3D SOUND APPLICATION FOR GAME ENVIRONMENTS

14/10/2014 – ALFRED TÅNG

Supervisor: Daniel Kade, Mälardalen University

Examinator: Thomas Larsson, Mälardalen University
Abstract

This report covers the creation and implementation of a 3D audio application using FMOD Ex API. The report will also cover a walkthrough of the basic principles of 3D and surround audio, examples of other uses of 3D audio, a comparison between available technologies today, both software and hardware and finally the result of the implementation of the 3D sound environment software, both server and client. The application was created to explore the use of 3D audio in immersive environments. There was no application like this available when this project was conducted. An inductive approach along with a form of rapid application development and scenario creation was used to achieve the results presented in this report. The implementation resulted in a working client and server software which is able to create a 3D sound environment. Based on a user evaluation the software proved to be quite successful. With the help of the implementation the user, or operator, can now create a sound environment for another user, or a listener. The environment is created and designed by the operator using the client side of the implementation and later played through the server side which is connected to a 4.1 speaker system. The operator can steer and add sounds from the client to an active environment and the listener can experience the change in real time. This project was conducted as a bachelor thesis in computer science at Mälardalens University in Västerås, Sweden.
Table of Figures

Introduction

3d Audio, Surround Audio and the CrossTalk

3D Audio
Surround Audio
Crosstalk
Summary

State-of-the-Art

Similar Projects
AudioChile
Cocktail – A MMAE Implementation
Summary

Recent Studies

Problem Definition

Challenge
Available Hardware
Available Software
Method
Approach
Obtaining Information
Scenarios

Comparison of Technologies

Audio Software
Summary
Audio Hardware
Headphones
Loudspeakers
Multiple users-problem
TABLE OF CONTENTS

Summary .................................................................................................................. 16
System Design ......................................................................................................... 17
  Design Process ...................................................................................................... 17
  Scenarios .............................................................................................................. 17
  Look and feel ........................................................................................................ 18
System Implementation ........................................................................................... 22
  Getting to know FMOD ....................................................................................... 22
  Prototype ............................................................................................................. 22
    Server .................................................................................................................. 22
    Client .................................................................................................................. 23
    Network ............................................................................................................. 23
  Problems and Obstacles ....................................................................................... 24
User Evaluation ....................................................................................................... 25
  Evaluation ............................................................................................................. 27
Discussion ................................................................................................................ 29
  Conclusion .......................................................................................................... 29
  Future Development ............................................................................................ 31
References ............................................................................................................... 32
Table of Figures

Figure 1 - At the left a bird sound is played through a normal surround system. The same bird sound is played in the right picture but know from a 3D sound system. The user can now determine the position of the sound more accurately. ................................................................. 4

Figure 2 - The crosstalk is the audio waves represented by Rx and Lx. ................................................................. 5

Figure 3 - Screenshot of the game Half-Life, released 8th November, 1998. (Valve Corporation, n.d.) ................................................................. 6

Figure 4 – Screenshots of different situations from AudioChile. (Sánchez & Sáenz, 3D Sound Interactive Environments for Blind Children Problem Solving Skills) ................................................................. 7

Figure 5 - Picture representing the test room. A) Speakers, B) The audio interface and C) The server. ................................................................. 16

Figure 6 - Sketch showing scenario A. .................................................................................................................... 17

Figure 7 - Sketch showing scenario B. .................................................................................................................... 18

Figure 8 - Screenshot of the server software. ......................................................................................................... 19

Figure 9 - Screenshot from Windows Metro simulator running the client software. ............................................. 20

Figure 10 - Concept of one of the discussed final designs. .................................................................................... 21

Figure 11 - Screenshot of the first test application using FMOD. ........................................................................ 22

Figure 12 - 1. User sends a connection request to specified ip-adress and port. 2. The server then answer the client with an ack-message. ..................................................................................... 23

Figure 13 - Picture showing the test setup. A - The audio interface, B - The server, C - The client 25

Figure 14 - Picture showing a test person placed at the sweet spot in the testing environment. A - Speakers connected to the audio interface. ..................................................................................... 26
INTRODUCTION

Introduction

This report describes the creation and implementation of a thesis to support a research in computer science conducted by Daniel Kade at Mälardalens Högskola (MDH) in Västerås. The basic idea behind Daniel’s research is to find easy ways to create a highly immersive environment for the user, where this thesis contributes to the task of exploring and implementing the basis of the sound experience research. The report will go through the basic principles of three dimensional audio (3D audio), software libraries and the implementation of a control panel software complete with server and client. The goal of this project is to develop the control panel software so the user can create, control and record a three dimensional sound environment for another person placed in the center of the newly created sound environment.

The client side of the control panel will be used as a tool to control the server side of the software. The server translates messages sent from the client to sound instructions which later being rendered with the use of the software library called FMOD Ex to a 4-way loudspeaker-system through an external soundcard. The three dimensional sound environment, which is generated from the sound instructions, is then experienced by another person standing in the focal point created by the speakers.
INTRODUCTION

3D AUDIO, SURROUND AUDIO AND THE CROSSTALK

3D audio and surround audio are actually two different things but often mistaken as the same. Much because the current trend in home entertainment where everything should be “High Resolution 3D Super Quality”, where the 3D attribute is used as a selling point and have nothing or very little to do with the sound image of the product.

3D Audio

3D audio is often used in games or other interactive applications and helps to create a more realistic environment. The 3D audio technology were created to mimic the real life audio. Compared to surround audio the 3D audio is a more complex and more immersive audio system. 3D audio is used to create a more vivid projection of the surrounding environment for the listener. Much like the surround audio, the listener is able to say which direction, a more precise direction, the sound is coming from. Besides the direction, the listener can comprehend the distance as well, much like we can in our normal lives. All in all, the simple characteristics of 3D audio comes down to this (Thorn, 2013):

- Volume and attenuation
  - To give the listener an idea of the distance
- Panning and direction
  - To give the listener an idea of the direction
- Reverberation and acoustics
  - To give the listener an idea of the surrounding environment

Even if a 3D sound system consist of only a stereo setup, like gaming headsets, the listener can experience these things really well due to software enhancements like “Razer Surround” (Inc., n.d.) or “EAX”. The software aims to simulate a surround sound system in the headphones using techniques like HRTF (Gardner G. W., 1997) to calculate how the sound should be played.

Surround Audio

The surround audio system is often used for home entertainment or by cinemas. The basic idea behind the surround system is to make the listener understand the basic direction of the sound to enhance the immersive experience. For example; from the back, the front or one of the sides. The listener cannot really understand how far or how near the sound is because it will always seem as if the sound is by the speakers or speaker, see Figure 1.
INTRODUCTION

Figure 1 - At the left a bird sound is played through a normal surround system. The same bird sound is played in the right picture but now through a 3D sound system. The user can now determine the position of the sound more accurately.

Crosstalk

Back in the early days of 3D audio, roughly at the time 3D games became popular, many people said that 3D audio was best experienced through headphones. This may be because it is much easier to isolate the 3D audio sources and deliver them to each ear without any interfering audio signals (Song, Zhang, Florencio, & Kang, 2010), also known as crosstalk. For example, let’s say that one sound is played in the left loudspeaker. This sound is only meant for the left ear but as you know you will hear it in the right as well. The same goes for the right loudspeaker. When these things happens the 3D audio effect fails and the sound you will hear will sound as a normal stereo sound where the two different sources, left and right, are mixed together. See Figure 2 for an illustrated explanation of the problem.

Per definition in the Collins English Dictionary (Dictionary.com), crosstalk is; unwanted signals in one channel of a communications system as a result of a transfer of energy from one or more other channels.
Figure 2 - The crosstalk is the audio waves represented by Rx and Lx.

Today it is possible to eliminate crosstalk by constructing special audio filters. (Gardner W.G., 1999) These type of filters creates a barrier between the two speakers almost like a virtual set of headphones. This technology makes it easier for everyone to experience 3D sound in a more comfortable and cheaper way without the use of headphones or complex surround setups.

Summary

Depending on the situation the two types of audio imagery systems presented earlier is used differently. In a movie, TV or other static entertainment where the immersion already is limited the surround audio system is most used. This is maybe because the surround audio model is fairly old and reliable and there is no need of you, the viewer or listener, to interact with the entertainment. In dynamic entertainment however, such as games or virtual reality, where immersion and interactivity plays a huge role one should think about using a 3D audio system. The 3D audio system has the best ability to recreate audio as we physically hear it in our everyday life, making the illusion of reality much more real than the surround audio option.
STATE-OF-THE-ART

State-of-the-Art

SIMILAR PROJECTS

3D audio has been well used in the gaming industry for years. Half-Life (Figure 3) was one of the first games in the industry that had support for 4 channel 3D audio. Today FMOD (Firelight Technologies PTY Ltd., n.d.), the software library that is used to achieve the results in this thesis, is used in a variety of games. Including AAA-titles such as Bioshock and Crysis. (Firelight Technologies PTY Ltd., 2013) It has also been integrated in the famous game engine Unity (Unity Technologies, n.d.).

FMOD is just one among many other engines available for the developers. 3D audio do not exist only in entertainment. 3D audio is a really good tool to use if the user is unable to experience visual feedback, for example when the user is driving. A lot of the research in 3D audio has been done to create an easier living for people with seeing disabilities (Sánchez, Baloian, Hassler, & Hoppe, 2003). With the help of 3D audio seeing impaired individuals can more easily visualize their surroundings or even use computers (Frauenberger & Noisternig, 2003). There are also other uses for 3D audio, including educational and work related, which are further explained below.
AudioChile

“3D Sound Interactive Environments for Blind Children Problem Solving Skills” (Sánchez & Sáenz, 3D Sound Interactive Environments for Blind Children Problem Solving Skills), is a project that uses 3D audio to help blind or seeing impaired children increase their cognitive and learning skills. The software they created, AudioChile, is a game where the user travels to different cities across Chile. The user then solves mazes and puzzles using impressions and feedback from both stereo and 3D audio together with simple on screen graphics (Figure 4).

The main purpose of their study was to see if young children with visual disabilities could easier develop certain problem-solving skills and also see if this type of virtual environment would be any useful in general. The 3D sound was used as a help to direct the user, localize different objects or characters and navigate the game world.

Cocktail – A MMAE Implementation

A project called “Cocktail” (Edlund, Gustafson, & Beskow, 2010), which was based at KTH in Stockholm, used 3D audio to create a flexible 3D audio environment that could be changed in real-time, much like the goal of this thesis. Cocktail relies on the so called “cocktail party effect” (Benyon, 2010), thereby the name. The effect occurs when the listener experience a lot of sounds at the same time and focuses on only one, then makes a judgment if it is important or not and finally moves on to the next sound.
The software creates a 3D soundscape using a random set of sounds that is being played all at once when the user starts. The software selects random sounds from a repository containing over thousands of sounds that has been configured in advance. Probabilities for a certain position of a sound, or for a certain sound to be played, can be configured at runtime. Cocktail then plays these sounds, which can be from a hundred to a thousand or more, all at once. This is made available through their sound engine MMAE. MMAE, compared to other normal sound engines, can compute thousands of 3D sound objects with very low computational cost. The software uses Snack Sound library (Sjölander, 2006) written by Kåre Sjölander at KTH, as its backbone.

**Summary**

There are different aspects from both of these projects that is interesting for this thesis. One thing is the area of application of 3D sound. “3D Sound Interactive Environments for Blind Children Problem Solving Skills” explains the 3D sound as a tool to widen the cognitive functions for young children. This hopefully means that the product of this thesis can serve as educational tool, in addition to its entertainment applications. Another interesting ability for the product of this thesis would be for it to be able to play sounds at the capacity of Cocktail.

**RECENT STUDIES**

In recent years the main focus in 3D audio research has been to enhance the quality of the 3D sound and make it more available to consumers, one of these studies is called S3A: Future Spatial Audio for an Immersive Listener Experience at Home (Research Councils UK, 2014). This research is a collaboration between several companies in the entertainment industry and researchers at the University of Surrey in the United Kingdom. The entertainment industries consist of companies from different branches of entertainment including TV, film and games.

The main focus of the research is to make 3D audio more practical and provide new platforms for UK’s entertainment industries to deliver high quality immersive sound to their users. Today, a high quality experience of 3D audio requires an advanced loudspeaker setup and the majority of listeners do not have this at home or together with their mobile device. By the time S3A is over the research team has hopefully found a way where 3D audio can be experienced at the same quality regardless audio setup.

The project is fully funded and as of today still active but due to its early state no results has yet been published. Dr. Tim Brookes, supervisor for the S3A project, has listed that the research is planned to continue until 2018 (University of Surrey, 2014).
Problem Definition

The time before this thesis was conducted similar tools was non-existent or not able to create the immersive sound environment that was required for the goal of this thesis. With the help of the solution provided in this thesis the user can create and move sounds in a 3D-environment from a client software which is then handled by a server software.

This thesis will serve as the audio part of a larger research project in immersive environments and hopefully provide a good foundation to further research in educational and entertainment purposes of immersive environments.

CHALLENGE

Create a client software which controls the 3D sound environment rendered by a server that receives signals from the client software. The server software should support scalable surround systems ranging from simple headphones to 7.1 setups or more (current sound interface supports only up to 7.1, see hardware specifications below). The client should be able to select sounds from a list of available sounds on the server and then add additional effects such as reverb, max/min volume etc.

Available Hardware

Audio Processing

There are two different systems used when processing audio. The loudspeaker system consists of four Genelek 8020C speakers connected to a Steinberg UR824 audio interface which is connected to the server. The other system consists of just one set of standard headphones connected to the standard sound card output of the server.

Genelek 8020C Speaker (Oy, n.d.)

- Active speaker
- Frequency range: 66 Hz - 20 kHz
- Dimensions: H x W x D, 242 x 151 x 142 mm
- Low frequency power: 20 W
- High frequency power: 20 W
**PROBLEM DEFINITION**

**Steinberg UR824 Audio Interface** (Technologies, n.d.)

- Connects via USB (24 bit/192 kHz)
- 8 line channels (output)
- 6 XLR channels (input)
- Optical I/O
- Supports surround setups up to 7.1 (MAC)
- Supports iPad

**Server**

If the purpose is to run the software with the Genelek speakers and the Steinberg interface the server is a Mac machine running OS X 10.7 or later with network support. This is because the Steinberg interface drivers don’t support surround setups in a Microsoft Windows environment. But for a basic setup with headphones a Windows laptop running Windows 8.1 has been used.

**Client**

The client software is running on a tablet PC running Windows 8.1 with an Intel Atom CPU and standard Wi-Fi and touch support.

**Available Software**

To control the output mixing from the Steinberg interface the accompanying software called dspMixFx is used. With this software the user can monitor and control how the interface should handle the various built-in channels.

Razer’s software Razer Surround is used at the server side to enhance the surround feeling when the user uses headphones instead of the loudspeaker system. Razer Surround is a virtual 7.1 surround engine that simulates a fully-fledged surround speaker system in a pair of regular headphones. (Inc., n.d.)
METHOD

At the start of the thesis three main parts were discussed and decided. The first part consisted of information gathering and literature review. In this part the main purpose was to find out if similar work had been done anywhere else and if so, how could their approach be beneficial for our ideas. Hardware and software options were also investigated at this stage.

After a decision was made about what hardware and which software library was most suited for this thesis the second part began. This part consisted mostly in meetings and discussions between student and supervisor about possible scenarios and how the software was going to be used. For a more in-depth look of the product of this part see Scenarios, page 15.

The last part was to implement a working demo, a prototype of the combined knowledge from the two earlier parts. The prototype was created to explore the capabilities of the ideas and the earlier developed scenarios.

Besides these three parts the student met with the supervisor each week throughout the thesis to discuss the progress and to get help in case of problems. It was also at these times the weekly planning was decided. The planning consisted of small tasks that was supposed to be done until the next meeting almost like scrum sprints (Schwaber & Sutherland, 2013).

Several different methods were used throughout the thesis. In the two first parts of the thesis standard methods for information gathering and literature review together with methods from interaction design were chosen, including brainstorming sessions and scenarios. During the implementation of the prototype this thesis incorporated two methods from computer science, Rapid Application Development (or RAD) and, as mentioned earlier, a bit of scrum. An inductive approach was used to create knowledge and conduct the research. The research was based on the thought to explore how a 3D sound environment can be used to create immersive experiences for users. To prove this reasoning and explore the capabilities of the implementation a quantitative user test was conducted (see User Evaluation, page 25).

Approach

There are a lot of different ways to approach a thesis and to know which way would be best for this thesis the question that is meant to be answered must first be recognized. With the help of an inductive approach together with a quantitative user test this thesis will try to give an answer if it is possible to create an immersive sound environment in a scenario where the user easily steers the sound in real-time for another user that is placed in the middle of the created sound environment. The result will be based on the user test (page 25) of the prototype explained in “System Implementation”, page 22.
Obtaining Information
At this stage the main goal was to achieve a greater understanding about the limitations and the requirements when working with 3D audio. A lot of information about earlier projects and 3D audio in general was obtained, the information was then sorted in relevancy to the thesis and unwanted information was disposed.

Research and obtaining information like this is a crucial part in a project. It is from here a researcher plan the next course of action and possibly anticipates unwanted pitfalls in the future of the project.

Scenarios
Scenarios created in the second part of the thesis served as a background or a sketch of instructions on how the final software was meant to be used. No considerations was taken to real world limitations when the first scenarios was created. As the last part progressed, the implementation part, the scenarios became more of a final goal than a sketch and more customized to fit certain limitations.

This method combined with an open mind, without the thought of limitations, can be a useful tool to create unique and well-functioning software.
Comparison of Technologies

AUDIO SOFTWARE

There are several different audio libraries that support 3D audio. Depending on what platform and which features are needed one is better than the other. To get a better understanding of which library was the most fitted for the thesis an investigation was made.

The libraries that were most interesting for this thesis were:

- IrrKlang (Ambiera, n.d.)
- FMOD Ex (Firelight Technologies PTY LTD., n.d.)
- OpenAL Soft (Strangesoft, n.d.)
- XAudio2 (Microsoft, n.d.)

The goal of the initial investigation was to find out which of these libraries that supported the following key features and checkpoints the best:

- How easy it was to locate the sound
- If it has a good quality output
- If it supports individual speaker assignment
- If it is possible to position multiple listeners in one sound environment
- If it is scalable, from stereo (two speakers) to e.g. 32 speakers
- If it is free

<table>
<thead>
<tr>
<th></th>
<th>IrrKlang</th>
<th>FMOD Ex</th>
<th>OpenAL Soft</th>
<th>XAudio2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Easy to locate the sound</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scale 1-5</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>Sound Quality</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scale 1-5</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Individual speaker</td>
<td>NO</td>
<td>YES</td>
<td>N/A</td>
<td>YES</td>
</tr>
<tr>
<td>assignment</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**COMPARISON OF TECHNOLOGIES**

<table>
<thead>
<tr>
<th></th>
<th>NO</th>
<th>YES</th>
<th>NO</th>
<th>YES</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Multiple Listeners</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NO</td>
<td>YES</td>
<td>NO</td>
<td>YES</td>
</tr>
<tr>
<td><strong>Scalable</strong></td>
<td>N/A</td>
<td>YES</td>
<td></td>
<td>YES (Only up to 7.1)</td>
</tr>
<tr>
<td><strong>Free</strong></td>
<td>YES (Non-commercial)</td>
<td>YES (Non-commercial)</td>
<td>YES</td>
<td>YES</td>
</tr>
</tbody>
</table>

*Table 1 – An overview of the comparison of the software libraries*

**Summary**

Investigation showed that some of these libraries was not interesting anymore. Some due to lack of further development and others due to missing key features like, 3D sound support or scalability, see Table 1. To get a deeper understanding about the possibilities, differences and limitations regarding the libraries that still was in the picture a simple test software was created. This software tested how easy it was to get the wanted results and the quality of sound the library provided.

After an extensive look into the remaining libraries only FMOD and XAudio2 were considered an option. The overall winner of the test were FMOD due to its easy implementation and good documentation. There is one thing to consider though, because of the bare bone structure of the XAudio2 API one could write more efficient and optimized code. Due to its bare bone characteristics, the API require more time to get simple functionality working and the XAudio2 is not cross platform compatible. The XAudio2 API is only supported in Windows and Xbox, when FMOD is supported on almost every well used platform today.

**AUDIO HARDWARE**

There are a variety of hardware options, ranging all the way from headphones to 9.2 systems and so on to choose from to when creating a functional 3D-sound environment. Following paragraph contains some general pros and cons regarding headphones and loudspeakers.

**Headphones**

Pros:

- Availability
- Cost
- Easy to achieve effect
COMPARISON OF TECHNOLOGIES

- Compatibility
- Easy to store
- Mobility
- Ability to close out surroundings (Isolation)

Cons:
- Lack the feeling of space (spacious-effect)
- Long use = uncomfortable

Loudspeakers

Pros:
- Availability
- Sound quality
- Can be cost effective, depending on situation (output effect etc.)
- Spacious-effect
- Larger effective area or “sweet spot”

Cons:
- Often big
- More peripherals = more cost
- Hard to close out surroundings (sound from other hardware etc.)

Multiple users-problem

When using headphones only one user can participate at a time unless multiple headphones are connected to the same system. This can be a great disadvantage if let’s say two players play a game in the same room. At the same time as sound is playing in player one’s headphones player two tries to communicate with player one but player one cannot hear player two’s voice because of the sound the game is making. This problem can however be eliminated if each player would use some type of voice input to the system.

If a loudspeaker setup would be used instead, both players would experience the same 3D effect and they would probably hear each other. Take note that when you are using a loudspeaker setup the only sound image that can be experienced is from one place in the game, this means that the two players must be in the exact same location at the same time if they play the game together in the same room at the same machine. This scenario could for example exist in a helicopter simulation where the two pilots almost sits on each other. Meaning that the two pilots would in reality almost have the same audio environment.
Summary

Depending on situation and purpose of the sound environment one is more suitable than the other. For example, if there isn’t so much space and the purpose is to get the listener fully immersed in a game or simulation one should consider going for headphones. This because headphones don’t take that much of space and it captures the listener in a more intense way than regular speakers. Headphones has the ability to "shoot" the sound more precise into the listener’s ears and isolate the listener from the ambient noise.

There is no really scientific answer if headphones are better than loudspeakers or the other way around. This debate has been going on for decades and will probably continue to do so. As of 2013, a trend showing that headphones have become more of a fashion item instead of a functional accessory. So today, if you look at it through the eyes of the general user, it should be headphones before loudspeakers. Not just any headphones, they should rather look good than sound good. (Arthur & Gibbs, 2013)

With that being said tastes differ from person to person, this is really up to the listener and what he or she prefers.

For this thesis however the focus is set on four loudspeakers placed in a square form. With its current measurements (from one speaker to another) this setup creates a sweet spot of roughly 0.09 square meters.

Figure 5 - Picture representing the test room. A) Speakers, B) The audio interface and C) The server
System Design

DESIGN PROCESS

Scenarios
From discussions with the supervisor two possible scenarios were created where this software could be used.

Scenario A

Figure 6 - Sketch showing scenario A.

Scenario A revolves around one user standing in the sweet spot and another user, which can be seen as a type of administrator, stands outside of the sound environment and controls the sound using the tablet. The administrator or user B creates a soundscape for User A with the help of the control panel software running on the tablet. The sound is then routed to the Steinberg interface and from there out to the speakers surrounding user A.
**Scenario B**

In this scenario user A, C and D has a mobile device, preferably a phone or something in that size. User B is still having the same role as in scenario A but instead of the server sending the audio signals through the external audio interface it streams sound directly to user A, C and D's mobile devices. User A, C and D then receives the sound in a pair of headphones connected to their own mobile device.

**Summary**

The scenario that was chosen as the goal for this thesis was scenario A (Figure 6). Scenario B (Figure 7) was ignored due to its added complexity, where the server streamed the sound to multiple users instead of playing the output directly to a set of speakers. This required the development of an application for a mobile device in addition to the already discussed server and client, which would be a too great task to finish in the given time.

**Look and feel**

During the whole thesis the prototype changed its design several times. Because of its permanent prototype stage appearances came second hand. The focus was more on the functionality and
debugging on both sides of the whole implementation (client and server). Even if the focus was set on functionality the ideas of the final look was kept to allow for future refinement of the user interface and experience.

**Server**

When looking at the current design of the server (Figure 8) one can quickly see that some of the features are really unnecessary for a final product. For the final product the server probably would have been converted into a service application, a console application or a simpler Form application. Then almost every current feature could have been removed and replaced with some simple command line options or something similar.

---

**Figure 8 - Screenshot of the server software.**
As the server, much of the current controls are implemented for easier debugging and evaluation. For example, the whole upper area of the application consists of a log (see Figure 9). The application logs everything that is happening, including network events and sound events, this feature is one of many that would not be consumer friendly. The general user should not really see or care of what is happening in the background.

Much like the scenarios a final look was discussed during the time that the thesis were conducted. The discussions were mainly about the client software because it was the biggest part which was meant to be seen by general users.
Future looks

One of the biggest differences between the concept (Figure 10) and the actual sketch (Figure 9) is that the panel representing the sound area, the area with the red dot in the lower left, has been centered and the whole log control has been removed. The colored dots represents different sounds in the sound area and the big black dot represents the listener. All of the active sounds is then represented in a list with their current status and coordinates. It was also discussed that the user should be able to save and load different sound scenarios and have the ability to “paint” a road or set waypoints that a selected sound could follow.

At the final stages of the client software the hopes was to have a final design like the one described above. As always, when developing user interfaces for the general consumer, one should put a great weight on user friendly placing of controls and to use well known design guidelines. This makes it easier for new users to feel familiar with the interface (Benyon, 2010) and furthermore it makes them utilize the software the way it’s meant to.
System Implementation

GETTING TO KNOW FMOD

With the help of source code from different FMOD examples that came with the API a simple application was created in Visual Studio 2013 using C#. The application was able to play a selected sound at given coordinates. This software became the base of the prototype’s server part but it was initially meant as a simple “get to know”-application to better grasp the understanding of the FMOD API. (Figure 11)

In this test application the user could play a sound from the hard disc drive and then control the coordinates of the sound using the keyboard. When the user started the application FMOD fetched system information like the name of the driver and the speaker setup that was configured on the machine it ran on. As the knowledge grew about FMOD under the time of the thesis, more parts were implemented like sound effects and support of multiple audio sources.

PROTOTYPE

During the whole period of the thesis the prototype evolved to a more complex application almost each week. From the start it could be considered as a simple audio player with very limited options. This simple audio player finally became a more complex 3D audio renderer with the support of different effects, audio formats and outputs.

The prototype consists of two main software parts. A client which can send instructions to the other part, the server. The implementation of these two parts were done in parallel to always be able to maintain functionality with each other when testing occurred. A more detailed explanation of the implementation process of these parts can be found below.

Server

The server was implemented using C# and Windows Forms. Instead of a Windows Forms implementation, the server could have been implemented as a service or a console application. But because it evolved from the first test application the form part was kept. One of the main
reasons to keep the existing test application was because it would be rather time consuming to copy-paste and special fit the code into a new project.

**Client**
Because the client software were planned to run on a Windows 8 tablet, the client was created as a Metro style application. The Metro interface was first introduced in Windows 8 and is meant to work as a seamless touch interface. This means that most of the user interface of Windows 8 is best suited for touchscreens. The applications written for Windows 8 Metro are more like mobile applications instead of a normal computer applications.

With this in mind the work began with creating the design that was discussed with the thesis supervisor (see Design Process). The design was implemented using XAML in Microsoft Blend and after all the controls and buttons were in place the code was implemented in Visual Studio.

**Network**
The connection between the server and client differs a little bit from each other. Because the client is a Windows Metro application the network API is not the same as a standard Form application. When Microsoft released the SDK for Windows Metro applications (Windows Store app) they changed the networking part in the API to be able to communicate with the network in the background when the app is running.

In a standard form application, like the server, the System.Net.Sockets namespace is used to create a TCP-connection. This namespace is not available in the Windows Store API so the client is using a namespace called Windows.Networking.Sockets.

In a more general sense, the connection between the client and server is a TCP based connection with acknowledge-messages (ack-messages). For example, when the client connects to the server an ack-message is sent from the server to let the client know that it is listening for instructions. This type of messaging is persistent throughout the whole session, e.g. sending/receiving sound instructions, stop all sounds and so on. The ack-messages makes it easier to keep the client and server synchronized with each other. See Figure 12.

**Figure 12** - 1. User sends a connection request to specified ip-adress and port. 2. The server then answer the client with an ack-message.
The messages that is sent from the client consists of simple strings. These strings are then decoded by the server and the right method is called. For example, if the client want to play a certain sound at a given position the message can look like this:

“SEDING SOUND^4^12^20^nameOfSound.wav^”

Each parameter is separated by a “^”. The first parameter tells the server which method should be called, the three after that tells at which coordinates the sound should be placed and the last tells the server which sound should be played.

PROBLEMS AND OBSTACLES

After the software was up and running it was time to connect it to the audio hardware. The software worked as expected when simple headphones was connected directly into the server’s built-in sound card.

The problem came later when the Steinberg interface was connected to a PC running Windows 8.1, the sound interface was unable to output surround when running under Windows. After a brief conversation with the Steinberg support it became clear that they do not support surround setups in Windows but in OS X it should run fine, even if they state support for both operating systems in their product description. With this set-back the server side of the application had to be run under OS X and because it had been developed using Windows forms and C# it had to be run with some sort of Windows emulation like Wine. (Wine HQ, n.d.)

After the server application finally was installed on the OS X machine running Wine a new problem emerged. Because of some limitations in the Wine engine the working surround setup in OS X could not be recognized in the “emulated” server software.

As a next step to get the server software running on the OS X machine was to try to port the whole application with the help of Mono API. Mono is a cross-platform implementation of Microsoft’s .NET framework which, if done right, would work on all major platforms without changing much of the code (Mono-Project, n.d.). Mono was a great tool to use because the server software was already written in C# and after the port almost every part of the server was kept and ran natively under OS X.

Even if it was rather easy to port the server from Windows forms to Mono all of these problems could have been eliminated from the start if Steinberg had given the correct information about their product to begin with. The server application would then be written in C or C++ and compiled for OS X to run as smooth as possible if the Steinberg interface were still considered as the sound hardware.
User Evaluation

To achieve a greater understanding of how close the software was to the initial idea and how well a user experienced the sound environment created by the software user tests were conducted. There are several different methods available to evaluate user experience, such as the Geneva Emotion Wheel (Scherer, 2005), PANAS (Watson, Clark, & Tellegen, 1988) and 3E (Tähti & Arhippainen, 2004). A more in-depth walkthrough of user-centered evaluation methodologies can be found in Handbook of Human-Computer Interaction (Karat, 1997). After discussions with the supervisor the most suitable evaluation method was to let one user at a time experience the sound environment and afterwards participate in a semi-structured experience interview (Longhurst, 2003) (also called soft interview). When conducting a soft interview it is important to keep the interview questions open. The soft interview is meant to be more in a conversational manner than a normal interview so instead of asking “Did you experience the sound to the left?” the questions was more “how”-based, like “How did you experience the sound to the left?”. This made the interview more like a discussion instead of a normal interview.

Figure 13 - Picture showing the test setup. A - The audio interface, B - The server, C - The client
Seven randomly selected people, both female (2) and male (5) in the ages of 22-28 tested the software. None of the people that tested the software had any experience of similar projects or 3D audio. The test consisted of three different scenarios and after every scenario, each of the participants filled out a questionnaire. After the participant read the questionnaire he or she was placed at the sweet spot (Figure 14) in the same setup as shown in Figure 5, page 11. The questionnaire was then discussed in a short soft interview.

The following text was placed in the beginning of the questionnaire to thank the participant and to ensure his or hers confidentiality when the results will be published:

“The purpose of this questionnaire is to gather information about the current state of the prototype. The information gathered with this questionnaire is vital and very valuable for us. Therefore, we assure you that we treat your information confidentially and thank you already in advance for your participation. Results from the study will be academically published, but no individual or opinions will be identifiable.”

Each participant was also told from the start that this test was voluntary and that they could cancel the test any time if they felt uncomfortable. These instructions along with the text in the questionnaire is very simple but very important things one must include when conducting tests like this. (Burnmeister, 2000)

Figure 13 shows the test setup. The server (B) was an older mac book running OSX Maverick. The tablet (C) was connected to the server and the black laptop was used only to deploy the client software to the tablet. The audio interface (A) input was connected to the server and its output connected to four speakers that can be viewed in Figure 14 (A), page 25.

In the first scenario the test person was told to locate and understand the direction of the sound placed in the environment. In the second scenario the test person was supposed to estimate how far the sound was from the test person’s position in relation to the speakers position and in the last scenario five different sounds were played and the test person was told to describe each sound, their position and how easy it was to separate the sounds from each other.
EVALUATION

Before the test began the participants showed great interest and seemed to be excited to experience the software, mostly because they never had any previous experience with 3D audio before. The time between the different scenarios of the test was short enough to keep the participants interested and focused. The overall timing of everything worked out quite well and each participant had enough time to fill out the current part’s questions in the questionnaire before the next part began.

The tests showed that six of seven persons thought a single sound was fairly easy to point out where it came from (Chart 1) and how far it was from the listener (Chart 2). The problem came later when the user was told to count the number of different sounds in one environment.
Five of seven testers were one sound away from the right number (Chart 3). This however can be because of sounds that were chosen for this scenario which were not easy to hear to begin with or due to the fact that they almost merged together when they were played in the same environment. In retrospect, a selection of more distinct sounds should have been chosen to create more accurate results, see Chart 4 and 5.

In conclusion, the overall experience of the test proved that this software is heading in the right direction. The implementation provided in this thesis proves that it is possible to create an immersive sound environment based on one of the scenarios (Figure 6) that were developed before the implementation process began. Throughout the test the participants seemed to enjoy the software and they seemed to be comfortable in the testing environment. Besides the poor sound choices in the last part of the test, the testing, viewed in a bigger picture, could not have went better.
Discussion

CONCLUSION
At the beginning of this thesis the main question was to find out if it was possible to create immersive sound experiences for a user in the sound environment and as the research went further this thesis proved to be unique. This means that there was no exact road to follow, because this had not been done before. The result of the user tests proves that it went the way it was supposed to. In other words, it is possible to create immersive sound experiences with the help of 3D sound.

The final product of the thesis proved to be unique in comparison to the state of the art presented earlier in this report. Even in the information gathering part of the thesis no information was found that came close to the result of this thesis. Because of the unique way 3D sound is being used in this thesis the result could serve as a good stepping stone in further research in easy controlled sound environments.

The only thing that is similar to the implementation presented in this thesis is the Cocktail project (see page 7). The most unique thing about this implementation compared to Cocktail is that Cocktail is fairly limited when it comes to customizing the sound space. As stated in the state of the art, in Cocktail the client has hardcoded presets to choose from when creating the sound space but in this implementation the client can choose whatever sound she likes. Other than Cocktail none of the other projects that was found came close to the same use or implementation as this thesis.

As stated earlier in this paper, the server software would have been written in C or C++ instead of C# if the information that the Steinberg interface did not support surround sound in a Windows environment had been know from the start. To solve this problem the server was implemented using Mono and if the server application were written in C or C++ it would have run natively under OS X, without any special treatment like this.

One thing that worked really well was when the server was running under Windows, like it was meant to, with headphones connected as output. When using it this way along-side the Razer Surround software the result is really good, the 3D feel is really there.

During the testing of the implementation the test persons were really curious on how it would feel to experience the sound environment created by the prototype and the four speakers (see page 25). Besides the test persons other people that heard about this thesis were really interested. If this software was combined with some sort of visual feedback it could become
really interesting and enjoyable but presented on its own it really feels like there is something missing.

Although the design and some parts of the implementation did not live up to the requirements of the decided scenario, this thesis have contributed to a good startup platform for future development and research in immersive game environments.

Working with sound in general has always been an interest of mine and the thought of this project, to work with 3D sound, was thrilling. At the start I did not know how sound in 3D space worked. I had only experienced it before when playing games and only worked with it in a simple way through Unity3D but no experience what so ever to code with audio API's.

When I started to integrate FMOD in the prototype it was really easy to understand the way it was meant to be used. FMOD has a really extensive documentation and there is a lot of developers that use FMOD which results in a wide range of forums (GameDev.Net LLC, n.d.) and 3rd-party guides (Katy, 2012-2013) that is available on the Internet. All this made it easier for me to implement FMOD, besides the sound quality this was one of the main reasons why I choose FMOD as the featured software library in this thesis.

One of the biggest disappointments was the audio interface from Steinberg. It was rather displeasing that it did not support surround capabilities in Windows. After this it was clear that I had to get my software working on OS X. It took a while but I learned to use OS X and got my software working, first through a simple “emulation” and later I learned to port it. Even if it was rather frustrating and time consuming I feel like I learned something. Now I have experience to work in OS X and I have experience in using cross-platform tools like Wine or Mono.
FUTURE DEVELOPMENT

It would be interesting to see this project as a final product. One of the major improvements that should be fixed is the support for OS X. The server needs to be rewritten in C or C++ to run natively under OS X. Right now the client has some performance issues when creating and steering a sound in the environment. One other thing to fix for a final version is the design of the client software, according to Figure 10.

If I get the opportunity and time I would help Daniel further in his research. It has been a great learning experience and hopefully, in a not so far future, we see products like this in our own homes.
References


Frauenberger, C., & Noisternig, M. (2003, July). 3D audio interfaces for the blind. In *Workshop on Nomadic Data Services and Mobility, Graz, Austria, March* (pp. 11-12).


http://www.surrey.ac.uk/schoolofarts/people/complete_staff_list/tim_brookes/index.htm

