

I-SPY: REMOTE OBJECT CAMERA CONTROLLER

MANOJ KUMAR . D

Master of Computer Applications
The Oxford College of Engineering
Bangalore, India.
dmanojkumar_1990@yahoo.in

Abstract—The need for a low-cost, scalable, cross-platform, distributed system for performing visual surveillance prompted the development of the Remote Object Camera Controller (ROCC). With a web cam and personal computer, a user can use ROCC to remotely monitor live video and archive video for later retrieval. This describes the use of the Java™ Programming Language, VLCJ, Real Time Streaming Protocol (RTSP) for constructing the ROCC system. Working with streaming video presents several challenges, including interfacing with a variety of web cams on different operating systems, efficiently streaming captured video, archiving video streams, and timely access to such archives.

Keywords- VLCJ, RTSP, visual Surveillance, Streaming.

I. INTRODUCTION

There is a growing need today to visually monitor an area from a remote location using a video camera or web cam. Distributed multimedia applications of this type require an efficient method of transporting real-time video from monitors to listening clients.

Streaming multimedia frameworks have become increasingly popular since the Internet is used as a media transportation layer. Working with streaming video presents several challenges, including interfacing with a variety of web cams on different operating systems, efficiently streaming captured video, archiving video streams, and timely access to such archives.

This project describes the use of the Java Programming Language, VLCJ and use of RTSP (Real Time Streaming Protocol) for constructing a low-cost, scalable, distributed system to perform surveillance.

Building a distributed system requires a great deal of network communication between distributed processes. The Real Time Streaming Protocol (RTSP) is a network control protocol designed for use in entertainment and communications systems to control streaming media servers.

II. USE OF TECHNOLOGY

A. VLCJ

The vlcj is an Open Source project that provides Java bindings for the excellent vlc media player from VideoLAN.

The bindings can be used to build media player client and server software using Java - everything from simply playing local media files to a full-blown video-on-demand streaming server is possible.

Support for playing media, especially video media, is generally poor on the Java platform. The Java Media Framework can be used to play some media, but it seems to be a neglected. For these reasons, the usual solution is to use a native media framework and bind to it using native libraries by using the Java Native Interface (JNI), or the Java Native API (JNA) to simplify JNI.

B. Real Time Streaming Protocol (RTSP)

The Real Time Streaming Protocol, or RTSP, is an application-level protocol for control over the delivery of data with real-time properties.

. It is also a client-server multimedia presentation protocol to enable controlled delivery of streamed multimedia data over IP network.

RTSP provides an extensible framework to enable controlled, on-demand delivery of real-time data, such as audio and video. Sources of data can include both live data feeds and stored clips.

This protocol is intended to control multiple data delivery sessions, provide a means for choosing delivery channels such as UDP, multicast UDP and TCP, and provide a means for choosing delivery mechanisms based upon RTP.

Figure 1 - RTSP Architecture

RTSP takes advantage of streaming which breaks data into packets sized according to the bandwidth available between client and server.

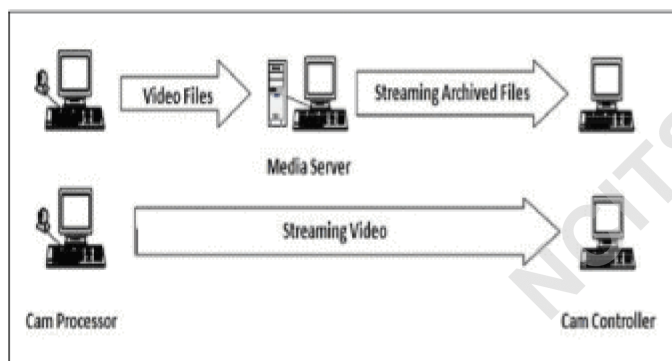
When the client has received enough packets, the user's software can be playing one packet, decompressing another, and downloading the third. This enables the user to listen or view the real-time file almost immediately, and without downloading the entire media file. This applies to live data feeds as well as stored clips.

III. IMPLEMENTATION OF TECHNOLOGIES

Remote Object Camera Controller consist of three Components.

- **Camera Processor:** It is used to get the Input (Live Video) from Web Cam then processed for Motion Detection and Stored.
- **Media Server:** It is responsible for archiving video files obtained from Cam Processor and used for streaming archived video files.
- **Camera Controller:** It allows the user to control any Cam Processor's Web Cam Remotely. The user may select any available archived video for viewing, can start and stop recording of the Live Video.

Figure 2 - Implementation Overview



A. VLCJ Implementation

VLCJ is implemented in both Camera Processor and Media Server components In order to get the Web Cam Input (Live Video) and processing it.

VLCJ Implementation Steps

- Load Native Library files "libvlc" using Java Bindings JNA (Java Native API) and VLC Media Player binding classes In order to run VLCJ.
- Use MediaPlayerFactory, EmbeddedMediaPlayer objects to register the Web Cam.
- Use JFrame, Canvas and CanvasVideoSurface to render the video on to the Screen.
- Use PlayMedia options to capture Live Video from Web Cam and save it to an "mpeg" file using sout transcode.

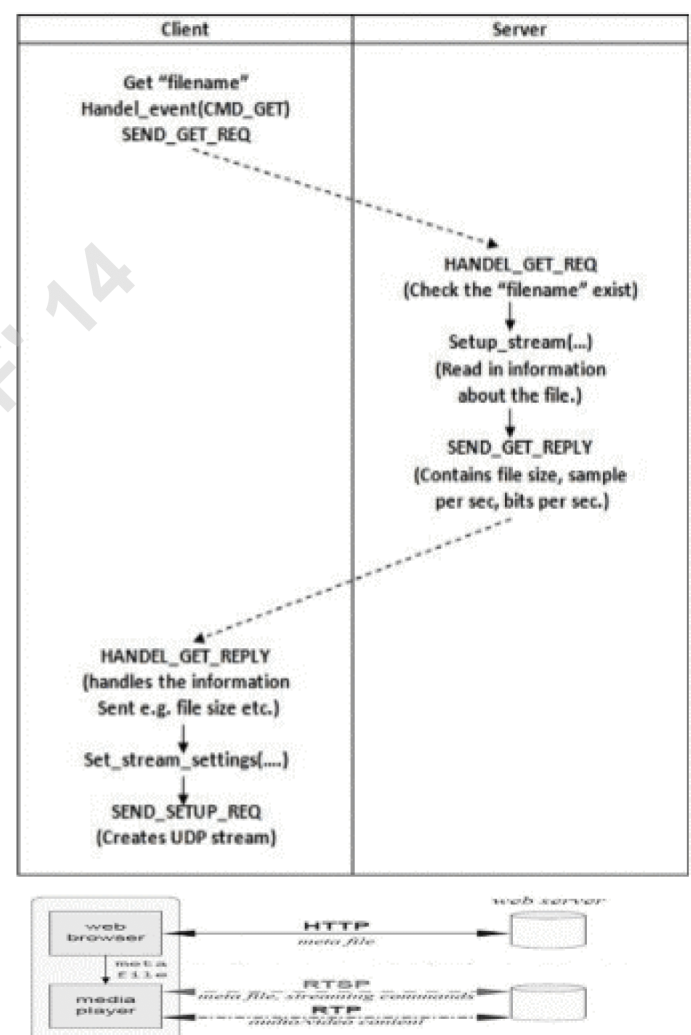
B. RTSP Implementation

RTSP is implemented in Camera Controller component In order to get the Live Video from Remote Web Cam and play archive video files from Media Server.

- Create RTSP Client and RTSP Server.
- An RTSP server needs to maintain state by default in almost all case. Both an RTSP server and client can issue requests.
- The Request-URI always contains the absolute URI.

Use RTSP methods "SETUP", "PLAY", "TEARDOWN"

Figure 3 - RTSP Implementation Steps



etc to communicate between Server and client.

IV. WORKING OF THE PROJECT

A complete Remote Object Camera Controller system involves all of the basic elements of creating, delivering, and ultimately playing the video content.

The main components of Remote Object Camera Controller used to accomplish Encoding Station, Video Server, Network Infrastructure, and Playback Client.

Steps Followed:

A. Capture

The first step in the process of creating streaming video is to "capture" the video from an analog source such as a camcorder.

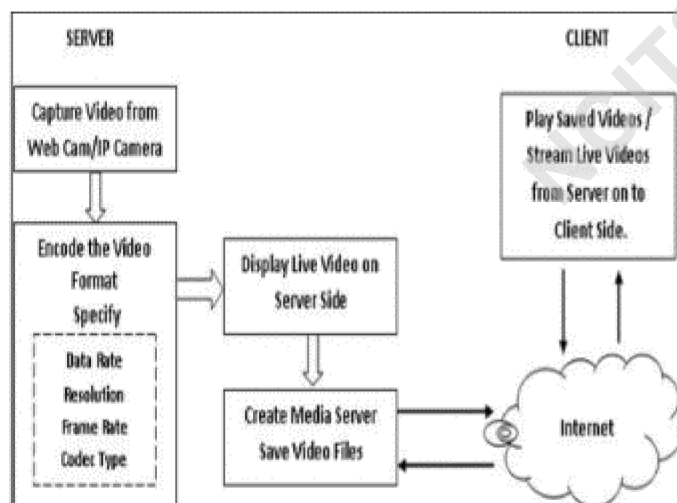
B. Encode

After the video is captured and is integrated with other media it may be encoded to the appropriate streaming file format. This generally involves specifying the desired output resolution, frame rate, and data rate for the streaming video file.

C. Server

The video server manages the delivery of video to clients using the appropriate network transport protocols over the network connection. The video server consists of a hardware platform that has been optimally configured for the delivery of real-time video plus video server software that runs under an operating system.

D. Play



Finally, at the client station the video player receives and buffers the video stream and plays it in the appropriate size window using a VCR-like user interface. The player generally supports such functions as play, pause, stop, rewind, seek, and fast forward.

Figure 4 - Remote Object Camera Controller Operation

V. FEATURES OF PROJECT

The key features of Remote Object Camera Controller are as follows

- This system can be used for performing "Visual Surveillance" of a particular perimeter, Office or Home.
- It provides a low-cost, scalable, cross-platform distributed system.
- If a Motion is detected inside the clients perimeter, It automatically sends an information to Client through message.
- It provides options for on-demand access of multimedia items such as stored real-time video files, and also provides live real-time video feeds.
- The Streaming of videos from Server to Client is done by Real Time Streaming Protocol.

VI. CONCLUSION

This system will definitely be a working prototype of a Security surveillance system at a low-cost. Just by installing Web Cam/ IP Camera to a Personal Computer and running the Remote Object Camera Controller application and providing an Internet Connection, User can access the system remotely without any complexity.

ACKNOWLEDGMENT

I am very thankful for my friends and lecturers in providing Information related to this Project.

REFERENCES

- [1] Java Network Programming 3rd Edition by Elliott Rusty Harold.
- [2] Internet Engineering Task Force (Internet-Draft) by H. Schulzrinne, A. Rao, R. Lanphier.
- [3] Information on Basic Streaming Technology and RTSP Protocol by California Software Labs (CSWL).
- [4] VLCJ Information provided in www.github.com and www.capricasoftware.com.