

# Streaming Server for Audio

Dheeraj Kumar & Abhishek Sugandhi & Sanjeet Khaitan

Computer Science and Engineering  
IIT Bombay  
dkarora@iitb.ac.in

Oct 31, 2004



# Introduction to Streaming Server

- Two Approaches for listening audio over Internet
  - Web Server Approach
    - Down-load and Play
    - Long Response time
  - Streaming Server Approach
    - Client Receives audio stream and plays it simultaneously
    - Do not wait for the whole song to download



# Existing Open Source Software

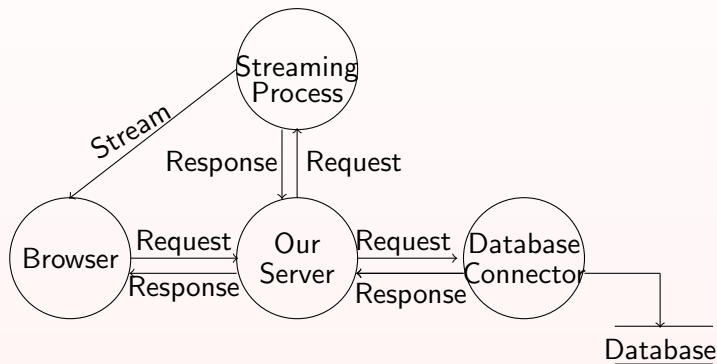
Existing Servers	
Server Name	Features
Fluid	No User Interface Available
Edna	Browsing of music collection possible
Lifestream	Composed of a streamer and a host
JStream	Similar to Edna
Streamsicle	Provides Playlist selector Web Interface



- To extend the 'fluid' Streaming Server
  - Currently does not have any user interface
  - We will create Web Interface for it
  - We will design database to store song and user information



## D.F.D. Level 0



# User Interface

- General User
- Registered User
- Authorized User
- Super User/Administrator



# General User

- Searching
- Listening the song



# Registered User

- General User Operation
- PlayList Operation
  - Adding a New PlayList
  - Deleting a Playlist
  - Editing a Playlist





# Authorized User

- Registered User Operation
- PlayList Operation
  - Adding a Song Attribute
  - Updating a Song Attribute
  - Upload Song



# Super User

- Authorized User Operation
- Add Authorized Users
- Edit User details



# Search Form

- Search for multiple conditions simultaneously
- Appearance
  - Checkbox before every field
  - Combobox after every field for holding domain values
  - Additional Textbox
- All Search are Substring Search



# Playlist Operation Form

- ComboBox for holding playlist names
- Remove playlist Button
- Create PlayList Button
  - Textbox for entering new playlist name



# General User Form

- It will be the starting form
- Divided in three frames
  - Left frame will be Search form
  - Right frame will be Login Form
  - Central frame will contain the list of songs selected



# Registered User Form

- This form will be displayed when you login as an registered user
- Divided into 3 frames
  - Left frame will be Search Form
  - Right frame will be PlayList Operation form
  - Central Frame will contain list of songs
    - Checkbox before every song
    - Select All Button
    - Add to PlayList Button
    - Remove from PlayList Button



# Authorized User Form

- This form will be displayed when you login as an authorized user
- Same as Registered User Form
- Central frame will have additional options like
  - Upload Song Option
  - Add attribute to songs
  - Modify attribute values of songs



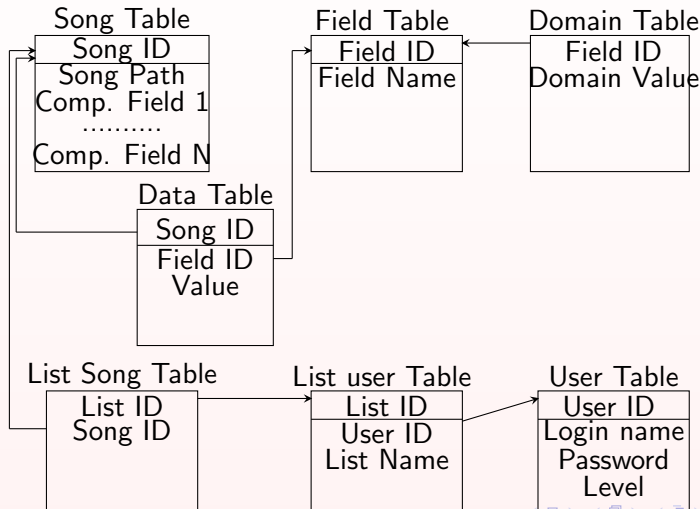
# Administrator Form

- This form will be displayed when you login as an administrator
- Same as Authorized User Form
- Central frame will have additional options of
  - Add Authorized User Option
  - Modify User information Option

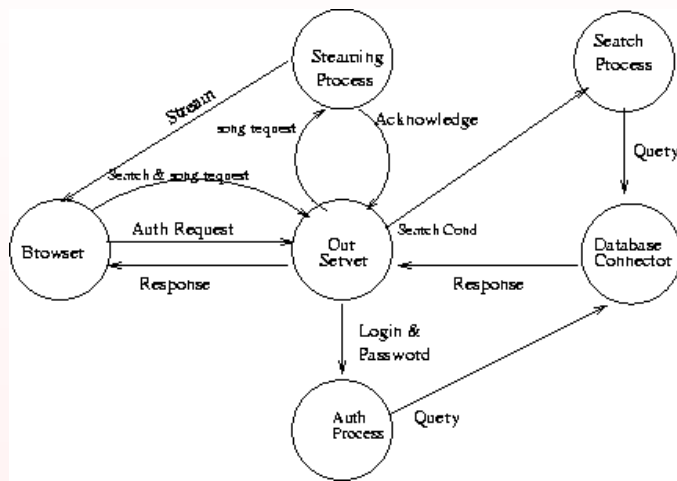




## Schema Diagram



## D.F.D Level 1



## Fluid : Mode of Operation

- Fluid Works in three mode
  - SingleCast Server : This Type of Server will send the stream to a specified host (address, port) using a TCP/IP connection
  - InterCast Server : This Type of Server can handle "any" number of client connections using TCP/IP connection
  - MultiCast Server : This Type of Server will send the stream to a specified group (address, port) using a UDP multicast connection.
- Each of these Server require a configuration file from which it gets the relevant information like client address , server port , playlist file path etc.



## Fluid : Type of Input Streams

- Fluid can fetch data from three type of Streams
- MP3InputStream : fetching data from mp3 files
- PlaylistStream : fetching data one by one from number of mp3 files
- RemoteInputStream : fetching data from remote m/c



# How to call fluid from our Server

## Procedure

Whenever a Client requests to play a list of songs ,we will create a playlist file and a configuration file, with the server port (which is not currently used by anyone ). We will also mention the client's ( address , port ) and playlist files path in the configuration file. Now what we require is to just create a new instance of fluid server with the configuration file passed as an argument . We will configure the server in single cast mode with playlist stream as input stream.



# Conclusions & Future Work

- Limitations
  - Fluid works only with MP3
  - Bandwidth required by the client depends on bit rate of the song
- Future Work
  - Support for other formats like ogg
  - Keeping track of user/server statistics
  - Changing the bit rate of song according to the Bandwidth



# References

- Web sites

- <http://fluid.sourceforge.net/> (Fluid Streaming Server)
- <http://www.linuxdevcenter.com/> (Streaming Media with Linux)
- <http://sourceforge.net/projects/edna/> (Edna – an MP3 server)
- <http://www.mp3-howto.com/> (The Linux MP3 & Ogg Vorbis HOWTO)



# Dummy Gnuplot

