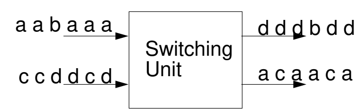
CSE and EE 461
Lecture 6

Summer Quarter 1998

Switching

- **Physical switching:** An old fashioned circuit switching network sets up physical path from one end to the other.
- **Logical switching:** A modern switching network sets up a logical path from one end to the other through switching units. ATDM is used.



- Management of resources - buffer space and outgoing bandwith.

  - on demand - datagram switched network - connectionless

  - preallocated - virtual circuit switched network - connection oriented

Switching Networks

- **Physical circuit switching**

  1. Call request - necessary resources are found
  2. Call accept - notification sender
  3. Data sent
  4. Disconnect

  There is little or no buffering at switches.
- **Message switching**

  Each message is sent independently. The entire message is buffered at switches.
- **Packet switching**

  Messages are broken up into packets. The packets are buffered at switches giving the potential for pipelining the message.

Packet Switching

- **Datagram service**

  - Each packet is routed to its destination independently

  - A message may have to be reconstructed at the end because packets may arrive out of order

  - The Internet Protocol (IP) is a datagram service
- **Internal virtual circuit service**

  - Packets follow a specified route which is set up by a call request

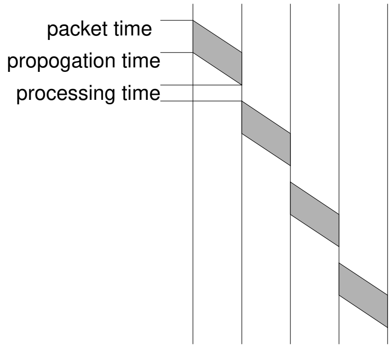
  - Packets do not arrive out of order so reconstructing a message is easier

  - Resources can be preallocated on a per circuit basis

  - Unlike a physical circuit there is packet buffering at switches.

  - The Asynchronous Transfer Mode (ATM) is an internal virtural circuit service.

Packet Switching Latency



packet time = p/b

prop time ~ distance

proc time ~ queuing

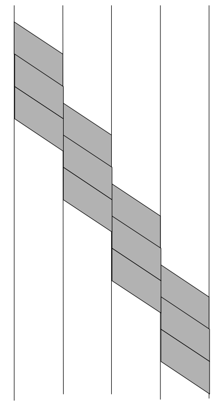
p = packet length

b = bit rate

h = number of hops

Time for first bit to arrive = (h-1)(packet time + proc time) + h(prop time)

Pipelining of packets



Total time = time for first bit to arrive +
sum of packet times
= (h-1)(packet time + proc time) +
h(prop time) + k(packet time)

kp = message size