Architecture and Mechanisms for High Quality Streaming Multimedia

Tze Sing Eugene Ng
Carnegie Mellon University

Joint work with Katie Guo, Markus Hofmann, Sanjoy Paul, and Hui Zhang
Outline

- Motivation
- Assumptions
- Project goals
- Architecture overview
- Basic mechanisms
- Related work
- Conclusion
Current Streaming Architecture

Unicast

Server

Internet

Receivers
Problems with Current Architecture

- Content provider
  - Server load increases linearly with the number of receivers
- Receiver
  - High start-up latency
  - Unpredictable playback quality
  - Poor performance with VCR operations
- ISP
  - Streaming multimedia flows lead to serious network congestion problems
Example: News Coverage of President Clinton’s Testimony

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

Multicast

Server

Internet

Receivers
Multicast for Live Broadcast

- Reduces both server load and network load
- Does not improve quality of service to receivers
Multicast for On-Demand Streaming

• Batching
  – Requests arriving within a time window $\Delta t$ are batched together and are served by one multicast session
• Reduces both server load and network load
• Does not improve quality of service to receivers
  – Start-up latency increases
A Fundamental Problem with Multicast

- Multicast works best when receivers are homogeneous and synchronous
- In reality, receivers are heterogeneous and asynchronous
Solution: Buffer, Buffer, More Buffer

- Buffering can mask both the heterogeneity and asynchrony of receivers
Assumptions

• Application environment
  – Live broadcasts
  – On-demand long streams
  – On-demand short clips
  – VCR operations allowed

• Network environment
  – Assume minimal QoS support
  – Can take advantage of underlying QoS if available

• Infeasible to replicate all streaming multimedia objects in their entirety
Project Goals

- Reduce start-up latency
- Improve playback quality
- Improve performance for VCR operations
- Reduce server load
- Reduce network load
New Streaming Architecture
New Streaming Architecture

Server

Helpers

Receivers
New Streaming Architecture

Server

Layer-4 Switches

 Helpers

 Receivers
Transparent Operation

Sender

Helper

Layer-4 Switch

Receiver
Transparent Operation

Sender

Layer-4 Switch

Helper

Receiver
Transparent Operation

Sender

Layer-4 Switch

Helper

Receiver
Transparent Operation

Sender

Helper

Layer-4 Switch

Receiver
Transparent Operation

Sender

Layer-4 Switch

Helper

Receiver
Helper Mesh Formation
Helper Mesh Formation

Buffer

2 $\Delta t$

$H1$

$R1$

t$_0$
Helper Mesh Formation

Buffer

2 $\Delta t$

$H^1$

$R_1$

$t_0$

$R_2$

$t_0 + \Delta t$
Helper Mesh Formation

Buffer

2 $\Delta t$

$H_1$

R1

R2

t$_0$

t$_0 + \Delta t$
Helper Mesh Formation

$H_1$

Buffer

2 $\Delta t$

t$_0$

t$_0$ + $\Delta t$

$R_1$

$R_2$

$\Delta t$

Buffer
Helper Mesh Formation

Buffer

$2 \Delta t$

$H_1$

$H_2$

$R_1$

$t_0$

$R_2$

$t_0 + \Delta t$

$R_3$

$t_0 + 2\Delta t$
Helper Mesh Formation

Buffer

2 $\Delta t$

$H1$

$H2$

$R1$

$t_0$

$R2$

$t_0 + \Delta t$

$R3$

$t_0 + 2\Delta t$
Helper Mesh Formation

Buffer

2 $\Delta t$

$H_1$

$H_2$

$R_1$

$t_0$

$R_2$

$t_0 + \Delta t$

$R_3$

$t_0 + 2\Delta t$
Helper Mesh Formation

Buffer

2 $\Delta t$

$H_1$

$H_2$

$R_1$

$R_2$

$R_3$

$t_0$

$t_0 + \Delta t$

$t_0 + 2\Delta t$
Helper Mesh Formation

Buffer

$2\Delta t$

$H_1$

$H_2$

$R_1$

$t_0$

$R_2$

$t_0 + \Delta t$

$R_3$

$t_0 + 2\Delta t$
Using Buffer in the General Case

$2 \Delta t$

$t_0$

$t_0 + \Delta t$

$t_0 + 6\Delta t$
Using Buffer in the General Case

2 $\Delta t$  \hspace{2cm} H1  \hspace{2cm} H2

R1  \hspace{2cm} R2  \hspace{2cm} R3

t_0  \hspace{2cm} t_0 + \Delta t  \hspace{2cm} t_0 + 6\Delta t
Using Buffer in the General Case

\[ t_0 \quad t_0 + \Delta t \quad t_0 + 6\Delta t \]

R1 \quad R2 \quad R3

H1 \quad H2

2\Delta t
Using Buffer in the General Case

\[ t_0 + 2\Delta t \]

\[ t_0 + \Delta t \]

\[ t_0 + 6\Delta t \]
Using Buffer in the General Case

4 \Delta t

2 \Delta t

\text{H1}

? 4 \Delta t \text{Catch-up Data}

R1

t_0

R2

t_0 + \Delta t

R3

t_0 + 6\Delta t
Using Buffer in the General Case

\[ t_0 + \Delta t \]

\[ t_0 + 6\Delta t \]

R1

R2

R3

4 \Delta t

2 \Delta t

4 \Delta t \text{ Catch-up Data}

?
Using Buffer in the General Case

\[ t_0 \]

\[ t_0 + \Delta t \]

\[ t_0 + 2\Delta t \]

\[ t_0 + 4\Delta t \]

\[ t_0 + 6\Delta t \]
Defining Parameters Using Segments

Segment
Defining Parameters Using Segments

Segment requested by receiver, $S_r$

Segment
Defining Parameters Using Segments

- $S_r$: Segment requested by receiver,
- $S_I$: Currently buffered data at helper
Defining Parameters Using Segments

- Segment
- Segment requested by receiver, $S_r$
- Currently buffered data at helper
- Additional buffer required to bridge the gap, $S_l - S_r$
Defining Parameters Using Segments

Catch-up Data

Segment requested by receiver, $S_r$
Currently buffered data at helper
Additional buffer required to bridge the gap, $S_l - S_r$
Helper Selection Considerations

- Additional buffer space required
  - Knowledge of temporal distance
- Additional network load incurred
  - Knowledge of network distance
- Additional system load incurred
  - Bandwidth and buffer consumption
  - For load balancing
- Data stream sharing or not
Data Stream Sharing or Not

Multicast

H1

H2

H3

Multicast

H1

H2

H3
Data Stream Sharing or Not

![Diagram showing multicast connections between nodes H1, H2, and H3.]
Data Stream Sharing or Not

Multicast

H1

H2

H3

H1

H2

H3
Data Stream Sharing or Not

Multicast

H1 → H2 → H3

H1 → H2

Multicast

H1 → H3

Tze Sing Eugene Ng        eugeneng@cs.cmu.edu        Carnegie Mellon University
Data Stream Sharing or Not

H1 -> Multicast -> H2

H1 -> Multicast -> H3

H2

H3

Tze Sing Eugene Ng  eugeneng@cs.cmu.edu  Carnegie Mellon University
Data Stream Sharing or Not

Multicast

H1

H2

H3

Multicast

H1

H2

H3
Data Stream Sharing or Not

Multicast

H1

H2

H3

Buffer $\Delta t$

Multicast

H1

H2

H3
Data Stream Sharing or Not

Multicast

Catch-up

Buffer $\Delta t$

Multicast
Data Stream Sharing or Not
Data Stream Sharing or Not

- Multicast
- Catch-up
- Buffer \( \Delta t \)

H1 \rightarrow H2
H1 \rightarrow H3
H1 \rightarrow H1
H2 \rightarrow H3
H3 \rightarrow Buffer \( \Delta t \)
H3 \rightarrow Buffer \( \Delta t \)
Data Stream Sharing or Not

H1

Multicast

H2

Catch-up

H3

Buffer $\Delta t$

H1

Multicast

H2

Catch-up

H3

Buffer $\Delta t$
Data Stream Sharing or Not

Multicast

Catch-up

Buffer $\Delta t$

H1

H2

H3

Multicast

Catch-up

Buffer $\Delta t$

H1

H2

H3
Data Stream Sharing or Not

- Multicast
- Catch-up
- Buffer $\Delta t$

H1 to H2
H1 to H3
H2 to H3
H1 to H3

Buffer $\Delta t$
Example Helper Selection Algorithm

```java
foreach region (local, regional, national, global) {
    if (! sHelper) {
        sHelper = findBestStreamSharingHelper(region);
    }
    if (! nHelper) {
        nHelper = findBestNewStreamHelper(region);
    }
    if (sHelper && nHelper) { break; }
}

if (bufferReq(sHelper) * netDistance(sHelper) <
    bufferReq(nHelper) * netDistance(nHelper)) {
    return sHelper;
} else { return nHelper; }
```
Algorithm in Action

Multicast

$H1$

$t_0 + \Delta t$

$H4$

$t_0 + 3\Delta t$

$H3$

$t_0 + 2\Delta t$

$H2$

$t_0 + \Delta t$

Multicast
Algorithm in Action

- Multicast at $t_0$
- $H1$ at $t_0 + \Delta t$
- $H4$ at $t_0 + 3\Delta t$
- Buffer $3\Delta t$
- $H2$ at $t_0 + \Delta t$
- Multicast
- $H3$ at $t_0 + 2\Delta t$
Algorithm in Action

Multicast

$H_1$ at $t_0 + \Delta t$

$H_4$ at $t_0 + 3\Delta t$

Buffer at $2\Delta t$

Multicast

$H_2$ at $t_0 + \Delta t$

$H_3$ at $t_0 + 2\Delta t$
Algorithm in Action

Multicast

Buffer 2Δt

H1

H4

Buffer 2Δt

H2

Multicast

H3

\[ t_0 + \Delta t \]

\[ t_0 + 2\Delta t \]

\[ t_0 + 3\Delta t \]
Algorithm in Action

Multicast

Buffer 2Δt

Multicast

Buffer 2Δt

H1

H2

H3

H4

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicast

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting

Multicasting
Algorithm in Action

- Multicast
- $t_0$
- $t_0 + \Delta t$
- $H1$
- $t_0 + 2\Delta t$
- $H3$
- $t_0 + 3\Delta t$
- $H4$
- $t_0 + \Delta t$
- Multicast
- Buffer $\Delta t$
- Buffer $2\Delta t$
- $H2$
State Distribution: What?

- Streams currently buffered, for each stream:
  - Current lowest segment sequence number
  - Bandwidth of this stream
  - Source of this stream: Multicast vs Unicast
- Overall buffer used and free
- Overall bandwidth used and free
State Distribution: How?

- TTL Scope
- Frequency of Reception
  - 15: Every Advertisement
  - 31: Every 2nd Advertisement
  - 63: Every 4th Advertisement
  - 127: Every 8th Advertisement

- Listening Helper
- Advertising Helper
Static Caching of Segments

- Short clips
- Hot segments
  - e.g. A particular song in a concert
- Initial segments
  - If receivers tend to start listening from the beginning
  - Further reduce start-up latency
- Catch-up data
- Requires a measure of “hotness”
Related Work

- **Chaining** [Hua, Sheu and Tavanapong, ICMCS ‘97]
  - Every receiver buffers a fixed amount of data
  - Late comers can be served out of receiver buffers
  - Forms a chain of receivers
  - Focuses on reducing server load
Related Work

- Patching [Hua, Cai and Sheu, ACM Multimedia ‘98]
  - Each receiver allocates buffer to catch-up with an on-going multicast session originating from the server
  - Patching refers to the need for catch-up data from the server
  - No buffer sharing, data always originates from server
Conclusion

• By using Helpers as data forwarding, buffering, and caching agents, we believe
  – Streaming multimedia quality can be enhanced
  – Server load can be reduced
  – Network load can be reduced
  – It’s a win, win, win
• Keys to success
  – Access pattern allows data sharing
  – Low mesh setup overhead, responsiveness is critical
  – Low state distribution overhead
• Huge design space remains to be explored
• System is currently being implemented in ns-2