CS640: Introduction to Computer Networks

Aditya Akella

Lecture 19 -
Multimedia Networking

The Road Ahead

- Multimedia requirements
  - Streaming
    - RTSP
  - Recovering from Jitter and Loss
  - RTP
    - RTCP

2

Application Classes

- Typically sensitive to delay, but can tolerate packet loss (would cause minor glitches that can be concealed)
- Data contains audio and video content ("continuous media"), three classes of applications:
  - Streaming stored content
  - Unidirectional Real-Time
  - Interactive Real-Time

3

Application Classes (more)

- Streaming stored content
  - Clients request audio/video files from servers and pipeline reception over the network and display
    - Interactive: user can control operation (similar to VCR: pause, resume, fast forward, rewind, etc.)
  - Streaming → start playing before all content arrives
  - Continuous playout: hard delivery constraints

4

Application Classes (more)

- Unidirectional Real-Time:
  - similar to existing TV and radio stations, but delivery on the network
  - Non-interactive, just listen/view
  - Delivery constraints still important
- Interactive Real-Time:
  - Phone conversation or video conference
  - More stringent delay requirement than Streaming and Unidirectional because of interactive nature
  - Video: < 150 msec acceptable
  - Audio: < 150 msec good, <400 msec acceptable

5

Multimedia Today

- Network is best-effort. But still…
  - Streaming applications delay of 5 to 10 seconds is typical and has been acceptable
  - Real-Time apps work well where there is plentiful bandwidth
- To mitigate impact of "best-effort" network and protocols, we can:
  - Use UDP, avoid TCP and its slow-start phase…
  - Buffer content at client, control playback, prefetch content to remedy delay variation
  - Adapt compression level to available bandwidth in the network
  - Send redundant information to make up for losses
  - Intelligent queueing tricks

6

Solution Approaches in IP Networks

- Just add more bandwidth enhance caching capabilities etc. (previous slide)!
- Need major change of the protocols :
  - Incorporate resource reservation (bandwidth, processing, buffering), and new scheduling policies
  - Set up service level agreements with applications, monitor and enforce the agreements, charge accordingly
- Need moderate changes ("Differentiated Services"):
  - Use two traffic classes for all packets and differentiate service accordingly
  - Charge based on class of packets
  - Network capacity is provided to ensure first class packets incur no significant delay at routers

7

Application Example: Streaming

- Important and growing application
  - Due to reduction of storage costs, increase in high speed net access from homes and enhancements to caching
- Audio/Video file is segmented and sent over either TCP or UDP
  - Web server
  - Streaming server
- Public segmentation protocol: Real-Time Protocol (RTP)
- User Interaction: Real-time Streaming protocol (RTSP)

8

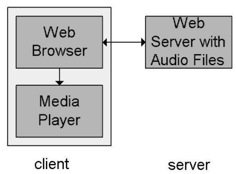
Streaming

- Helper Application: displays content, which is typically requested via a Web browser; e.g. RealPlayer; typical functions:
  - Decompression
  - Jitter removal
  - Error correction: use redundant packets to be used for reconstruction of original stream
  - GUI for user control

9

Streaming From Web Servers

- Audio: in files sent as HTTP objects
- Video (interleaved audio and images in one file, or two separate files and client synchronizes the display) sent as HTTP object(s)
- A simple architecture is to have the Browser request the object(s) and after their reception pass them to the player for display
  - No pipelining



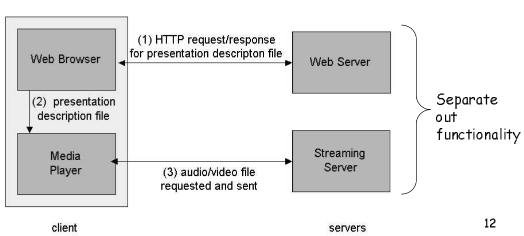
Streaming From Web Server

- Alternative: set up connection between server and player, then download
- Web browser requests and receives a Meta File (a file describing the object) instead of receiving the file itself;
- Browser launches the appropriate Player and passes it the Meta File;
- Player sets up a TCP connection with Web Server and downloads the file using HTTP

11

Using a Streaming Server

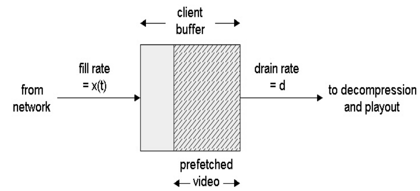
- This gets us around HTTP, allows use of UDP vs. TCP and the application layer protocol can be better tailored to Streaming; many enhancements options are possible …



12

Options When Using a Streaming Server

- UDP: Server sends at a rate (Compression and Transmission) appropriate for client; to reduce jitter, Player buffers initially for 2-5 seconds, then starts display
- Use TCP, and sender sends at maximum possible rate under TCP; retransmit when error is encountered; Player uses a much large buffer to smooth delivery rate of TCP



13

Real Time Streaming Protocol (RTSP)

- For user to control display: rewind, fast forward, pause, resume, etc…
- Out-of-band protocol (uses two connections, one for control messages (Port 554) and one for media stream)
- As before, meta file is communicated to web browser which then launches the Player;
  - Meta file contains "presentation description file" which has information on the multi-media content

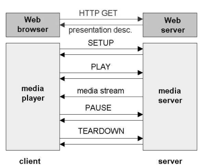
14

Presentation Description Example

```
<title>Xena: Warrior Princess</title>
<session>
      <group language=en lipsync>
            <switch>
               <track type=audio
                     e="PCMU/8000/1"
                     src = "rtsp://audio.example.com/xena/audio.en/lofi">
               <track type=audio
                     e="DVI4/16000/2" pt="90 DVI4/8000/1"
                     src="rtsp://audio.example.com/xena/audio.en/hifi">
            </switch>
         <track type="video/jpeg"
                     src="rtsp://video.example.com/twister/video">
      </group>
</session>
```

15

RTSP Operation



- C: SETUP rtsp://audio.example.com/xena/audio RTSP/1.0
  Transport: rtp/udp; compression; port=3056; mode=PLAY
- S: RTSP/1.0 200 1 OK
  Session 4231
- C: PLAY rtsp://audio.example.com/xena/audio.en/lofi RTSP/1.0
  Session: 4231
  Range: npt=0- (npt = normal play time)
- C: PAUSE rtsp://audio.example.com/xena/audio.en/lofi RTSP/1.0
  Session: 4231
  Range: npt=37
- C: TEARDOWN rtsp://audio.example.com/xena/audio.en/lofi RTSP/1.0
  Session: 4231
- S: 200 3 OK

16

Real-Time (Phone) Over IP's Best-Effort

- Internet phone applications generate packets during talk spurts
- Bit rate is 8 KBytes, and every 20 msec, the sender forms a packet of 160 Bytes + a header
- The coded voice information is encapsulated into a UDP packet and sent out
- Packets may be arbitrarily delayed or lost
  - When to play back a chunk?
  - What to do with a missing chunk?

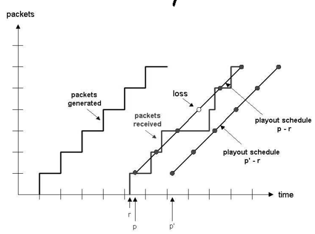
17

Removing "Jitter"

- Decision on when to play out a chunk affected by network "jitter"
  - Variation in queueing delays of chunks
- One option: ignore jitter and play chunks as and when they arrive
  - Can become highly unintelligible, quickly
- But jitter can be handled using:
  - sequence numbers
  - time stamps
  - delaying playout

18

Fixed Playout Delay



- Trade-off between lost packets and large delays
- Can make play-out even better with “adaptive play-out”

19

Recovery From Packet Loss

- Loss interpreted in a broad sense: packet never arrives or arrives later than its scheduled playout time
- Since retransmission is inappropriate for Real Time applications, FEC or Interleaving are used to reduce loss impact and improve quality
- FEC is Forward Error Correction
  - Simplest FEC scheme adds a redundant chunk made up of exclusive OR of a group of n chunks
    - Can reconstruct if at most one lost chunk
  - Redundancy is 1/n, bad for small n
  - Also, play out delay is higher

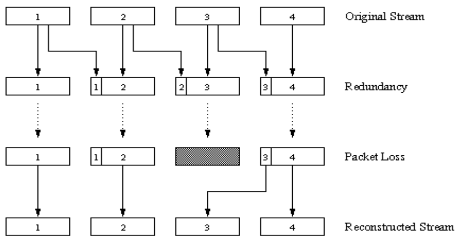
20

Another FEC Mechanism

- Send a low resolution audio stream as redundant information
- Upon loss, playout available redundant chunk
  - Albeit a lower quality one
- With one redundant low quality chunk per chunk, scheme can recover from single packet losses

21

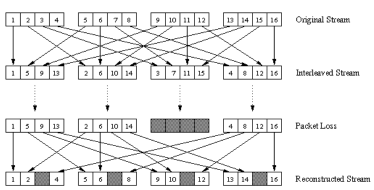
Piggybacking Lower Quality Stream



22

Interleaving

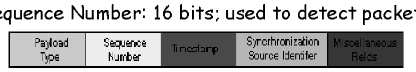
- Divide 20 msec of audio data into smaller units of 5 msec each and interleave
- Upon loss, have a set of partially filled chunks
- Has no redundancy, but can cause delay in playout beyond Real Time requirements



23

Real-Time Protocol (RTP)

- Provides standard packet format for real-time application
- Application-level; Typically runs over UDP
- Specifies header fields for identifying payload type, detecting packet loss, accounting for jitter etc.
- Payload Type: 7 bits, providing 128 possible different types of encoding; eg PCM, MPEG2 video, etc.
- Sequence Number: 16 bits; used to detect packet loss



RTP Header

24

Real-Time Protocol (RTP)

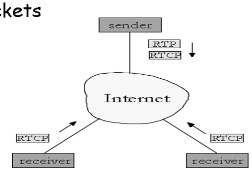
- Timestamp: 32 bytes; gives the sampling instant of the first audio/video byte in the packet; used to remove jitter introduced by the network
- Synchronization Source identifier (SSRC): 32 bits; an id for the source of a stream; assigned randomly by the source

RTP Header

25

RTP Control Protocol (RTCP)

- Protocol specifies report packets exchanged between sources and destinations of multimedia information
- Three reports are defined: Receiver reception, Sender, and Source description
- Reports contain statistics such as the number of packets sent, number of packets lost, inter-arrival jitter
- Used to modify sender transmission rates and for diagnostics purposes



RTCP Bandwidth Scaling

- If each receiver sends RTCP packets to all other receivers, the traffic load resulting can be large
- RTCP adjusts the interval between reports based on the number of participating receivers
- Typically, limit the RTCP bandwidth to 5% of the session bandwidth, divided between the sender reports (25%) and the receivers reports (75%)

27