RESEARCH AND DESIGN OF A MOBILE STREAMING MEDIA CONTENT DELIVERY NETWORK

Susie Wee, John Apostolopoulos, Wai-tian Tan, Sumit Roy

Streaming Media Systems Group
HP Labs, Palo Alto, CA USA

ABSTRACT

Delivering media to large numbers of mobile users presents challenges due to the stringent requirements of streaming media, mobility, wireless, and scaling to support large numbers of users. This paper presents a Mobile Streaming Media Content Delivery Network (MSM-CDN) designed to overcome these challenges. The MSM-CDN is a network overlay consisting of overlay servers on top of the existing network; these overlay servers are control points that facilitate end-to-end media delivery and mid-network media services. This paper presents an overview of the MSM-CDN system architecture, and describes the testbed prototype that we built based on these architectural principles. The MSM-CDN provides a new platform for media delivery, and we describe a number of research directions related to the MSM-CDN.

1. INTRODUCTION

Advances in next-generation cellular networks and wireless LANs are bringing higher bandwidths to mobile users. These higher bandwidths naturally create the demand for media-rich applications, which in turn create requirements for a media delivery infrastructure that can handle the challenges of real-time streaming media; user mobility; highly dynamic, error-prone wireless channels; and scaling to large numbers of users.

Even large-scale delivery of web pages has resulted in difficulties such as network congestion and server overload. Content delivery networks (CDNs) were developed to overcome these problems and improve the overall performance of the network. The additional requirements for delivering streaming media to mobile users over wired and wireless networks further exacerbate these problems and motivates the need for a new solution.

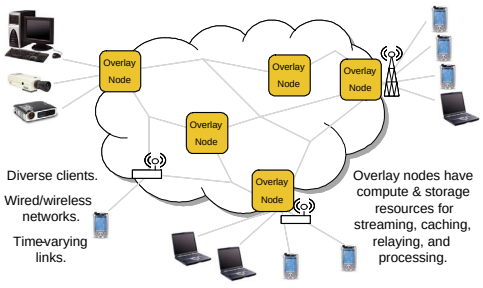

Figure 1: Overlay architecture on the existing underlying network.

In this paper, we present a mobile streaming media content delivery network (MSM-CDN) designed to provide large-scale media delivery services to mobile users. The MSM-CDN is a virtual overlay network placed on top of today's IP networks, as shown in Figure 1. The MSM-CDN is a set of managed or self-managed overlay nodes that work together to deliver media streams to mobile users. These overlay nodes provide valuable control points in the network for improving end-to-end streaming performance.

The MSM-CDN is designed to enable large-scale end-to-end media delivery. Furthermore, the MSM-CDN provides a new platform for research in media delivery. In this paper, Section 2 describes the MSM-CDN architecture. Section 3 discusses research areas related to the MSM-CDN. Section 4 highlights the MSM-CDN testbed prototype which we built. We conclude with final remarks.

2. MSM-CDN SYSTEM ARCHITECTURE

An end-to-end media delivery system must satisfy a number of system requirements. First, a system must be *interoperable* with existing infrastructure. Also, it must be *flexible* to allow customization for different system requirements. In such a way, systems can be deployed incrementally, and adapted for evolving user and system usage patterns. Such flexibility is typically accomplished through *modular* design. Finally, it must be *self-manageable* or *manageable* to allow monitoring of system performance by network operators and users.

This section describes the architecture of a mobile streaming media content delivery network (MSM-CDN) designed to overcome the challenges of large-scale, mobile streaming media delivery, while satisfying the requirements stated above. The MSM-CDN was designed as a network overlay to leverage the connectivity provided by an underlying network. A key design choice was to develop an adaptive architecture that makes the MSM-CDN applicable to a variety of networks and interoperable with other systems, while still allowing customizations for the different requirements of each network.

The MSM-CDN architecture is described in three parts: modular components, component interfaces, and system management.

2.1. MSM-CDN modular components

The MSM-CDN architecture is based on modular components that interact with one another. This modularity allows the system to be deployed incrementally over time in a manner that adapts to user, network, and system load. A more detailed view of the MSM-CDN is shown in Figure 2. As mentioned earlier, the MSM-CDN is a collection of overlay nodes, each of which has compute and storage resources. Each overlay node can consist of a collection of overlay

servers or managers. In the simplest case, an overlay node can be a single overlay server or a single manager. In other cases, a node may contain a cluster of overlay servers and possibly a local manager.

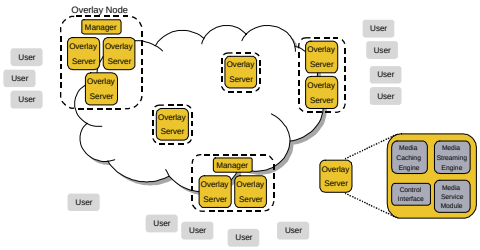

Figure 2: System architecture consisting of servers and managers.

The *overlay servers* are the basic building block of the MSM-CDN. The overlay servers can be used to store or cache media streams in the middle or edge of the network, and to relay media streams across the network. Replicating media content across multiple overlay servers leads to improved streaming performance, and also prevents server overload and improves scalability since the media can be delivered from multiple nodes. Furthermore, the overlay servers have the added role of constantly monitoring their surroundings and sharing their information with other overlay servers and managers. Section 2.4 describes the overlay server in further detail.

The *managers* can be used to gather and analyze system statistics and control various parts of the MSM-CDN. Managers can also be used for operational purposes such as adding or reconfiguring overlay servers. Note that management functionalities can be incorporated into the overlay servers themselves. When this occurs, explicit managers are not needed, but the overlay servers can still work in cooperation with one another.

2.2. MSM-CDN component interfaces

MSM-CDN components interact with one another through their component interfaces. These interfaces allow the components to work cooperatively as they deliver media streams to mobile users. They also allow the system to be reconfigured to handle changing user patterns and time-varying network and system loads.

Media transport is done through streaming and data transfer interfaces. *Streaming interfaces* allow overlay servers to receive streaming inputs from media sources such as streaming servers, live media recorders, or other overlay servers and to send streaming outputs to media players or other overlay servers. Thus, live media streams can be relayed across a network from streaming sources to media clients through one or more overlay servers. *Data transfer interfaces* allow overlay servers to receive and send media files or media segments in a file transfer mode. This interface allows overlay servers to transfer entire media streams or segments of media streams with web servers and with other overlay servers. As one example, the data transfer interface can be used by the overlay server to prefetch media streams for predicted user requests.

Control/management interfaces allow overlay servers and managers to query other MSM-CDN components for information such as content usage statistics, server load, and network congestion. The control interface also allows managers and overlay servers to give and receive commands to and from other MSM-CDN components. This interface allows the overlay servers to cooperate and act as a system to collect and analyze statistics, predict behaviors from these statistics, and perform tasks to serve predicted user requests in a resource-efficient manner.

2.3. MSM-CDN system management

Manageability was a key design goal of the MSM-CDN. MSM-CDN management can be divided into two functions, (1) system monitoring, measurement, and analysis (through queries) and (2) system control (through commands). Both these functions can be performed between components through the management/control interface. The control interface allows the system to accept and give requests and commands. Since each overlay server tracks its own statistics, it can respond to queries for content usage, server load, and network conditions. Also, overlay servers can respond to commands for moving content, beginning and ending streaming sessions, and processing streams. These commands and requests can be from other overlay servers or managers.

The flexibility and modularity of the MSM-CDN architecture allows it to be configured for *centralized or distributed* operation, *push or pull* mode, and its *scalablity and adaptability* allow it to be customized for a large range of deployment scenarios.

2.4. MSM-CDN overlay server functionalities

The basic functionalities of the overlay server include streaming, caching, content distribution, resource monitoring, resource management, and signaling. Furthermore, an overlay server may also have advanced media service functionalities that can perform session management of streaming media sessions and media processing operations on cached and relayed media streams.

The overlay server's basic *streaming* functionalities include the ability to send and receive media streams through its streaming interface. A single scheduler is employed to coordinate the streaming of multiple sessions, thereby allowing inter-session optimization. The overlay server supports session control functions such as start, stop, and pause for outgoing streams and record for incoming streams.

The overlay server has *caching* functionalities that allow it to store and retrieve media streams to and from the disk. For overlay servers that are close to a client, caching can achieve reduced latency for media access, and reduce total amount of network traffic. The cached content can also be locked for a specified period of time to prevent eviction from cache replacement policies.

The overlay server has *content distribution* functionalities that allow it to transfer media streams to and from other overlay servers or standard web or streaming servers. A received media stream can be cached by the overlay node, and/or relayed to other nodes. For instance, the overlay servers can perform overlay-level multicast, or stream splitting, where the overlay server relays a single stream to multiple recipients [1].

The overlay server has *resource monitoring and management* functionalities that allow it to monitor and log its observations over time, and share these logs with other overlay servers and managers through the control/management interface. Each overlay server tracks the requests that it receives, its server load, and observed network conditions. The logs of many overlay servers can be gathered and analyzed to improve the performance of the MSM-CDN.

Overlay servers have *peering* functionalities that allow them to query or track the contents of other overlay servers for requests that are not in its cache. It can have parent/child or sibling relationships with other overlay servers. The overlay server peering relationships can be changed through the management interface.

An overlay server may also have advanced *media service* functionalities that allow it to manage media sessions and process media streams. Media processing operations include media transcoding, to adapt media streams for diverse client capabilities and changing network conditions [2, 3, 4]; and media segmentation, to enable efficient delivery of long media streams by caching media segments aligned with user viewing statistics. Overlay servers can also perform midstream handoff of streaming sessions as well as midstream handoff of live transcoding sessions [5].

3. MSM-CDN RESEARCH

The MSM-CDN was designed to enable large-scale end-to-end media delivery. It also represents a new platform for research. This section describes a number of research directions for the MSM-CDN.

Media distribution across MSM-CDN infrastructure: Placing media content on an overlay server close to a requesting client can lead to the media being streamed over a shorter network path, thus reducing the startup latency of a streaming session, the probability of packet loss, and the total network usage. This motivates the need for media distribution algorithms that optimize system performance based on the predicted demand. These optimizations can be performed by aggregating measured statistics and developing predictive prefetching algorithms based on statistical analysis. Specifically, the prediction can be based on the content request patterns monitored, logged, and reported by the overlay servers through the control and management interface. Thus, appropriate cache allocation is a key problem, as is the pre-distribution of media content across a caching infrastructure, and the selected locations of the caches [6].

Media caching: Closely related is the problem of media caching on the overlay servers. The goal of improving cache hit rate makes it desirable to store large numbers of media streams on the overlay servers. However, since media streams can require large amounts of storage, storing entire media streams in a cache is clearly inefficient. Thus, the media caching problem involves determining what media streams [7] or media stream segments should be cached [8]. These decisions can be based on a number of factors such as media popularity, size, cacheability, and other factors such as premium content versus free content. Media distribution and caching are critical components of the MSM-CDN because they can lead to considerable improvements in resource utilization and system reliability.

Client request redirection/server selection: When a client requests some content, it must be directed to a server for serving the content. This can be achieved through a number of mechanisms, such as manipulating DNS entries or dynamic SMIL modification [9]. This operation also requires a system monitoring and management component for finding the "best" overlay server based on a number of metrics, including content availability, server load, and network load. An architecture was designed for monitoring the server and network load of overlay servers and assigning requests to the least-loaded, available edge server [10].

Media streaming: Streaming involves the delivery of long, continuous media streams, and desires highly predictable bandwidths, low delay, and preferably no losses. In particular, mid-stream disruption of a streaming session can be highly distracting. There are a variety of research opportunities in adaptive streaming in overlay nodes for improving system performance [11].

Stream scheduling: A number of opportunities lie in the general area of stream scheduling. These scheduling problems have a number of flavors, where the basic idea is scheduling the packet transmissions for a media stream over a channel that may exhibit time-varying available bandwidth, loss rate, and delay. While early streaming systems simply transmitted media packets in consecutive order without regard for the importance of individual packets, significant benefits result by simply accounting for the priority of each packet, e.g. I, P, or B frame or scalable layer. Further benefits result from rate-distortion optimized packet scheduling, which decides which packet should be transmitted at each transmission opportunity, as a function of estimated channel conditions and client feedback [12, 13]. In addition, low-complexity stream scheduling algorithms can exploit periodic coding structures in the encoded video [14].

Mid-network stream adaptation for diverse clients and network conditions: A streaming system must be able to deliver media streams to a diverse range of clients over heterogeneous, time-varying networks. In many scenarios, the downstream network conditions and client capabilities are not known in advance. In these scenarios, it is useful to be able to dynamically adapt streaming media to match the available bandwidth and capabilities of the specific client device. A number of approaches can be used to solve this problem. Multiple file switching switches between media files coded at different data rates [15]. Scalable coding stores base and enhancement streams that can be sent in a prioritized fashion [16]. Transcoding adapts pre-compressed streams into formats better suited for downstream conditions. These methods provide different tradeoffs in terms of flexibility, compression efficiency, and complexity [2, 3, 4].

Wireless streaming: Wireless channels are a shared, highly dynamic medium, leading to unpredictable, time-varying available bandwidth, delay, and loss rates. A key opportunity lies in optimizing wireless streaming algorithms from the overlay servers. Since an overlay server can be co-located with the wireless basestation, it can more readily adapt the streaming to the wireless channel variations.

End-to-end security: A related problem involves the desire to provide end-to-end security for a streaming session, while also supporting mid-network transcoding. While these properties appear to be mutually exclusive, a careful co-design of the compression, encryption, and packetization can enable mid-network transcoding while preserving end-to-end security (secure transcoding) [17].

Streaming to many clients: Another key problem is supporting popular events via multicast or one-to-many communication. IP Multicast is currently not supported in the Internet, but application-layer multicast can be provided by the overlay network. In addition, diverse clients requiring different bit rate versions of the same content can be supported by using scalable coding and sending different layers on different multicast trees–each receiver joins the appropriate multicast tree(s) based on the desired content [1]. Similarly, multiple multicast trees can provide different amounts of FEC for error control, where each client selects the desired amount of FEC [18].

Robust streaming using distributed infrastructure and diversity: The distributed infrastructure of the MSM-CDN also provides an opportunity to explicitly achieve *path diversity* or *server diversity* between each client and multiple nearby overlay servers. For example, multiple servers can send different streams over different paths (partially shared and partially not) to each client, thereby providing various forms of diversity which can overcome congestion or outage along a single path, and improved fault tolerance. This may be achieved using multiple description (MD) coding as an MD-CDN [19] (using various MD codecs, e.g. [20, 21, 22]) or single description or scalable coding with FEC [23, 24, 25]. It can also be achieved by using multiple wireless basestations or 802.11 access points [26].

Handoff of streaming sessions: Streaming media delivery differs from webpage delivery in that streaming sessions are often long lived. The long-lived nature of streaming sessions combined with

user mobility raises the technical issue of midstream handoffs of streaming sessions between overlay servers. Furthermore, when the streaming session involves transcoding, mid-stream hand-off of the transcoding session may also be required [5].

Dynamic load balancing of long-lived streaming sessions: The midstream handoff capability is also useful for enabling improved dynamic load balancing and fault tolerance. As more streaming sessions are started, it may be useful to rebalance the streaming or transcoding sessions. For example, the overlay nodes can be used to explicitly route streams by using application-level forwarding where the overlay servers act as relays [20]. By combining this with the mid-stream handoff capability, streams can be dynamically re-routed to alleviate network congestion and improve load balancing.

Advanced media services: In addition to media delivery, the MSM-CDN overlay can be used to perform in-network media processing services on delivered media streams. Example media services include transcoding and video or audio processing operations such as background removal or noise reduction. Offering such services requires the development of mid-network media processing algorithms and a media services architecture (MSA) that is coupled with the media delivery architecture [27].

4. MSM-CDN TESTBED PROTOTYPE

A mobile streaming media testbed was designed and built to demonstrate the capabilities of the MSM-CDN. The testbed consists of a number of basic entities including content servers with pre-encoded content and a live MPEG-4 encoder/streamer; streaming media clients running on laptops and PDAs (HP iPAQs) connected over 802.11; and streaming overlay servers and management servers that together form an adaptive MSM-CDN. These entities together perform the functionalities of content distribution and caching, streaming, resource monitoring, resource management, and signalling. To facilitate interoperability with different clients, the streaming server supports both 3GPP PSS [28] and ISMA [29] specifications. These specifications establish interfaces between streaming clients and servers and provide guidelines for the use of RTSP, SDP, and RTP. Interoperability has been demonstrated with 3GPP-compliant servers and with QuickTime, RealPlayer, and third-party 3GPP-compliant players. Signalling between MSM-CDN overlay nodes is performed using SOAP/XML. The testbed has nodes in the US and Japan. The overlay nodes perform live stream splitting (application-layer multicast) and media service functions such as live in-network transcoding of MPEG-4 video streams [2]. Live streaming upload (RTSP Record) from the mobile client to the infrastructure is supported. Segment-based media caching is used to improve cache usage while being transparent to the clients. The system performs client redirection to overlay servers using a portal server architecture with SMIL-based URL rewriting [9]. The management server is a service location manager (SLM) [10] that assigns client-requested streaming, transcoding, or media service sessions [27] to the best overlay server based on network and system resource usage. Additional details on the MSM-CDN system design and prototype are given in [30].

5. SUMMARY

We presented an architecture for a mobile streaming media content delivery network (MSM-CDN) designed to deliver media to large numbers of mobile users. This system handles challenges that arise due to the stringent requirements imposed by streaming media, mobility, and wireless. The system architecture was designed to be flexible, modular, and interoperable with other systems and underlying networks. The MSM-CDN provides a new platform for advanced research to improve the end-to-end system performance. An MSM-CDN testbed prototype was built based on these architectural principles, and the prototype was tested with 3GPP-compliant origin servers and media clients.

6. REFERENCES

[1] S. McCanne, V. Jacobsen, and M. Vetterli, "Receiver-driven layered multicast," *ACM SIGCOMM*, Aug 1996.

[2] S. Wee, B. Shen, and J. Apostolopoulos, "Compressed-domain video processing," *HP Labs Tech Report (HPL-2002-282)*, October 2002.

[3] A. Vetro, C. Christopoulos, and H. Sun, "Video transcoding architectures and techniques: An overview," *IEEE Signal Processing Magazine*, March 2003.

[4] S. Wee, J. Apostolopoulos, and N. Feamster, "Field-to-frame transcoding with temporal and spatial downsampling," *IEEE ICIP*, October 1999.

[5] S. Roy, B. Shen, V. Sundaram, and R. Kumar, "Application level hand-off support for mobile media transcoding sessions," *ACM NOSSDAV*, May 2002.

[6] L. Qiu, V. Padmanabhan, and G. Voelker, "On the placement of web server replicas," *INFOCOM*, 2001.

[7] M. Miao and A. Ortega, "Scalable proxy caching of video under storage constraints," *IEEE Journal on Selected Areas in Communications*, Sept 2002.

[8] S. Sen, J. Rexford, and D.F. Towsley, "Proxy prefix caching for multimedia streams," in *IEEE INFOCOM*, 1999.

[9] T. Yoshimura, Y. Yonemoto, T. Ohya, M. Etoh, and S. Wee, "Mobile streaming media CDN enabled by dynamic SMIL," *Inter. World Wide Web Conf.*, May 2002.

[10] S. Roy, M. Covell, J. Ankcorn, S. Wee, M. Etoh, and T. Yoshimura, "A System Architecture for Mobile Streaming Media Services," in *IEEE Mobile Distributed Computing Workshop*, May 2003.

[11] M.-T. Sun and A. Reibman, Eds., *Compressed Video over Networks*, Marcel Dekker, 2001.

[12] P. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," *IEEE Trans. on Multimedia*, Submitted Feb 2001.

[13] B. Girod, J. Chakareski, M. Kalman, Y. Liang, E. Setton, and R. Zhang, "Advances in network-adaptive video streaming," *Tyrrhenian Inter. Workshop on Digital Communications*, Sept 2002.

[14] S. Wee, W. Tan, J. Apostolopoulos, and M. Etoh, "Optimized video streaming for networks with varying delay," in *IEEE ICME*, August 2002.

[15] G. Conklin, G. Greenbaum, K. Lillevold, A. Lippman, and Y. Reznik, "Video coding for streaming media delivery on the internet," *IEEE Trans. CSVT*, 2001.

[16] H. Radha, M. van der Schaar, and Y. Chen, "The MPEG-4 fine-grained scalable video coding method for multimedia streaming over IP," *IEEE Trans. on Multimedia*, March 2001.

[17] S.J. Wee and J.G. Apostolopoulos, "Secure scalable streaming enabling transcoding without decryption," *IEEE ICIP*, Oct. 2001.

[18] W. Tan and A. Zakhor, "Video multicast using layered FEC and scalable compression," *IEEE Trans. Circuits and Systems for Video Technology*, March 2001.

[19] J.G. Apostolopoulos, T. Wong, W. Tan, and S.J. Wee, "On multiple description streaming with content delivery networks," *IEEE INFOCOM*, June 2002.

[20] J.G. Apostolopoulos, "Reliable video communication over lossy packet networks using multiple state encoding and path diversity," *VCIP*, January 2001.

[21] A.R. Reibman, H. Jafarkhani, Y. Wang, M.T. Orchard, and R. Puri, "Multiple description video coding using motion-compensated temporal prediction," *IEEE Trans. Circuits and Systems for Video Technology*, March 2002.

[22] Y. Wang and S. Lin, "Error resilient video coding using multiple description motion compensation," *IEEE Trans. Circuits Systems for Video Tech*, June 2002.

[23] T. Nguyen and A. Zakhor, "Distributed video streaming over internet," *SPIE Multimedia Computing and Networking 2002*, January 2002.

[24] A. Majumdar, R. Puri, and K. Ramchandran, "Distributed multimedia transmission from multiple servers," *IEEE Inter. Conf. Image Processing*, Sept. 2002.

[25] J. Chakareski and B. Girod, "Rate-distortion optimized packet scheduling and routing for media streaming with path diversity," *IEEE DCC*, April 2003.

[26] A. Miu, J. Apostolopoulos, W. Tan, and M. Trott, "Low-latency wireless video over 802.11 networks using path diversity," *IEEE ICME*, July 2003.

[27] M. Harville, M. Covell, and S. Wee, "An architecture for componentized, network-based media services," in *IEEE ICME*, July 2003.

[28] 3rd Generation Partnership Project, *Transparent End-to-End Packet Switched Streaming Service (PSS): 3GPP TS 26.233 & 26.234*, 2002.

[29] Internet Streaming Media Alliance (ISMA), *www.isma.tv*.

[30] S. Wee, W. Tan, J. Apostolopoulos, S. Roy, M. Etoh, and T. Ohya, "System design of a mobile streaming media content delivery network," *Under Review*.