Network Layer

- Network Layer: the most complex layer
  - Requires the coordinated actions of multiple, geographically distributed network elements (switches & routers)
  - Must be able to deal with very large scales
    - Billions of users (people & communicating devices)
  - Biggest Challenges
    - Addressing: where should information be directed to?
    - Routing: what path should be used to get information there?

Packet Switching

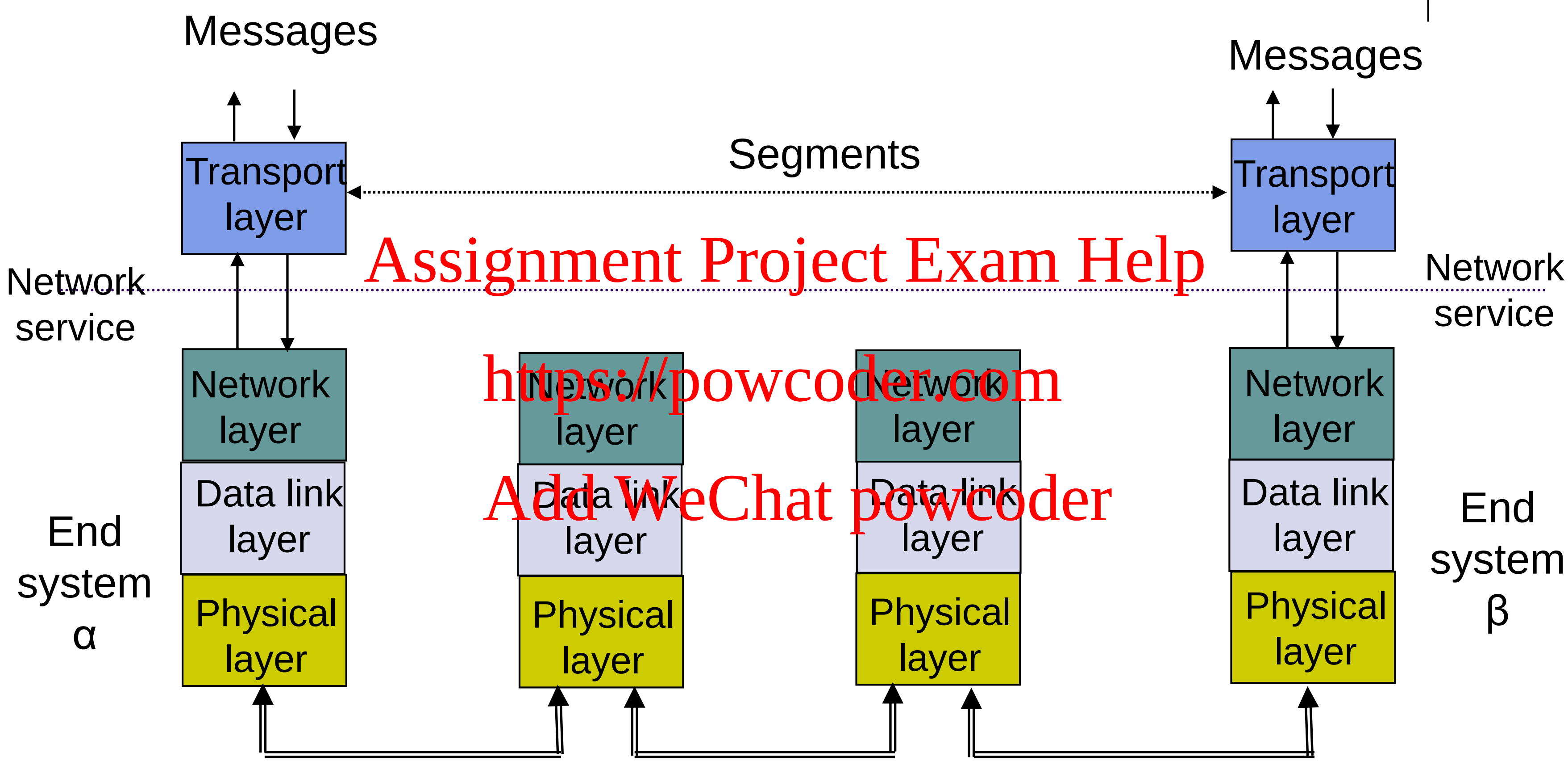
- Transfer of information as payload in data packets
- Packets undergo random delays & possible loss
- Different applications impose differing requirements on the transfer of information

Perspectives of Packet Networks

- External View of the network
  - Services that the network provides to the transport layer
  - Services are independent of the underlying network
  - Whether the network service requires setting up of connections
  - Whether data transfer requires any quality-of-service guarantees

- Internal Operation of the network
  - Considers physical topology of the network and its interconnection
  - Approaches used to direct information – datagram, virtual circuit
  - Addressing and routing procedures
  - Deal with congestion inside the network
  - Traffic management inside the network

Network Service



- Network layer can offer a variety of services to transport layer
- Connection-oriented service or connectionless service
- Best-effort or delay/loss guarantees

Connectionless vs. Connection-oriented

Connectionless :

- Only two basic interactions between transport and network layer
  - Request to network layer to send a packet
  - Indication from network layer that a packet has arrived
- User can request packet transmission at any time
  - No need to inform network layer ahead of time
- Responsibility for error control, sequencing and flow control on transport-layer

Connection-oriented :

- Connection-setup required
  - Network layer must be informed about the new flow to be sent to the network
  - Network layer maintains state information about the flows it is handling
- Allows usage and quality-of-service negotiations
  - Network resources may be allocated
- Connection-termination required
- Complex than connectionless service

Network Service vs. Operation

Network Service

- Connectionless (UDP)
  - Datagram Transfer
- Connection-Oriented (TCP)
  - Reliable and possibly constant bit rate transfer

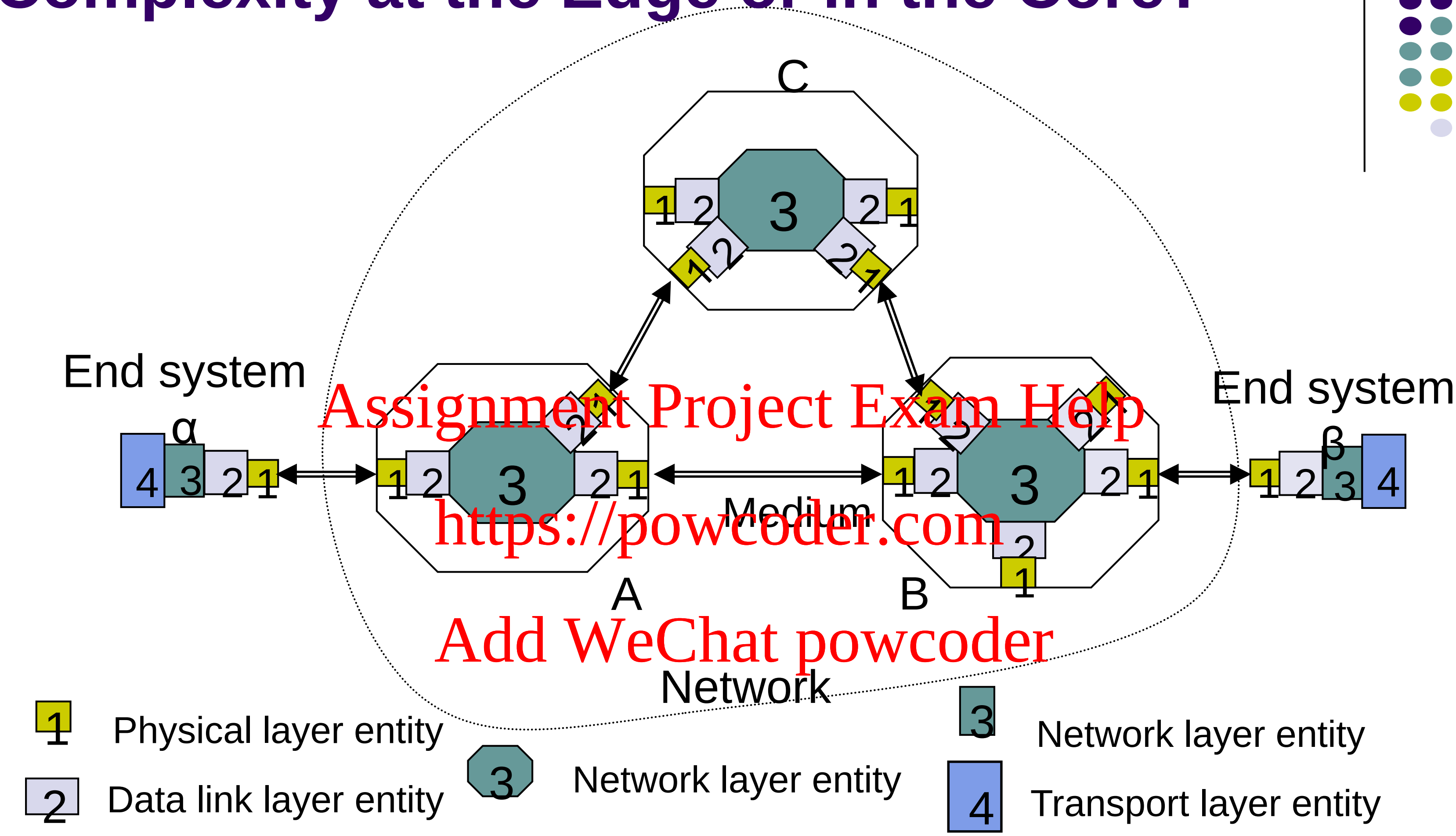
Internal Network Operation

- Connectionless
  - Datagram operation
  - IP
- Connection-Oriented
  - Virtual Circuit operation
  - Telephone connection
  - ATM

Various combinations are possible

- Connection-oriented service over Connectionless operation
- Connectionless service over Connection-Oriented operation
- Context & requirements determine what makes sense

Complexity at the Edge or in the Core?



Need for the network to grow to very large scale –

- keep the core of the network simple (connectionless packet network)
- provide necessary complexity at the edge

Network Layer Functions

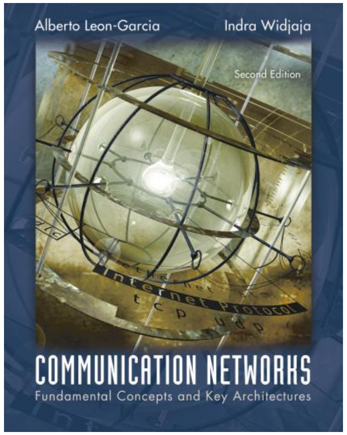
Essential

- **Routing**: mechanisms for determining the set of best paths for routing packets requires the collaboration of network elements
- **Forwarding**: transfer of packets from NE inputs to outputs
- **Priority & Scheduling**: determining order of packet transmission in each NE

Optional: congestion control, segmentation & reassembly, security

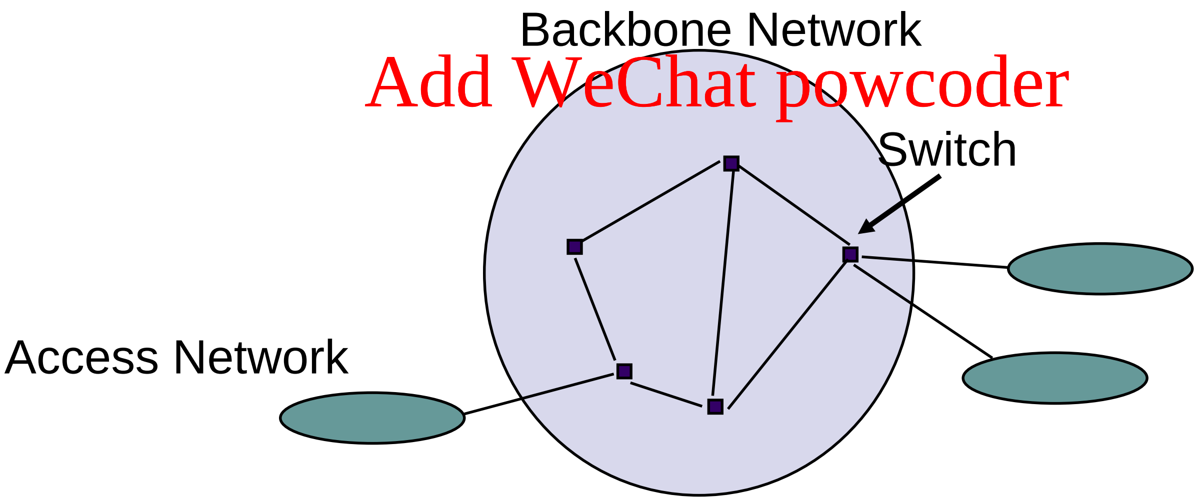
Packet-Switching Networks

Datagrams and Virtual Circuits

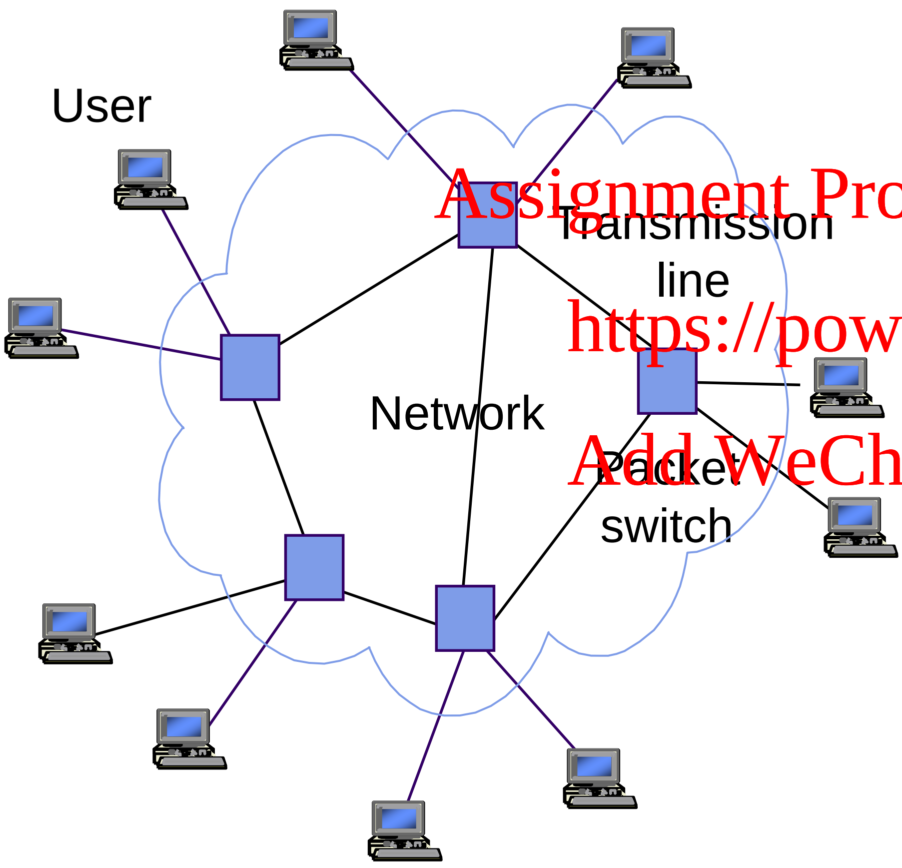


The Switching Function

- Dynamic interconnection of inputs to outputs
- Enables dynamic sharing of transmission resource
- Two fundamental approaches:
  - Connectionless
  - Connection-Oriented: Call setup control, Connection control



Packet Switching Network



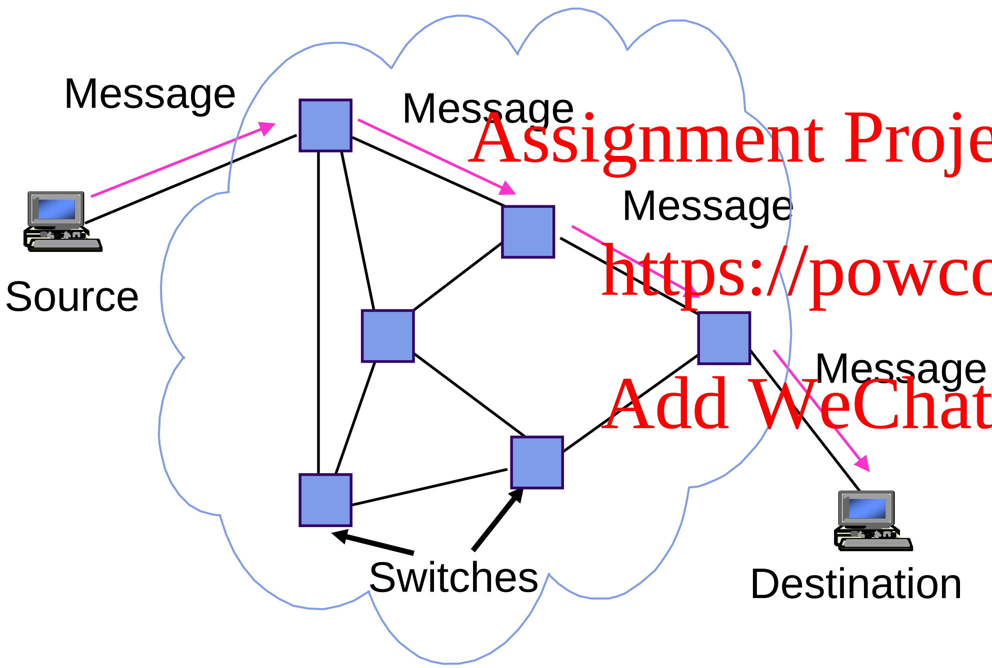
Packet switching network

- Transfers packets between users
- Transmission lines + packet switches (routers)
- Origin in message switching

Two modes of operation:

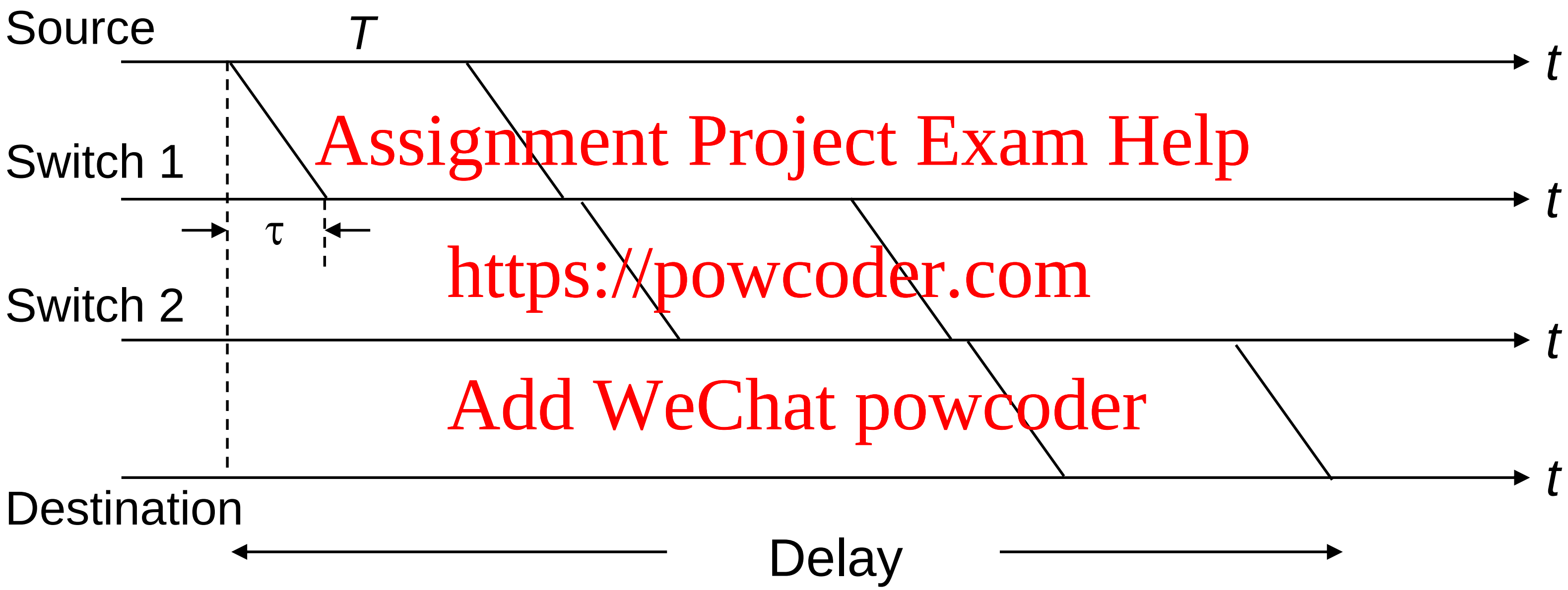
- Connectionless
- Virtual Circuit

Message Switching



- Message switching invented for telegraphy
- Entire messages multiplexed onto shared lines, stored & forwarded
- Headers for source & destination addresses
- Routing at message switches
- Connectionless

Message Switching Delay

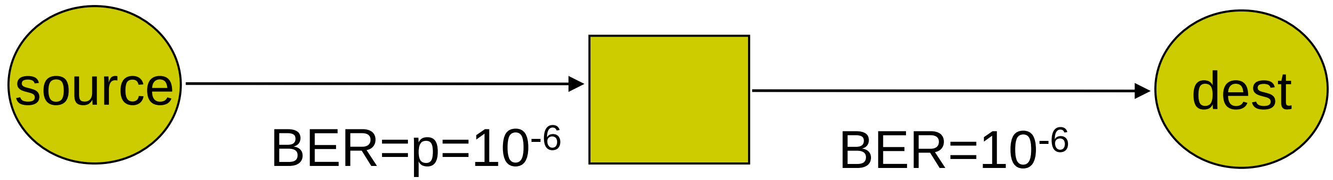


Minimum delay = $3\tau + 3T$

Additional queueing delays possible at each link

Long Messages vs. Packets

1 Mbit message



How many bits need to be transmitted to deliver message?

- Approach 1: send 1 Mbit message
- Probability message arrives correctly

$$P_c = (1-10^{-6})^{10^6} \approx e^{-10^6 10^{-6}} = e^{-1} \approx 1/3$$

- On average it takes about 3 transmissions/hop
- Total # bits transmitted ≈ 6 Mbits

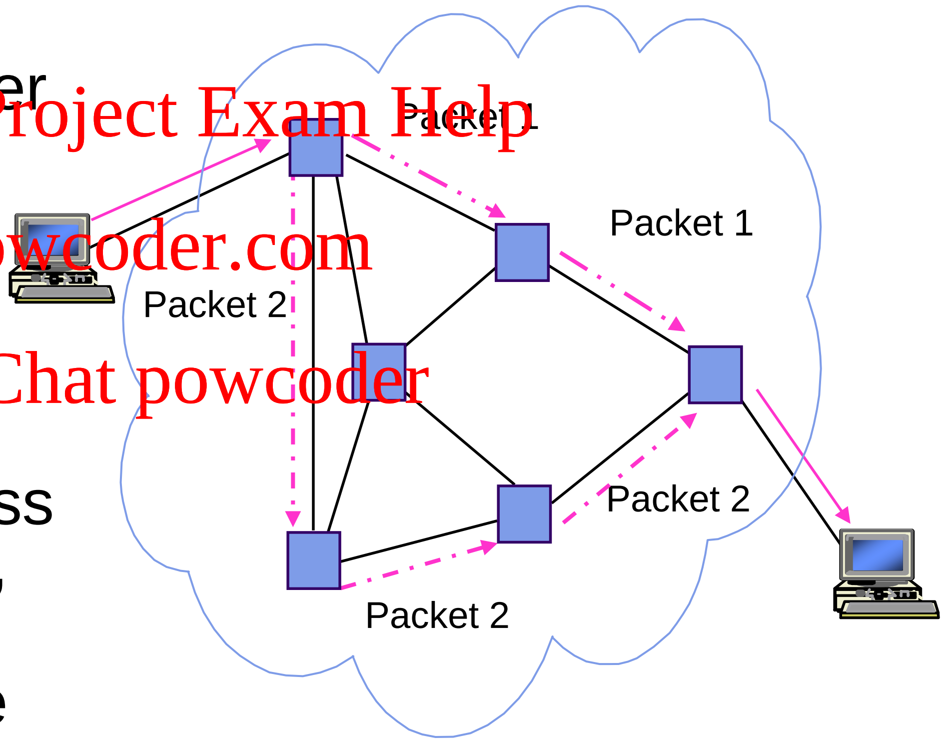
- Approach 2: send 10 100-kbit packets
- Probability packet arrives correctly

$$P_c' = (1-10^{-6})^{10^5} \approx e^{-10^5 10^{-6}} = e^{-0.1} \approx 0.9$$

- On average it takes about 1.1 transmissions/hop
- Total # bits transmitted ≈ 2.2 Mbits

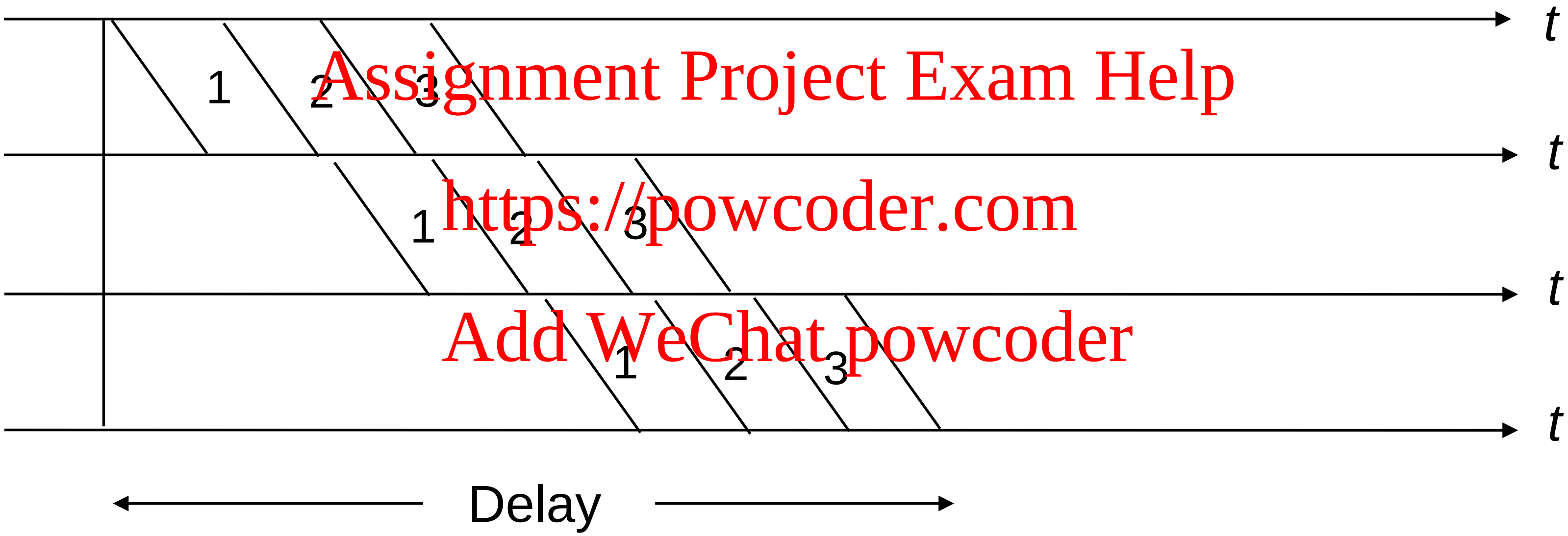
Packet Switching - Datagram

- Messages broken into smaller units (packets)
- Source & destination addresses in packet header
- Connectionless, packets routed independently (datagram)
- Packet may arrive out of order
- Pipelining of packets across network can reduce delay, increase throughput
- Lower delay that message switching, suitable for interactive traffic



Packet Switching Delay

Assume three packets corresponding to one message traverse same path

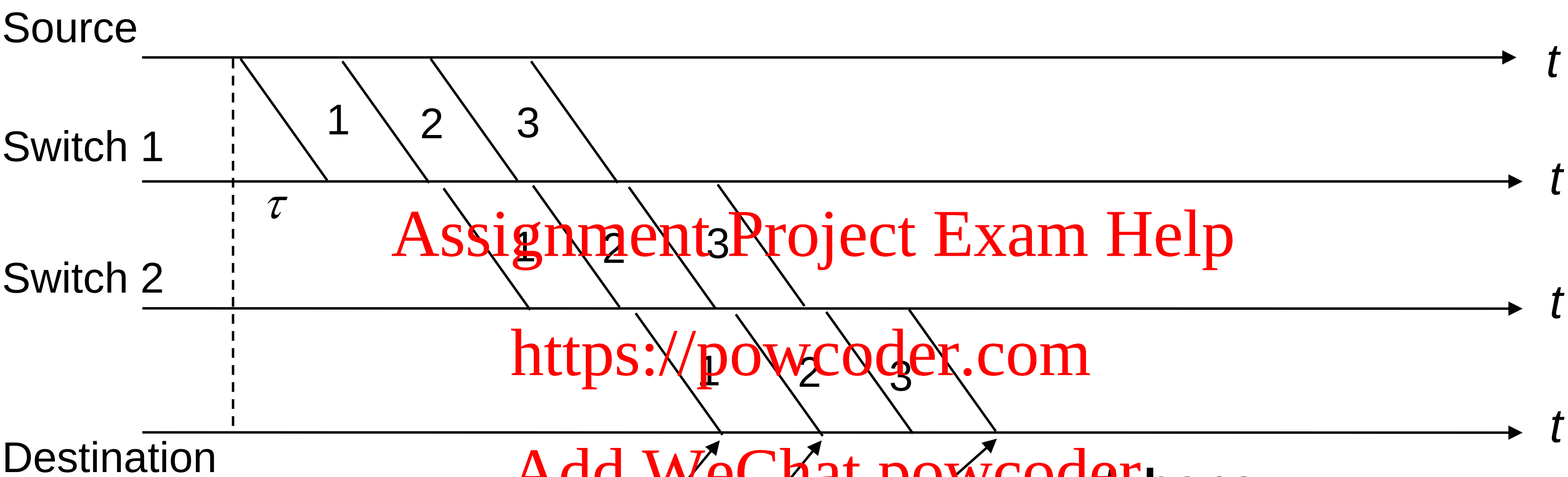


Minimum Delay = $3\tau + 5(T/3)$ (single path assumed)

Additional queueing delays possible at each link

Packet pipelining enables message to arrive sooner

Delay for k-Packet Message over L Hops



3 hops

$3\tau + 2(T/3)$ first bit received

$3\tau + 3(T/3)$ first bit released

$3\tau + 5\ (T/3)$ last bit released

***L* hops**

$L\tau + (L\text{-}1)P$ first bit received

$L\tau + LP$ first bit released

$L\tau + LP + (k\text{-}1)P$ last bit released
where $T = k\ P$

Routing Tables in Datagram Networks

Destination address	Output port
0785	7
1345	12
1566	6
2458	12

- Route determined by table lookup
- Routing decision involves finding next hop in route to given destination
- Routing table has an entry for each destination specifying output port that leads to next hop
- Size of table becomes impractical for very large number of destinations

Example: Internet Routing

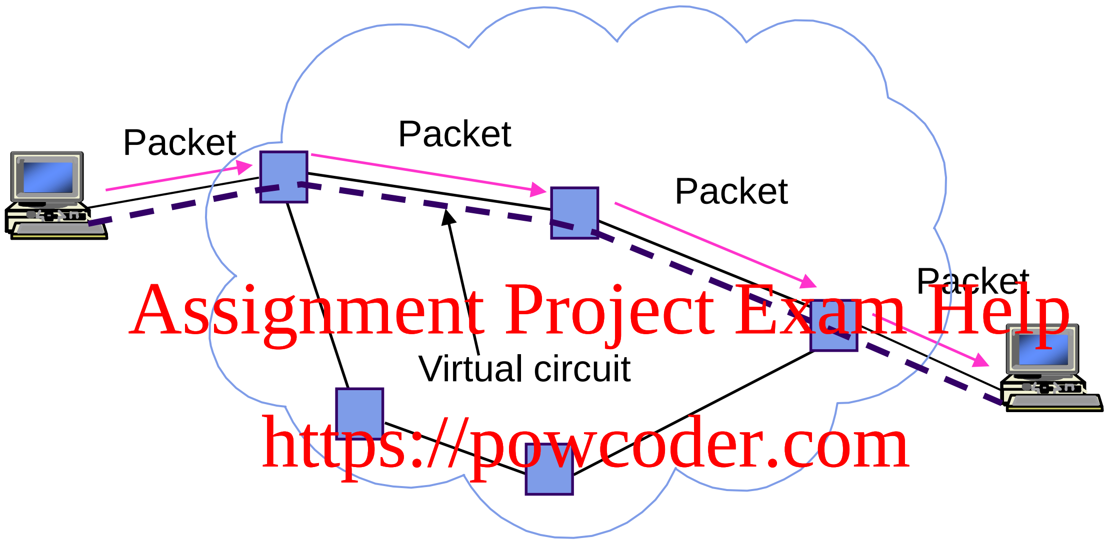
- Internet protocol uses datagram packet switching *across networks*
  - Networks are treated as data links

- Hosts have two-port IP address:

  - Network address + Host address

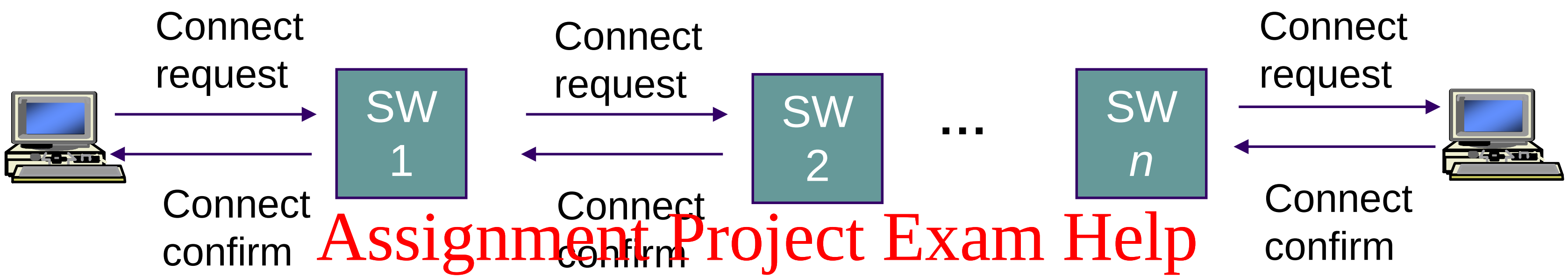
- Routers do table lookup on network address
  - This reduces size of routing table
- In addition, network addresses are assigned so that they can also be aggregated
  - Discussed as CIDR in Chapter 8

Packet Switching – Virtual Circuit



- Call set-up phase sets up pointers in fixed path along network
- All packets for a connection follow the same path
- Abbreviated header identifies connection on each link
- Packets queue for transmission
- Variable bit rates possible, negotiated during call set-up
- Delays variable, cannot be less than circuit switching

Connection Setup

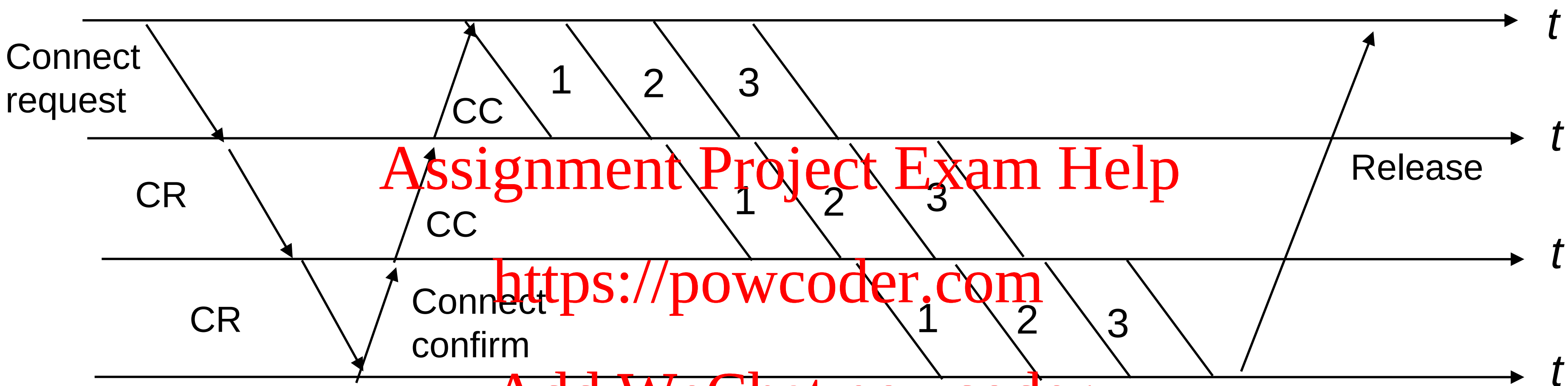


- Signaling messages propagate as route is selected

- Signaling messages identify connection and setup tables in switches

- Typically a connection is identified by a local tag, Virtual Circuit Identifier (VCI)
- Each switch only needs to know how to relate an incoming tag in one input to an outgoing tag in the corresponding output
- Once tables are setup, packets can flow along path

Connection Setup Delay



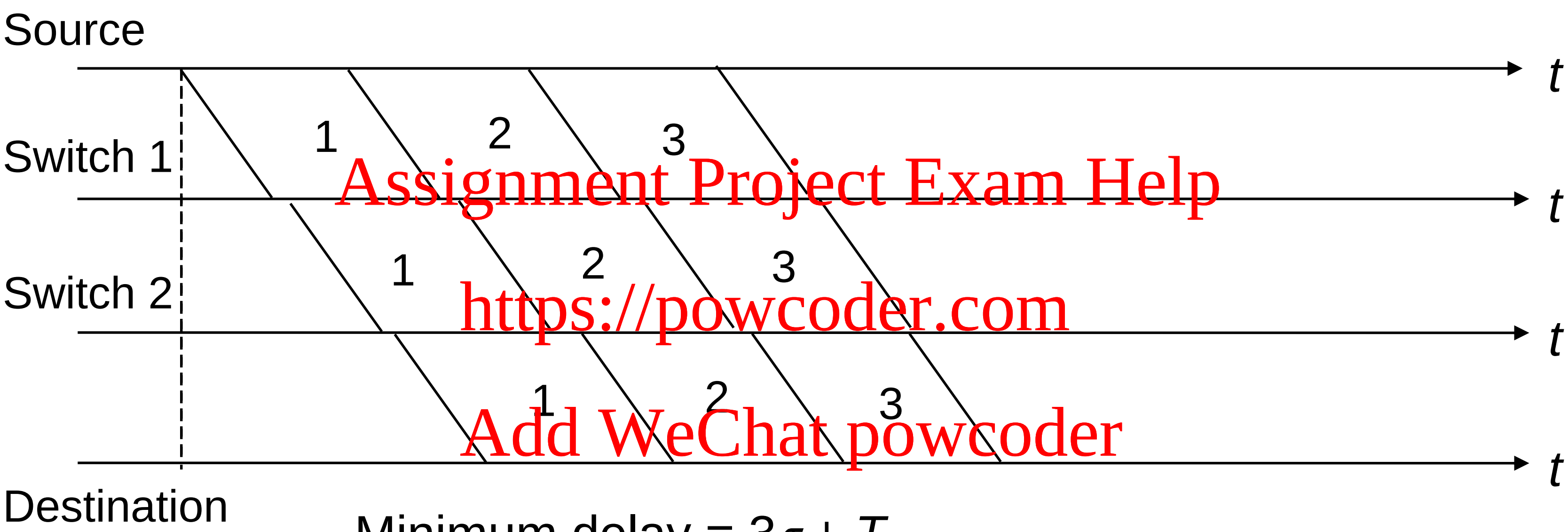
- Connection setup delay is incurred before any packet can be transferred
- Delay is acceptable for sustained transfer of large number of packets
- This delay may be unacceptably high if only a few packets are being transferred

Virtual Circuit Forwarding Tables

Input VCI	Output port	Output VCI
12	13	44
15	15	23
27	13	16
58	7	34

- Each input port of packet switch has a forwarding table
- Lookup entry for VCI of incoming packet
- Determine output port (next hop) and insert VCI for next link
- Very high speeds are possible
- Table can also include priority or other information about how packet should be treated

Cut-Through switching



Minimum delay = $3\tau + T$

- Some networks perform error checking on header only, so packet can be forwarded as soon as header is received & processed
- Delays reduced further with cut-through switching

Message vs. Packet Minimum Delay

- Message:

$$L\,\tau + L\,T \qquad = L\,\tau + (L-1)\,T + T$$

- Packet

$$L\,\tau + L\,P + (k-1)\,P \qquad = L\,\tau + (L-1)\,P + T$$

- Cut-Through Packet (Immediate forwarding after header)

$$= L\,\tau + T$$

Above neglect header processing delays

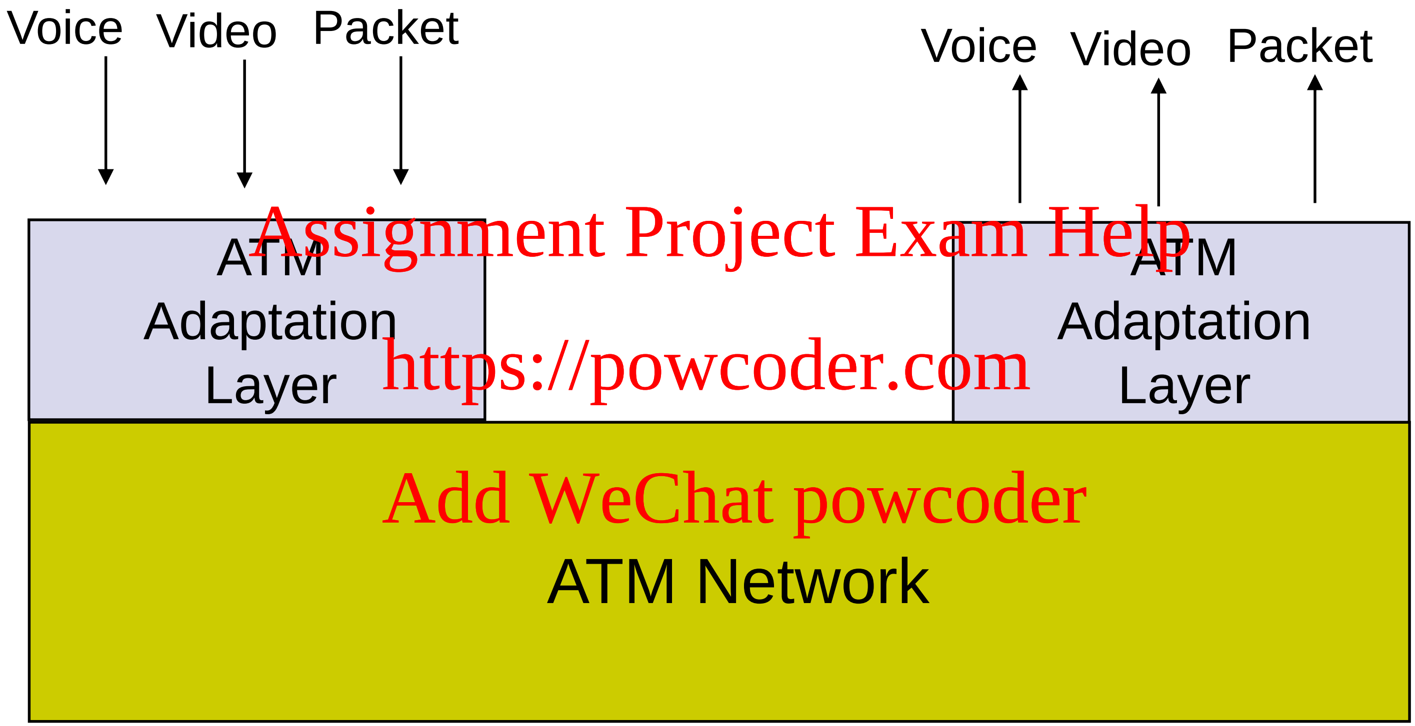
Example: ATM Networks

- All information mapped into short fixed-length packets called *cells*
- Connections set up across network
  - Virtual circuits established across networks
  - Tables setup at ATM switches
- Several types of network services offered
  - Constant bit rate connections
  - Variable bit rate connections

Asynchronous Tranfer Mode (ATM)

- Packet multiplexing and switching
  - Fixed-length packets: “cells”
  - Connection-oriented
  - Rich Quality of Service support
- Conceived as end-to-end
  - Supporting wide range of services
    - Real time voice and video
    - Circuit emulation for digital transport
    - Data traffic with bandwidth guarantees
- Detailed discussion in Chapter 9

ATM Networking



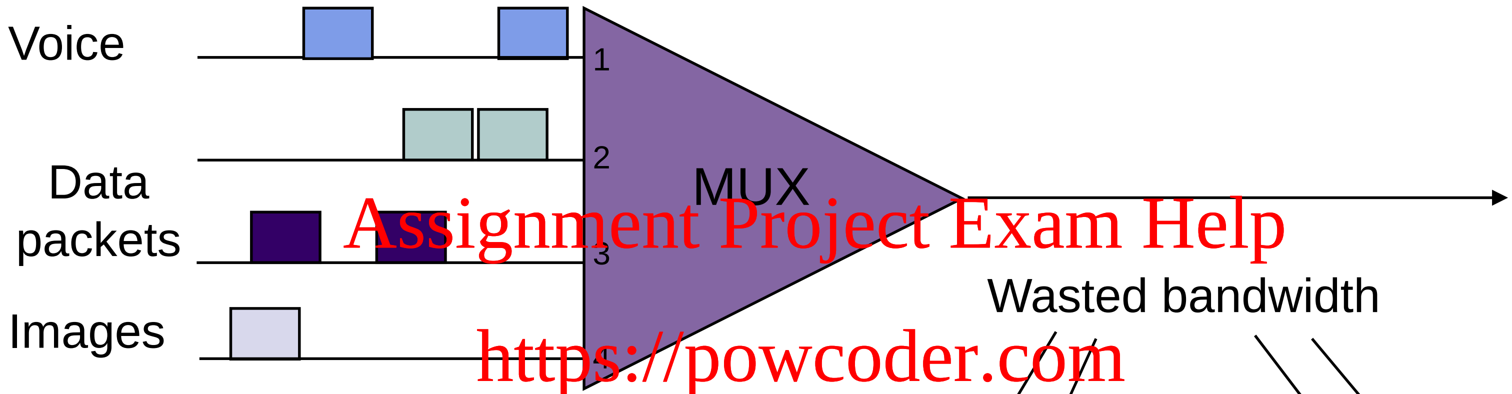
- End-to-end information transport using cells
- 53-byte cell provide low delay and fine multiplexing granularity
- Support for many services through ATM Adaptation Layer

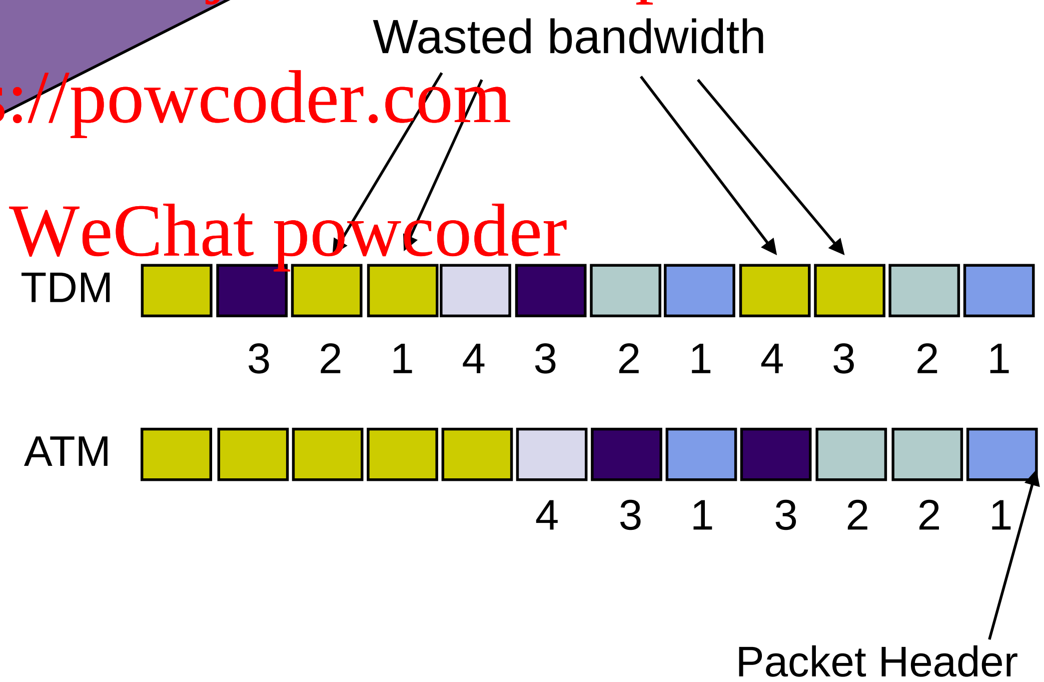
TDM vs. Packet Multiplexing

	Variable bit rate	Delay	Burst traffic	Processing
TDM	Multirate only	Low, fixed ✓	Inefficient	Minimal, very high speed
Packet	Easily handled ✓	Variable	Efficient ✓	Header & packet processing required *

* In mid-1980s, packet processing mainly in software and hence slow; By late 1990s, very high speed packet processing possible

ATM: Attributes of TDM & Packet Switching

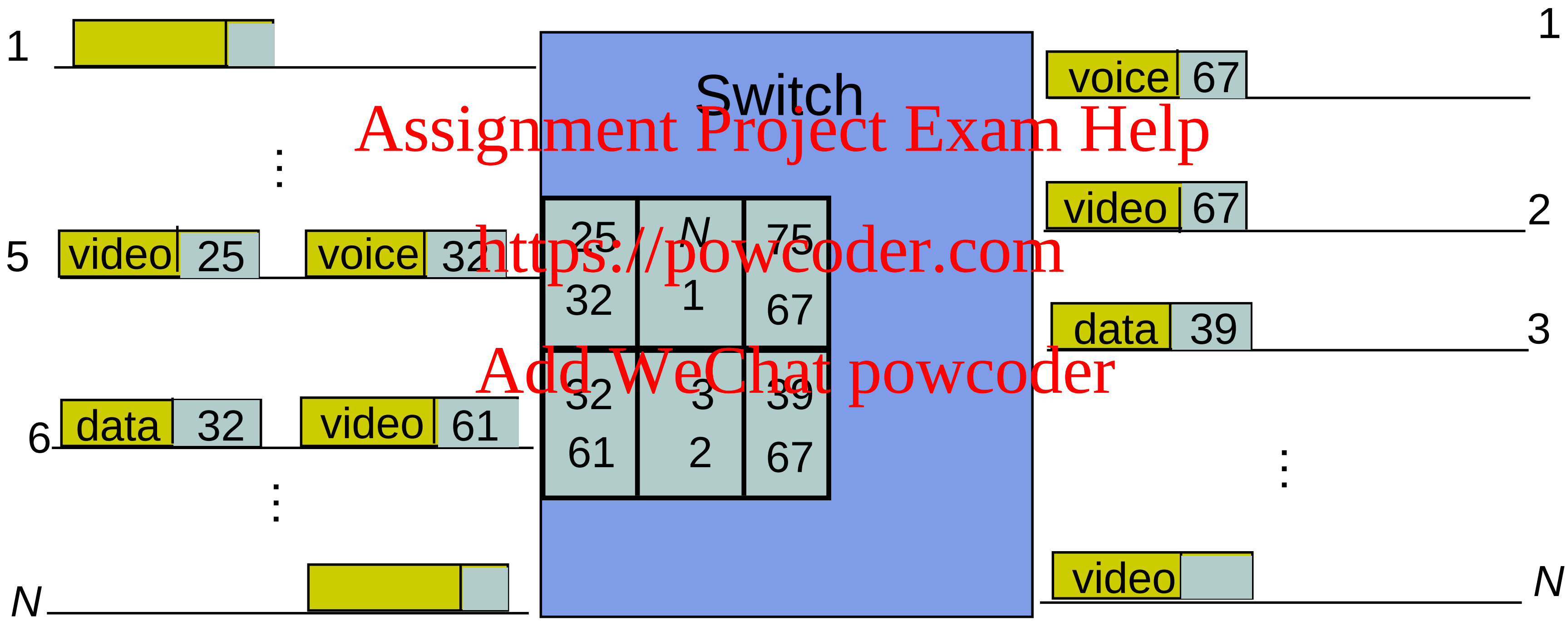




- Packet structure gives flexibility & efficiency
- Synchronous slot transmission gives high speed & density

ATM Switching

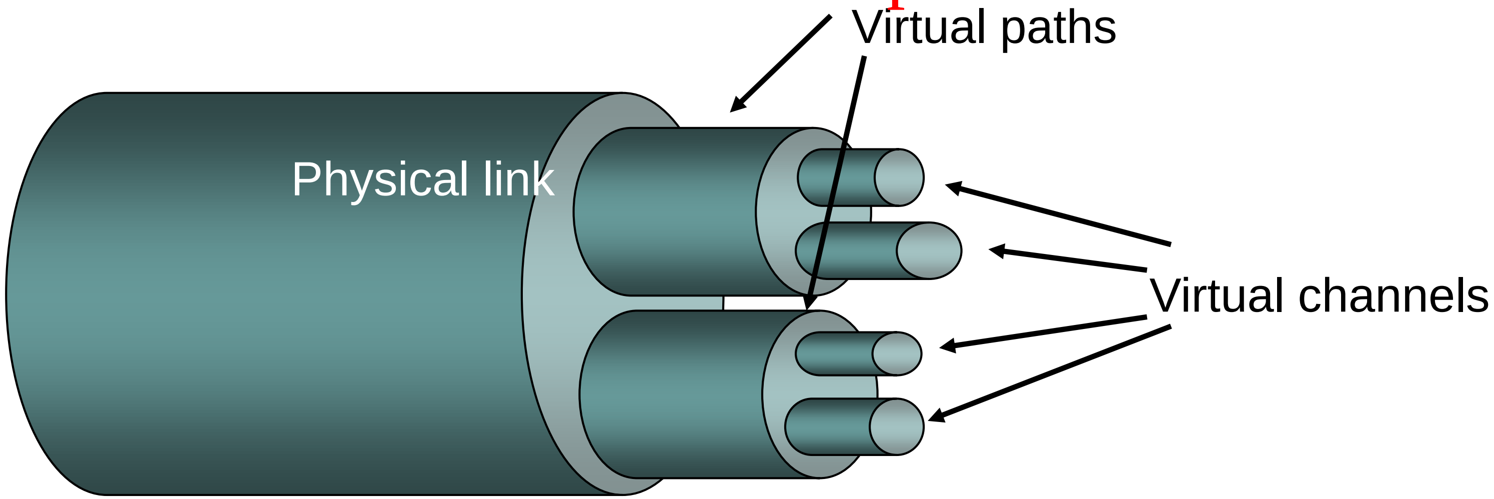
Switch carries out table translation and routing



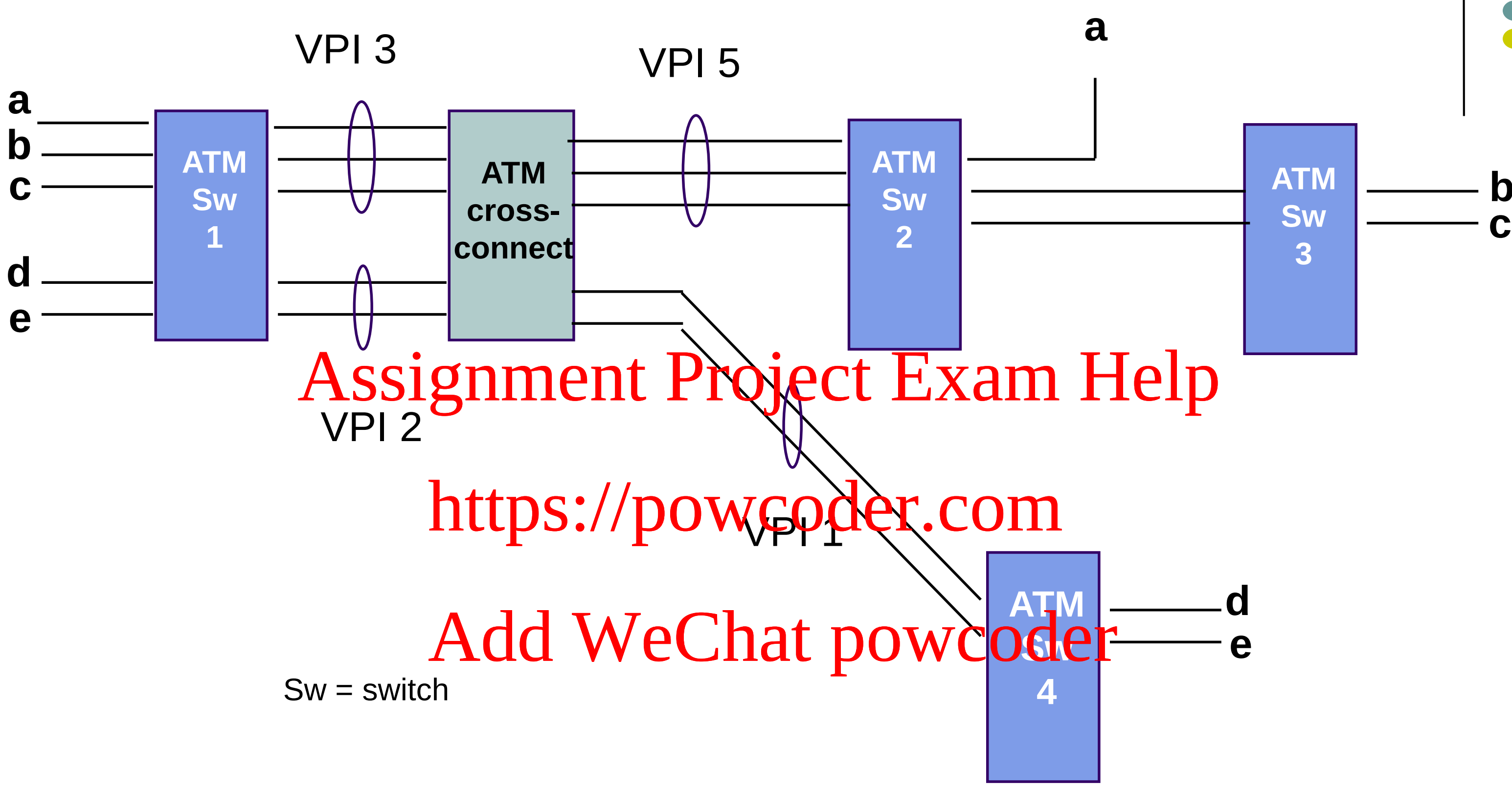
ATM switches can be implemented using shared memory, shared backplanes, or self-routing multi-stage fabrics

ATM Virtual Connections

- Virtual connections setup across network
- Connections identified by locally-defined tags
- ATM Header contains virtual connection information:
  - 8-bit Virtual Path Identifier (VPI)
  - 16-bit Virtual Channel Identifier (VCI) – local identifier
- Powerful traffic grooming capabilities
  - Multiple VCs can be bundled within a VP (flows that have a common path through the network are grouped together)
  - Similar to tributaries with SONET, except variable bit rates possible



VPI/VCI switching & multiplexing



- Connections a,b,c bundled into VP at switch 1
  - Crossconnect switches VP without looking at VCIs
  - VP unbundled at switch 2; VC switching thereafter
- VPI/VCI structure allows creation virtual networks
  - Can support large number of connections – provides scalability

MPLS & ATM

- ATM initially touted as more scalable than packet switching
- ATM envisioned speeds of 150-600 Mbps

- Advances in optical transmission proved ATM to be the less scalable: @ 10 Gbps

  - Segmentation & reassembly of messages & streams into 48-byte cell payloads difficult & inefficient

  - Header must be processed every 53 bytes vs. 500 bytes on average for packets
  - Delay due to 1250 byte packet at 10 Gbps = 1 μsec; delay due to 53 byte cell @ 150 Mbps ≈ 3 μsec
- MPLS (Chapter 10) uses tags to transfer packets across virtual circuits in Internet