What is Streaming?

- Streaming media refers to media types with time constraints and a continuous data flow.
  - Video and audio transmission.
- Playback of streaming media starts while data is being received, i.e. it is not necessary to download the entire file before playback starts.
  - Different from MP3 downloads (e.g. using Napster).
- Streaming media is becoming more and more important on the Internet.
  - Commercial usage,
  - Private usage.
- “The streaming media server, advertising, subscription services, and online music shipment market is expected to grow to $14.9 billion by 2009.”
  (Source: Web page of Global Information, Inc.)
Application Examples

- Live, non-interactive applications:
  - Internet radio, news broadcasts, sporting events, etc.,
  - Content is typically not recorded in advance,
  - Loose delay constraints.
- Live, interactive applications:
  - Teleconferencing, video/audio phones, distributed games, etc.,
  - Tight delay constraints to support interactivity.
- Stored, non-interactive applications (on-demand):
  - Movies over the Internet, news archives, video clips on homepages, etc.,
  - Content recorded in advance; loose delay constraints.

Application Software

- Public domain software for streaming over the MBone:
  - VideoConference (VIC),
  - INRIA Video Conferencing System (IVS),
  - Visual Audio Tool (VAT),
  - Robust Audio Tool (RAT),
  - Network Voice Terminal (Nevot).
- Dominant players in the commercial world:
  - Microsoft (Windows Media Player, Silverlight),
  - Adobe (Flash),
  - Apple (QuickTime Player).
Example: RealPlayer ("click play button")

Example: RealPlayer ("buffering")
Streaming on the Internet - Today

- Current streaming technology is in a relatively primitive stage, despite great research work.
- Dominant market players are Microsoft, Adobe, and Apple.
- Most streaming applications are based on unicast transport:
  - Server load increases linearly with the number of clients.
  - Bandwidth intensive multimedia flows lead to serious network congestion.
  - Clients experience unpredictable playback quality and high start-up latency.
Example: Clinton’s Testimony

- CNN audio and video quality became unbearable for most people at around 1:00 pm on August 17, 1998.
- Link to video stream removed from CNN Web page by 1:15 pm.
- Other news servers were also unreachable.

Using IP Multicast for Streaming Media

- IP multicast replicates data packets at branching points in the network.
- Benefits of using IP multicast technology:
  - Reduction of server load,
  - Reduction of network load.
- Problems with multicast-based streaming:
  - IP multicast is not yet widely deployed (lack of business model, management issues, does not yet work properly across domains, etc.).
  - Multicast is useful for live broadcasts; it is not directly applicable to on-demand style services (because it requires synchronous receivers).
  - Multicast does not address the problem of high start-up latency.
Caching of Streaming Media

- Client requests for streaming media are handled by the local cache:
  - *Cache miss*: Request is forwarded to the origin server, which starts playback through the cache; the cache relays the data and simultaneously stores the stream.
  - *Cache hit*: The cache starts playback of the requested stream.

- Benefits of caching:
  - Reduced server load,
  - Reduced network load,
  - Improved playback quality,
  - Improved start-up latency.

Which Would You Rather Be Watching?

A loading bar...  ... or the real image
Connecting
Buffering
Buffering
1 sec
2 sec
3 sec
4 sec
5 sec
Playing

Advanced Networking (SS 16): 13 – Caching Techniques for Streaming Media
Server no longer reachable
Please try again later

Network Congestion..............
The Answer is easy…

The real image !!!
Caching Techniques for Streaming Media

- Why is streaming media different from conventional web traffic?
  - Different signaling protocols (HTTP vs. RTSP),
  - Object size: caching of streaming objects in their entirety does not scale,
  - Time constraints of continuous media.
- Classical caching techniques are not feasible for streaming media.
- Innovative streaming solutions will adapt to user behavior and will scale with increasing demand of streaming media.
  - Fast prefix transfer reduces access delay,
  - Scalability is achieved by segmentation of streaming objects,
  - Dynamic caching increases throughput through request aggregation.

Streaming Media Protocols

**IETF Standard Streaming Protocols**
  - Control protocol for streaming (like a VCR remote control),
  - Superset of HTTP 1.1 with streaming extensions.
  - Framework for transport of real-time data.
- RTCP: The RTP Control Protocol.
  - Enables feedback on session quality/status.
- DASH

**Widely-Used Proprietary Protocols**
  - Real Networks proprietary protocol corresponding to RTSP.
- RDT: Real Data Transport protocol.
  - Real Networks proprietary protocol corresponding to RTP.
- MMS: Microsoft’s proprietary protocol for streaming.
Real Time Streaming Protocol (RTSP)

- RTSP controls transmission of streaming objects (“control for remote VCRs”).
- RTSP is different from HTTP.
  - Data carried out of band (RTP),
  - RTSP server is state full,
  - Uses different methods (PLAY, PAUSE, RECORD, etc.).
- RTSP inherits from HTTP.
  - Re-use HTTP extensions.
    - Authentication,
    - Cache control.
- Media stream and sessions defined by a RTSP URL.
  - rtsp://www.streamcorp.com:554/startrek/autrack

Layering Of Streaming Protocols

Application Layer  
HTTP  |  RTSP  |  RTP  |  RTCP
-----------------------------------------------
Transport Layer    
TCP  |  UDP
-----------------------------------------------
Network Layer      
IP
RTSP Operations

HTTP GET
Session description

RTSP
Server

RTSP Client

RTSP
Client

RTSP
Server

SETUP

PLAY
RTP audio
RTP video
RTCP
PAUSE
TEAR DOWN

On The Need Of Client Side Buffering (1)

Server

Client

Playback
Interruption
Buffered Playback

- Streaming applications maintain a playback buffer to absorb jitter.
- Playback starts when the playback buffer is filled completely.
Example: Access Delay

<table>
<thead>
<tr>
<th>Station</th>
<th>Buffer Delay [sec]</th>
<th>Connection Delay [sec]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vintage Rock</td>
<td>1.3</td>
<td>6.1</td>
</tr>
<tr>
<td>PLJ</td>
<td>1.2</td>
<td>6.3</td>
</tr>
<tr>
<td>FM Champl</td>
<td>2.1</td>
<td>6.6</td>
</tr>
<tr>
<td>NDR 2</td>
<td>3.1</td>
<td>6.4</td>
</tr>
<tr>
<td>SRA-FM</td>
<td>4.0</td>
<td>7.6</td>
</tr>
</tbody>
</table>

- Buffer Delay [sec]
- Connection Delay [sec]

Fast Prefix Transfer

- Fast prefix transfer at the beginning of the data transfer decreases buffer delay.
- Knowledge of the buffer size at the client allows for optimal improvement.
Segmentation of Streaming Objects (1)

**Without Segmentation**
- Initial cache content
- Cache content after receiving request for object B

**With Segmentation**
- Initial cache content
- Cache content after receiving request for object B

- Division of streaming objects into smaller segments.
- Segments can be cached and replaced independently.

Segmentation of Streaming Objects (2)

- Cooperation among caches allows the use of segments from multiple caches
- Timing relation among segments must be controlled:
  - Intelligent fetching of segments,
  - Intelligent cache replacement to avoid gaps.
Cache Replacement for Streaming Media

- Goal: fine grained cache replacement, while avoiding data gaps.
- Cache replacement algorithm:
  - Determine the victim segment,
  - Determine the chunk to which the victim belongs,
  - Eject the last segment of the victim chunk.

Dynamic Caching

- Asynchronous requests characterized by temporal distance $\Delta$.
- Dynamic cache bridges temporal distance.
  - Enables use of multicast even for asynchronous requests.
- Size of dynamic cache independent from size of media object.
- Dynamic caching requires data patching.
Simulation Results

- Network simulation based on MCI backbone topology using network simulator ns-2.
- Servers hold 100 different streaming objects.
  - Access to streaming objects according to Zipf distribution.
- Playback rate: 1.33 Mbps.
- Size of streaming objects: 1 GB.
- Cache size: 5 GB.
- Simulation time: 10 hours.
- Time interval between two successive client requests: 10 sec.

Benefits of Segmentation