# High-Speed Networks: Introduction (3)

**Abdulaziz Almulhem**

---

# Recap

- Networking technologies
- The growth in speed (ATM)
- Networking Models (layered archit.)
- Network Protocols
- Internetworking

---

# This lecture

- Traffic characterization
- Switching techniques
- Internetworking, again.

---

# Characteristics of Traffic

- Traffic arrival rate and variability
- Connection duration
- Distribution of message length
- Allowable delay and variability of delay
- Required reliability

---

# Examples of Traffic Types

- Interactive terminal-to-computer sessions
  - » low message rate
  - » message length short
  - » delay requirement moderately strict
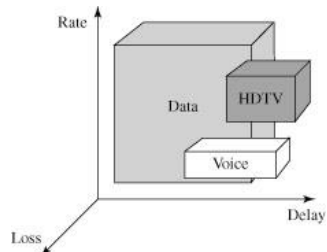  - » required reliability high

---

# Traffic Examples (cont.)

- File transfer sessions
  - » message rate low
  - » message length very long
  - » delay requirement very relaxed
  - » required reliability very high

## QoS requirements of different types of traffic

## QoS Scale
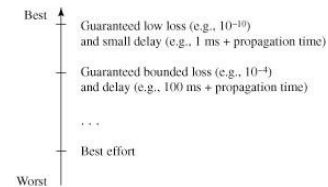
## Circuit Switching

- When session is set up, path is chosen and bandwidth allocated on each link (by FDM or TDM).
  - » If no path with sufficient BW, call is rejected
  - » Advantage: once call is accepted, BW is guaranteed; no queuing
  - » Disadvantage: inefficient utilization of transmission capacity if traffic is bursty

## Packet Switching

- Store and forward
- Statistical multiplexing
  - » No fixed allocation of BW
  - » Packets from different sessions combined into single queue for each outgoing link
  - » Full transmission capacity of link dedicated to single packet
- Advantage: full utilization of link capacity whenever traffic is present

## Connectionless versus Connection-Oriented Routing

- Virtual circuit routing
  - » connection-oriented
  - » fixed path (but not fixed BW) assigned at start of session; all packets follow same path
  - » Example: ATM
- Datagram routing
  - » packets in session are routed independently
  - » Example: IP

## Internetworking

- Heterogeneous working platforms
- Two important functions:
  - » Fragmentation and reassembly
  - » Encapsulation/decapsulation

Figure 2.8   Configuration for TCP/IP Example

---

**1. Preparing the data.** The application protocol prepares a block of data for transmission. For example, an email message (SMTP), a file (FTP), or a block of user input (TELNET).

**2. Using a common syntax.** If necessary, the data are converted to a form expected by the destination. This may include a different character code, the use of encryption, and/or compression.

**3. Segmenting the data.** TCP may break the data block into a number of segments, keeping track of their sequence. Each TCP segment includes a header containing a sequence number and a frame check sequence to detect errors.

**4. Duplicating segments.** A copy is made of each TCP segment, in case the loss or damage of a segment necessitates retransmission. When an acknowledgment is received from the other TCP entity, a segment is erased.

**5. Fragmenting the segments.** IP may break a TCP segment into a number of datagrams to meet size requirements of the intervening networks. Each datagram includes a header containing a destination address, a frame check sequence, and other control information.

**6. Framing.** A frame relay header and trailer is added to each IP datagram. The header contains a connection identifier and the trailer contains a frame check sequence.

**Peer-to-peer dialogue.** Before data are sent, the sending and receiving applications agree on format and encoding and agree to exchange data.

**Peer-to-peer dialogue.** The two TCP entities agree to open a connection.

**Peer-to-peer dialogue.** Each IP datagram is forwarded through networks and routers to the destination system.

**Peer-to-peer dialogue.** Each frame is forwarded through the frame relay network.

**7. Transmission.** Each frame is transmitted over the medium as a sequence of bits.

---

**10. Routing the packet.** IP examines the IP header and makes a routing decision. It determines which outgoing link is to be used and then passes the datagram back to the link layer for transmission on that link.

**Peer-to-peer dialogue.** The router will pass this datagram onto another router or to the destination system.

**9. Processing the frame.** The frame relay layer removes the header and trailer and processes them. The frame check sequence is used for error detection. The connection number identifies the source.

**11. Forming LLC PDU.** An LLC header is added to each IP datagram to form an LLC PDU. The header contains sequence number and address information.

**12. Framing.** A MAC header and trailer is added to each LLC PDU, forming a MAC frame. The header contains address information and the trailer contains a frame check sequence.

**8. Arriving at router.** The incoming signal is received over the transmission medium and interpreted as a frame of bits.

**13. Transmission.** Each frame is transmitted over the medium as a sequence of bits.

---

**20. Delivering the data.** The application performs any needed transformations, including decompression and decryption, and directs the data to the appropriate file or other destination.

**19. Reassembling the data.** If TCP has broken the user data into multiple segments, these are reassembled and the block is passed up to the application.

**18. Processing the TCP segment.** TCP removes the header. It checks the frame check sequence and acknowledges if there is a match and discards for mismatch. Flow control is also performed.

**17. Processing the IP datagram.** IP removes the header. The frame check sequence and other control information are processed.

**16. Processing the LLC PDU.** The LLC layer removes the header and processes it. The sequence number is used for flow and error control.

**15. Processing the frame.** The MAC layer removes the header and trailer and processes them. The frame check sequence is used for error detection.

**14. Arriving at destination.** The incoming signal is received over the transmission medium and interpreted as a frame of bits.