Volume 4, Issue 4, April 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Streaming Media

Renukadevi[1], G.munjunath[2] Jagadevi.Gudda[3]

Assistant Professor in CSE Department SLN Engineering College[1] Raichur, India

Senior Software Engineer Ericsson Television Southampton UK[2], India

Assistant Professor in Electronics Department, S.B.College of Science Gulbarga[3], India

Abstract: In Broadcast live TV, the server streams the content (Audio/Video) on a particular IP address and the clients join to that IP address. This mechanism is called as multicast and the session is called multicast session. Streaming Media Server shall support one-to-many and many-to-many real-time communication over an IP infrastructure in a network. It scales to a larger receiver population by requiring neither prior knowledge of a receiver's identity nor prior knowledge of the number of receivers. Multicast uses network infrastructure efficiently by requiring the source to send a packet only once, even if it needs to be delivered to a large number of receivers. The nodes in the network (typically network switches and routers) take care of replicating the packet to reach multiple receivers such that messages are sent over each link of the network only once. The most common low-level protocol to use multicast addressing is User Datagram Protocol (UDP).

Key words: Streaming media, streaming media server,

I. INTRODUCTION

Using the technologies which were earlier available during the early era of 90s it was very difficult and was almost impossible to send/share a recorded videos across the world wide web. In the later part of 90s people tried to share videos file using ***"Traditional download"*** mechanism. In 1999, the phenomenal popularity of the 2 1/2-minute movie trailer for "Star Wars: The Phantom Menace"—a whopping big 25 MB file at 480×216 resolution (good quality)—was proof that media-hungry Internet audiences are willing to be forgiving of long waits for large files.

Web-based technologies emerged as a powerful tool of communication and collaboration for people living in the different part of the world in the era after 2000. At this point the technology that has been largely used by people and organizations to share video was **"Progressive download"** mechanism.

So, the middle of 2005 was a time where people don't only wanted to share the videos but they also wanted to have the rich media experience while watching a program over internet and this demand obviously was not ridiculous as by this time sufficient infrastructure and technology was there in place to broadcast such high-quality videos over internet for a large set of users.

To address all kind of issues the new phenomena that emerged was "Streaming Approach". Streaming approach can be clearly defined by following three primary characteristics combined:

- Streaming media technology enables real-time or on-demand access to audio, video, and multimedia content via the Internet or an intranet:
- Streaming media is transmitted by a media server application, and is processed and played back by a client player application, as it is received:
- A streamed file is received, processed, and played simultaneously and immediately, leaving behind no residual copy of the content on the receiving device:

Present System

1. No control over the video once it has been delivered to consumer.
2. No content protection.
3. No control over piracy.
4. No quality control.
5. No insight about end-user experience.
6. No collaboration options while watching a video content.
7. No method to deliver real-time broadcasting of events.

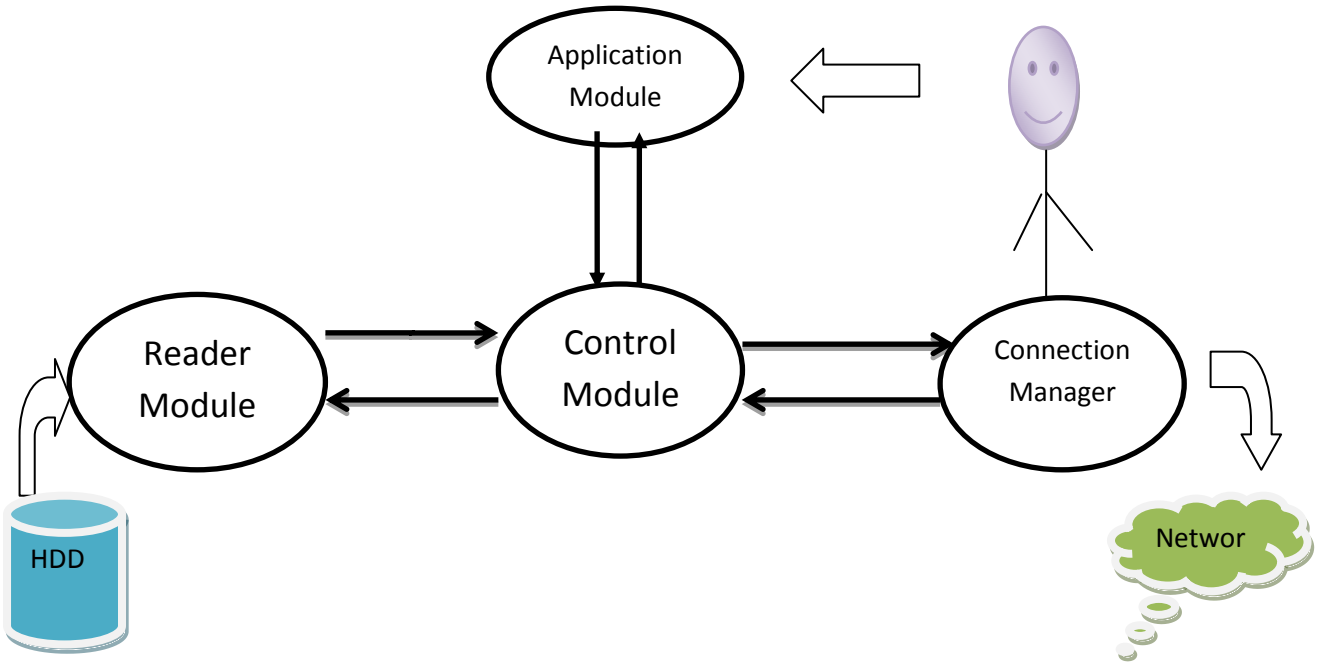
Proposed System

The purpose of this work is to describe about high level design of RTP based streaming media server. The streaming media server shall stream live/stored content over the network. The media content shall be encapsulated in the form of MPEG2 transport stream. The content shall be streamed by making use of RTP protocol. Streaming server shall be Windows based software, which is used to serve the needs of the client on the network. The server shall support following functionality:

- Multicast session in which the server streams stored content on a particular IP address and clients join to that IP address.
- Console based user interface that allows the user to interact with the server.
- Support multiple clients.

II. SYSTEM DESIGN

This section describes various components involved in the system and their interactions as shown in below figure:



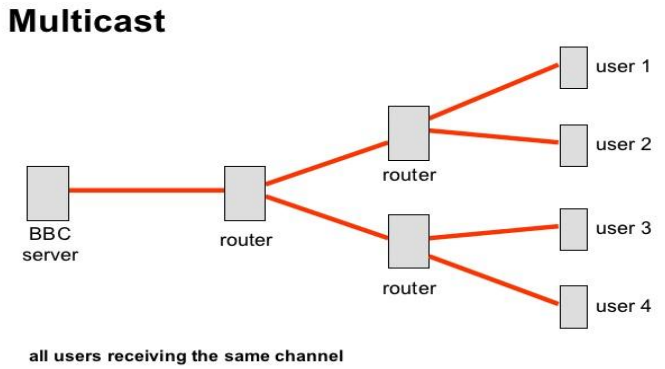
Above figure shows high level diagram for streaming media server

1. Connection Module:

As the name suggests, connection manager is responsible for managing connections with the client. Apart from this module also provides interface to send RTP packets over the network. This module is used for establishment of socket connection between client and server. It informs the controller module that connection is established. Here we use multicasting method and the UDP protocol to transfer the data using sockets.

MULTICAST:

Multicast is a method of one-to-many transmission which is often deployed in IP applications of streaming media. Multiple viewers can simultaneously tap into a single transmission from one source.



Television programming is a perfect analogy. When you want to watch HBO you tap into the HBO feed on the cable network, not to a dedicated signal of HBO. If your cable box is authorized, you get access to the existing stream of HBO on the cable network.

Reader Module:

This module is responsible for reading MPEG2 transport stream and encapsulate RTP header and transmit the packets over the network. This module provides interface to control manager to start/stop streaming RTP packets. The control manager notifies reader module about the filename and multicast address to which the RTP packets shall be

transmitted. Each RTP packet can accommodate at most 7 TS packets; each TS packet is 188 bytes long. This module forms the RTP header as per the standard, which includes adding the RTP timestamp in each RTP packet.

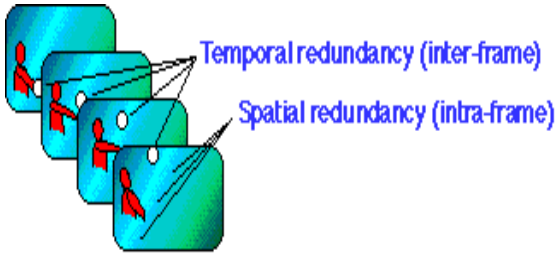
It is used to read the request from the client, process the request and serves those requests. Once the connection is established the reader module starts reading the MPEG2 transport stream stored from HDD.

Real Time Protocol (RTP): The real-time transport protocol (RTP) provides end-to-end delivery services for data with real-time characteristics, such as interactive audio and video or simulation data, over multicast or unicast network services. Applications typically run RTP on top of UDP to make use of its multiplexing and checksum services; both protocols contribute parts of the transport protocol functionality. However, RTP may be used with other suitable underlying network or transport protocols. RTP supports data transfer to multiple destinations using multicast distribution if provided by the underlying network.

The MPEG-2 video compression algorithm achieves very high rates of compression by exploiting the redundancy in video information. MPEG-2 removes both the temporal redundancy and spatial redundancy which are present in motion video.

Temporal redundancy arises when successive frames of video display images of the same scene. It is common for the content of the scene to remain fixed or to change only slightly between successive frames.

Spatial redundancy occurs because pats of the picture (called *pels*) are often replicated within a single frame of video.



Clearly, it is not always possible to compress every frame of a video clip to the same extent - some parts of a clip may have low spatial redundancy (e.g. complex picture content), while other parts may have low temporal redundancy (e.g. fast moving sequences).

Application Module:

This module shall provide console based user interface to

a. Start/Stop streaming the transport stream.
b. Specify the filename and the multicast ipaddress and port number. Alternatively this information can be read from the configuration file.

The application manager notifies control manager as soon as user inputs the command. These events shall be processed as per the state machine. The application shall also store some of the details such as ipaddress, port number and state (start/stop)of the server in its database. This shall be useful to check some of the invalid options from the user.

Controller Module: The control manager manages and coordinates various modules in the system by means of event based mechanism. The control manager shall have the state machine, which shall be used to notify appropriate module(s) whenever certain event occurs.

III. TIMING AND BUFFER CONTROL

PCR - Program Clock Reference

Synchronization of the receiver System Time Clock (STC) with the transmitter STC depends on transmitting PCRs through a constant-delay portion of the system. Thus, PCRs are inserted following the encoder buffer and extracted before the receiver buffer. PCRs are inserted with a maximum interval of 100ms.

STC - System Time Clock

The System Time Clock (STC) uses a 27 MHz oscillator with a worst case tolerance of +/- 810 Hz. [Note that this is a looser tolerance than that of the NTSC color subcarrier The STC first divides the the 27MHz by 300, giving a 90kHz clock that is counted by a 33 bit counter to give the base STC value. The remainder is taken as a 9-bit value that may be used as an STC extension.

SCR - System Clock Reference

System Clock Reference (SCR) is a time stamp in the program stream, as opposed to the Program Clock Reference (PCR), which appears in the transport stream. In most common cases, the SCR values and PCR values function identically; however, the maximum allowed interval between SCRs is 700ms, while the maximum between PCRs is 100ms.

PTS - Presentation Time Stamp

The Presentation Time Stamp (PTS) indicates the instant at which an access unit should be removed from the receiver buffer, instantaneously decoded, and presented for display. Any delay in a practical decoding or display process can be compensated for if it is fixed. Therefore, if a practical decoder has a variable delay, it must include buffering to restore a constant delay at the output. Note that the display itself may include a delay, for example, the time for the vertical scanning of a CRT to reach the middle of the picture.

DTS - Decode Time Stamp

The Decode Time Stamp (DTS) indicates the time at which an access unit should be instantaneously removed from the receiver buffer and decoded. It differs from the Presentation Time Stamp (PTS) only when picture reordering is used for B pictures. If DTS is used, PTS must also be provided in the bit stream.

Buffer Synchronization and Startup

Buffer synchronization and startup can be handled by means of time stamps plus an a priori constraint on the maximum buffer size, rather than explicit use of the VBV values.

Low Delay Mode

MPEG provides for a low delay mode by setting certain parameters. For low delay, the total buffer delay must be minimized. The low delay mode is signalled with a flag, low_delay, in the sequence header

Decoder STC Synchronization

Decoder System Time Clock (STC) synchronization is the process of setting both frequency of the STC 27 MHz oscillator and the value of the STC counter so that the counter matches the program clock references (PCRs) in the incoming data stream. This effectively matches the local 27 MHz clock to the encoder's 27 MHz clock.

IV. SYSTEM INTEGRATION

During the system integration process we take all the independently developed sub-systems and put them together to make up a complete system. Integration can be done in two ways

a) Big bang approach
b) Incremental approach

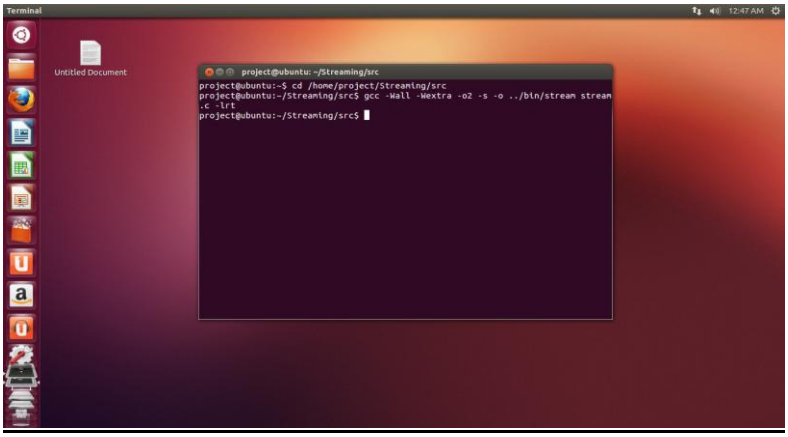
In this work we used incremental approach & developed sub-systems are

1) Connection Module
2) Reader Module
3) Application Module
4) Control Module

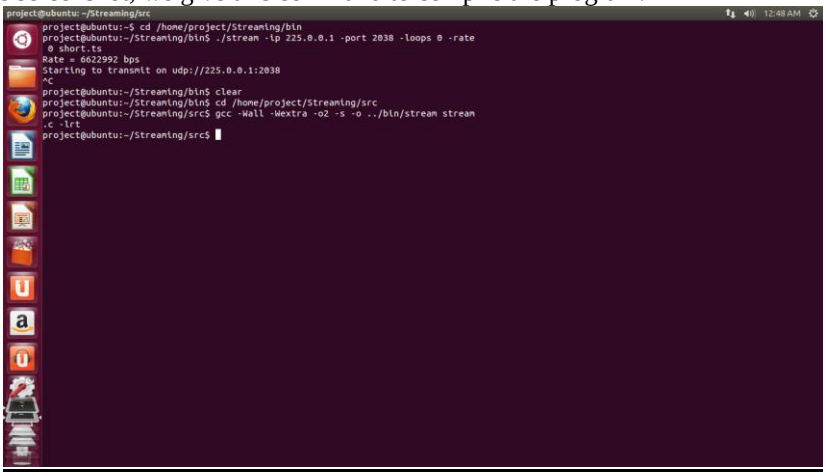
- First the above sub-systems are tested independently to verify their functionality.
- The Control module and connection module are integrated. After they are integrated the system function starts.

V. Results

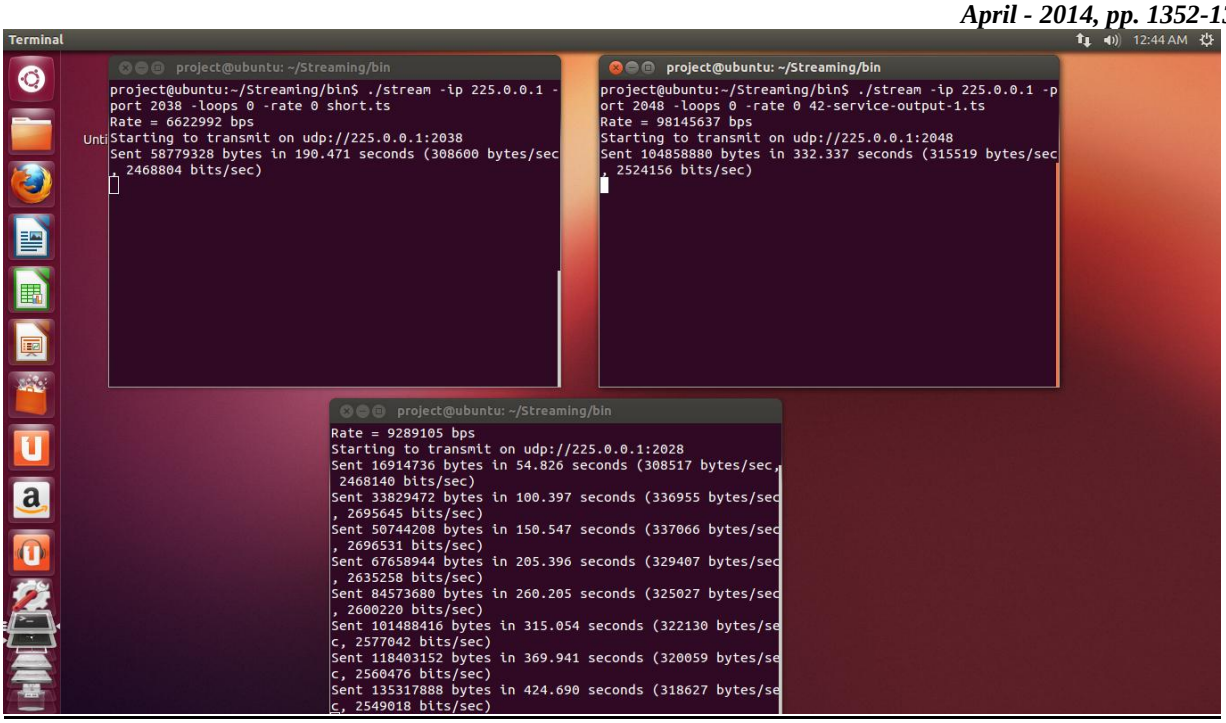
SERVER SIDE:



As it can be seen in the screenshot, we give this command to compile the program.

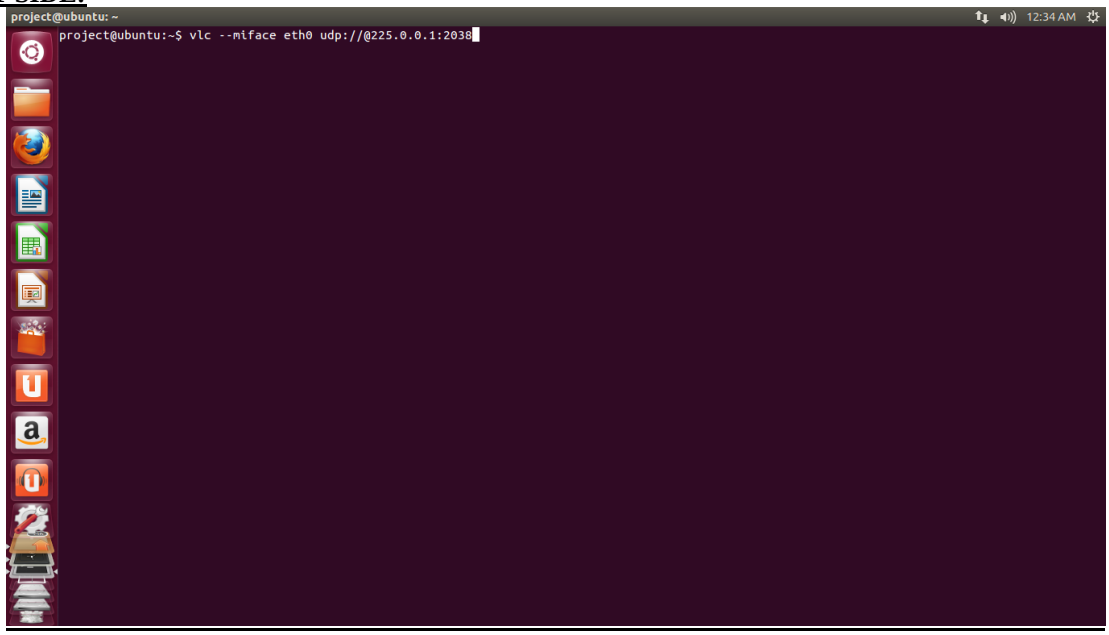


Above screenshot shows the command to execute the program which will transfer the TS file to the client



As shown in the above screenshot, we can transfer more than one file at time. Here three files are transferred on three different terminals.

CLIENT SIDE:



As it can be seen in the screenshot, this command opens the VLC on the client.

VI. CONCLUSION

Streaming media is a relatively novel technology that enables the Internet user to experience multimedia presentations on the fly. The concept of streaming media has experienced a dramatic growth and has become a viable commercial proposition in a relatively short time. This technology is likely to continue to change the way we experience not only the Internet but also media delivery in general. Indeed, many believe that "streaming media will be the biggest thing since television".

It is certain that streaming technology may fundamentally change the way we produce, distribute and consume media. Streaming media is likely to change the economics of production and distribution profoundly. It can embrace radio and television and yet, at the same time, could offer much more flexibility to the user. The main purpose of this article has been to introduce some basic components on the streaming-server (delivery) side, including the new streaming protocols that facilitate streaming delivery over the Internet. Many related topics have not been covered or even mentioned, including the means of improving Internet delivery by using caching, distributed servers and multicasting. The area of multimedia codecs and players has not been touched upon either. These, and many other technical and non-technical areas, will be covered in one or more follow-up articles.

References

[1] http://www.cod.edu/it/streamingmedia/intro.htm
[2] http://www.j2mesalsa.com/media/intro.php(diagram)
[3] http://beej.us/guide/bgnet/output/html/multipage/index.html
[4] http://www.erg.abdn.ac.uk/future-net/digital-video/mpeg2-trans.html
[5] http://www.ee.oulu.fi/~skidi/teaching/internet_multimedia_2002/rtp.pdf
[6] http://www.bretl.com/mpeghtml/timebuf.HTM
[7] http://tech.ebu.ch/docs/techreview/trev_292-kozamernik.pdf

