UNIVERSITÄT KLAGENFURT

ITEC: Institute for Information Technology

Softwarepraktikum
Prof. Dr. Laszlo Böszörmenyi
Projektleiter: DI Mike Kropfberger, DI Peter Schojer

MuViTrans
A Multi Video Transcoder

Project members:
Birgit Antonitsch
0060646
bantonit@edu.uni-klu.ac.at

Helga Fiebiger
0160002
hfiebige@edu.uni-klu.ac.at

Sigrid Kuchler
0160005
skuchler@edu.uni-klu.ac.at

Table of contents

1 Project description 4

1.1 Milestones 4

1.2 Responsibility 4

2 Design of MuViTrans 5

2.1 Class hierarchy 5

2.2 Structure of a configuration file: 6

3 GUI 8

3.1 Main GUI 8

3.1.1 Menu bar 9

3.1.2 Buttons 11

3.2 Destination Video 14

3.2.1 Job file Buttons 14

3.2.2 General Job file properties 15

3.2.3 Extended job file properties 18

3.3 Destination Audio 20

3.3.1 Job file properties 20

3.4 Progress 22

4 Possible Future extensions 23

4.1 Destination Video 23

4.2 User Define 24

4.3 Destination Audio 24

Table of figures

Figure 1: Class Diagram 5

Figure 2: MultiTranscoder 8

Figure 3: WorkingDirectory 10

Figure 4: ProjectName 10

Figure 5: Delete Job 12

Figure 6: SystemFile 12

Figure 7: DestinationVideo General 14

Figure 8: User Define 16

Figure 9: DestinationVideo Extended 18

Figure 10: DestinationAudio 20

Figure 11: Progress 22

1 Project description

MuViTrans - the Multi Video Transcoder - is an application build on top of the ViTooki library with the focus on easy transcoding of multimedia streams. A powerful but still simple to use GUI assists novel as expert users in defining for each stream a set of transcoding jobs. Via the GUI, the user can then combine input files and transcoded streams into new system files. Due to the fact that transcoding is a time consuming operation, MuViTrans separates job definition from transcoding and system file creation. Users first specify the work to be done, then the transcoding is initiated either through the GUI or the work is saved to a project file and executed at a later time point by a command line tool. As input format AVI, MPEG-1, MPEG-4, YUV and MP3 is supported, as output format MPEG-4.

1.1 Milestones

The project “MuViTrans” started in October 2003 and took one year (two semesters).

Implementation of a command-line-tool, which changes resolution and color of an input stream	finished 10/2003
Design of the GUI	12/2003
Design of class hierarchy	12/2003
Implementation of functionality of GUI	2/2004 -7/2004
Implementation of class hierarchy	finished 3/2004
Review of the GUI design	5/2004
Implementation of transcoder	7/2004
Testing & error correction	5/2004

1.2 Responsibility

All members (Antonitsch, Fiebiger, Kuchler) are responsible for the GUI design.

Antonitsch: Implementation of the functionality of the GUI, especially the main GUI, which gives the possibility to load sources, create system files and opens the forms for

creating jobs. Further more the form “WorkingDirectory” and “ProjectName” were implemented.

Fiebiger: Implementation of the functionality of the GUI, especially creating and editing audio jobs or video jobs with all kinds of settings. Further more the form “UserDefine” and “Progress” were implemented.

Kuchler: Design and implementation of the class hierarchy. Implementation of the transcoder and responsibility for the batch file support.

2 Design of MuViTrans

2.1 Class hierarchy

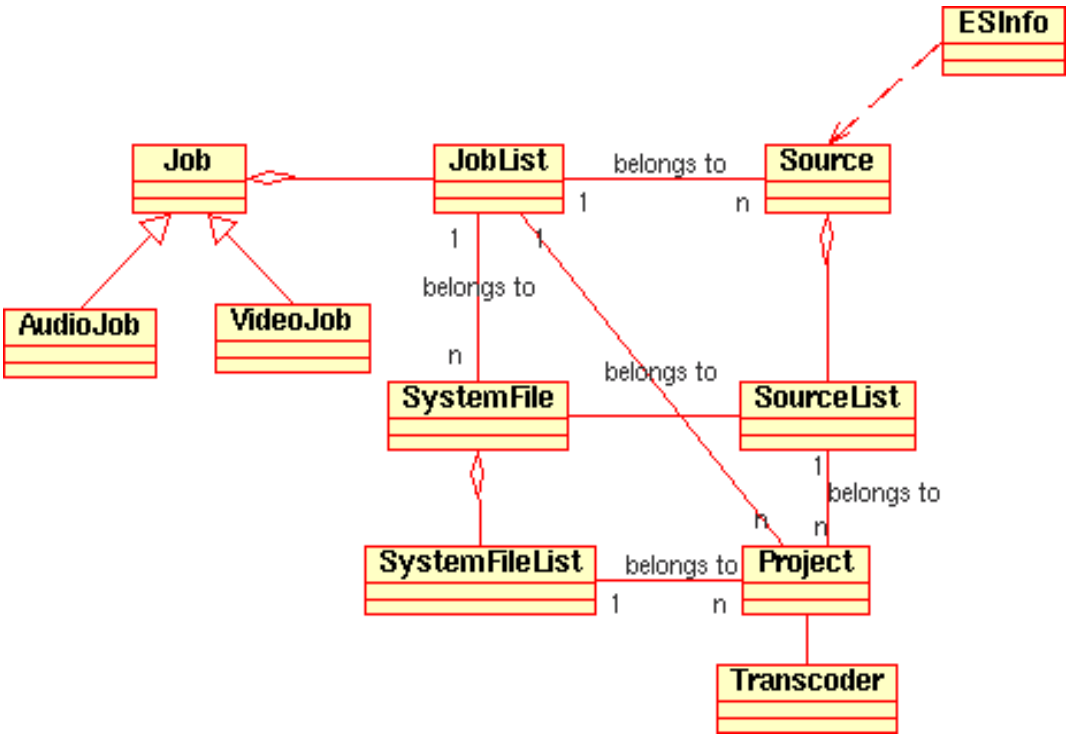

Figure 1: Class Diagram

Figure 1 shows the class hierarchy of MuViTrans. Every project has a name and three lists: a source list, a job list and a system file list.

The source list contains all sources which are used in this project. A source consists of an elementary stream, a job list and a reference counter, which checks how often the source is referenced.

The job list contains audio- and video jobs. Every job has a unique jobID, a name, a source and an output filename. With video jobs one can specify parameters like bit rate, width, height or target codec. With audio jobs parameters like sample rate or codec can be specified.

The system file list contains system files that will be created during transcoding. A system file consists of several elementary streams that are multiplexed into one file. When defining a system file, one can use original elementary streams (sources) as well as the output of a transcoding job. Each system file needs an output name and the parameters interleave and streamable influence how the system file is multiplexed.

To be able to save and restore projects, a simple text based format was defined, that contains the definition of all sources, jobs and system files. When loading an existing project all jobs and sources will be created and all lists will be filled up automatically.

2.2 Structure of a configuration file:

```
<Project>
        name=transcode.pro
<Source list>
        <Source>
                name=../../media/coastguard_9_6.mp4
                streamID=1
        </Source>
        <Source>
                name=../../media/coastguard+jazzpiano.mp4
                streamID=4
        </Source>
</Source list>
<Joblist>
        <Videojob>
                jobID=1
                name=ersterjob.mp4
                outFile=ersterjob.mp4
                width=1024
                . . .
        </Videojob>
```

```
        <Audiojob>
                jobID=2
                name=zweiterjob.mp3
                outFile=zweiterjob.mp3
                targetBit rate=96
                . . .
        </Audiojob>
</Joblist>
<System file list>
        <System file>
                name=sysfile1
                . . .
                <Source list>
                        <Source>
                        name=../../media/coastguard_9_6.mp4
                        streamID=1
                        <Joblist>
                                jobID=1
                                jobID=2
                        </Joblist>
                        </Source>
                </Source list>
        </System file>
</System file list>
</Project>
```

3 GUI

3.1 Main GUI

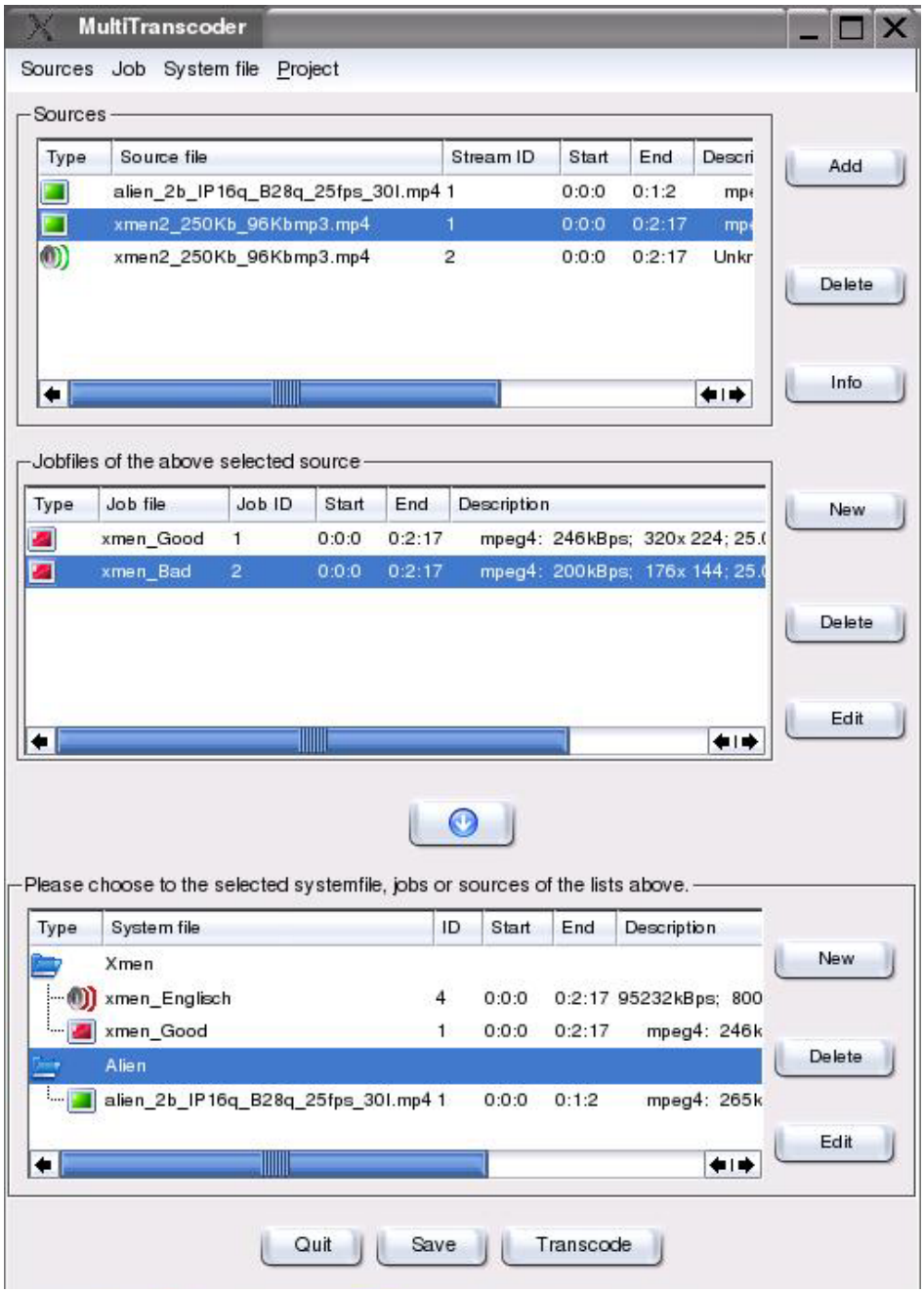

Figure 2: MultiTranscoder

Figure 2 shows the main GUI which is shown to the user, when he starts MuViTrans. The main GUI has three lists. The first list is the source list and allows one to add sources from the hard disk to the GUI. A source has following properties:

- **Type**

  An elementary stream can be of type audio stream or video stream. An icon will show the type of the stream.

   Is an audio stream

   Is a video stream

- **Source file**
  The name of the source.

- **Stream ID**
  Each source has an id and the combination of name and id is unique.

- **Start – End**
  For example: Your source is a video and has several elementary streams. Then each elementary stream has a start and end time in relation to the video. This is particularly useful for DVD where several videos can be contained in one file.

- **Description**
  This is a special string, which shows additional, stream type specific properties like codec, bit rate, sample rate, size, channels…

The second list contains the jobs already defined for the selected source and the third list contains all system files. What a job or a system file is will be explained later.

Note that icons for a job are red, whereas source icons are green.

 Is an audio job

 Is a video job

 Is a system file

If you have already added sources and jobs to your main window and you select one source file there will be only shown the job files of the selected source.

3.1.1 Menu bar

General all buttons, that are shown in the main GUI, can also be activated via the menu bar. Additionally, there are the following menu bar items:

- **Project -> Working Directory**
  Figure 3 shows the form "Working Directory" which allows one to set a working directory where all created jobs are saved. If this working directory is not set, the jobs will be saved, where the project is saved.

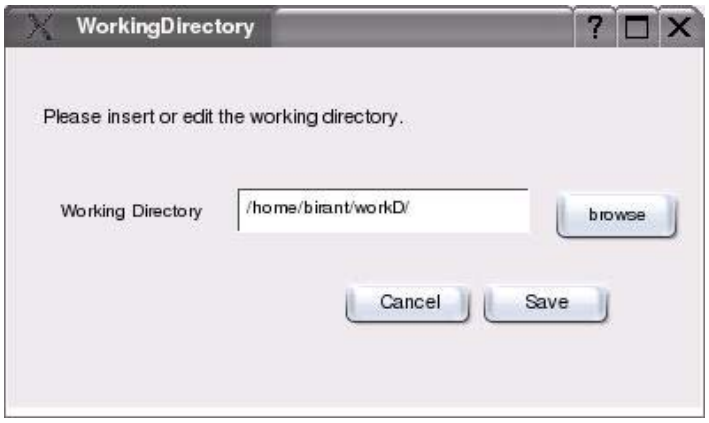

Figure 3: WorkingDirectory

- **Project -> Load Project**
  Loads a previously saved project.

- **Project -> Save Project**
  It saves the current project.

- **Project -> Rename Project**
  Opens a small GUI, which is shown as Figure 4. In this form one can edit the project name.

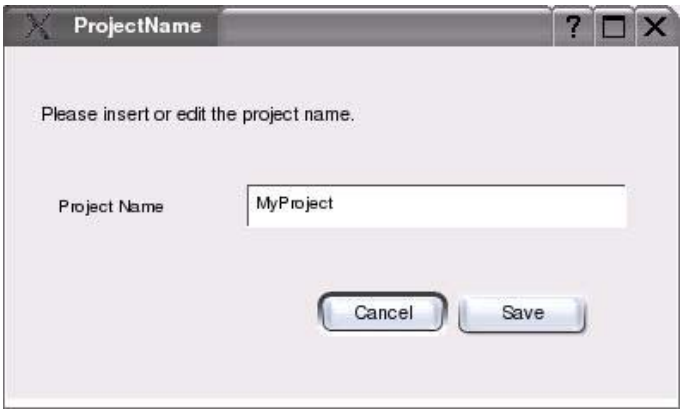

Figure 4: ProjectName

3.1.2 Buttons

Buttons related to sources:

Add: This button opens a browse window where one can add a source file to the project. A source can be for example an audio or a video stream.

Delete: Removes the currently selected source file from the source list. When pressing this button, a conformation window pops up. If one confirms, the source will not only be deleted from the source list but from the complete project. If one has already added this source file to a system file, the source will also be deleted from each system file. If this source is used by jobs, they will also be deleted. So please be careful with this button.

If no source is selected the delete button is not active.

Info: Select a source file and press this button and a window pops up, showing all properties of the selected source file. In this window one has not the possibility to create a new Job. Press the button “New”, right beside the list of job files to activate that.

If no source is selected the info button is not active.

Buttons related to job files:

New: If one has selected a source file of the source list above, one can make a new Job. A job allows one to define the properties of the elementary stream that should be created from the source. For example, one can make a video job with good and bad quality. If the button new is pressed, a GUI pops up. Depending on the selected source the GUI “DestinationAudio” or “DestinationVideo” appears. These GUIs allow one to create several jobs and once finished, pressing the ok button, returns the user to the main GUI “MultiTranscoder”.

If no source is selected, the new button is not active.

Delete: Pressing this button, the window shown in Figure 5 pops up asking for confirmation. If confirmed, the selected job will be deleted from the job file list and also

from the project. If this job file is already used in system files, the job file is deleted from all affected system files.

If no job is selected, the button is not active.

Figure 5: Delete Job

Edit: Pressing this button and depending on the selected job file, the GUI "DestinationAudio" or "DestinationVideo" appears. This GUI shows the properties of the selected job and one can change the properties of this job.

If no job is selected the button is not activate.

Buttons related to system files:

New: If one presses this button, the window shown in Figure 6 appears. There one can name the system file and set the properties "Interleave" and "allow streaming".

The name of a system file must not be empty and must be unique.

Figure 6: SystemFile

Delete: If one has selected a job file or a source file of a system file, one will deleted this file from the system file. But this file is not deleted from the list of sources or the list of job files.

Selecting a system file and pressing the delete button, will delete the system file with all sources and job files. But it has no effects on the source list and job file list above.

If no file of the system file list is selected the button is not active.

Edit: If one selects an existing system file, the same window like “System file New” appears. The difference is that the GUI is already filled and one can make changes.

One can’t select a source file or a job file and press the button edit. Use the corresponding “Edit” button instead. After editing a job/source, the system file list is updated.

Arrow Button:

First one should have added a system file in the system file list. This system file has to be selected. Then select a source file or a job file of the lists above and press the button. The selected source file or job file will now be added to the selected system file.

If no system file and source or job is selected, the button is not activated.

Quit: If you want to quit everything *without* saving, press this button.

Save: With this button all sources, jobs, system files and their properties are saved. A window pops up allowing one to choose directory location and file name.

Transcode: Click this button and your project will be transcoded.

3.2 Destination Video

This form shown in Figure 7 is called by the MultiTranscoder form based on a source file. The source file is an elementary stream. Within the Destination Video form one can create jobs for this stream. First, one configures the job, and then it is saved either with “OK” or with “Save Job”. Further jobs are inserted with the “Save Job” button, existing ones are updated with the “Update” button. Note that “Cancel” throws away all jobs created in this editing session and that “Ok” saves all jobs. By calling this form again with the same input file the saved jobs will be there again.

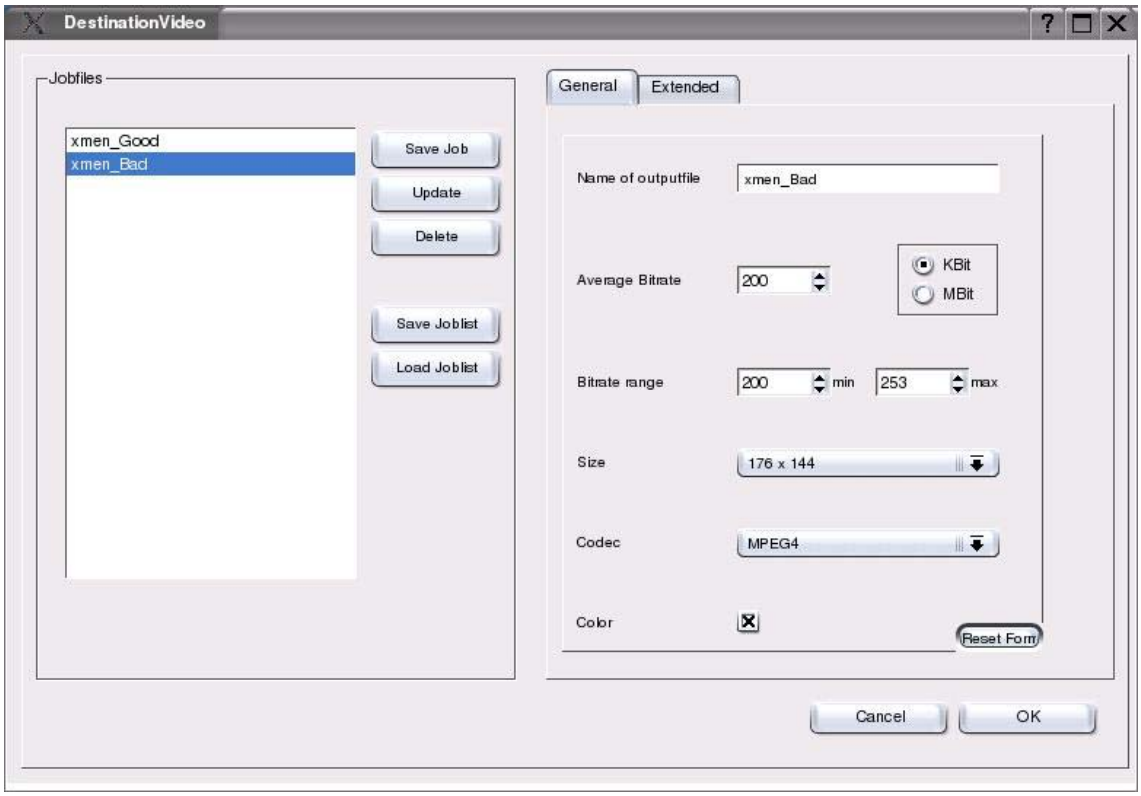

Figure 7: DestinationVideo General

3.2.1 Job file Buttons

Save Job: First, one configures all settings, then one clicks the button „Save Job“ and a new job with these settings and the name will be saved locally.

Update: First, one selects a job, then, after changing the to-be-updated settings, one clicks this button and the selected job in the job list will be updated. Note that an undo is not possible. An updated job can't be set back to its initial settings with the “Cancel” button.

Delete: Irrevocably deletes the selected job. If the list “job files” has no jobs, or no job is selected, nothing happens.

Save Job list: Allows one to save the current job list to the hard disk, in a source file independent manner. Thus, this job list can serve as a template for other sources that require the same set of jobs.

Load Job list: With load job list one can load a previously saved template. The jobs defined in the template are added to the job list of the current source. The names of the jobs are updated, so that they contain the name of the current source.

Reset Form: This button sets the form back to the first settings, so that the properties of the source are shown.

OK: By clicking on the “OK“ button two things happen.

1. All jobs listed in the job list on the left hand side are saved and also shown in the MultiTranscode form.

2. All settings one has finally made in the job are saved as a new job, if there is no job selected in the current job list. Otherwise the selected job will be updated.

Cancel: Every new job created during this editing session will be deleted. Updates to already existing jobs can not be cancelled and remain in the list.

3.2.2 General Job file properties

Name of output file: Initially the name of the input file is set as the name. One can also put file paths in there but do not begin with a „/“. When one presses the return key when the cursor is located in the text field of the output file, a new job is created with the name which is currently written in this text field. The settings of the other fields will be copied from the old job. All other settings on the register card “General” are read out from the input stream. After adjusting all settings, one can click the button “Save Job” and a new job with these settings will be made. If one tries to create a new job with the same name, the name will be modified to <name>-1, at the second time <name>-2 and

so on. This guaranties unique job names. This is important because the name of a job will be used as the output file.

Average Bit rate: Initially the average bit rate is read out from the input stream. If the bits per second are less then the maxValue of the bit rate input field*1024, then the checkbox “KBit” is set. Otherwise the “MBit” checkbox is set. For example, one saves a job with a bit rate of 1 MBit. When the job is called again, the average bit rate will show 1024 “KBit”.

Bit rate range: These fields are also bound on the “KBit” and “MBit” checkboxes. The fields are not read out from the input stream. Initially, both min and max values will be set to the same amount as the average bit rate value. If these values are modified, the min value has to be less or equal to the average bit rate and the max value has to be higher or equal to the average bit rate. MuViTrans will automatically adjust these values for the user.

Size: Initially the resolution of the input stream is read out. If one wants to change it, one can either use one of the values defined in the push down menu or choose “USER DEFINE”. By selecting this item a new form pops up, which is called “UserDefine”.

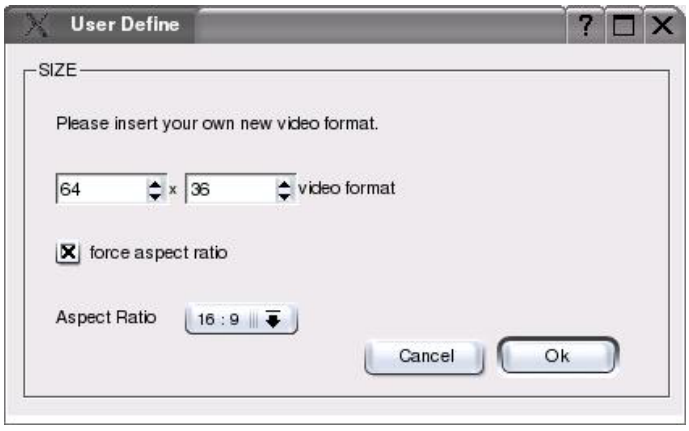

Figure 8: User Define

User Define: This form, shown in Figure 8, allows a user to select a custom width and height of a video. Initially, the values of the input stream will be set. If this form is

opened with a selected job in the job list, the settings of the job are shown. If one wants to keep the aspect ratio, activate the “force aspect ratio” checkbox. This will enable the aspect ratio field.

Aspect Ratio: If the input format allows to specify an aspect ratio, this value is extracted from the input stream, otherwise it is calculated from the dimension of the video. This value, for example 1.777, is equal to the aspect ratio 16:9. There are only some different aspect ratios the form supports. These are: “16 : 9”, “4 : 3”, “11 : 9” and “1 : 1”. If an input stream has an aspect ratio, which the form does not support, the item “Unknown” will be selected.

Codec: Here one can select the codecs used for producing the output stream. If none of these codecs matches, “UNKNOWN CODEC” will be selected initially. There are many codecs you can select for a video stream.

Color: If an input stream has a color, one can create a job, which is grey scaled. If the input stream is not colored, the checkbox is set disabled.

3.2.3 Extended job file properties

These fields are initially not set.

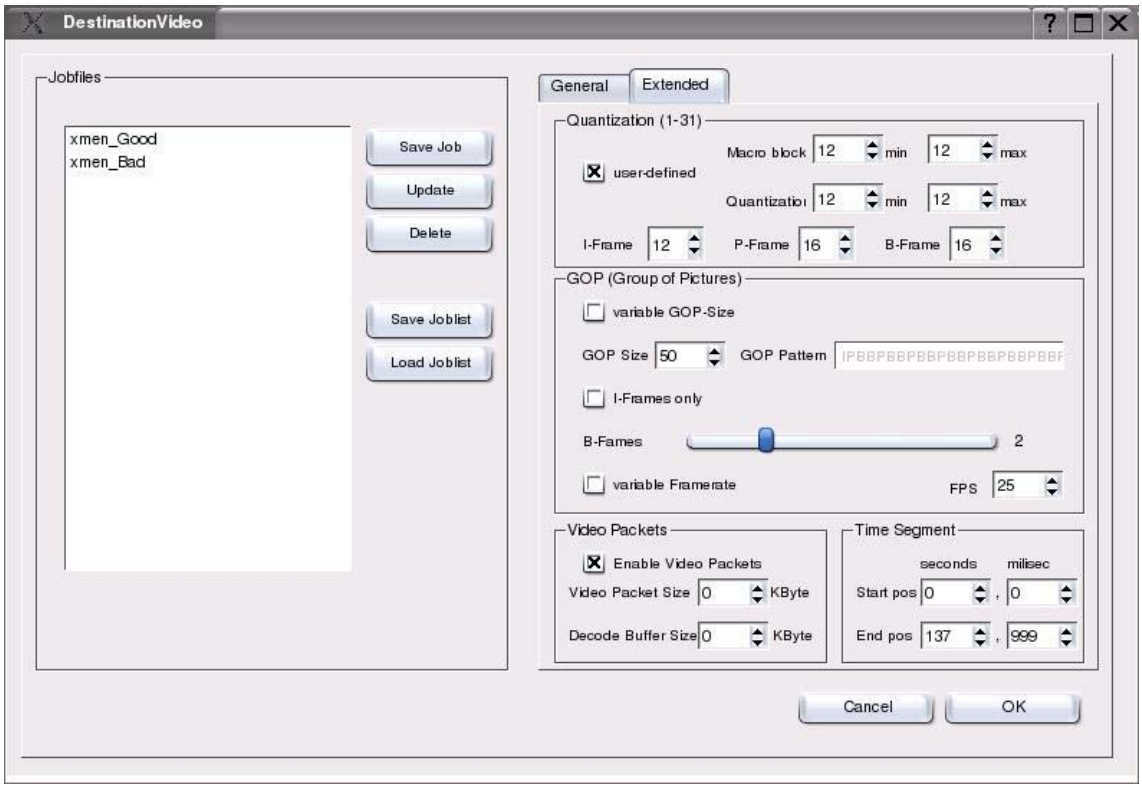

Figure 9: DestinationVideo Extended

Quantization: Quantization is a procedure where the codec accuracy will be adapted to the perceptive faculty of the human people. Because of the laziness of the human eyes, the viewer doesn't recognize the lower accuracy of the data at special details. The macro block is the smallest decode able item which is possible. You can set here values for the minimum and the maximum quantization of the macro block. Setting the quantization range is also supported for frames for each frame type (I, P, B) and different quantization level are possible. If the check box “user defined“ is not selected, one is not able to set anything.

GOP: Here some checkboxes exclude others. If “variable GOP-size” is enabled, the GOP size field and the checkbox “I-frames only” are set disabled. The field GOP Pattern shows how the frame pattern of the output stream. The more the B-frame slider is raised, the more B-frames the video stream will contain. If “I-frames only” is clicked, every field in the GOP except variable frame rate and FPS is set disabled. If variable

frame rate is clicked, one cannot rise or increment the FPS value. This value shows how many frames per second the video will have.

- **I-Frames (I = intra)**
  These are independent frames. They are regularly contained in a MPG video to enable the start of a decoder e.g. at a change of a scene.
- **P-Frames (P = predicted)**
  These are individual frames which contain the difference to the previous frame.
- **B-Frames (B = bidirectional)**
  These frames where calculated by the previous and the following frame.

Video Packets: If one wants to enable video packets then click on the checkbox next to "Enable Video Packets". This enables the field "Video Packet Size", where one can set a value in bytes how large one video packet should be. Also the decoder buffer size can be set here. This value is given in KBytes and tells the encoder, how large the decoder buffer at the client side will be.

Time Segment: This options allow to truncate videos to a given time range. The start and end position can be selected in seconds and milliseconds. Initially the start position is set to 0 and the end position is set to the duration of the video.

3.3 Destination Audio

This form is called by the MultiTranscoder form and enables to build, update or delete audio jobs. The source file is an elementary stream. Within the Destination Audio form one can create jobs for this stream. This form works similar to the Destination Video form. See Section 3.2.1 for an explanation of all buttons.

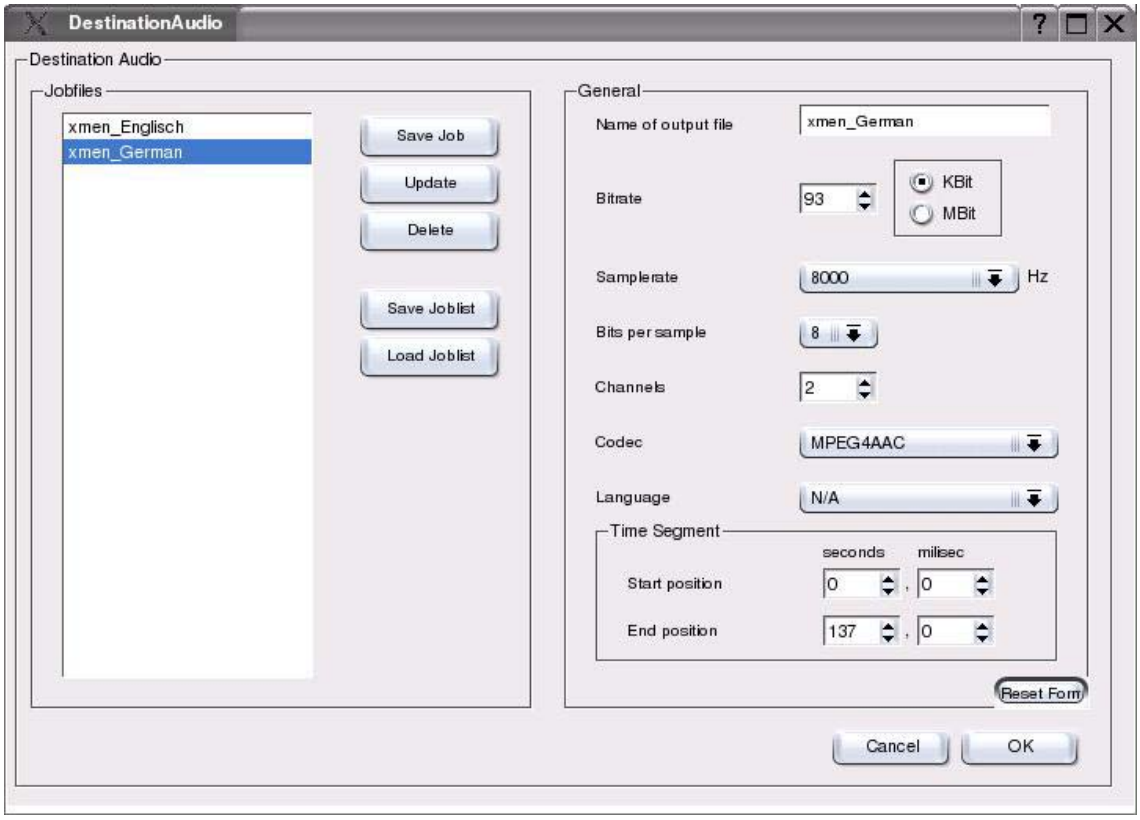

Figure 10: DestinationAudio

3.3.1 Job file properties

Name of output file: Initially the name of the input file is set as the name. One can also put file paths in there but do not begin with a „/“. When one presses the return key when the cursor is located in the text field of the output file, a new job is created with the name which is currently written in this text field. The settings of the other fields will be copied from the old job. All other settings on the register card “General” are read out from the input stream. After adjusting all settings, one can click the button “Save Job” and a new job with these settings will be made. If one tries to create a new job with the same name, the name will be modified to <name>-1, at the second time <name>-2 and so on. This guaranties unique job names. This is important because the name of a job will be used as the output file.

Bit rate: Initially the average bit rate is read out from the input stream. If the bits per second are less then the maxValue of the bit rate input field*1024, then the checkbox “KBit” is set. Otherwise the “MBit” checkbox is set. . For example, one saves a job with a bit rate of 1 MBit. When the job is called again, the average bit rate will show 1024 “KBit”. It is similar to the bit rate field in video jobs but of course with lower values (e.g. 128 KBit). Currently KBit will be enough, but in the future MBit might be needed.

Sample rate: Here you can select the sample rate (e.g. 44.1 KHz). But not all values are given in Hz. The value contains how often the sound will be scanned.

Bits per Sample: This shows the range of values the amplitude can have. So, here you can select how many bits per one sample you want to have. A lower value gives lower quality.

Channels: Here one can select the number of channels the output stream should have. For example, one for mono, two for stereo.

Codec: Here one can select codecs. If none of these codecs matches, “UNKNOWN CODEC” will be selected initially. There are many codecs you can select for an audio stream.

Language: This field shows which language the input stream has. If the input stream has no information on the language, one can set here the language in case it is known.

Time Segment: This options allow to truncate audios to a given time range. The start and end position can be selected in seconds and milliseconds. Initially the start position is set to 0 and the end position is set to the duration of the audio.

3.4 Progress

When clicking the transcoding button from the MultiTranscoder form, the transcoding process will start. Figure 11 shows the Progress form, which shows how much transcoding process is already completed.

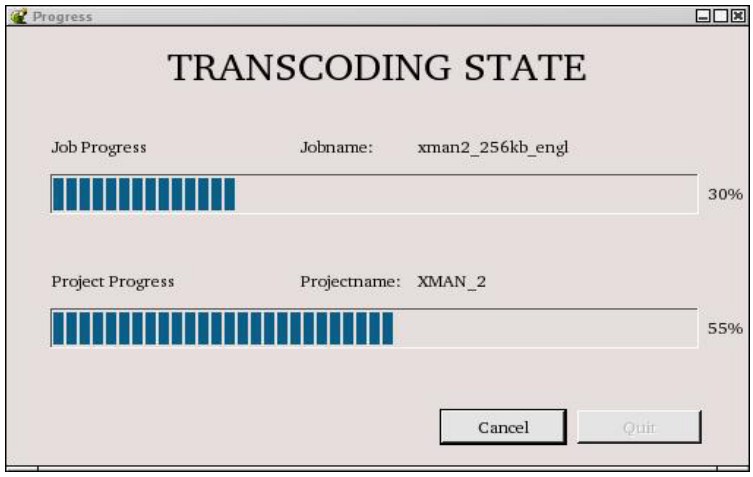

Figure 11: Progress

Job Progress: Here the current state of the Job which is transcoded at this time is shown. The name of the Job is written next to the field “Jobname”. Also the percentage of the job will be shown on the right hand side of the progress bar.

Project Progress: Here the current state of the whole project will be shown. The name of the project is written next to the field “Projectname”. Also the percentage of the project state will be shown on the right hand side of the progress bar.

Cancel: The “Cancel” button stops the transcoding process but the jobs which have already been transcoded will not be deleted.

Quit: When the transcoding process has finished the “Quit” button is enabled.

4 Possible Future extensions

All extensions one can make with little code change are commented in the DestinationVideoImpl.cpp- and DestinationAudioImpl.cpp file: //Extension <extension name>

This was made for easier future extensions and changes.

4.1 Destination Video

Bit rate range

In the future it will be possible to save in a stream the possible minimum and maximum bit rate. One searches for the text //Extension Bit rate min max.

Write:

```
i_bit rate_min=esinfo->getMinBandwidth();

i_bit rate_max=esinfo->getMaxBandwidth();
```

Instead of:

```
i_bit rate_min=i_bitsPerSecond;

i_bit rate_max=i_bitsPerSecond;
```

Codec

The codecs for the videos are stored locally in an array. If one wants to append some codecs in the push down menu of the Destination Video GUI, one has to add the codecs at the end of the array before „UNKNOWN CODEC“.

This array can be found under the definitions in the DestinationVideoImpl.cpp file at //Extension Codec.

Color

Maybe in the future it will also be possible to color an input stream which is grey scaled. To extend that function, search for //Extension Color.

Comment the line “color->setChecked(false);” and one can set a grey scaled video to a colored video.

4.2 User Define

Aspect Ratio

To add new aspect ratios to the User Define GUI, add the new aspect ratios at //Extension Aspect Ratio 1.

There one can see two arrays, one for the width and one for the height. Simply extend these arrays with the new values.

Furthermore, add the float value of the aspect ratio to an “if” statement, appending a new “else if“ branch.

These changes have to made at //Extension Aspect Ratio 2.

4.3 Destination Audio

Codec

The codecs for the videos are stored locally in an array. If one wants to append some codecs in the push down menu of the Destination Audio GUI, one has to add the codecs at the end of the array before „UNKNOWN CODEC“.

This array can found right below the definitions in the DestinationAudioImpl.cpp file at //Extension Codec.

Language

In the future it will be possible to define for a given input stream a language. For that feature we made an array containing strings of languages. To extend this array, search for //Extension Language in DestinationAudioImpl.cpp.

But don’t forget to update the value of the definition MAX_LANGUAGE!