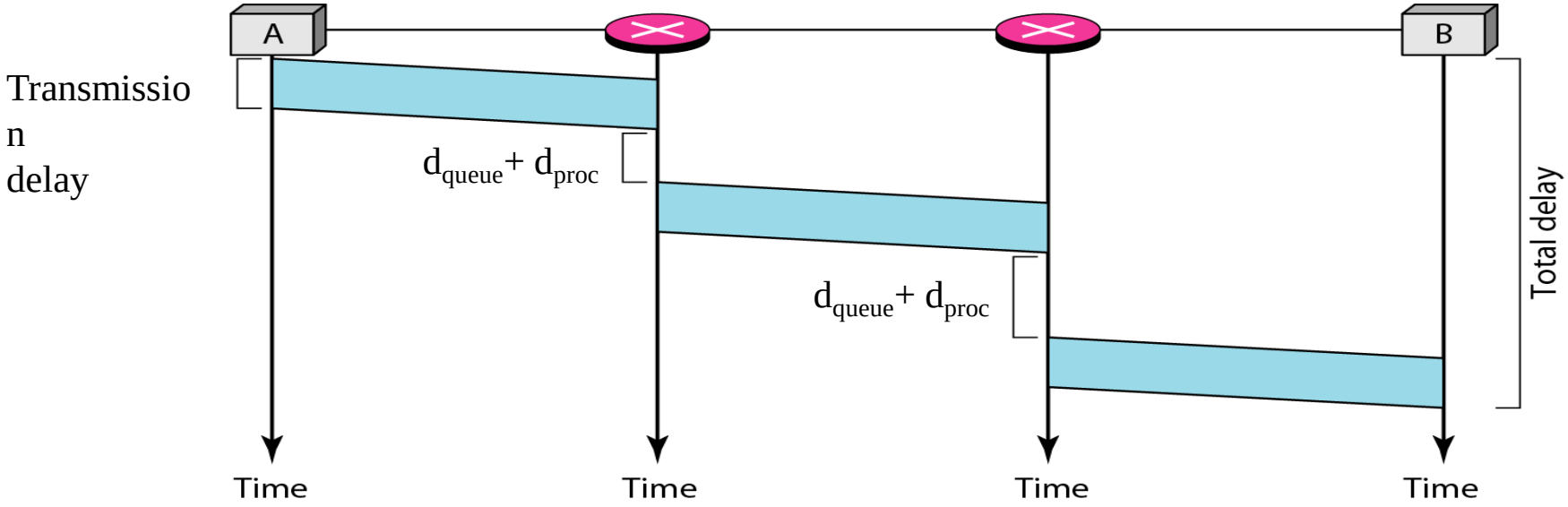


# Connectionless Packet-Switched (Cont.)

- Remember that circuit-switched network is not optimized for burst traffic, i.e., data
- Similarly connectionless packet switching is not optimized for stream type (real time) traffic
  - E.g. voice over internet may not have a good quality, because the internet congestion results in some voice packets to be dropped or face excessive delays
- Multiple end systems may be sending packets through the network nodes, paths are not dedicated, however each packet has the sender and receiver addresses

# Connectionless Packet-Switched – Delay

- Compare this with the delay in circuit switching. There is no call setup, or teardown (disconnect) however there is data processing and queuing delays which do not exist in circuit switching case



# Packet or Circuit Switched? – Efficiency Comparison

- Packet switching is a lot more efficient, mainly due to the following reasons:

  1) The links (trunks) that are high bandwidth and very costly, are shared by a far more number of end systems. Circuit switching technology is very inefficient for burst type in that sense

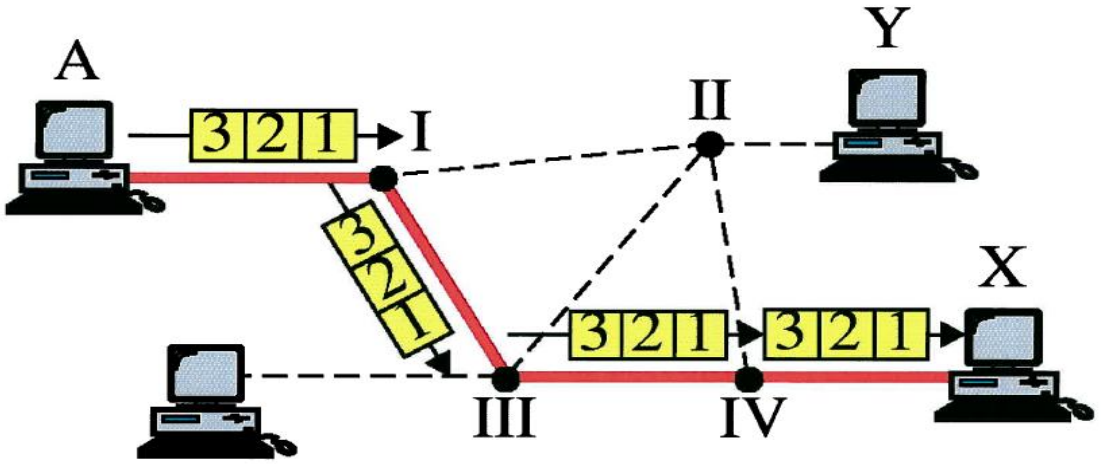
  2) Packet switching technology is more fault-tolerant, meaning if some nodes or links on the path become defective or do not function any longer, communication could still be possible with rerouting the packet. In circuit switching if the circuit does not function any longer, the calls will be disconnected

# Store-and-Forward Delay

- **Transmission delay** is also referred to as the **store-and-forward delay**
- Example: There are N links between two hosts A and B, each of rate R bps. Assume there is only one packet in the network. Also ignore the propagation delay. What is the total delay?
  - Packet must be first transmitted from host A to the first link. This takes L/R seconds. It must be transmitted on each of the N-1 remaining links, i.e., it must be stored and forwarded N-1 times, each time with store-and-forward delay of L/R. Thus the total delay is NL/R

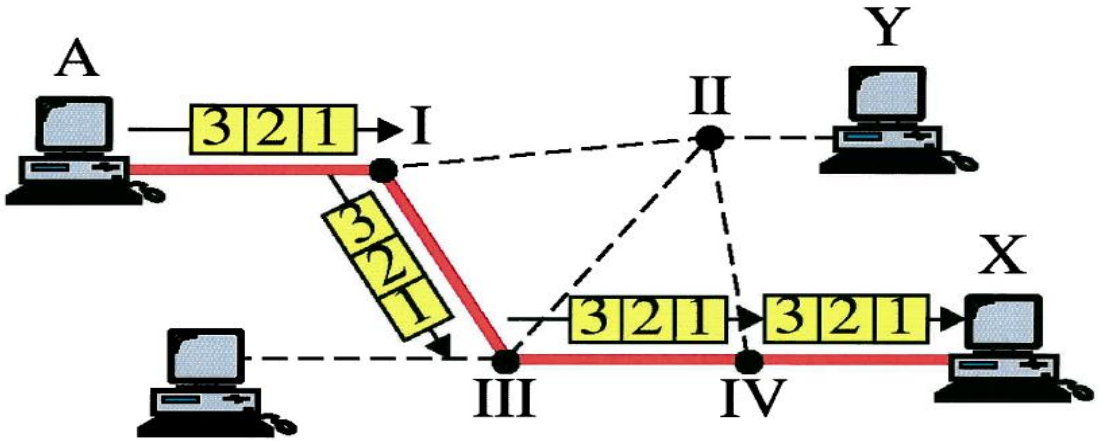
# Virtual Circuit Switching (Connection-Oriented Packet Switching)

- The notion of packet switching in virtual circuits implies that store-and-forward is still the concept
- Virtual means something that gives you the impression of the real but it is not real! Therefore in case of virtual circuit switching, it looks like circuit switching, but it really is not!
- In virtual circuit, before the communication starts, a virtual connection (but not dedicated) is going to be set up



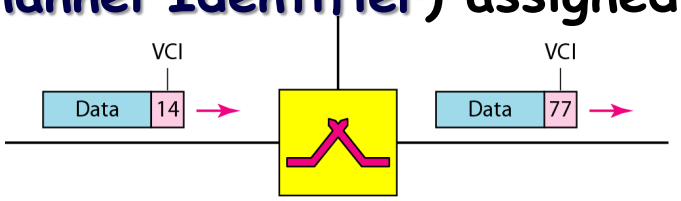
# Virtual Circuit Switching (Cont.)

- There is a call request packet and a call accept packet that establish the connection. Again note that connection is not dedicated, if it were, it would be circuit switching
- While being transmitted, the packets will follow the same virtual connection that was set up, but based on store-and-forward
- Can the packets be received out of order? No, they will be in order, because they are following the same route

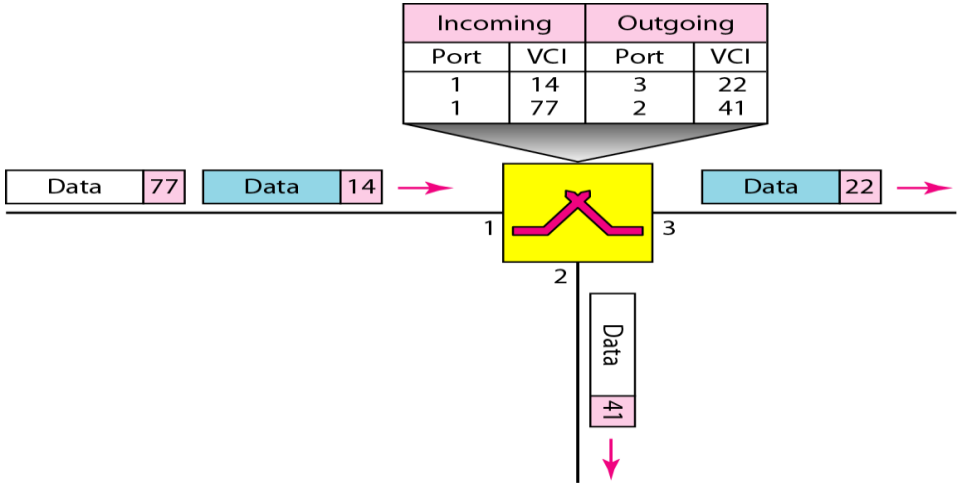


# Virtual Circuit Switching (Cont.)

- Instead of destination address, every frame contains a **VCI** (**Virtual Circuit** or **Channel Identifier**) assigned by the network

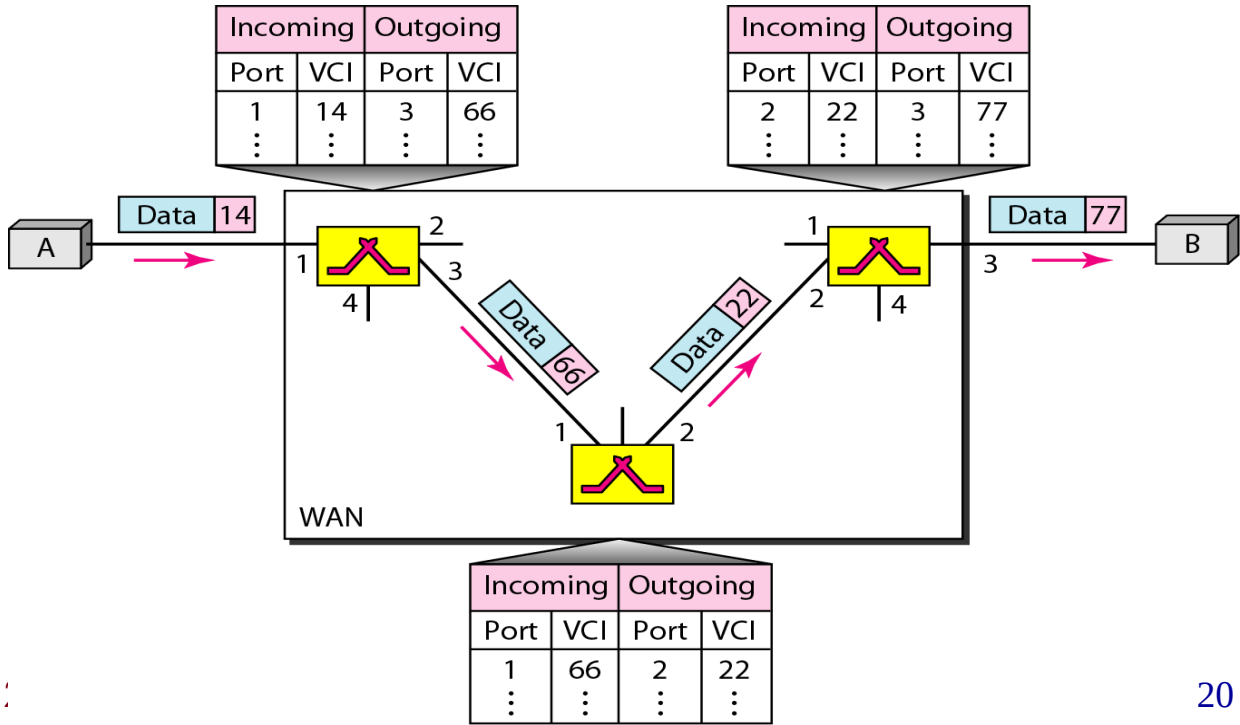



- A VCI-based lookup table is used to fwd the frame. Destination address is not required, because all the packets are going to follow the same virtual connection (route)
- In case a node drops a packet, it will inform the sender so that the packet can be transmitted. This is no longer a best effort service; because of the added quality, this service is more costly



# Virtual Circuit Switching (Cont.)

- After the communication is done, the virtual connection is cleared, similarly to what is done in a real circuit switching
- The packets will follow the same route, so the routing decision is done only during the virtual connection, afterwards routers won't make routing decisions. In virtual circuit switching if something goes wrong with one of the links that is along the virtual connection, the virtual connection will be dropped



# Virtual Circuit Switching – Efficiency

- Question: Compare the reliability and efficiency of virtual circuit switching with those of the packet-switching
- Virtual circuit switching is less flexible
- It is more reliable as it gurantees that the packets are going to be received in the same order as they were transmitted
- Call setup time includes the call request time (with request packet) and call accept time (with accept packet)
- Virtual circuit switching uses the concept of store-and-forward, meaning packets are stored, then forwarded; also both request packet and accept packet need to be stored and forwarded in the intermediate nodes, so there is some processing time involved for both setup and accept times

# Virtual Circuit Switching – Efficiency (Cont.)

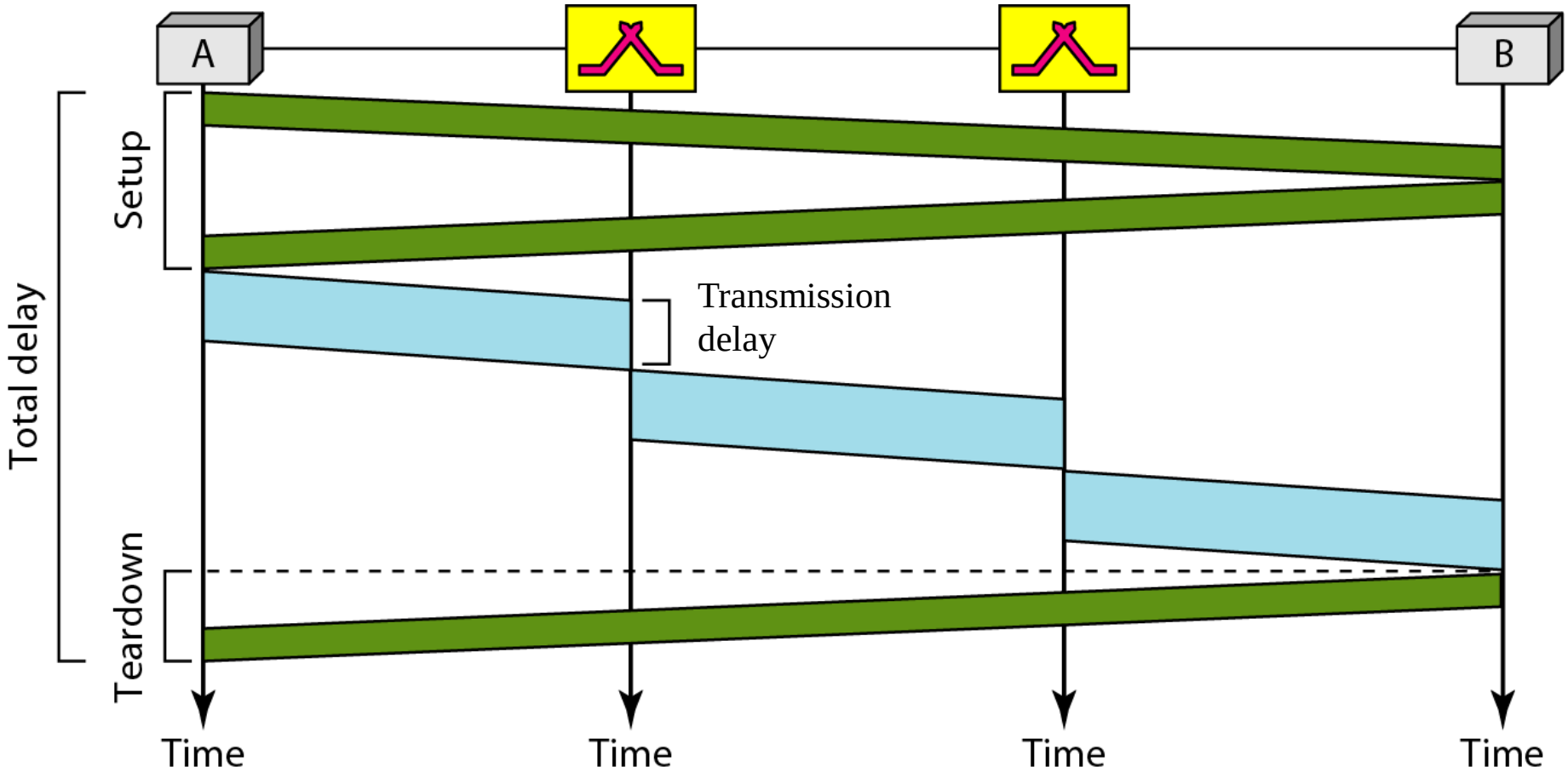
- In circuit switching the nodes (switches) need to do some processing to establish the dedicated path, therefore the call setup time includes some processing time, however there is no store-and-forward concept (because the resources have already been dedicated) and the call accept signal traverses the nodes back to the caller without any processing

- In short in call setup time, the request times for both virtual circuit switching and circuit switching look similar, however the call accept times are different

# Virtual Circuit Switching – Efficiency (Cont.)

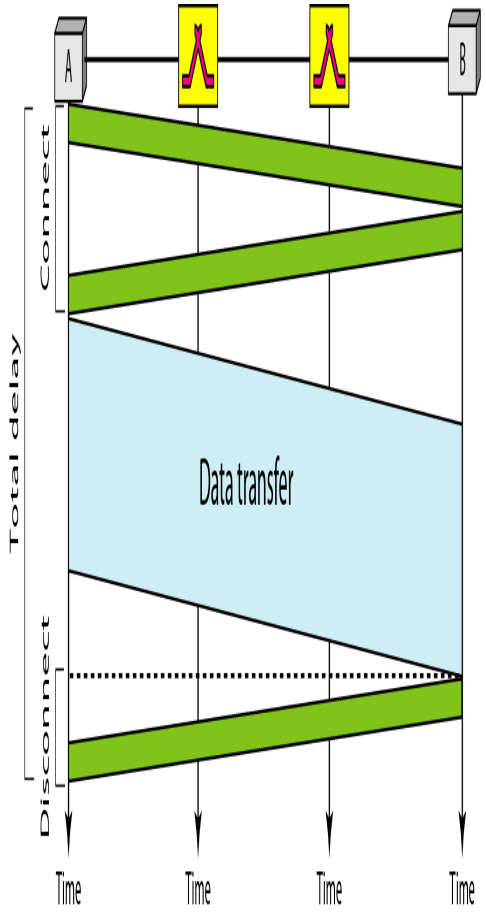
- Its data transfer phase is different from that in circuit switching. In circuit switching there is no store-and-forward perception and the circuit is dedicated, so there is no processing (break in time) during data transfer, however in virtual circuit switching there is still some processing for data even if the virtual connection is already established. The processing and queuing delays are negligible
- Therefore data transfer phase is similar to that in packet switching (where "store-&-forward" exists, the resources are not dedicated, so there are some processing,) however there is a difference: In packet switching, they are not following the same route, whereas in virtual circuit switching they are
- Reminder: there is no call setup in connectionless packet switching (e.g., IP)

# Virtual Circuit Switching – Delay

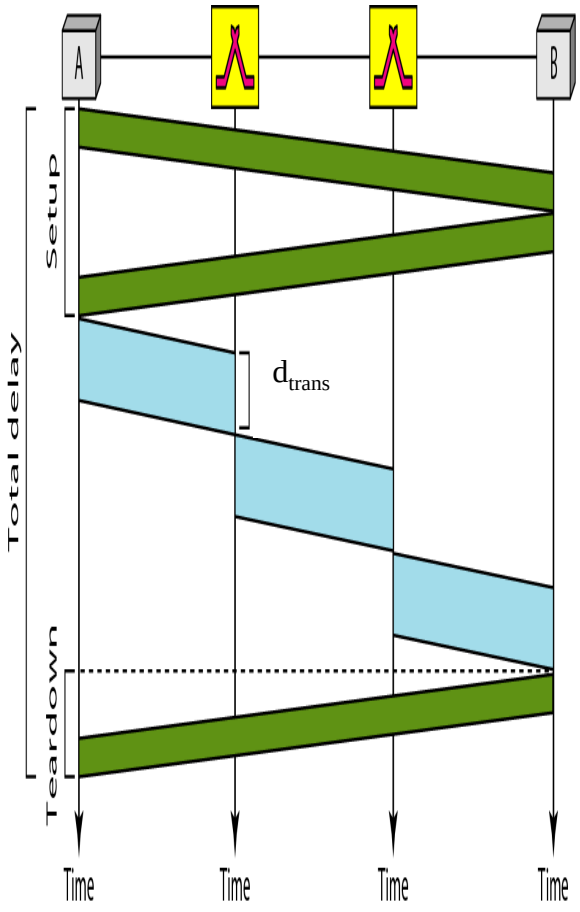


# Delay Comparison

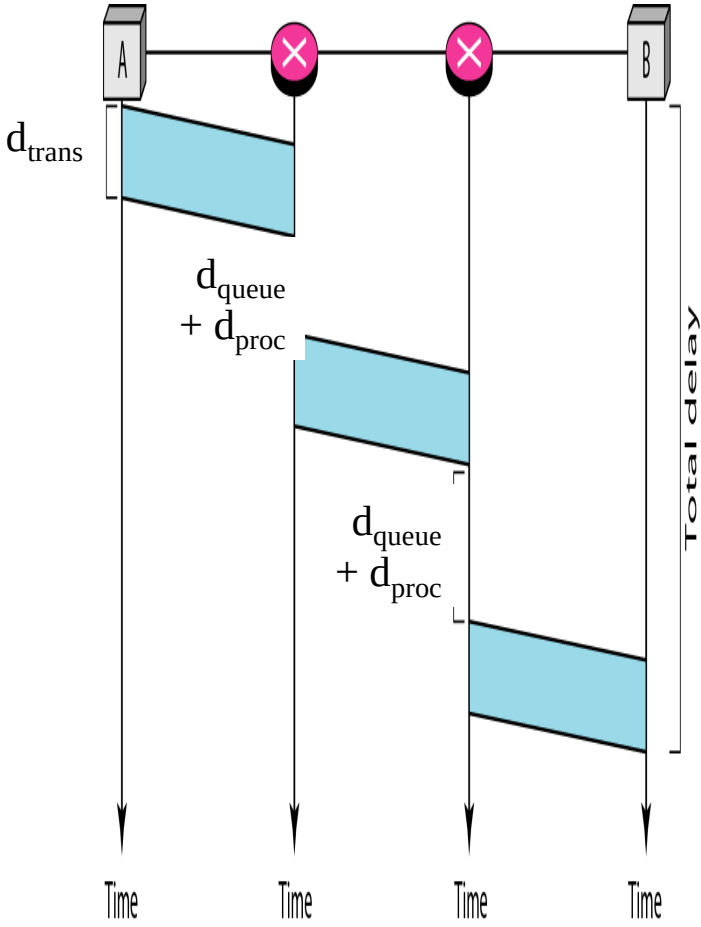
**Circuit Switching**




**Virtual Circuit Switching**

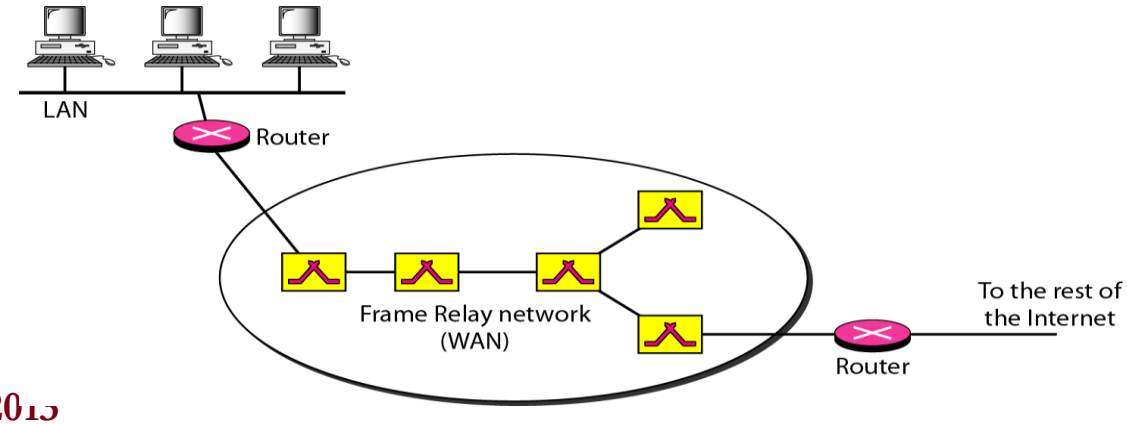



**Connectionless Packet Switching**



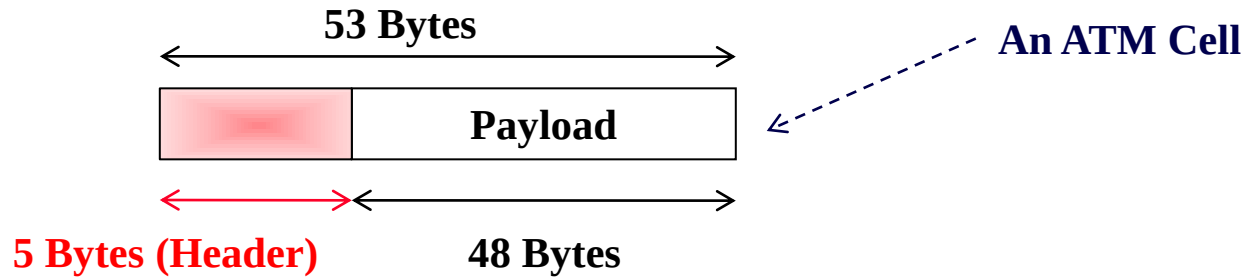
# Virtual Circuit Switching – FR

- FR (Frame Relay) is a virtual circuit wide-area network (developed in late 80s)
- It was mainly designed to address the drawback of X.25 a virtual circuit switching network that performed switching in Network layer and was low bandwidth
- FR is less expensive than traditional WANs and operates at high bandwidths of 1.55Mbps or 44.376Mbps, so it can be used instead of T1 and T3 lines
- FR operates in Physical and Data Link layers, therefore it can be used as a backbone network



# Virtual Circuit Switching – ATM

- Packets and frames are in general variable in length, e.g., FR frames are variable in size. ATM (Asynchronous Transfer Mode) is a cell relay protocol with fixed size units called cell . A cell is in Data Link Layer:




- 53 bytes is considered a short unit. Note that ATM was developed primarily for voice; and shorter units are preferred for voice
- ATM protocol is faster than the IP protocol. ATM combined with Synchronous Optical Networking allows high-speed interconnection of all networks; however ATM is more expensive and this is the main reason ATM technology is not as flexible/popular as the IP