Multimedia Digital Library: Performance and Scalability Issues

LakshmiPratha, Manjunath Mattam, Vamshi Ambati, Raj Reddy

lakshmipratha@iiit.net , manju@cmu.edu , vamshi@cmu.edu , rr@cmu.edu
International Institute of Information and Technology,
Gachibowli, Hyderabad, AP 500019 INDIA
Wean Hall 5325, School of Computer Science, Carnegie Mellon University,
Pittsburgh PA 15213-3891

Abstract – **Digital Libraries have become the source of information sharing across the globe for education, research and knowledge. The full potential of the Digital Libraries is realized when any citizen can have access to all human knowledge from any location. In today's scenario the multimedia has become an effective information sharing resource, people of all ages can understand more clearly by seeing / hearing rather than reading. Based on the tremendous growth in the various technologies, the establishment of Multimedia Digital Libraries is not an impossible task. But there are some critical challenges like performance and scalability, that prevail in the Multimedia Digital Library. In this paper, we are critical challenges and are providing a solution framework through an architectural representation.**

IndexTerms **– Architectural Framework, Perforrnance, Scalability, Media Server.**

I. BACKGROUND

Today more emphasis is given to the information captured in the form of Images, Video and Audio. These huge collections of Images, Audio and Video form the Multimedia Information Store best known as **Multimedia Digital Library (MDL)**. Some of the Videos and audios that are stored in Multimedia Digital Library constitutes the lectures of world's top most universities, seminars or conferences on various research topics, and news clips that are pre-recorded, and so on. This digitized Multimedia content has received wide attention in recent years due to enhanced internet services, protocols, storage options and applications.

Due to rapid growth in digital media and network technology, development of multimedia digital libraries has been an interesting research focus. Many research works have set a path of success like **Informedia II** *which provides a high visualization quality for videos*, [1] **MDLTR and ASIS MDL** *are the other research works, that provides indexing, searching using keywords in Videos*. [2] All these above tools and techniques concentrate on the visualization, summarization and database technologies but the performance and scalability issues are not considered as the primary area of attention.

In this context performance and scalability represents a smooth playback of video / audio in the client machine, with just-in-time delivery of the digitized content to a number of users. This problem is referred to in the multimedia information and networking community as the need for guaranteed Quality of Service (QoS), where the details of such guarantees depend on volume of data and playback specifications. Also as the numbers of users increase, application that control the delivery of the multimedia content has to be scaled up to support more users.

In this paper, we focus on performance and scalability issues, highlighting in what ways they affect the access to a Multimedia Digital Library and provide a remedial approach through an architectural framework. In section two we highlight these issues and related technology which can address the issues. In section three we propose the architectural framework that best suits the Multimedia Digital Library and in section four we bringout the strengths and weaknesses of the proposed architectural framework.

II. QOS ISSUES IN MULTIMEDIA DIGITAL LIBRARY

QoS in a Multimedia Digital Library can be highlighted in terms of both functional and non-fucntional requirements. In this paper, we address the critical non-functional requirements / quality attributes of the Multimedia Digital Library, the Performance and the Scalability.

A. Performance Issue

A Performance quality requirement defines a metric that states the amount of work an application must perform in a given time, also called as through put [4]. In the real-time systems performance is considered very critical, for example if a programmed robot fails to perform a specified functionality within a stipulated time undesirable thing can happen. But when it comes to web-based applications, performance is not considered a critical factor, but little they realize that it may also lead to undesirable things.

Performance usually manifests itself in the following measures:

a. Throughput
b. Response time

The above two measures that have major affect on the multimedia digital library. When streaming the audio/video files, client system slows down due to various reasons like file size, disk space, congestion, bandwidth and so on. This affects the thoroughput and response time of the system.

Throughput in the Multimedia Digital Library

Throughput is a measure of the amount of work an application must perform in one unit time. Work is typically measured in transactions per second (tps) / messages passed per second (mps). In the case of Multimedia contents which are in the form of an audio / video, the throughput is in terms of frames per second. Because of the relatively large file size of the media content there is a tremendous delay during the client access.

To avoid such delays we adapt the concept of striping from RAID (Redundant Array of Inexpensive Disks) technology for storing the digitized content in the Multimedia Digital Library. The striper component should segment the single media file and store them into different disks. The definition of the **Segment** in an MDL might vary depending upon the choice of the user and the information the fragment holds. The process of striping in our MDL enables spreading demand over multiple disks thereby avoids overloading of any particular disk.

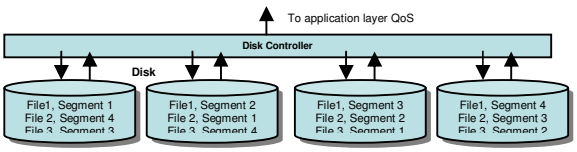

Figure 1: Submitted media file is stripped and segments are stored across multiple disks.

Striping improves throughput over storage of complete files on particular disks, by preventing one or two media file retrieval from dominating use of a particular disk and limiting the number of simultaneous users of files on that disk. Also the retrieval speed is aided by accessing the disks in parallel. [3]

Response Time in a Multimedia Digital Library

A web-based multimedia digital library has another important challenge of performance measure which is the response time. Unlike the transmission of textual data the application has to transmit the multimedia data. The response time is most often associated with the time an application takes to respond to some input.

Considering the scenario, when a student requests the MDL application to view a segment of an one-hour lecture, the following process takes place:

a. The entire lecture is either getting ready for streaming or is downloaded via a cache mechanism

b. Then the user using his player has to select the location /time stamp of the lecture he/she wants to view.

In order to avoid the frustration of the user and to improve the response time, we introduce the concept of segmentation and a metadata for each segmented unit. This segmentation is done considering the transcript of the video/audio file and the keywords. The segmented files are striped across the disks, the corresponding metadata for each segment is stored in the Metadata server. Whenever the user requests for a particular segment, it can be retrieved and streamed rather than the complete video/audio file. This saves time and enhances the transaction between the request made and the multimedia server.

We also analysed the usage of different protocols and recommend protocols like Real-time streaming protocol (RTSP), Multi Media Streaming (MMS) as an alternative to HTTP Streaming. These protocols best suits the multimedia applications since it enhances the response time in a network congestion scenario. RTSP (as shown in Figure 2) is one of a potpourri Real time protocol (RTP) which is specifically oriented to delivery of media streams in applications such as real-time exchange and downloading of music and video, this way user can play very soon after transfer begins rather than waiting until the entire file is transferred. [3]

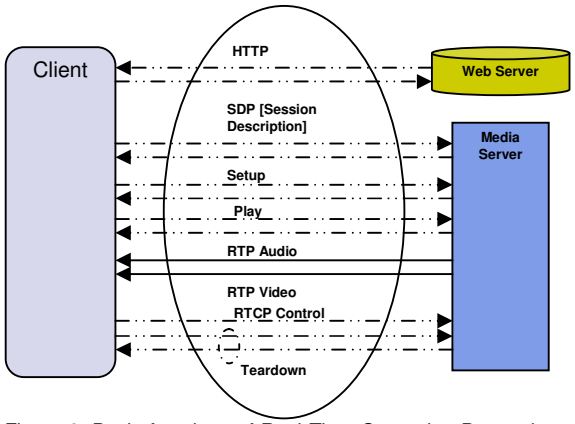

Figure 2: Basic functions of Real Time Streaming Protocol in working.

These specific features like striping, segmentation and support for multiple protocols should be implemented in the Media Server.

B. Scalability Issue

Scalability can be represented as how well a solution to some problem will work when the size of the problem increases. In the multimedia digital library we are highlighting the issue of scalability in terms of

a. Request Load
b. Data Size
c. Simultaneous Connections
d. Bandwidth

The concept of Request Load and the Data Size are addressed by the striping and segmentation mechanisms.

Now the major concern is the simultaneous connections and bandwidth that need to be established in a web-based multimedia digital library. In general the content delivery can take place either by providing the downloadable file or by streaming. Streaming via multicast or broadcast mode will reduce the flexibility for client and also faces barriers at router levels.

In order to provide that flexibility for the clients we propose an on-demand unicasting which is the generic streaming mode. When a server provides a media stream that a client has requested, the buffer in the client system is a key component for smooth packet flow.

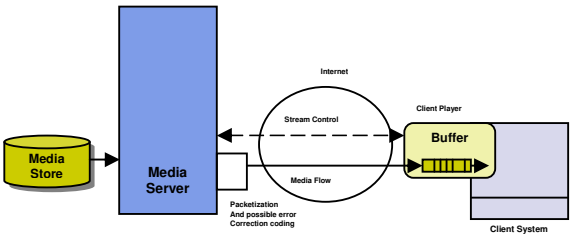

Figure 3: Unicast Streaming from a source server

In this scenario, a media player interface of the Multimedia Digital Library application provides multiple buffers which enhances the smooth packet flow and web-based interface for the client flexibility.

Each user whenever he/ she requests for accessing a video/audio file, the media server via an application sends the segmented video/audio files. The player interface provided by the MDL, uses multiple buffers that are large enough to hold one movie/audio segment. The buffers are filled and emptied in a complementary manner as shown in the following figure:

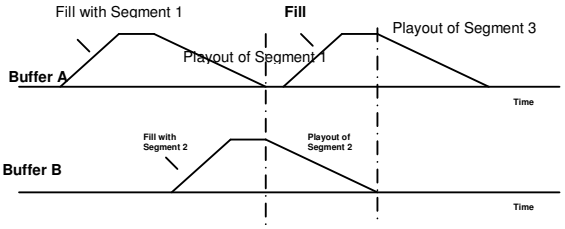

Figure 4: Just-in-time Continuous Stream Download to an end user

Number of client connections will be limited based on bandwidth and average rate at which clients connect to the media server. The average rate at which clients stream content from server is hard to determine because average can be skewed by high number of slow client to server connections or moderate number of fast client to server connections.

Maximum effective client connections = Maximum bandwidth/Average stream rate.

Maximum bandwidth must not reach 85% capacity of network card. Thus the simultaneous uninterrupted audio/video, controlled playback, fairness among the users, and efficient use of network resources is achieved via the media player provided by the Multimedia application server. On-demand Unicasting, media player and the bandwidth increases the scalability of the Media Server.

III. Architectural Framework For Media Digital Library

The proposed architectural framework provides a solution that can be applied with any technology such as J2EE and/or .NET, Web Services etc. We concentrate on the framework and its workflow based on the above discussed issues.

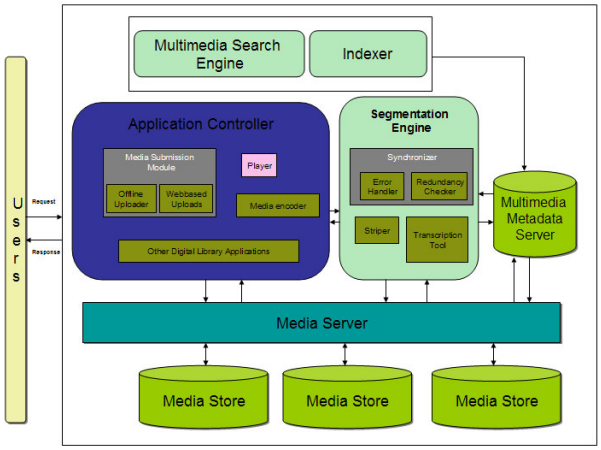

Figure 5: Architectural Framework addressing the Performance and Scalability Issue of MDL

The above framework is based on the web-based client server architecture. The entire set of components illustrated within a boundary represents the server applications, their components, sub components and the connectors between each of the components.

A. *Architectural Component Specifications*

The major components of the above proposed architectural framework are the Application controller and the segmentation engine

Application Controller

The application controller is the **communication point** between the external users and the System. This components consists of the following sub components:

a. Media Submission Manager
b. Digital Library Manager
c. Media Player Interface and Media Encoder

Media submission Manager, performs the activity of uploading the media content onto the Media Server. It interacts with the Segmentation Engine Component and provide the media content as an input to the Segmentation Engine component. It handles multiple requests for the upload actitivity. It handles offline uploads and also online uploads.

Digital Library Manager, performs the normal digital library operations that are unique for the general digital library, which consists of the details of the

digitization centres and other image processing and other relevant units related to the media content creation. Media Player Interface, acts as an interface between the client side player and the server. Media Encoder, encodes different formats in which the videos are uploaded and maps them to the formats standardized by the MDL.

Segmentation Engine

The Segmentation Engine is the core component that addresses the **performance issue** of the Multimedia Digital Library. It consists of the Synchronizer, Metadata Generator and the Striper.

Synchronizer, performs the synchronization of the audio, video files and the synchronization of the segmented files for a smooth playback, the output of the synchronizer is sent to the media player for the playback.

Metadata Generator, performs the unique metadata generation for each segment of a lengthy video or for every video file and its segments. It stores the metadata information in the metadata server. It enhances the searching and indexing mechanism and also the access to the media content in the media server.

Metadata is a crucial part of MDL for it helps in idenfying the information required by the user. Metadata entry for the segments in a video is difficult and depending on the granularity of the segment, or size of the video. However there are other semi-automatic ways of capturing metadata which needs to explored. For instance, during a lecture or a presentation, kestrokes, mouseclicks, voice, presentation content could act as cues in obtaining such metadata.

Based on the metadata the segments are created and these segments are given as an input to the striper. Striper performs the striping operation and segments each video / audio files and stores them in the media server.

The other components of the Multimedia Digital Library have already been taken care of by the existing digital library applications, hence emphasis on these other components like Searching, Indexing have not been given.

B. Sample Usage Scenario Of The Framework

The workflow of the solution can be explained in two modes:

The first mode is where the user requests for viewing and the second mode is where the user submits the video/audio file to be stored.

Let us consider the first mode, where the user requests for a video clip to be viewed. This request is sent to the application controller, which invokes its sub component the player interface. The player interface based on clients request/ protocol/ application, talks with the segmentation engine to provide the segmented video clip. The segmentation engine communicates with the Multimedia indexer and the search engine that uses the metadata server and provides the resource location to the synchronizer component. The synchronizer component synchronizes the striped segments and sends these segments to the player interface. The digital content is now visible in the client machine via a web browser or a client-side multimedia player. In this mode there might be n- number of simultaneous users who will be accessing the application controller. And the application controller by means of the Media Server module performs the unicasting operation.

The second mode, wherein the user request is to upload the video/audio files, the application controller invokes the media submission manager. Media Submission manager with help of Media Encoder converts all the video/audio formats to the supported media format. Media Submission Manager sends the submitted files to segmentation engine, to stripe the audio/video contents into segments and store them across the various disks. Media Encoder and segmentation engine work together to create unicast publishing points.

IV. Strengths And Weaknesses Of The Architectural Framework

The wide acceptance and usage of an MDL is dependent upon the value added services that it provides. Application developers / service providers can easily integrate new services with the existing framework and there is a great seperation of concerns brought in by the application controller. This leads to easy development of services as the developers now need not concern about the QoS or other finer details of the MDL like segmentation, metadata etc.

Advantages to this architectural solution is that the components specified in the framework can be implemented using the J2EE (like JMF, EJB, etc.) technology that provides key run-time features such as threading (to provide concurrent, high performance execution environment), Caching (to hold a huge size of caches since the size of cache can be specified in the deployment descriptor) and Connection Pooling (to manage a pool of database connections that enables efficient access).

The identified disadvantage is the transcription of the Video files and its segments that goes as an input to the metadata generator. The transcription has to be done at the users side and the accuracy of the transcriptions and segments identification cannot be determined in the current scenario.

V. Conclusion

We have addressed the performance and scalability issues that are unique to Multimedia Digital Library. We analysed the technology that best fits the Multimedia Digital Library and proposed an architectural solution framework. This framework can be implemented for a web-based Multimedia Digital Library using opensource technolgies and applications.

ACKNOWLEDGEMENTS

Our sincere thanks to Prof. Raj Reddy, Carnegie Mellon University, Pittsburgh, USA and the contribution made by the Digital Library teams at International Institute of Information Technology, Hyderabad, India and IISc Bangalore. We also acknowledge the efforts of the reviewers and their feedbacks on this paper.

REFERENCES

[1] Gerhard Weikam, *Quality of Service guarantees for Multimedia Libraries and Beyond.* http://www-dbs.cs.uni-sb.de/

[2] *Distributed Multimedia Digital Libraries for Medical Education*, http://www.cs.purdue.edu/vdbms/papers/medlib.pdf

[3] Stephen Weinstein: *The Multimedia Internet, Communications and Technology,* Consultants Summit, NJ .

[4] Ian Gortan, *Essential Software Architecture*, National ICT, Australia, Springer Publications.