First Stage Project Report
for
"Audio Streaming Server"

Abhishek Sugandhi	Dheeraj Kumar	Sanjeet Khaitan
04305016	04305019	04305018
M.Tech.(CSE) Ist yr.	M.Tech.(CSE) Ist yr.	M.Tech.(CSE) Ist yr.
IIT,Bombay	IIT,Bombay	IIT,Bombay

September 10, 2004

Department of Computer Science and Engineering,
IIT Bombay

Contents

1 Overview **3**

2 Existing Softwares **3**
2.1 Icecast 3
2.2 Edna 3
2.3 Litestream 4
2.4 JStream 4
2.5 Streamsicle 4
2.6 Apache::MP3 Module 4
2.7 Fluid 4

3 Our Enhancements **5**
3.1 Understanding the Fluid's Code 5
3.2 Designing the Proper User Interface 5
3.3 Designing the Administrator Interface 5
3.4 Support For Different file Formats 5

4 System Requirements **6**
4.1 Software Requirements 6
4.2 Hardware Requirements 6

1 Overview

Until recently, audio on the Web was primarily a download-and-play technology. You had to first download an entire media file before it could play. It was like pouring milk into a glass and then drinking it. But because media files are usually very large and take a long time to download, the only content found on the Web was short 30-second clips often even shorter. Even these files could take 20 minutes or longer to download. In other words, it took a long time to pour the milk, and then it would barely quench your thirst.

Listening audio files that stream, is more like drinking straight from the carton; streaming media files begin playing almost immediately, while the data is being sent, without having to wait for the whole file to download. Other than a few seconds of delay before the file starts to play, you don't have to wait to start listening, no matter if the file lasts 30 seconds or 30 minutes.

As audio streaming over the Internet has become more popular, two primary methods for streaming content have emerged.[3] The first method is the Web server approach, in which a standard Web server is used to supply data to the client. The second method is the streaming media server approach, in which a specialized streaming server delivers the data to the client.While Web server streaming can be an effective interim solution, a streaming server is more efficient and flexible and provides a better user experience.

2 Existing Softwares

There are lots of software for streaming audio.Some of them are listed below.[5]

2.1 Icecast

Icecast, the project, is a collection of programs and libraries for streaming audio over the Internet. This includes:

- icecast, a program that streams audio data to listeners
- libshout, a library for communicating with Icecast servers
- IceS, a program that sends audio data to Icecast servers

2.2 Edna

Edna is a small, lightweight Python based streamer with built-in web interface.[2] Edna allows you to access your MP3 collection from any networked computer. This software streams your MP3s via HTTP to any MP3 player that supports playing off a remote connection (e.g. Winamp, FreeAmp, Sonique, XMMS).

Edna supports:

- Browsing your music collection from a remote computer.
- URL support to allow people to stream specific songs.

- Automatic playlist construction for entire directories of songs, including a ”shuffle” ability.
- The merging of multiple music collections while keeping the files on the remote computer.

2.3 Litestream

Litestream is similar in it’s operation to Icecast. The system is composed of the streamer and the source.

2.4 JStream

JStream is similar to Edna except written in Java.

2.5 Streamsicle

Streamsicle is a mixture of Fluid RC2 and a Tomcat server. It allows you to stream MP3’s and provides a playlist selector web interface.

2.6 Apache::MP3 Module

The Apache:MP3 module allows the Apache WebServer to stream MP3’s. It does not offer the same level of features as Fluid, Litestream & Icecast.

2.7 Fluid

Fluid is a MP3 streaming server written in JAVA.[4]

Not only is support added for many different ways of streaming, there is also a plug-in API for easy creation of streamers handling other kinds of media.

You can either listen for incoming clients, send out the information blindly using multicast technology or direct the stream to a specific destination.

No proper Web Interface.

3 Our Enhancements

We want to enhance 'Fluid' by adding some more features to it.. We are dividing our work in the following steps...

3.1 Understanding the Fluid's Code

Its the main time consuming task.

3.2 Designing the Proper User Interface

We will provide a good Web based Graphical User Interface to the user, There will be two type of users

General User A general user can listen to specific songs, search for different audio, create playlists or can listen to any live audio. The search method which will be provided will be quite adaptive where he/she can find audio files based on various attributes, like file name(wild cards will be implemented), last number of accesses, size of file, artist, composer, album, raga etc. They will also have authority to change values for some of the attributes of a song, but such permission will be very limited.

Authorized User These user are very special type of user. In addition to the above facilities, they can change the tags related to each songs (audio file) and can add additional information about the song. The songs classification will grow as more and more users add new information to them. This will help to find a collection of songs in a more classified way.

3.3 Designing the Administrator Interface

We will also provide administrator interface. We have also planned to make the new system as a portal where users can be created and only authorized users will be allowed to access the server. A user will also be able to customize his environment, like his own playlist, favorites etc. A user will also be allowed to add new audio to the system. The Administrator will be responsible for creating new users, viewing statistics etc.

3.4 Support For Different file Formats

Until now Fluid supports only mp3 file format. As mp3 format has been patented so world is moving from mp3 to ogg format. So our next task will be to take it into consideration , and we will try to implement support for ogg and rm format.

4 System Requirements

4.1 Software Requirements

- Linux OS
- JDK installed
- mysql
- MP3 decoder

4.2 Hardware Requirements

- Audio Enabled System
- Network support - Preferred connection are xDSL, cable modem, or T1/T3 lines

References

[1] Dave Phillips. *Streaming Media with Linux.* World Wide Web, http://www.linuxdevcenter.com/.

[2] Greg Stein. *Edna – an MP3 server.* World Wide Web, http://sourceforge.net/projects/edna/.

[3] Greg Stein. *Web Server vs. Streaming Server.* World Wide Web, http://www.microsoft.com/windows/windowsmedia/compare/webservvstreamserv.aspx.

[4] Lars Samuelsson. *Fluid Streaming Server.* World Wide Web, http://fluid.sourceforge.net/.

[5] Phil Kerr. *The Linux MP3 & Ogg Vorbis HOWTO.* World Wide Web, http://www.mp3-howto.com/.