# Mixxx: Towards Novel DJ Interfaces

Tue Haste Andersen
Department of Computer Science
University of Copenhagen
DK-2100 Copenhagen
Denmark
haste@diku.dk

## ABSTRACT

The Disc Jockey (DJ) software system Mixxx is presented. Mixxx makes it possible to conduct studies of new interaction techniques in connection with the DJ situation, by its open design and easy integration of new software modules and MIDI connection to external controllers. To gain a better understanding of working practices, and to aid the design process of new interfaces, interviews with two contemporary musicians and DJ's are presented. In contact with these musicians development of several novel prototypes for DJ interaction have been made. Finally implementation details of Mixxx are described.

## Keywords

DJ, software, interaction, visualization, controllers, augmented reality.

## 1. INTRODUCTION

The role of a Disc Jockey (DJ) is to select and mix music during playback. The audience can be listening on the radio, on a dance floor, or in an ambient club setting. By playing pieces of music and mixing them together, the DJ seeks to please or provoke the listeners, depending on the setting and audience.

Even today many professional DJ's are still using turntables as playback medium instead of digital based interfaces, such as computers and CD players. This is especially true for the artistic or music-producing DJ's. There are several reasons for this, one of them is fashion and habitude, another is that much contemporary music is still released first (and in many cases only) on vinyl. However, another reason, more important to the work presented here, is the superior interaction possibilities of a DJ turntable when compared to other playback devices. The degree of control and feedback is something that is hard to achieve with todays commercial digital solutions.

In general, DJ systems can be divided into four classes: the analogue systems, digital WIMP (Windows, Icons, Mouse, and Point-and-click) based interfaces, digital interfaces emulating the interaction of analogue interfaces, and finally a new class of digital systems which goes beyond the WIMP interfaces without being limited by the analogue design metaphors. The novel interfaces should not merely be a reproduction of the analogue interfaces, and not limiting the user to only one way of solving a given problem. In this way, the new designs can be used in ways that they have not intentionally been designed for, just as the turntable was not originally designed with scratching in mind.

This paper presents a software system supporting research for this last class of DJ systems. Mixxx [2] is a digital DJ system, which allows for experimentation with interfaces. Several new controllers and visual interfaces have been prototyped and demonstrated using Mixxx, which also will be presented here. To evaluate these new interfaces, Mixxx has to be used in real performance and production situation, and can therefore be regarded as not only a research tool, but also a set of performance tools. The focus of this study will be on the artistic use of DJ interfaces, although it is likely to be of direct importance in areas such as radio and TV production, and other places where sound playback requires a high degree of timing.

The paper is organized as follows: Section 2 describe briefly the setup used by typical DJ's, along with related work. Section 3 presents two interviews with contemporary electronic musicians. A short description of their working practices is given, along with observations of direct importance to the design of new interfaces. Section 4 gives an overview of Mixxx, and the related interface technologies developed, which is followed by a description of the implementation in section 5. Finally a conclusion is given in section 6.

## 2. BACKGROUND

In this section I will briefly describe the interfaces used by DJ's today, followed by a short overview of related research and commercial solutions available.

### 2.1 Live DJ interaction with audio

A traditional DJ setup includes two turntables and a mixer. The control parametes available include continuous variable playback speed, sound level, filtering, and a cross fader for mixing between the two sound sources. Special CD players are also present in most setups. DJ CD players are most notably different from normal CD players in that they have speed control and abilities to store a specific position in a song. Even though the CD players have been available for a long time, and CD's are more convenient to transport when compared to vinyl records, the vinyl still seems to be the preferred playback medium. Digital DJ software solutions are also available, where it is possible to select and mix sounds stored in MP3 format. These solutions often depend on the use of a mouse, and in some cases external knobs connected through MIDI. However, these solutions seem to impose the same problems on the DJ's as the DJ CD players do. When comparing to the turntable, the DJ is loosing visual feedback by the loss of reflection from the grooves in the vinyl, the ability to scratch, and the ability to skip quickly and precisely through a song, by either spinning the record by hand, or by moving the stylus.

The mixer allows for mixing of different audio sources into one, independent control of filters and sound effects, and a special output channel for headphones, which the DJ can use for listening and changing audio before it is mixed with the audio output. This is especially important in disciplines such as *beat mixing*, in which DJ's match two consecutive songs in beat before mixing them together, forming a smooth transition in tempo and pitch from one song to another.

## 2.2 Related work

Research in interfaces of musical instruments is a growing area, with many new ideas on how to interact with novel and old instruments. The work presented here takes a somewhat different approach from other instrument studies, in the sense that it focuses on interaction with pre-recorded audio, instead of generating audio by itself.

Tools for navigating in audio have been explored outside a musical context, mostly in relation to searching in recorded speech. SpeechSkimmer [4] is an example of such a system, where knowledge about pronunciation is used to provide visual cues of where new topics in a conversation are introduced.

A number of commercial digital DJ solutions are available. Most of them provide a GUI for selecting, mixing and controlling playback speed and position, and some have external interfaces with sliders, knobs and rotary controllers, similar to the DJ CD players. One product stands out, namely Final Scratch [20] from Stanton, which produces software and a dedicated DSP box, for connecting the traditional turntables to the computer. Instead of using vinyl as playback medium, special records are used for which the DSP box picks up the position, and sends it to the computer. On the computer, the DJ selects tracks, and in this way is able to use the old turntables, while having the comfort of not carrying hundreds of vinyl discs. The solution is clever, and certainly a step forward when compared to other digital solutions. However, it does not provide any further novel additions to the user interface, apart from a standard scrolling waveform display.

Research on DJ's working practices conducted in an academic setting is rare. Hansen [9] studied specific turntable techniques used by the turntable instrumentalists or the so-called turntablists, musicians who use the turntable as an instrument. These techniques are often employed and invented in the genres of hip-hop, and to a lesser extent techno.

Examples of new music performance tools include Audiopad [18] and Block Jam [17] that makes use of tangible interfaces in the playback and control of music on a sample-based sequencer. These interfaces can be used in a DJ situation to control the arrangement of different tracks and properties of a musical piece as stored in a sequencer, going from a linear playback of a musical piece to a non-linear playback where the DJ or musician is in control. These interfaces solve interaction problems of WIMP interfaces and bring new elements of control in the hands of the DJ. However, music is most often distributed as one linear piece, and thus requires manual segmentation to enable playback and control of individual elements of the musical piece.

# 3. INTERVIEWS

To gain a better understanding of current practices and uses of equipment used by professional electronic musicians and DJ's, two interviews were conducted. The interviews were made as contextual interviews [6], they took place at the DJ's own place and lasted between one and a half to two hours. The interviews were video taped for later analysis and reference. Although only two participants were used, this study made it clear that a number of methods are developed and used individually by each musician, and thus it might be difficult to paint a general picture, even from a large set of interviews.

In the following, the two interviews are described and key issues discovered during the interviews are highlighted. The description is based on observations made during the actual interview, or during analysis of the recorded videos. Both direct verbal information from the interviewed persons, and indirect analysis of the way people act and interact with the instruments are used as basis for the following description.

The first interview was done with an electronic musician who uses only turntables and hardware synthesizers and sequencers in the production and performance of music. This means that no computer with a WIMP interface is involved, only computers with custom interfaces, typically based on a number of buttons, knobs, sliders, dials and a minimum of LCD displays.

A typical composition session by this musician is carried out by playing with the music. Rhythms and bass tracks are programmed on synthesizers, changed and reworked until something which the musician is satisfied with is reached. Then the rhythm tracks are recorded on a harddisk recorder, and experimentation with the music continues using turntables and keyboards. If something goes wrong, the work is usually completely redone, instead of trying to edit some of the recorded tracks. Preparation for a DJ session includes selecting a set of interesting records, and maybe practice with other musicians if they are involved. Often when several DJ's are playing live together, they set up their equipment on a line, with front to the audience. However, this musician preferred to be able to look at his companion, at least while practicing.

A number of important observations to keep in mind when designing new DJ interfaces were found during the interview:

1. The tempo is constantly adjusted on the turntable to keep the record in synchronization with other sound sources, be it live musicians or a drum machine. By constantly making small adjustments to the playback speed slider, the record is kept in perfect sync with other sources. A slider is superior to +/- pushbuttons with LCD displays, primarily because it can be operated without looking at it, and large changes to the playback speed can be done quickly when searching for the right speed in a new track. The speed slider is also used to adjust phase of beat, instead of stopping the record by holding it with a finger for a very small period of time. This technique is avoided because it results in large transient changes to the pitch when longer cords are played on the record.

2. Visual feedback from the light reflections in the grooves is in general not used. They can be hard to see in a club with limited light sources.

3. When using vinyl records, it is not necessary to know the music on the records in advance. The content can easily be reviewed live, by moving the stylus through the record manually while listening on the headphones.

4. A record is most easily started by scratching over the beat and releasing when in phase with the other sound sources.

5. Composition of songs is primarily done manually. When using drum and bass synthesizers, each part of a song is programmed in a bank. Instead of pre-programming the order of each part, switching between parts is done manually. The reason for this, according to the DJ, is because it is easier, and gives more freedom for improvisations. It was observed that the use of modes [19] on some synthesizers can easily lead to confusion.

6. The filters on the mixer are often used to replace a given frequency band from one track with the same frequency band from another track. Also the filters can be used to make solos, by constantly adjusting them. This is something, according to the DJ, which can be more easily done using WIMP based sequencers when making recordings.

The second musician interviewed works by using WIMP based interfaces extensively, and also plays some acoustic instruments and uses turntables. Different programs were used in this interview including Muzys [16] and ProTools [8]. This musician primarily works by arranging samples of recorded music in a sequencer, both for production and live usage. A number of interesting observations in his usage of the interfaces were also made:

1. Synchronization of a sample with other tracks is done automatically by the program from information about where the beats are in a sample. The beat points is found by the program and adjusted manually. The adjustments are done solely from the visualization of the sampled waveform, not by listening.

2. Time stretching is problematic with respect to sound quality, and therefore pitching is often used, where the pitch is changed along with the playback length.

3. In live sessions, different parts are loaded into the program, and activated by MIDI controls or by clicking with the mouse. New samples cannot be brought in live, since there is no time for adjusting the beat of a sample.

## 3.1 Discussion

The two interviews may not be representative for how electronic musicians and DJ's work in general, but give insight into the process of composition and improvisation using the analogue and digital equipment described previously. The first musician is extremely dependent on how the interfaces are constructed, and uses them to form habits, while the second musician to a much higher extent depends on the features available in WIMP interfaces, at the tradeoff of loosing some control in a live situation. Both depend on the ability to easily search or navigate in the audio, either to synchronize it with other sources, or to get an overview of its contents. It seems that to be able to do the synchronization in a live situation, high precission is required, e.g. as when the record is started by scratching over the beat. The musician having waveform displays available, in some situations relies solely on them, while the first musician only uses haptic and auditory feedback from the instruments.

While WIMP based interfaces are general, they also have many problems, especially in situations where huge requirements is put on the humans shoulder in form of reaction time and precission [19]. When working towards ubiquitous computing, the tangability and feel of human computer interfaces becomes important [10]. Especially in a DJ and other musical situations it seems important, because a high degree of accuracy is required within a narrow time frame. To build new controllers and interfaces, it is therefore of primary importance to understand and possibly model the feel and tangability of the existing analogue interfaces.

For both of the interviewed musicians it also seems of importance to treat computers as individual instruments or sound producing tools. By using each computer as a tool to produce a certain kind of sound, the musicians are able to configure and arrange the tools in their own way, enabling arbitrary physical arrangement and interconnection. The tool based work practice has nothing to do with feel, but may be of importance to creativity imposed by the musicians on the music. By connecting and arranging the tools in different ways, different types of music can be produced. Thus, the musician is not limited by the intended use of the instrument for which it was designed. In this way an instrument designed for one way of producing music can become a broader sound producing tool. The analogue Roland TB303 (Transistor Bass) synthesizer is a good example of an instrument which was designed to be used as a stand-in for a real bass player, but became one of the classic instruments in techno music. The modular analogue synthesizers are an example of a type of instrument which in contrast was designed to be open ended, by letting the musician rewire the analogue circuit producing the sounds.

# 4. DESIGN

The main design goal of Mixxx is to make it possible to conduct interaction studies of novel interfaces in relation to the DJ situation [1]. However, to be able to evaluate novel interfaces and interaction techniques Mixxx must be used in realistic settings of performance and production of music. For this reason Mixxx can be regarded as both a set of performance tools and as a means of studying DJ performances. The studies could for instance be based on quantitative evaluation of controllers, controller mapping, visualization techniques or qualitative studies on the DJ situation in general. By performing such studies it is hoped that new and improved interfaces and ways of interacting with media can be reached.

As such Mixxx is designed to enable for open ended tool based composition as discussed in previous section. The code is modular and enables for graphical and physical interfaces at many levels. As a performance tool Mixxx is currently emulating a traditional DJ setup with mixer, and two playback devices enabling mapping to MIDI controllers and parameter visualizations.

An overview of Mixxx is shown in figure 1. Different interfaces have been proposed and prototyped based on the two interviewed musicians, and a GUI is provided for most of the available controls. The implemented GUI has features resembling other commercial software available, and can be used in comparative studies with other types of interaction. Configuration and track selection is provided solely through the GUI. The prototypes include a controller with the same interface as a DJ mixer, and a rotary controller based on the turntable metaphor. Furthermore a visualization prototype using a Fisheye [7] in waveform displays is shown, along with an augmented turntable currently under development. The augmented turntable brings many of the ideas presented in this article together in one device.

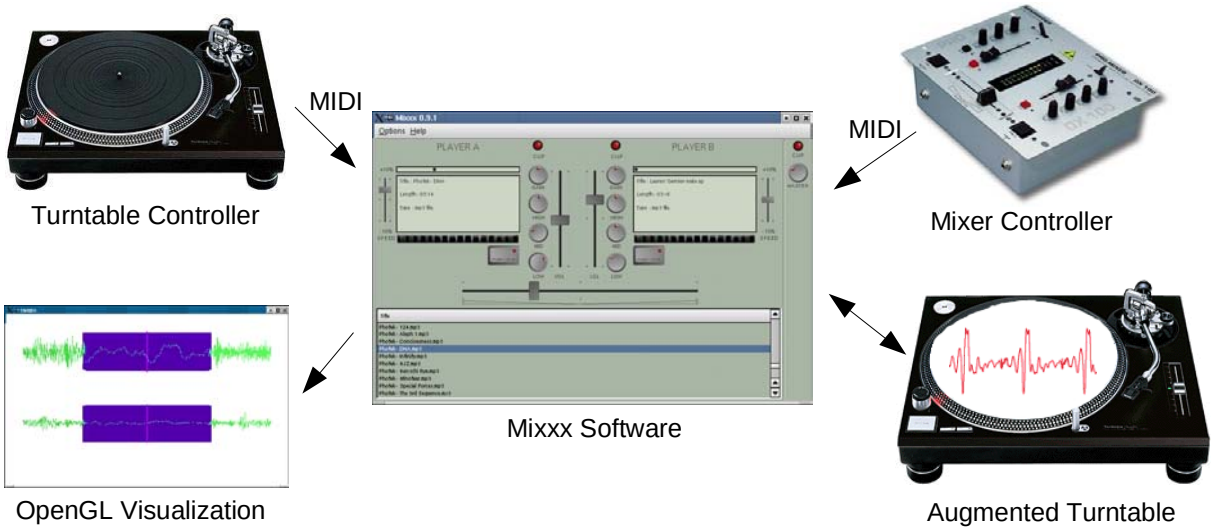


**Figure 1: Overview of Mixxx with the various interaction modules developed so far.**

## 4.1 Mixer interface

As a first step in developing new controllers, a controller box was prototyped with an interface similar to a standard DJ scratch mixer. The prototype is built by replacing the electronics of such a mixer with hardware that converts the controller's analogue output to MIDI messages.

This approach of using external knobs and sliders is identical to commercial MIDI hardware, with the only exception that this interface has the exact look and feel of a scratch mixer.

## 4.2 Turntable metaphor

For searching and synchronization of audio, the analogue turntable seems superior to most digital equipment, although no good visualization techniques are available. Experiments with different rotary controllers have been done, in particular with a modified turntable. The turntable controller works by sensing and outputting the velocity of the deck plate[1] over MIDI. The accuracy of the turntable, when measured as the minimum movement at the edge of the deck plate causing a MIDI velocity message, is less than one millimeter. In this way the turntable can be used to navigate in digital audio, however with another feel that the traditional turntable setup, where a moter is giving active force feedback and the plate is not necessarily following the vinyl when dragging the record back and forth.

As mentioned in section 2 the use of the turntable metaphor is not new. In Final Scratch the metaphor is used, but in a different way. In FinalScratch the turntable stays closer to the analogue DJ turntable, and thus it might be easier for a professional DJ or musician to transfer his or her skills to the FinalScratch setup. However, the purpose of Mixxx and the research conducted with Mixxx, is not to replace the turntable as instrument, but to find new and better ways of navigating and giving new inspiration to the artists. In this sense, staying completely true to the turntable metaphor is likely to limit more than open new design possibilities.

## 4.3 The AudioFish

The AudioFish, earlier presented in [3], is the Fisheye visualization technique [7] applied to waveforms. Different signal values are displayed over time, for each piece of music played back. The idea is to use the fisheye to zoom into a region near the playback position, while still being able to see far into the future and past of the waveform. In the lower left part of figure 1 a screenshot of the AudioFish is shown, where only the waveform is used. This could possibly be overlayed with other time dependent parameters such as information about beat [15], pitch and timbre [11]. On the figure, two tracks are played back, showed one over another. By arranging the tracks as parallel displays they can be compared visually, and thus can be used for synchronizing two or more tracks in time.

The parameter visualization serves a number of purposes:

1. Provides cues of the structure of a song without the need to listen to the song, e.g. by showing energy and tonality as function of time;

2. Allows for matching of parameters from different audio sources using comparative displays;

3. Supports collaborative work through overlay of real-world objects with visualizations of song parameters (Augmented Reality). In this way several musicians can see the parameters more easily than with a traditional computer screen.

The size and zoom of the Fisheye can be changed dynamically, but initial evaluation suggests that this should be done automatically, e.g. as function of the playback speed.

## 4.4 The Augmented Turntable

The Augmented Turntable currently in development, is a combination of the AudioFish with the turntable controller. Using a computer display projected down on the deck plate, the deck plate can be used as a visualization area. By modifying the AudioFish from being plotted in a Cartesian coordinate system, to a circular plot of the waveform in a polar coordinate system [22], the viewing area is used optimally, and the notion of a vinyl groove is reused.

The use of the area on the turntable as visualization supports collaboration to a much higher degree than the use of various external controllers coupled with a WIMP interface. Removing the computer screen means that the musician can

[1]The deck plate is the actual spinning turntable.

now work like a typical DJ artist, and easily have eye contact with other musicians. The use of the circular plot even means that there is no correct angle for the displayed image to be viewed at.

A future possibility is to mount sensors on the deck plate to facilitate a simple point-and-click system, for selecting different parts of a track, or make track selection possible without involving another interface.

The projection is done using a projector above the turntable. This is somewhat primitive but may in the future be replaced by light emitting polymers. Another solution would be to use head mounted see-through displays and the AR ToolKit [12]. This would however require that every person using the turntable would have to wear a display. In all, the projected image currently seems to be a simple and inexpensive solution.

# 5. IMPLEMENTATION

Mixxx is developed in C++ using the QT toolkit [21] and PortAudio [5]. By utilizing these libraries, Mixxx is able to compile and run on MacOS X, Windows, Linux and other Unix derivatives. The program is released under the General Public License, and is freely available for download.

Figure 2 gives an overview of the different modules in Mixxx that are executed in four different thread classes. The internal processing in Mixxx is separated in different objects derived from `EngineObject`, each representing a processing module. Each module takes a buffer as input, and provides a buffer as output. The buffers can be samples, SDIF frames, or other data structures. The modules are similar to other module based processing systems like the Linux audio plugin format LADSPA [13]. Each module is mapped to one or more controller entries, each represented by an object derived from a `ControlObject`. Each entry can be assigned a MIDI value, and thus every GUI control is easily mapped to a MIDI controller. MIDI was chosen as the communication channel to external controllers because of its wide usage. If, however, better time resolution is required, the speed of the serial channel can be changed, at the loss of compliance with MIDI equipment. This may be of interest in research on the required time resolution.

Playback is done in the `Player` object, which request a buffer of samples. To process the samples the `Player` calls a list of `EngineObject`s. The `EngineObject`s signals to the file I/O thread if more samples are needed from the file. Reading and decoding of MP3 files is handled in `SoundSource` objects.

The AudioFish is implemented as a separate module in Mixxx, `MixxxVisual`, running in the main GUI thread. The AudioFish is written using OpenGL to perform the zooming operation directly on the graphics card.

## 5.1 Latency

The latency of the total system is governed by several factors:

- Sound card latency
- Operating system and driver architecture used
- MIDI controller
- Processing speed of one block of audio

The sound card latency can be adjusted dynamically in a preference panel. The system currently uses PortAudio [5] as interface to the platform dependent audio API. On Linux the latency of the total system can thus be below 5 ms, with the right kernel configuration. Lower latencies might be achievable, but has not been researched further. Low latency is also achievable on MacOS X and on Windows.

## 5.2 Controllers

The external MIDI controllers are built using modified hardware, like a turntable or a mixer. For sensing and sending out MIDI events a PIC chip from Microchip [14] was used. The model used is a PIC16F874 with a number of analogue inputs, and digital inputs and outputs. The programs running on the chips are written in C.

In the case of the mixer described in section 4.1, the sliders, potentiometers and buttons are connected to proper inputs on the PIC. The turntable was built using an old belt driven model. The moter was replaced with a high resolution rotary encoder, giving 200 impulses per revolution. Because of the gearing between the deck plate and the rotary, this gives a sub millimeter resolution when moving the deck plate on the outer edge. The PIC chip in the turntable was programmed to output the velocity of the deck plate.

The source code and diagrams for the PIC based controllers will be available on the Mixxx website [2].

# 6. CONCLUSIONS

DJ equipment and work practices have been studied by interviewing two musicians. Although two interviews clearly are not representative for a whole culture, they give insight into the process of DJ'ing and composing music using modern instruments. The use of instruments as tools with open-ended designs, compared to instruments designed for only one purpose is considered important.

Based on these studies the digital DJ system Mixxx has been presented, and demonstrated as an open and extensible base for future research in interaction studies. A number of interface prototypes for Mixxx have been built. Some of these prototypes are based around the turntable metaphor, focusing on tools which can be used by one or more players in a collaborative setting without limiting the interaction to that of analogue equipment.

# 7. ACKNOWLEDGMENTS

Mixxx was developed in collaboration with Ken Haste Andersen. The AudioFish was developed together with Kenny Erleben. Thanks to August Engkilde and DJ Kruzh'Em for the interviews, Kristoffer Jensen and Declan Murphy from the Music Informatics Group at DIKU for inspiring discussions and comments. Finally also a thank to the people at the Doctoral Colloquium, NordiCHI 2002 in Aarhus, for discussions and comments.

# 8. REFERENCES

[1] T. H. Andersen. Live DJ interaction with sound. Doctoral Colloquium at NordiCHI, Aarhus, Denmark, October 2002.

[2] T. H. Andersen and K. H. Andersen. Mixxx. http://mixxx.sourceforge.net/, January 2003.

[3] T. H. Andersen and K. Erleben. Sound interaction by use of comparative visual displays. In *Proceedings of the Second Danish HCI Symposium*. HCØ Tryk, November 2002.

[4] B. Arons. SpeechSkimmer: A system for interactively skimming recorded speech. *ACM Transactions on Computer-Human Interaction*, 4(1):3–38, March 1997.

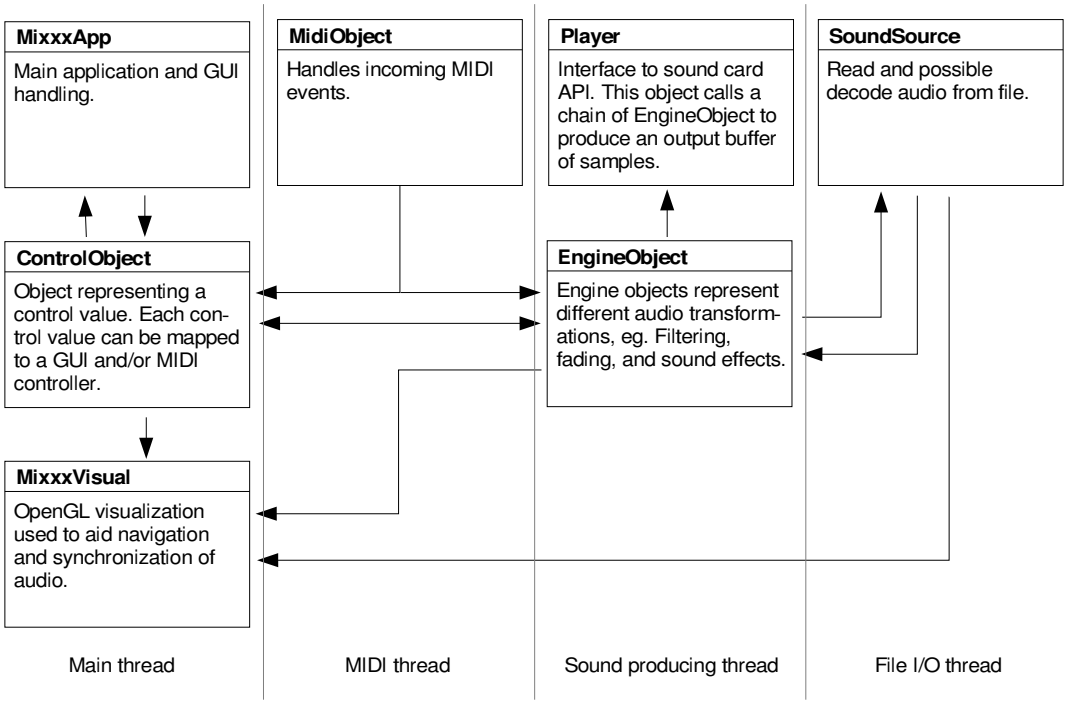


**Figure 2: Overview of the Mixxx architecture, with different modules, executed in four different thread classes.**


[5] R. Bencina and et.al. PortAudio - Portable Audio Library. `http://www.portaudio.com`, January 2003.

[6] H. Beyer and K. Holtzblatt. *Contextual Design*. Morgan Kaufmann Publishers, 1998.

[7] S. Card, J. Mackinlay, and B. Shneiderman, editors. *Readings in Information Visualization*. Morgan Kaufmann Publishers, 1999.

[8] DigiDesign Inc. ProTools. http://www.digidesign.com/, January 2003.

[9] K. F. Hansen. Playing the turntable: An introduction to scratching. *Speech, Music and Hearing*, 42:69–79, 2001. KTH, Stockholm, Sweden, TMH-QPSR.

[10] H. Ishii and B. Ullmer. Tangible bits: Towards seamless interfaces between people, bits and atoms. In *Proceedings of CHI*, March 1997.

[11] K. Jensen. *Timbre Models of Musical Sounds*. Ph.D. dissertation, Department of Computer Science, University of Copenhagen, 1999. Report no. 99/7.

[12] H. Kato, M. Billinghurst, and I. Poupyrev. Virtual object manipulation on a table-top AR environment. In *Proceedings of ISAR 2000*, October 2000.

[13] LADSPA developers. Linux Audio Developer's Simple Plugin API (LADSPA). http://www.ladspa.org/, January 2003.

[14] Microchip Inc. Microchip. http://www.microchip.com/, January 2003.

[15] D. Murphy, T. H. Andersen, and K. Jensen. Conducting audio files via computer vision. In *Proceedings of the Gesture Workshop, Genova*, 2003.

[16] Muzys Team. Muzys. http://www.muzys.com/, January 2003.

[17] H. Newton-Dunn, H. Nakano, and J. Gibson. Block Jam. In *Proceedings of the SIGGraph*, 2002. Abstract.

[18] J. Patten, B. Recht, and H. Ishii. Audiopad: A tag-based interface for musical performance. In *Proceedings of the Conference on New Interfaces for Musical Expression*, 2002.

[19] J. Raskin. *The Humane Interface*. Addison-Wesley, 2000.

[20] Stanton. Final Scratch. http://www.finalscratch.com/, January 2003.

[21] Trolltech AS. QT. `http://www.trolltech.com`, January 2003.

[22] F. Vernier, N. Lesh, and C. Shen. Visualization techniques for circular tabletop interfaces. In *ACM Advanced Visual Interfaces (AVI)*, May 2002.