

MULTIMEDIA TRANSCODER

(*Video Converter using FFmpeg*)

Summer Internship 2012

Submitted in fulfilment of internship project

By

Multimedia Transcoder Team

Under the Guidance of

Prof Deepak Phatak



Department of Computer Science and Engineering

Indian Institute of Technology, Bombay
Mumbai

ACKNOWLEDGMENT

We the summer interns team of Multimedia Transcoder are over helmed in all humbleness and gratefulness to acknowledge our depth to all those who have helped us to put our ideas and assigned work, well above the level of simplicity and into something concrete.

We all thank whole heartedly Prof. Deepak Phatak for selecting us a part of his valuable project, constantly motivating for doing better and showing complete confidence in our work.

We are indebted to our Project Manager Mr. Parag Tiwari for his constant help and support in light of this project. We are very thankful to our mentor Mr. Vivek Nadar for his valuable suggestions, guidance and encouragement. Also in the process, we learnt a lot other technical and non technical things from them

We would like to thank Mr. Bikas Chhatri for making our stay here as summer interns comfortable and for all his administrative help.

Finally we also like to thank all other colleagues working in different projects under Prof. Phatak for helping us at small problems as well as critical junctures.

TEAM MEMBERS

Name	College	e-mail
Nikhil Choudhary	Sardar Vallabhbhai National Institute of Technology-Surat	U09co247@coed.svnit.ac.in
Lakshmi Kumari Pogiri	Shri Vishnu Engineering College for Women-Bhimavara	lakshmi.IT30@gmail.com
Surbhi Mahajan	Dr.B.R Ambedkar National Institute of Techonology-Jalandhar	surbhimahajan543@gmail.com

MENTORS

Mr.Parag Tiwari-Project Manager at IITB

Mr. Vivek Nadar-Jr. Software Engineer at IITB

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included. We have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Multimedia Transcoder Team

Date: 02-07-12

ABSTRACT

Nowadays video is being produced and consumed in more component representation formats, more device types and over variety of networks than ever. Transcoding is a process of translating or converting one coded signal representation to another. However most of the time transcoding becomes computationally intensive process and complex task if done through command line/terminal. Due to this, one might need a graphical user interface which is easy to use by anyone .This “Multimedia Transcoder” project aims at providing a user-friendly GUI to easily and quickly convert videos/audios from one format to another , specify different codecs,resolutions,aspect ratios,frame rates ,bit rates.

CONTENTS

1	Introduction	7
1.1	Purpose	7
1.2	Scope	8
1.3	About Codecs and Formats	8
1.4	Abbreviations,Definitions	9
1.5	Drawbacks of Transcoding	10
2	Design Objectives and Functionality	
2.1	Design And Implementation Of GUI	11
2.2	2.2.1 FFmpeg	16
2.2.2	FFmpeg options used in “AV Transcoder”	17
2.2.3	Projects Using Ffmpeg	20
3	Application Requirements And Features	
3.1	Features Of AV Transcoder	
3.2	Software and Hardware Requirements	
3.3	Steps to install ffmpeg in ubuntu	
4	Diagrammatic Description	
4.1	Data Flow Diagram	
4.2	Usecase Diagram	
4.2.1	Usecase diagram for Video-Video converting	
4.2.2	Usecase diagram for Video-Audio Converter	
4.2.3	Usecase diagram for Codec	
4.3	Class Diagram	
4.4	Sequence Diagram	
4.4.1	Sequence diagram for Video-Video Conversion	
4.4.2	Sequence diagram for Video-Audio Conversion	
4.4.3	Sequence diagram for Play	
5	Conclusion and Future Scope	
5.1	How is “AV Transcoder” useful for Aakash?	
5.2	Conclusion	

Bibliography

1.

INTRODUCTION

Rich media resources transmitted over wired and wireless networks are becoming more and more important for our life. Users of devices, such as computers, PDAs, mobile phones, require accessing the Internet and networks from anywhere and at any time regardless of resources, network conditions, and capabilities of client devices. Intelligent media conversions, such as scalability, transcoding and modality changes, are the important means for multimedia adaptation and Universal Multimedia Access (UMA) .

Real-time transcoding in a many-to-many way (any input format to any output format) is becoming a necessity to provide true search capability for any multimedia content on any mobile device, with over 500 million videos on the web and a plethora of mobile devices.

1.1

PURPOSE

Transcoding is defined, in general, as the conversion of one compressed signal to another. Transcoding is the direct digital-to-digital data conversion of one encoding to another, such as for movie data files or audio files. This is usually done in cases where a target device (or workflow) does not support the format or has limited storage capacity that mandates a reduced file size, or to convert incompatible or obsolete data to a better-supported or modern format. The research work concerning transcoding is mainly focused on efficiency, complexity, latency, processing power, error resilience, bit rate change, resolution change, frame rate change, picture quality and so forth. To solve the problems, the coding methods, algorithms and international coding standards used for design and implementation of transcoding architectures are critical.

Due to the expansion and diversity of multimedia applications , present communication infrastructure comprised of different underlying networks and protocols and new and new PDA's being launched everyday into the market like the recently developed Aakash Tablet ,there has been a growing need for a single video being available in different formats and codecs supported by different platforms. New software codecs are used in difffernt consumer products to make them cheaper and more effective ,eg the latest being H264. For this there is a need of software that would quickly ,easily and efficiently convert a previously compressed video signal into another one with different format,such as different bit rate, frame rate, frame size, or even compression standard. An already available software for this purpose is FFmpeg. It is a complete, cross-platform solution to play, convert and stream audio and video. It has the leading multimedia framework, able to decode, encode, transcode, mux, demux, stream, filter and play pretty much anything that humans and machines have created. It supports the most obscure ancient formats up to the cutting edge,no matter if they were designed by some standards committee, the community or a corporation. But it is too difficult to be used by a common-user. It has to be executed using command-line /terminal which is not very convenient for everyone . “**AV Transcoder**” is an easy yet a professional solution for this. It is basically a simple and comfortable interface to FFmpeg .

It does all the work using FFmpeg in backend. User doesn't need to remember the complex commands . Everything can be done with click.

The main purpose of most video converter program is to convert media already on your computer to a more compatible format on other computers or digital devices. Video converter software is particularly useful when you own an digital device eg Aakash, but have a video with settings that cant be played on that. A conundrum many are running into is they are uploading and editing their home videos on their home computer, but they can't transfer and play them on their mobile or tablet. This is where AV Transcoder comes into picture.

1.2 SCOPE

“AV Transcoder” is compatible with Windows XP/Vista/7 and Ubuntu 11.10 and above.” For Ubuntu the FFmpeg version currently being used is FFmpeg 0.7.13 “peace”. For windows FFmpeg version used is FFmpeg-git-c-103dc0 32-bit Static. It can be upgraded further with newer versions. New Formats and Codecs can also be added in future

1.3 About Codecs and Formats

Before going into details of software some basic questions that should be answered are:

➤ Which codec to choose?

There are four principal goals to consider when choosing a codec:

1. The size of the compressed file
2. The speed of compression
3. The speed of decompression
4. The quality of the final video/audio/image.

As you are starting to see, the list of codecs we can use for video (fast compression and fast decompression) is starting to dwindle.

- The first, and best, rule to keep in mind when choosing a video codec is to select the codec that matches the format of the video you are going to convert.
- The second rule is if you are selecting a codec to integrate multiple video formats choose the codec that provides the highest quality.
- The third rule is to select your codec based upon your final output format.

➤ What are Formats and Specifications?

The word “format” is used for many things in video, but it tends to confuse people for this very reason. So let's clarify some definitions real quick:

Video Format: This include filetypes, like MPEG-1 (MPG), MPEG-2 (MPG), Windows Media (WMV), H.264 (MP4), Divx and Xvid (AVI), FlashVideo (FLV), etc

Playback Format: This refers to the final product used when viewing the video. Examples include DVD-Video (DVD), VideoCD (VCD)

Video Specifications: For example, DVD-Video specification uses MPEG-1 or MPEG-2 with certain resolutions and settings only — any old MPEG-1 or MPEG-2 will not suffice.

➤ **What is the difference between a codec and a video format?**

First-time users often stumble when trying to figure out the difference between codecs and formats. A codec is a method for encoding or decoding data--specifically, compressed data. Though the word *codec* has become somewhat generic, the term was originally a shortened form of *compressor-decompressor*. That's what codecs do: they take digital media data and either compress them (for transport and storage) or decompress them (for viewing or transcoding.)

1.4

ABBREVIATIONS,DEFINITIONS

3GP	Third Generation Partnership Project
AAC	Advanced Audio Coding
Ac3	Dolby Digital Sound File
AMR	Adoptive Multirate
AVI	Audio Video Interleave
DV Standard	Digital Video
FFmpeg	Fast Forward Motion Picture Experts Group
FFplay	Fast Forward Play
FLV	Flash Video
OGG	Operation Good Guys
OGV	Ogg Video
MPEG	Motion Picture Experts Group
MP4	Moving Picture Expert Group-4
NTSC	National Television System Committee
PAL-VCD	Phase Alternating Line Video Compact Disc
SVCD	Super Video Compact Disc
WAV	Windows Wave
WMA	Windows Media Audio
WMV	Windows Media Video

1.5**DRAWBACKS OF TRANSCODING**

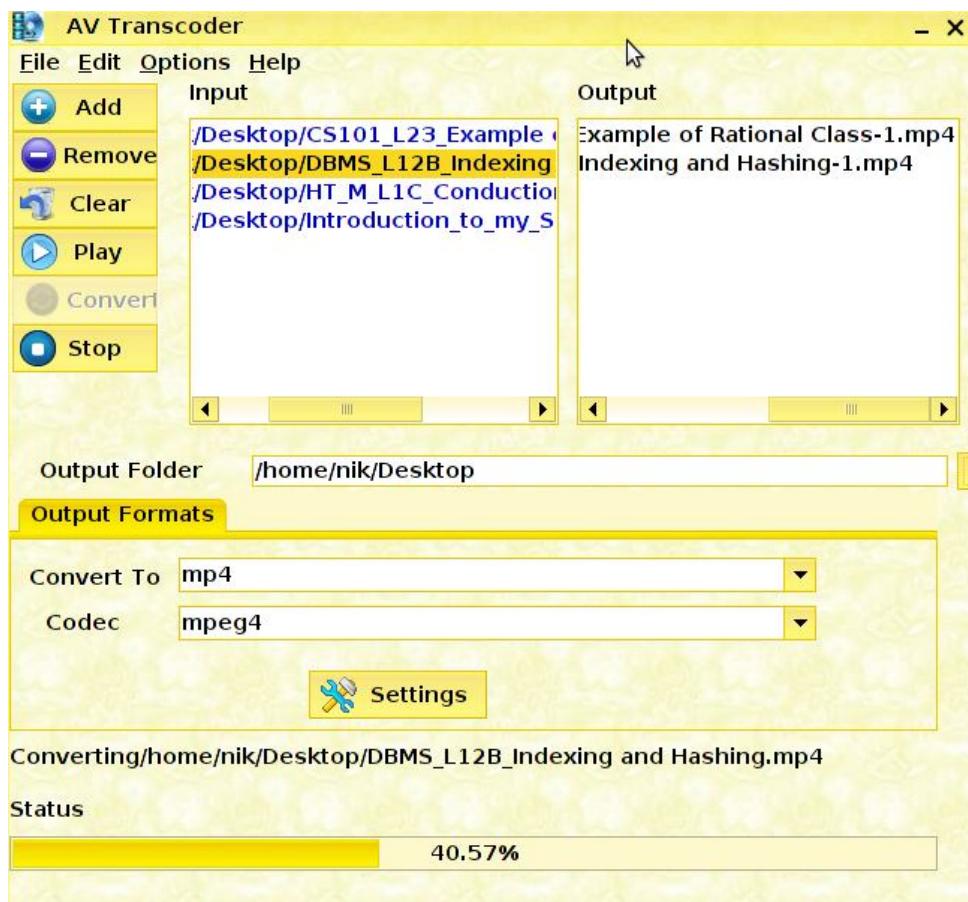
The key drawback of transcoding in lossy formats is decreased quality. Compression artifacts are cumulative, so transcoding causes a progressive loss of quality with each successive generation, known as digital generation loss. For this reason, transcoding is generally discouraged unless unavoidable.

It is better to retain a copy in a lossless format (such as TTA, FLAC or WavPack for sound), and then encode directly from the lossless source file to the lossy formats required. For image and digital audio editing, one is advised to capture or save images in a raw or uncompressed format and edit (a copy of) that version, only converting to lossy formats for distribution.

Although the loss is unavoidable, it can be minimized by choosing the appropriate resolution and bit-rate.

2 DESIGN OBJECTIVES AND FUNCTIONALITY

2.1 DESIGN AND IMPLEMENTATION OF GUI



- **Add Button:** It is used to add files to the input List by browsing. Multiple selection can be done.
- **Remove Button:** It is used to remove the selected file from input/output List.
- **Clear Button:** It is used to clear the entire input and output List.
- **Play Button:** It is used to play the selected input/output file from List. It uses FFplay for playing the file.
While Playing the following keys are available:
Q and ESC – quit.
F – toggles Fullscreen.
P and SPACE – Pause.
A – cycle audio channel.
V – cycle video channel.
T – cycle subtitles.
W – Show audio waves.

Left/Right – Seek backward/forward 10 seconds.

Up/Down - Seek backward/forward 1 minute

Page up/Page down- seek backward/forward 10 minutes

Mouse click-seek to percentage of file corresponding to fraction of width.

- **Convert Button:** It is used to convert the input files to different formats and codecs.
 - It runs the FFmpeg on files in the List
 - Different formats can be selected from **Convert To** option in **Output Formats** tab.
 - Different Codecs can be selected from **Codec** option.
- **Stop Button:** It is used to stop the conversion of selected file . Conversion of next input file from List will start subsequently.
- **Output Folder:** The default output folder is the current working directory . The user can choose any destination folder by browsing using (...) button.
- **Output Formats:**
 - **Convert To:** It is used to select the different output formats from the listed ones. The various formats already available are:

Mp4,avi,flv,wmv,ogv,webm,aac

User can add more by edit option that is explained later.

- **Codec:** It is used to select the different codec from the listed ones.

Svc,dvd,dv,dv50,pal-vcd,ntsc-svcd,mpeg-4,mpeg-4 highly compressed,H264,H264-highly compressed.

User can add more by edit option that is explained later.

- **Settings Button:**

Additional options can be selected by using settings button.
- **Video:**
 - **Video Bit rate:** It is used to select different bitrates from listed values.

Bit rate is a measure of the rate of information content in a video stream. It is quantified using the bit per second(*bit/s* or *bps*) unit or *Megabits per second (Mbit/s)*. A higher bit rate allows better video quality. For example VideoCD, with a bit rate of about 1 Mbit/s, is lower quality than DVD, with maximum bit rate of 10.08 Mbit/s for video. HD (High Definition Digital Video and TV) has a still higher quality, with a bit rate of about 20 Mbit/s.

The various Bit rates available are:
"200 ", "350 ", "500 ", "700 ", "850 ", "1000 ", "1024 "
 - **Video Frame Rate:** It is used to select different frame rates from listed values.

Frame rate, the number of still pictures per unit of time of video, ranges from six or eight frames per second (*frame/s*) for old mechanical cameras to 120 or more frames per second for new professional cameras. PAL (Europe, Asia, Australia, etc.), while

- NTSC (USA, Canada, Japan, etc.) specifies 29.97 frame/s. Film is shot at the slower frame rate of 24photograms/s, which complicates slightly the process of transferring a cinematic motion picture to video. The minimum frame rate to achieve the illusion of a moving image is about fifteen frames per second.

Various Frame rates available are:

"15 ","20 ","24 ","25 ","30 ","60"

- **Resolution:** It is used to set the different resolutions for videos.

The display resolution of a digital television or display device is the number of distinct pixels in each dimension that can be displayed. It can be an ambiguous term especially as the displayed resolution is controlled by all different factors in cathode ray tube (CRT), flat panel or projection displays using fixed picture-element (pixel) arrays.

It is usually quoted as width × height, with the units in pixels: for example, "1024 × 768" means the width is 1024 pixels and the height is 768 pixels. This example would normally be spoken as "ten twenty-four by seven sixty-eight" or "ten twenty-four by seven six eight".

Various resolutions available are:

"128x96","176x144","160x128","290x240","320x240","352x288","360x240","640x480","720x576","704x576","1024x768","1408x1152"

- **Aspect Ratio:** It is used to set the aspect ratio for video.

Aspect ratio describes the dimensions of video screens and video picture elements. All popular video formats are rectilinear, and so can be described by a ratio between width and height. The screen aspect ratio of a traditional television screen is 4:3, or about 1.33:1. High definition televisions use an aspect ratio of 16:9, or about 1.78:1. The aspect ratio of a full 35 mm film frame with soundtrack (also known as the Academy ratio) is 1.375:1.

Ratios where the height is taller than the width are uncommon in general everyday use, but do have application in computer systems where the screen may be better suited for a vertical layout. The most common tall aspect ratio of 3:4 is referred to as *portrait mode* and is created by physically rotating the display device 90 degrees from the normal position. Other tall aspect ratios such as 9:16 are technically possible but rarely used. (For a more detailed discussion of this topic please refer to the page orientation article.)Pixels on computer monitors are usually square, but pixels used in digital video often have non-square aspect ratios, such as those used in the PAL and NTSC variants of the CCIR 601 digital video standard, and the corresponding anamorphic widescreen formats. Therefore, an NTSC DV image which is 720 pixels by 48.

Aspect ratios available are:

"4:3","16:9"

➤ **Audio:**

- **Audio Bit rate:** It is used to select different bitrates from listed values.

It is the rate of compression used to encode the file in the audio aspect. The higher the rate is, the better the quality of sound is.

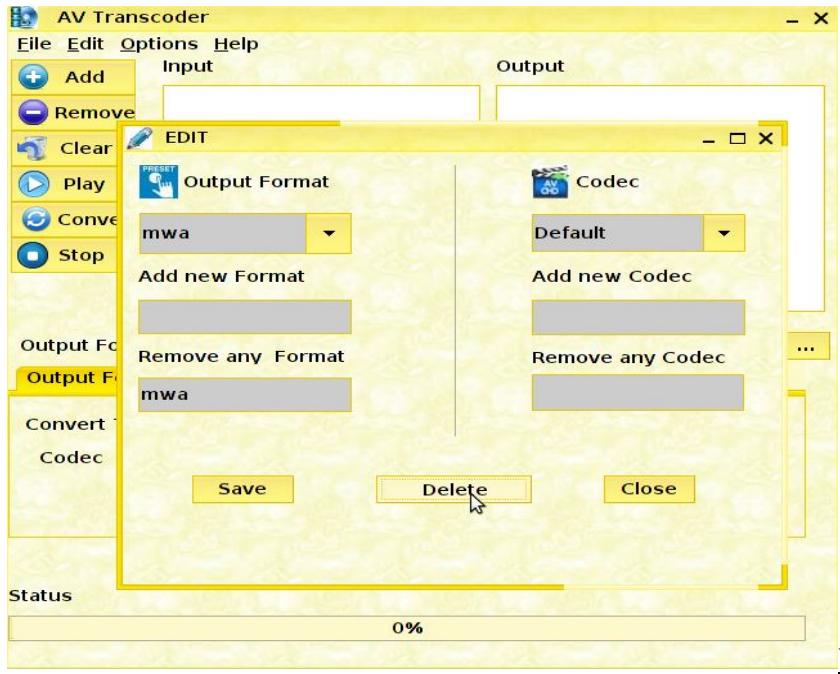
Audio Bit rates available are:

"32","64","128","150","160","200"

- **Sample Rate:** It is used to select different frame rates from listed values. The audio sampling rate is number of samples per second that are used to digitize a particular sound. Most digital video (DV) cameras can record at several audio sampling rates, where higher rates produce better results.
Sample rates available are:
"8000", "11025", "16000", "22050", "32000", "44100", "48000"
 - **Audio Channels:** It is used to set number of audio channels to either one or two. By default it is 2.
A "channel" in audio is just one separate stream of audio information. Mono audio sources have one channel. Stereo sources have two (left and right.)
- **Split:** It is used to split video/Audio files by specifying **Start time** and **Duration**.
- **FFmpeg:** It is used to specify any additional commands and options that an advanced user wants to execute using ffmpeg.
- **Status:** It shows the progress of video/audio which is being converted. Percentage of converted video is displayed .
- **File Menu:**
 - **Add:** It is used to add the files to the input List.
 - **Exit:** It is used to exit from the program.
- **Edit Menu:**
 - **Presets:** It is used to open the preset editor.
 - **Themes:** It is used to change the LookAndFeel of GUI.
- **Options Menu:**
 - **Additional Options:** It is used to show the additional options tab same as settings button.
 - **Start Converting:** It is used to start the conversion process.
 - **Stop Converting All:** It is used to stop the converting process of all the files in the List.
- **Help Menu:**
 - **User Manual:** It is used to show the user manual of this project.
 - **FFmpeg Website:** It is used to browse the FFmpeg website.
 - **About:** It shows additional information .

EDIT (Format and Codec)

This option is provided so that user can update formats and codecs in future because every now and then new codecs are launched and different PDA's coming in market support certain specific formats and codecs.



- **Add new preset :** A new format can be added by typing in textfield .
- **Add new Codec:** A new codec can be added by typing in textfield.
- **Remove any preset:** It is used to delete a format from the List of output formats.
- **Remove any Codec:** It is used to delete a codec from the List of codecs.
- **Save Button:** It is used to add the preset and/or codec in json file.
- **Delete Button:** It is used to delete the preset and/or codec from json file.
- **Close Button:** It is used to close the editor window . The changes made are reflected in **Convert to** and **Codec** option of video converter

2.2.1

FFmpeg (The backend Software used)

FFmpeg is a complete, cross-platform solution to record, convert and stream audio and video. It includes libavcodec - the leading audio/video codec library.

ffmpeg reads from an arbitrary number of input "files" (which can be regular files, pipes, network streams, grabbing devices, etc.), specified by the -i option, and writes to an arbitrary number of output "files", which are specified by a plain output filename. Anything found on the command line which cannot be interpreted as an option is considered to be an output filename.

Each input or output file can in principle contain any number of streams of different types (video/audio/subtitle/attachment/data). Allowed number and/or types of streams can be limited by the container format. Selecting, which streams from which inputs go into output, is done either automatically or with the -map option (see the Stream selection chapter).



FFmpeg provides various tools:

- ffmpeg is a command line tool to convert multimedia files between formats.
- ffplay is a simple media player based on SDL and the FFmpeg libraries.
- ffprobe is a simple multimedia stream analyzer.

and developers libraries:

- libavutil is a library containing functions for simplifying programming, including random number generators, data structures, mathematics routines, core multimedia utilities, and much more.
- libavcodec is a library containing decoders and encoders for audio/video codecs.
- libavformat is a library containing demuxers and muxers for multimedia container formats.
- libavdevice is a library containing input and output devices for grabbing from and rendering to many common multimedia input/output software frameworks, including Video4Linux, Video4Linux2, VfW, and ALSA.
- libavfilter is a library containing media filters.
- libswscale is a library performing highly optimized image scaling and color space/pixel format conversion operations.
- libswresample is a library performing highly optimized audio resampling, rematrixing and sample format conversion operations.

2.2.2**“FFmpeg options used in “AV Transcoder”**

- ‘-i *filename (input)*’

input file name

- ‘-target *type (output)*’

Specify target file type (vcd, svcd, dvd, dv, dv50). *type* may be prefixed with pal-, ntsc- or film- to use the corresponding standard. All the format options (bitrate, codecs, buffer sizes) are then set automatically.

Split options

- ‘-t *duration (output)*’

Stop writing the output after its duration reaches *duration*. *duration* may be a number in seconds, or in hh:mm:ss[.xxx] form.

- ‘-ss *position (input/output)*’

When used as an input option (before -i), seeks in this input file to *position*. When used as an output option (before an output filename), decodes but discards input until the timestamps reach *position*. This is slower, but more accurate.
position may be either in seconds or in hh:mm:ss[.xxx] form.

Video options

- ‘-vf *frames number (output)*’

Set the number of video frames to record. This is an alias for -frames:v.

- ‘-r[:*streamSpecifier*] *fps (input/output,per-stream)*’

Set frame rate (Hz value, fraction or abbreviation). As an input option, ignore any timestamps stored in the file and instead generate timestamps assuming constant frame rate *fps*. As an output option, duplicate or drop input frames to achieve constant output frame rate *fps* (note that this actually causes the fps filter to be inserted to the end of the corresponding filtergraph).

- ‘-s[:*streamSpecifier*] *size (input/output,per-stream)*’

Set frame size. As an input option, this is a shortcut for the ‘video_size’ private option, recognized by some demuxers for which the frame size is either not stored in the file or is configurable – e.g. raw video or video grabbers. As an output option, this inserts the scale video filter to the end of the corresponding filtergraph. Please use the

scale filter directly to insert it at the beginning or some other place. The format is ‘wxh’ (default - same as source).

- ‘-aspect[:*streamSpecifier*] *aspect* (*output,per-stream*)’

Set the video display aspect ratio specified by *aspect*. *aspect* can be a floating point number string, or a string of the form *num:den*, where *num* and *den* are the numerator and denominator of the aspect ratio. For example "4:3", "16:9", "1.3333", and "1.7777" are valid argument values.

- ‘-vcodec *codec* (*output*)’

Set the video codec. This is an alias for -codec:v.

Audio options

- ‘-aframes *number* (*output*)’

Set the number of audio frames to record. This is an alias for -frames:a.

- ‘-ar[:*streamSpecifier*] *freq* (*input/output,per-stream*)’

Set the audio sampling frequency. For output streams it is set by default to the frequency of the corresponding input stream. For input streams this option only makes sense for audio grabbing devices and raw demuxers and is mapped to the corresponding demuxer options.

- ‘-ac[:*streamSpecifier*] *channels* (*input/output,per-stream*)’

Set the number of audio channels. For output streams it is set by default to the number of input audio channels. For input streams this option only makes sense for audio grabbing devices and raw demuxers and is mapped to the corresponding demuxer options.

- ‘-acodec *codec* (*input/output*)’

Set the audio codec. This is an alias for -codec:a.

2.2.3

PROJECTS USING FFMPEG

FFmpeg is used by many open source (and proprietary) projects, including ffmpeg2theora, VLC, MPlayer, HandBrake, Blender, Google Chrome, YouTube, FaceBook, and others.

FFmpeg handles the HTML5 Video and Audio media content in the Origyn Web Browser for MorphOS Operating System.

A number of graphical user interfaces for FFmpeg have been developed, including Avanti, Winff, and the Miro Video Converter

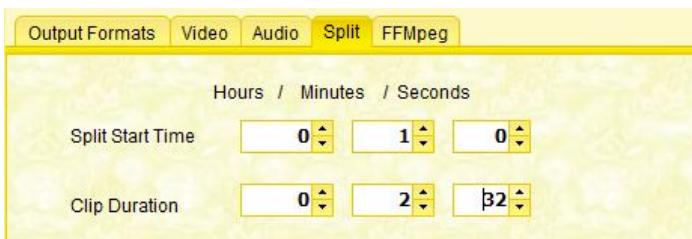
3 DESIGN CONSIDERATIONS AND FEATURES

3.1 Features Of AV Transcoder

- Converts multiple different files all to the same format at once.
- Extracts audio from video.
- Converts between audio formats.
- Easy access to common conversion options such as bitrate, frame size, frame rate, aspect ratio.



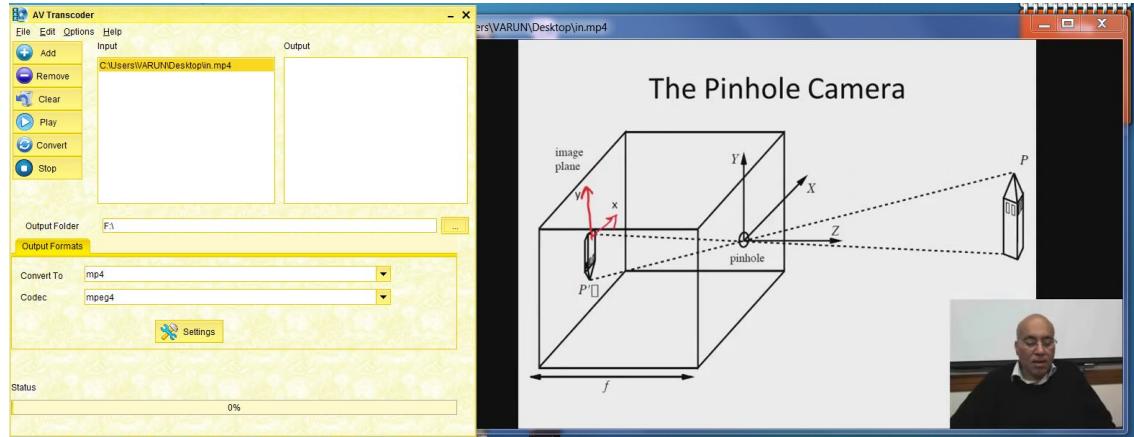
- Specification of additional command line parameters for advanced users .
- Splits video by specifying the start time and duration and change the settings at same time.



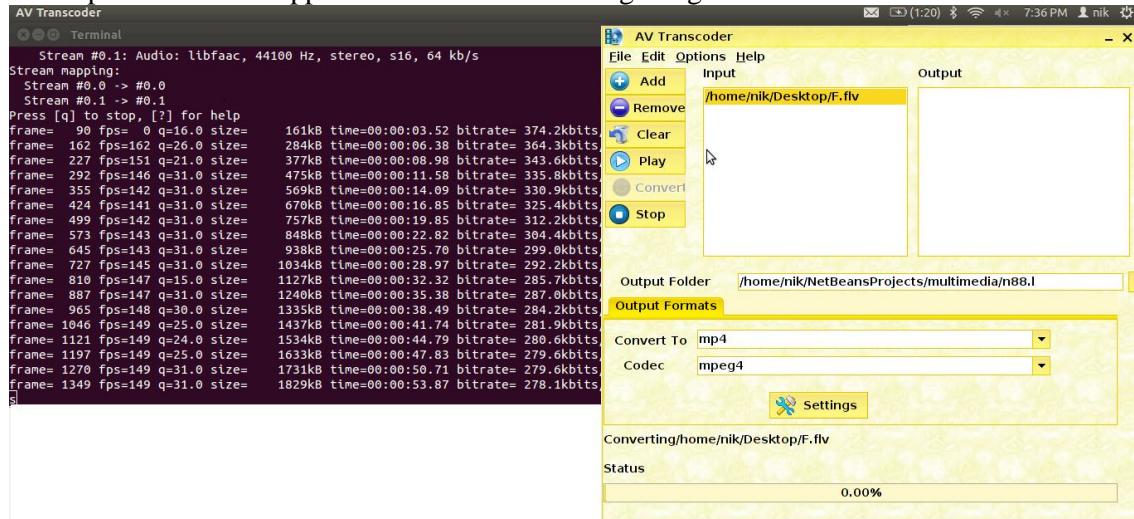
- For same format there are multiple options eg for mpeg there is standard and compressed format.
- Support for H264 is also provided.
- User can directly set the settings as svcd,pal-vcd,dvd,ntsc-dvd,dv50 etc.
- Percentage of Video being converted is displayed side-by-side and progress is shown using progress bar.

AV TRANSCODER

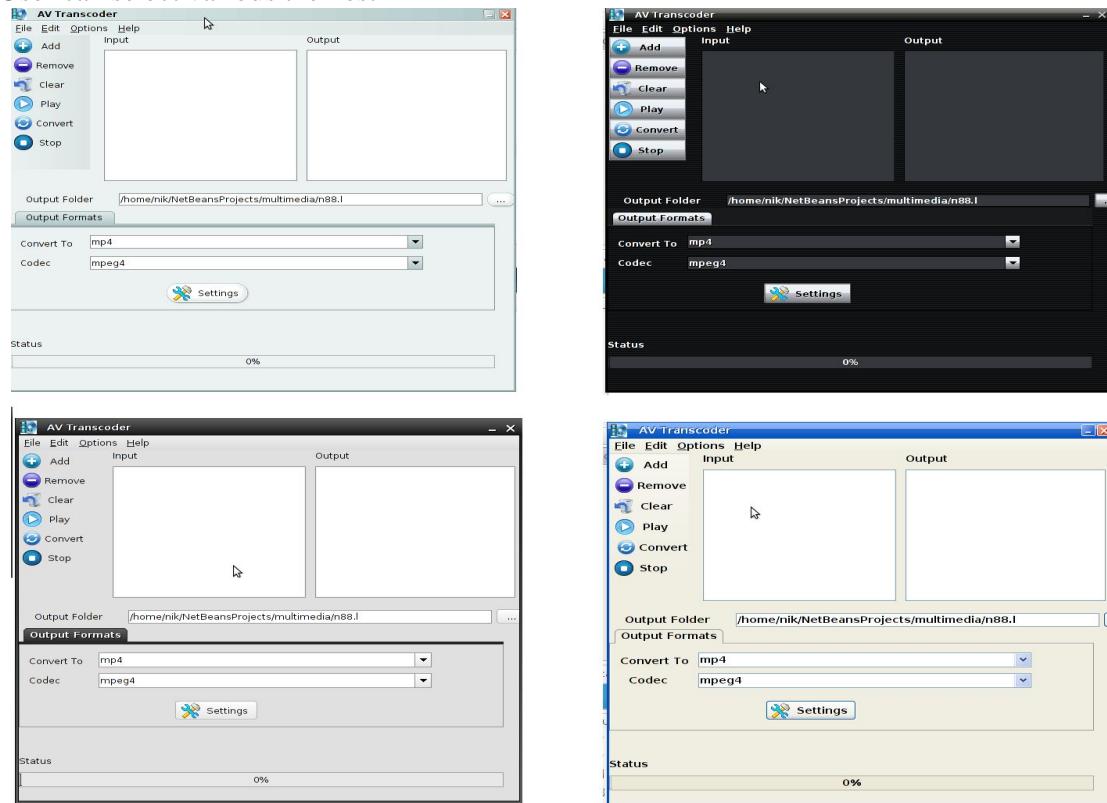
- Video/Audio can be directly played from input/output list of videos by selecting from list.



- User is provided with support to view how video is getting converted.



- User can select various themes.



3.2

Software and Hardware Requirements

- Pentium III-based PC or greater
- Windows and Ubuntu
- Better the processor faster will be the conversion
- 50 MB available hard disk space
- FFmpeg

3.3

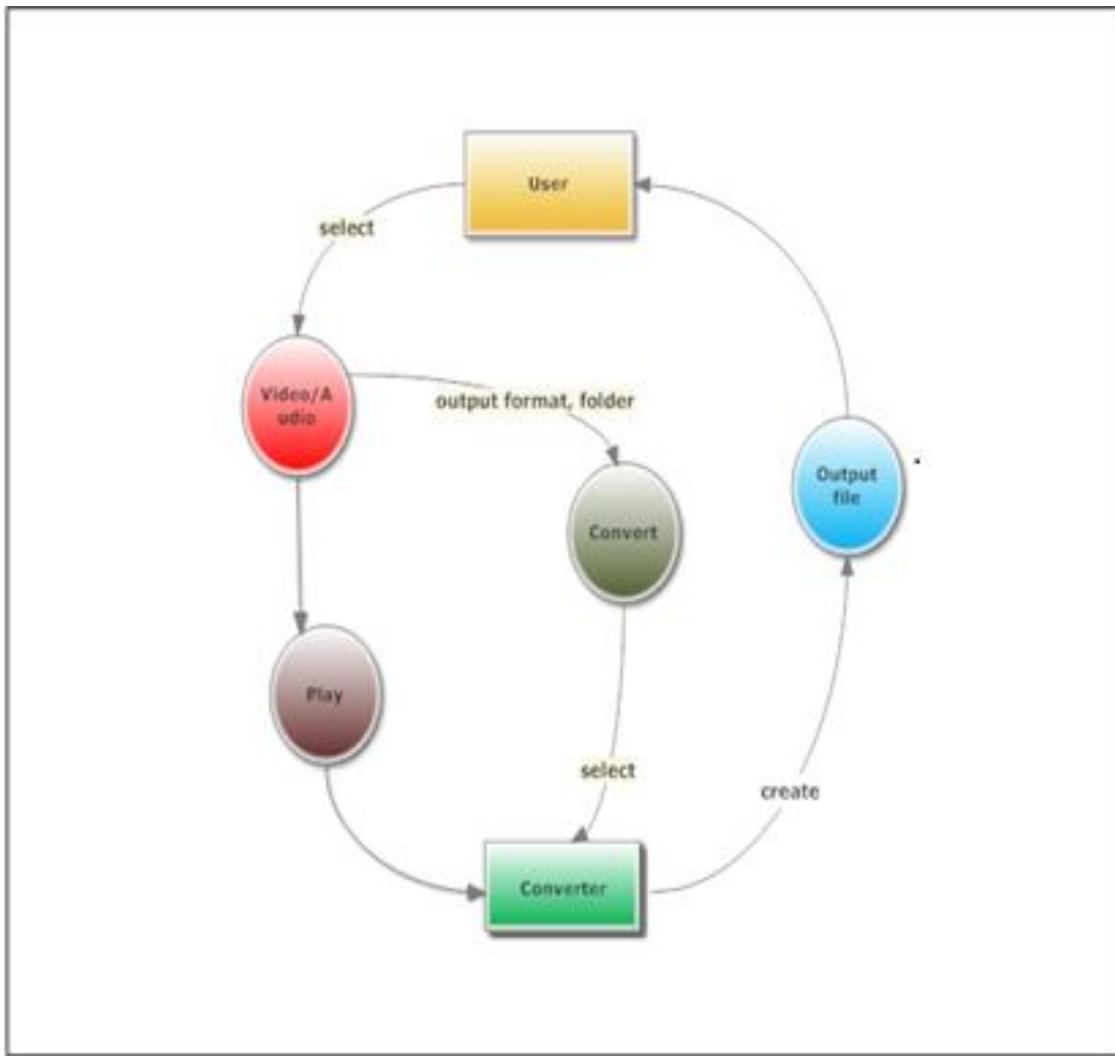
Steps to install ffmpeg in ubuntu

1: download ffmpeg from following link-<http://ffmpeg.org/releases/ffmpeg-0.7.12.tar.bz2>
extract the file

2: Go to terminal:-

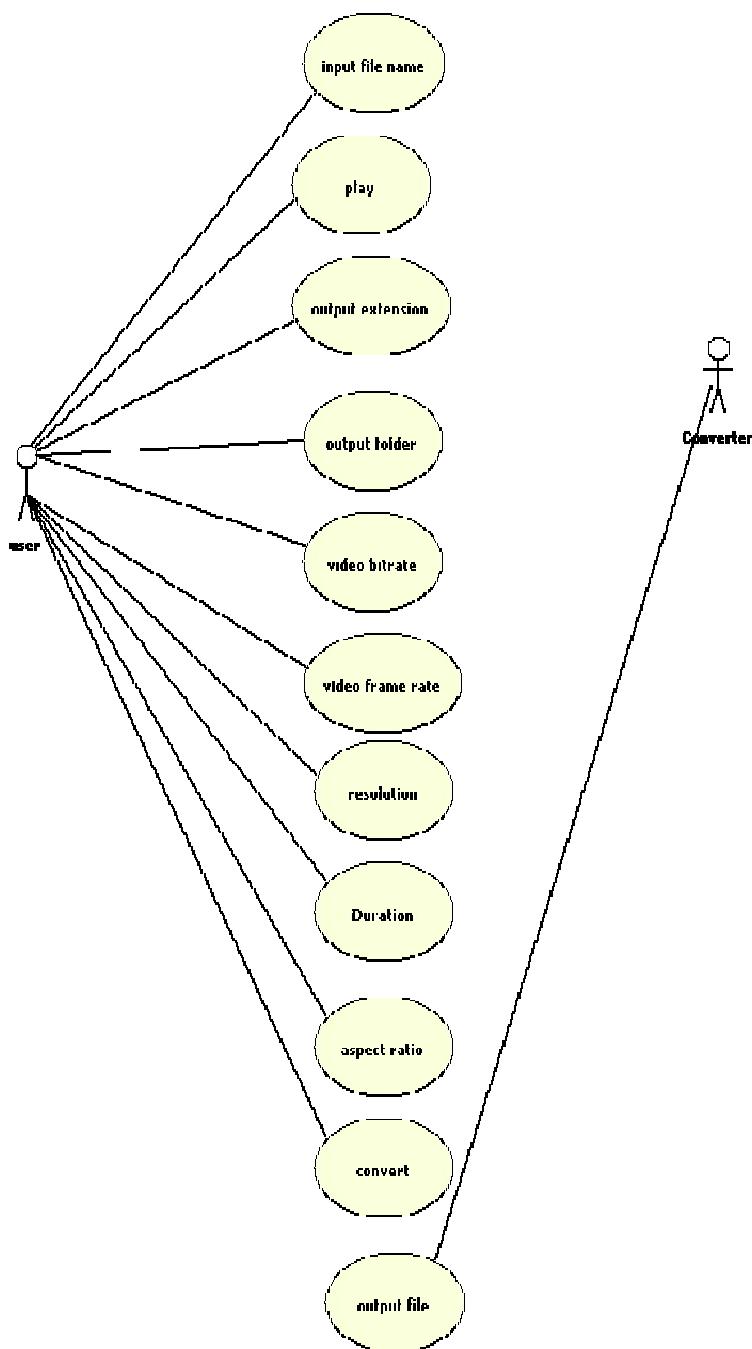
change the current working directory to the place where ffmpeg is installed
 sudo apt-get remove ffmpeg
 sudo apt-get install yasm
 sudo apt-get install libmp3lame-dev
 sudo apt-get install libfaac-dev
 sudo apt-get install libtheora-dev
 sudo apt-get install libvorbis-dev

```
sudo apt-get install libvpx-dev
sudo apt-get install libxvidcore-dev
sudo apt-get install libxvidcore-dev
sudo apt-get install libopencore-amrnb-dev
sudo apt-get install libopencore-amrwb-dev
sudo apt-get install liba52-dev
sudo apt-get install libx264-dev
apt-get update
apt-get install libx264-dev x264
./configure --enable-gpl --enable-version3 --enable-nonfree --enable-postproc --enable-libfaac --enable-libmp3lame --enable-libopencore-amrnb --enable-libopencore-amrwb --enable-libtheora --enable-libvorbis --enable-libvpx --enable-libx264 --enable-libxvid enable-x11grab
make
make install
```

DIAGRAMMATIC DESCRIPTION**4.1 Data Flow Diagram**

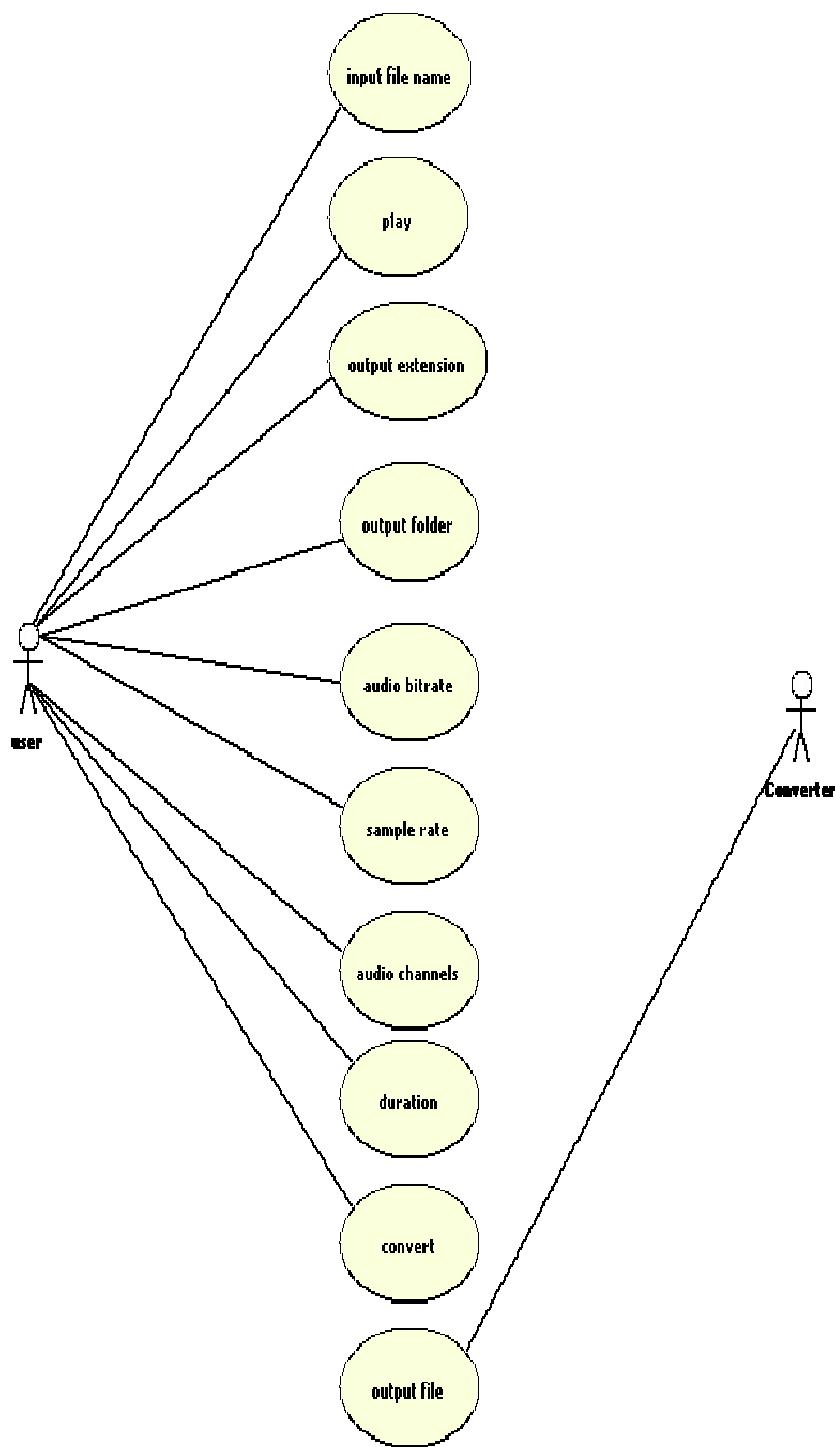
4.2 Usecase Diagram:

4.2.1 Usecase diagram for video-video converting:

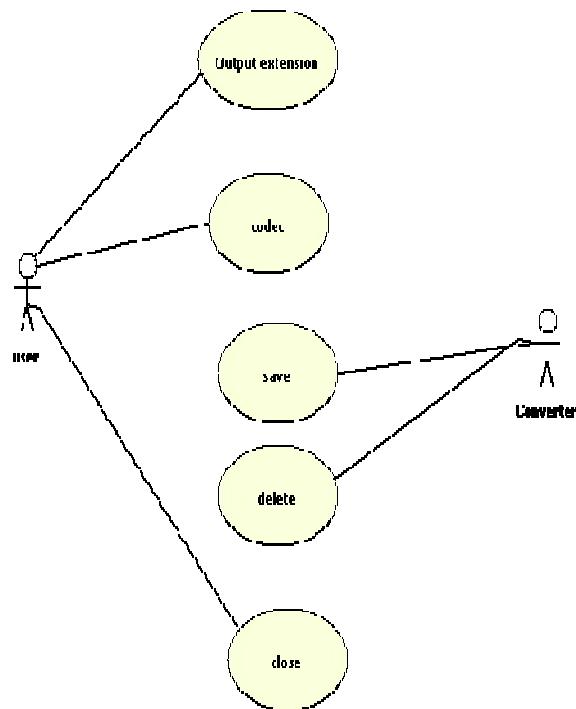


4.2.2 Usecase diagram for Video-Audio Converter:

AV TRANSCODER

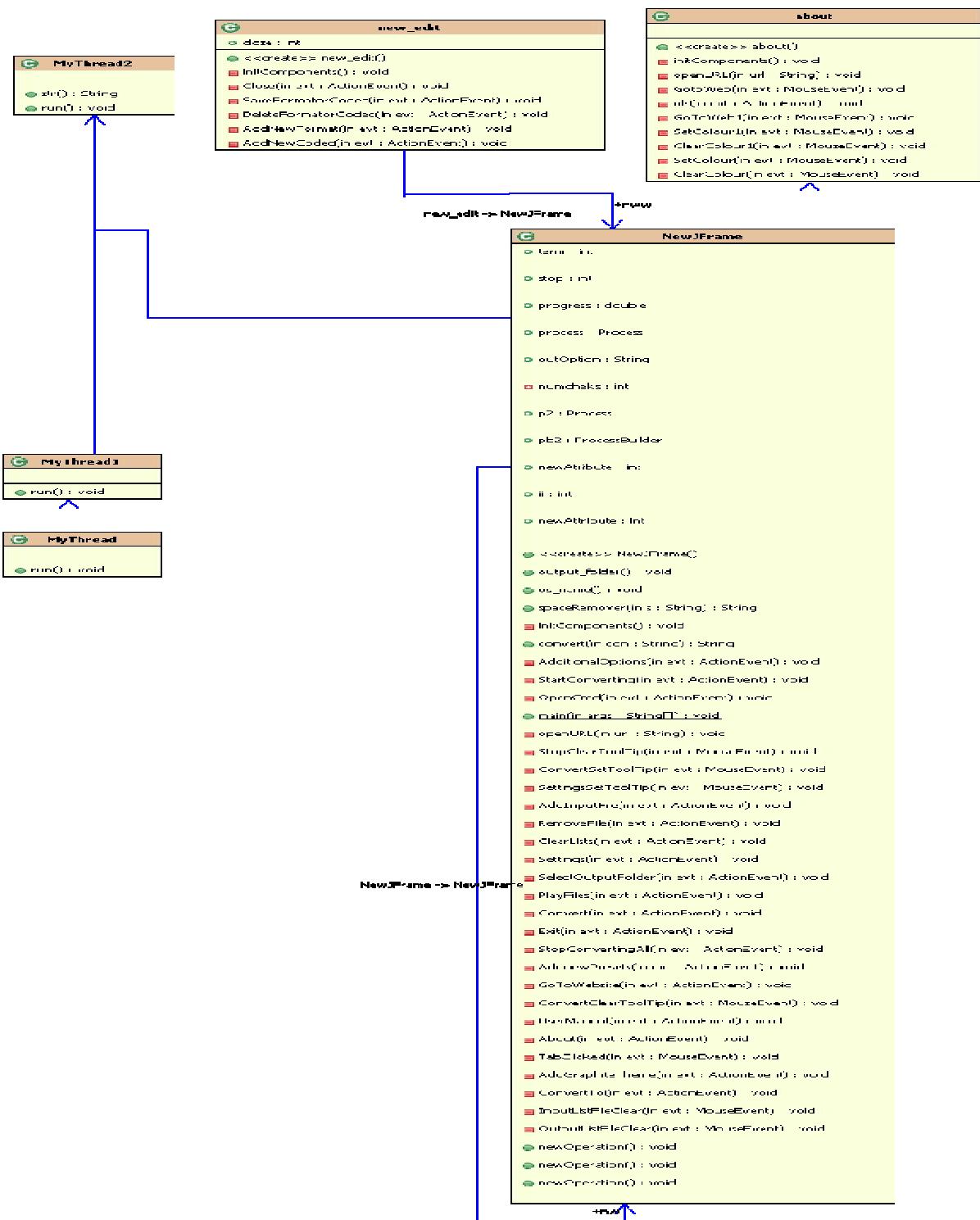


4.2.3 Usecase diagram for Codec:



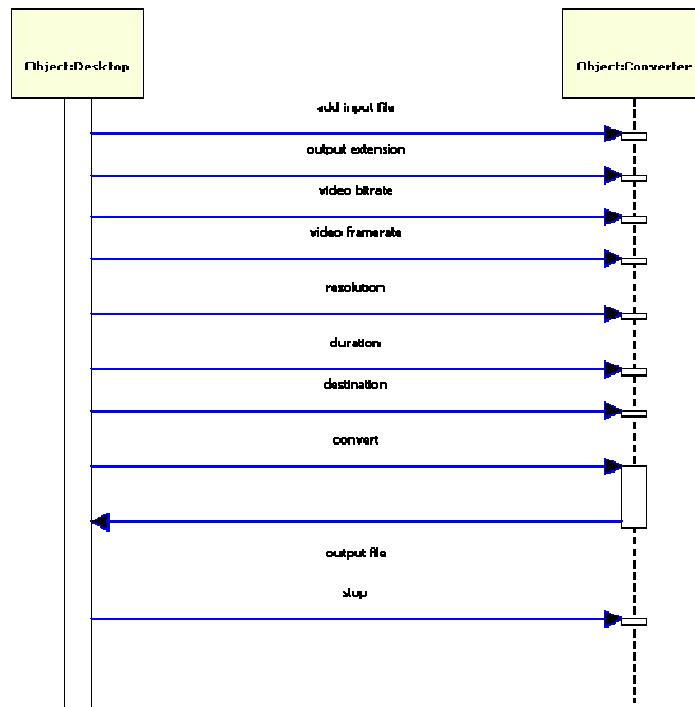
4.3 Class Diagram:

AV TRANSCODER

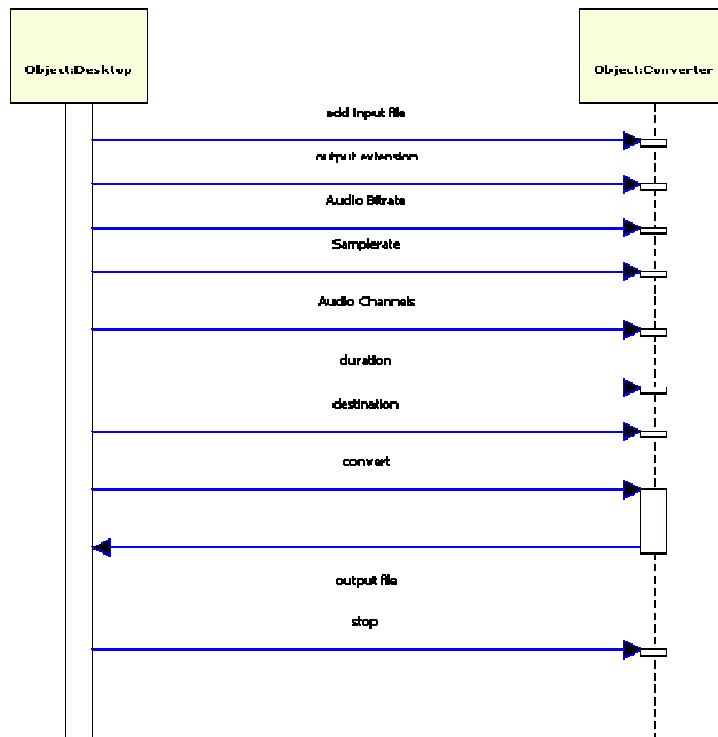


4.4 Sequence Diagram:

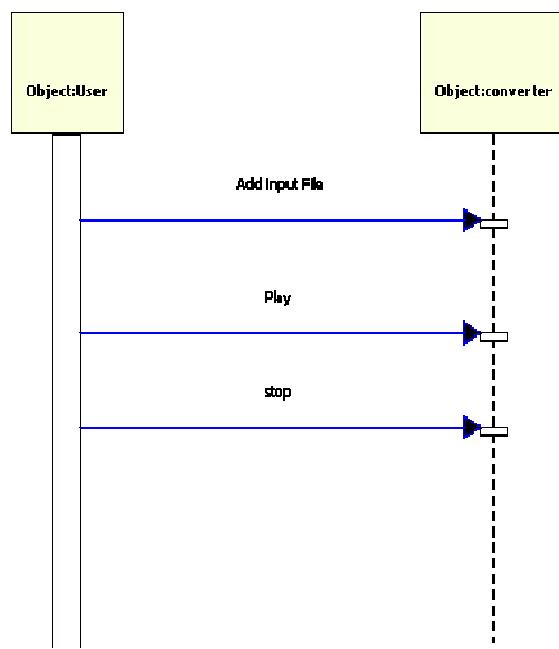
4.4.1 Sequence diagram for Video-Video Conversion:

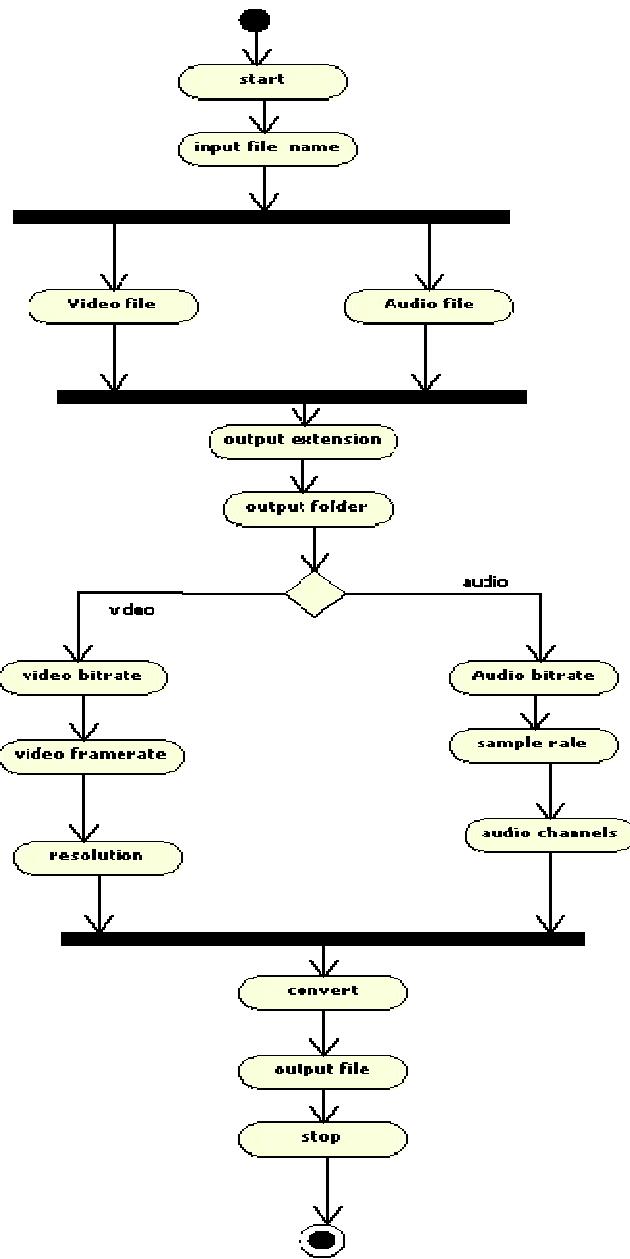


4.4.2 Sequence diagram for Video-Audio Conversion:



4.4.3 Sequence Diagram For Play_:



4.5 Activity Diagram:

5.1 How is “AV Transcoder” useful for Aakash?

- Aakash tablet supports mp4 and 3gp formats and with “AV transcoder” videos from other formats can be easily converted to the these formats.
- The compressed video will have its size reduced so it will require less memory ,thus many videos can be stored in limited memory space of Aakash.
- User can set the settings like frame rate,video bit rate,resolution,aspect ratio, audio sample rate, audio bit rate ,audio channel depending upon quality of video required in Aakash.

5.2**CONCLUSION**

AV Transcoder offers to crisscross video formats in a matter of seconds. We are able to easily change the file type of our videos with this impressive program.

Its interface is very user-friendly and welcoming, even for someone who has limited experience with videos and file conversion. Its helpfully labeled commands practically walks us through the entire process . Once we'd picked the videos we wanted to convert, we are able to choose from many file formats to convert them into and it has made the conversion incredibly easy.. The program breaks down the process into simple steps so even inexperienced users can quickly convert all sorts of files and for advanced users there are additional options. It is compatible with both Windows and Linux.

In nutshell, AV Transcoder has great future scope especially when new PDA's like Aakash are being launched having less memory for which compressed videos with high resolution are needed.

BIBLIOGRAPHY

- www.ffmpeg.org
- Transcoding -Wikipedia
- Java Swing (Robert Eckstein, Marc Loy and Dave Wood)
- JAVA-Complete Reference
- stackoverflow.com
- Codec-Wikipedia