Chapter 7
Multimedia Networking

A note on the use of these ppt slides:
We're making these slides freely available to all (faculty, students, readers). They're in PowerPoint form so you can add, modify, and delete slides (including this one) and slide content to suit your needs. They obviously represent a lot of work on our part. In return for use, we only ask the following:
- If you use these slides (e.g., in a class) in substantially unaltered form, that you mention their source (after all, we'd like people to use our book!)
- If you post any slides in substantially unaltered form on a www site, that you note that they are adapted from (or perhaps identical to) our slides, and note our copyright of this material.

Thanks and enjoy! JFK / KWR

All material copyright 1996-2009
J.F Kurose and K.W. Ross, All Rights Reserved
Multimedia and Quality of Service: What is it?

multimedia applications: network audio and video ("continuous media")

QoS network provides application with level of performance needed for application to function.
Chapter 7: goals

**Principles**
- classify multimedia applications
- identify network services applications need
- making the best of best effort service

**Protocols and Architectures**
- specific protocols for best-effort
- mechanisms for providing QoS
- architectures for QoS
Chapter 7 outline

7.1 multimedia networking applications
7.2 streaming stored audio and video
7.3 making the best out of best effort service
7.4 protocols for real-time interactive applications RTP,RTCP,SIP

7.5 providing multiple classes of service
7.6 providing QoS guarantees
MM Networking Applications

Classes of MM applications:
1) stored streaming
2) live streaming
3) interactive, real-time

Fundamental characteristics:
- typically delay sensitive
  - end-to-end delay
  - delay jitter
- loss tolerant: infrequent losses cause minor glitches
- antithesis of data, which are loss intolerant but delay tolerant.
Streaming Stored Multimedia

Stored streaming:
- media stored at source
- transmitted to client
- *streaming*: client playout begins *before* all data has arrived
- timing constraint for still-to-be transmitted data: in time for playout
Streaming Stored Multimedia: What is it?

1. video recorded
2. video sent
3. video received, played out at client

Cumulative data

network delay

streaming: at this time, client playing out early part of video, while server still sending later part of video
Streaming *Stored* Multimedia: Interactivity

- **VCR-like functionality**: client can pause, rewind, FF, push slider bar
  - 10 sec initial delay OK
  - 1-2 sec until command effect OK

- Timing constraint for still-to-be transmitted data: in time for playout
Streaming *Live* Multimedia

**Examples:**
- Internet radio talk show
- live sporting event

*Streaming* (as with streaming *stored* multimedia)
- playback buffer
- playback can lag tens of seconds after transmission
- still have timing constraint

**Interactivity**
- fast forward impossible
- rewind, pause possible!
Real-Time Interactive Multimedia

- **applications:** IP telephony, video conference, distributed interactive worlds

- **end-end delay requirements:**
  - **audio:** $< 150$ msec good, $< 400$ msec OK
    - includes application-level (packetization) and network delays
    - higher delays noticeable, impair interactivity

- **session initialization**
  - how does callee advertise its IP address, port number, encoding algorithms?
Multimedia Over Today’s Internet

TCP/UDP/IP: “best-effort service”

- no guarantees on delay, loss

But you said multimedia apps require QoS and level of performance to be effective!

Today’s Internet multimedia applications use application-level techniques to mitigate (as best possible) effects of delay, loss
How should the Internet evolve to better support multimedia?

Integrated services philosophy:
- fundamental changes in Internet so that apps can reserve end-to-end bandwidth
- requires new, complex software in hosts & routers

Laissez-faire
- no major changes
- more bandwidth when needed
- content distribution, application-layer multicast

Differentiated services philosophy:
- fewer changes to Internet infrastructure, yet provide 1st and 2nd class service

What’s your opinion?
A few words about audio compression

- analog signal sampled at constant rate
  - telephone: 8,000 samples/sec
  - CD music: 44,100 samples/sec
- each sample quantized, i.e., rounded
  - e.g., 2⁸ = 256 possible quantized values
- each quantized value represented by bits
  - 8 bits for 256 values

- example: 8,000 samples/sec, 256 quantized values --> 64,000 bps
- receiver converts bits back to analog signal:
  - some quality reduction

Example rates
- CD: 1.411 Mbps
- MP3: 96, 128, 160 kbps
- Internet telephony: 5.3 kbps and up
A few words about video compression

- video: sequence of images displayed at constant rate
  - e.g. 24 images/sec
- digital image: array of pixels
  - each pixel represented by bits
- redundancy
  - spatial (within image)
  - temporal (from one image to next)

Examples:
- MPEG 1 (CD-ROM) 1.5 Mbps
- MPEG2 (DVD) 3-6 Mbps
- MPEG4 (often used in Internet, < 1 Mbps)

Research:
- layered (scalable) video
  - adapt layers to available bandwidth
Chapter 7 outline

7.1 multimedia networking applications
7.2 streaming stored audio and video
7.3 making the best out of best effort service
7.4 protocols for real-time interactive applications RTP, RTCP, SIP
7.5 providing multiple classes of service
7.6 providing QoS guarantees
Streaming Stored Multimedia

Application-level streaming techniques for making the best out of best effort service:
- client-side buffering
- use of UDP versus TCP
- multiple encodings of multimedia

Media Player
- jitter removal
- decompression
- error concealment
- graphical user interface w/ controls for interactivity
Internet multimedia: simplest approach

- audio or video stored in file
- files transferred as HTTP object
  - received in entirety at client
  - then passed to player

audio, video not streamed:
- no, “pipelining,” long delays until playout!
Internet multimedia: streaming approach

- browser GETs metafile
- browser launches player, passing metafile
- player contacts server
- server streams audio/video to player
Streaming from a streaming server

- allows for non-HTTP protocol between server, media player
- UDP or TCP for step (3), more shortly

Diagram:
- Web Browser
- Web Server
- Media Player
- Streaming Server

(1) HTTP request/response for presentation description file
(2) presentation description file
(3) audio/video file requested and sent
Streaming Multimedia: Client Buffering

- client-side buffering, playout delay compensate for network-added delay, delay jitter
Streaming Multimedia: Client Buffering

- client-side buffering, playout delay compensate for network-added delay, delay jitter
Streaming Multimedia: UDP or TCP?

**UDP**
- server sends at rate appropriate for client (oblivious to network congestion!)
  - often send rate = encoding rate = constant rate
  - then, fill rate = constant rate - packet loss
- short playout delay (2-5 seconds) to remove network jitter
- error recover: time permitting

**TCP**
- send at maximum possible rate under TCP
- fill rate fluctuates due to TCP congestion control
- larger playout delay: smooth TCP delivery rate
- HTTP/TCP passes more easily through firewalls
Streaming Multimedia: client rate(s)

Q: how to handle different client receive rate capabilities?

- 28.8 Kbps dialup
- 100 Mbps Ethernet

A: server stores, transmits multiple copies of video, encoded at different rates
Interactive Multimedia: Internet Phone

Introduce Internet Phone by way of an example

- speaker’s audio: alternating talk spurts, silent periods.
  - 64 kbps during talk spurt
  - pkts generated only during talk spurts
  - 20 msec chunks at 8 Kbytes/sec: 160 bytes data

- application-layer header added to each chunk.
- chunk+header encapsulated into UDP segment.
- application sends UDP segment into socket every 20 msec during talkspurt
Internet Phone: Packet Loss and Delay

- **network loss**: IP datagram lost due to network congestion (router buffer overflow)

- **delay loss**: IP datagram arrives too late for playout at receiver
  - delays: processing, queueing in network; end-system (sender, receiver) delays
  - typical maximum tolerable delay: 400 ms

- **loss tolerance**: depending on voice encoding, losses concealed, packet loss rates between 1% and 10% can be tolerated.
- consider end-to-end delays of two consecutive packets: difference can be more or less than 20 msec (transmission time difference)
Providing **Multiple Classes of Service**

- thus far: making the best of best effort service
  - one-size fits all service model
- alternative: multiple classes of service
  - partition traffic into classes
  - network treats different classes of traffic differently (analogy: VIP service vs regular service)
- granularity: differential service among multiple classes, not among individual connections
- history: ToS bits