Remote Music Collaboration

Onusa Yamchareon and Gero Herkenrath

December 21, 2005

Abstract

This work deals with systems for remote collaboration on music. It provides a general understanding of why such systems are in need and how exactly collaboration on music can be understood by introducing three different aspects of the field. After explaining several problems to be solved when dealing with these aspects it will look on six systems as examples. Each system is explained and will be discussed afterwards to give an impression on how they solve the before mentioned issues. Tha last part of this chapter will then give a conclusion together with the "MC-Spave" a design space like overview of the six systems.

Contents

1 Introduction

When we think about music we will probably imagine a song of our very own taste and consider it as one single piece. On a second thought it is, however, more than just one single thing, it is a combination of different instruments each playing another melody, composing the song as a whole. Of course in most cases that means that actually several people worked together to produce that specific piece of music. This paper will now introduce computersystems that help people to remotely collaborate on music. It will explain why it is important these days to provide such systems and give several examples of existing systems or systems in development. Additionally it will explain the different aspects of collaboration on music in general, that is what exactly can be collaborated on, and it will explain why it is socially important to suport collaboration on music.
The rest of this paper is organized in the following structure: First we explain the already mentioned social importance for systems supporting remote collaboration on music. Next we will deal with the different possible ways to collaborate on music, that is introduce the terms composition, improvisation and performance. After this we will continue looking at the problems that each of these aspects of collaboration on music induces, together with some general problems one has to solve when designing a system. This together provides a base to evaluate the six example systems that are listed in the following section of the paper. The conclusion will then present what we call the music-collaboration space, a kind of specific design space in which the presented systems are embedded and that may give ideas for future projects that deal with parts of that space that are not yet developed very much.

2 Collaboration - Why?

The perhaps most imminent question people who have no experience with systems for collaboration on music may be "Why do we need these systems?" There are two different answers to that question. The first one is of a social kind, thus perhaps convincing most people. In past times music had a much greater role in everyones everyday life. Even people who could not play an instrument or were not very talented in making music and singing had to do with it. People sang at work, at festivities or even for sad reasons like a funeral. In our days much of these social aspects of making music have changed. Playing an instrument or even singing is considered to be a stylized activity requiring practice and accuracy [NBK04]. This can be considered as some kind of social loss and one motivation to support collaboration on music this is to bring back the making of music together with others into our everyday life. Since the parameters of this everyday life have changed with times some things are to be kept in mind when dealing that problem. First of all the technical knowledge has grown, so it is just logical to see how this can improve collaboration, but there is another aspect, too. People's lifes have become more hectic and mobility was never as much possible as today. We have friends that may live very far away from our homes and that we would like to collaborate with, due to the social nature of making music. Or we may want the possibility to quickly arrange a jam session to make music with others. Since it is not a good option to always carry around our instruments or travel long distances just for making music, systems enabling remote collaboration are in great need. From there it is only a small step to not only make these systems usable for professional artists, but also for average people who might not play an instrument. We will present examples for such systems later in this paper. Of course this means that

the second great motivation for creating music-collaboration systems can only be to help professional or semi-professional musicians then. These people who already integrated music a bit further in their social life will probably agree that it is more contending to make music together with others rather than playing for oneself alone. For them it is not so important to have systems helping in the creation of music but creating some kind of virtual space where they can meet with other musicians to collaborate. Most of these systems follow traditional ways of representing music and/or rely on the use of normal instruments as well.

3 Composition - Improvisation - Performance

These motivations already give a slight impression what aspects belong to collaboration on music. We will split them into three terms, composition, improvisation and performance. A system that is supposed to help in the collaboration on music must deal with at least one of these aspects. It can either support in the process of composing music together with others or improvising music together or actually giving a performance in front of an audience with the performers collaborating via that system from remote locations. Where the first and third term may be self-explanatory we would like to give some notes on the second one so the reader may better understand the rest of this paper. With improvisation we not only mean the term that is used by professional musicians, i.e. referring to improvised soli in e.g. a jazz jam session but also the more common meaning, i.e. creating music by try and error. Thus systems of this category may totally discard traditional ways of representing or creating music so even non-musicians can improvise sounds together. The results of this kind of improvisation may, of course, sound well or not, but the idea behind iit is not to make pleasant music in the first place, but more to bring people to music in the first place. The later to be mentioned system of Daisyphone will probably best explain this idea. The last important thing needed for a good understanding of these terms is that they tend to merge, especially the last two. For example would a system, that supports musicial improvisation in the professional musicians' understanding, but also let's the collaborators perform in front of an audience be falling into both categories. The same goes for systems that let users compose a song by improvising it bit for bit; it would fit into the criteria of composition and improvisation as we defined them. This means that the categories describe a one-dimensional space of systems, from composition over common-sense improvisation, professional-sense improvisation to performance. The idea of this space together with the most problematic of establishing such systems remotely will be presented in the last chapter of this work.

4 Problems of remote music collaboration

For remote music collaboration, there are many problematic aspects that the systems aiming to support it should be concerned about. In our study we would like to categorize these problems, according to the types of music collaboration: composition, improvisation and performance.

For people to do composition, the musical ideas should be able to be shared widely in the format that everybody in the group can understand. The different skill level or the difference of individually skillful instruments is not an obstacle to do collaboration. Scheduling the work and setting composition policies is another facet that should be included in the

system design.
For music improvisation, the need of real-time communication between musicians is very significant. There may be hundreds of times that musicians have to revise their work during performing music together. Also, the agreement of performance - e.g. which instrument plays a major role in each music part, or who performs solo in the particular part - should be handled. Furthermore, ordinary audio recording features are main functionalities of the systems purposed to support such this collaboration.
Not far away from what counts for improvisation, the real-time communication and audio recording feature should also be considered in the systems for music performance. At the same time, special requirements for performance are also things like providing a natural feeling of performing with remote partners, the feedback from audience, and more synchronizing between local and remote audio data.
Additionally, we would like to evaluate the systems also with other common concerns, beside the issues demonstrated above. The learning curve is a tool to represent how much intellectual effort the systems demand to be used. How general social interaction among participants is supported by the systems might be another way to determine their considerations.

5 Systems

In this section we will give six examples of existing systems supporting the remote collaboration on music. Each system will be described in a rather general way, so one gets an idea of the general idea behind it. Afterwards we will then discuss each system following the criteria explained earlier. INterested readers may want to take a look at the references in case they wish to get more detailed information on a certain system.

5.1 myvirtualband.com

The first example system we will introduce illustrates the probably most naive approach to create a system for collaboration on music. Myvirtualband.com [KS] is a project that was started in 2004 by Kelly Senecal and Scott Mason. These two musicians wanted to establish a platform for remote collaboration on music. The result clearly fits in our composition section, since it is basically a bulletin board and a FTP service and thus provides no real-time collaboration.

5.1.1 How does it work?

Members of myvirtualband.com can record their music with the computer in any form they like and then upload them as mp3 to a FTP server. After that they just open a thread in the bulletin board to find other musucians who wish to contribute to their piece. Interested people can then download the file and add more content to it in whatever method they prefer, usually by recording additional lines of music. Then they re-upload the new mp3 again and so on.

5.1.2 Discussion

Though this idea does not specifically introduce any new technique or idea that distinguishes it from systems for other forms of computer aided remote collaboration it has some benefits that need to be mentioned. First of all a bulletin board and FTP server

are widely known and accepted services in todays internet community, so most new users don't have to learn any knew system specific things. This results in a very low learning curve, making the systems attractive for more traditional musicians that don't want to have to learn a lot of knew things just to collaborate with others. Second the idea of composing the music by recording it bit by bit means that the artists don't have to agree on a specific representation of the composition, like bass or violine key. Third the bulletin board provides not only a medium to agree on what and how to work on, but also a well-known platform for social interaction. The latter is an important point every musician will probably agree on, since it is only natural for people to want to know with whom they work. However this naive apporach has also many disadvantages. The most obvious may be the large delay users may experience. It just takes a lot of time to record a track, mix it, upload it, download a new version again and so on. This means it takes a lot of time for people to agree on changes and make suggestions. Another issue is that the resulting music is nothing more than an already recorded audio track, there is not nescessarily and form of written representation of the music so others can play it on their own. Something other systems we will see also deal with is totally missing in this idea, i.e. a system inherent policy for users. Theoretically a collaborator could add to or remove from the work anything they want, even things that formerly were agreed on not to change. If the other users then don't keep copies of former work states of their own the whole piece may be ruined; there is no undo function. Of these three problems the first one is probably the worst, since missing real-time probably confines the users' creativity.

5.2 Digital musician

Digital musician was officially launched in August 2005. With the excellent combination of a community website and a VST (Visual Studio Technology) software plug-in, it allows users to remotely work as a team and produce music online.

5.2.1 How does it work?

Digital musician.net (DMN), a community website, works as users' information center and online community. To be described, users will find system information, a tutorial, guideline, or up-to-date news in the website. Concurrently, online forum, searching mechanism to find collaborators and self advertisement strongly support the communication among users.
"A VST plug-in + DSL internet connection = a revolutionary music production tool" [dm]. Digital musician link (DML), a VST plug-in, mainly supports real-time music collaboration. The plug-in is used together with general stream applications such as Steinberg Cubase and Garageband. The connection between users is realized by using a peer-to-peer method. After connecting to their partners through DML, the real-time communication is provided via DML functions: video/audio conference, text chat. In composition, audio and MIDI data are sent to participants' sequencing program accurately in sync. The musicians can also record and playback data. These functionalities result in kind of a split up studio, where the producer is on the one side and the artist on the other, although of course with Digital Musician, both can control the sequencer.

5.2.2 Discussion

Digital musician aims to support the collaboration of composition. Deliberately, many functions are provided to fit users' needs; real-time communication is realized by video/audio conference and text chat, social interaction is provided by the community website etc. The mechanism to solve the latency problem is done by the idea of time-stamping. The audio and MIDI data will be time-stamped by DML. Thus, the sequencer on the arrival side is capable of knowing where the position of audio and MIDI data should be. In addition, since DML is a plug-in to any sequencing program users are familiar with, that means the learning curve is quite low. However, with peer-to-peer connection, users can only connect to one collaborator at a time.

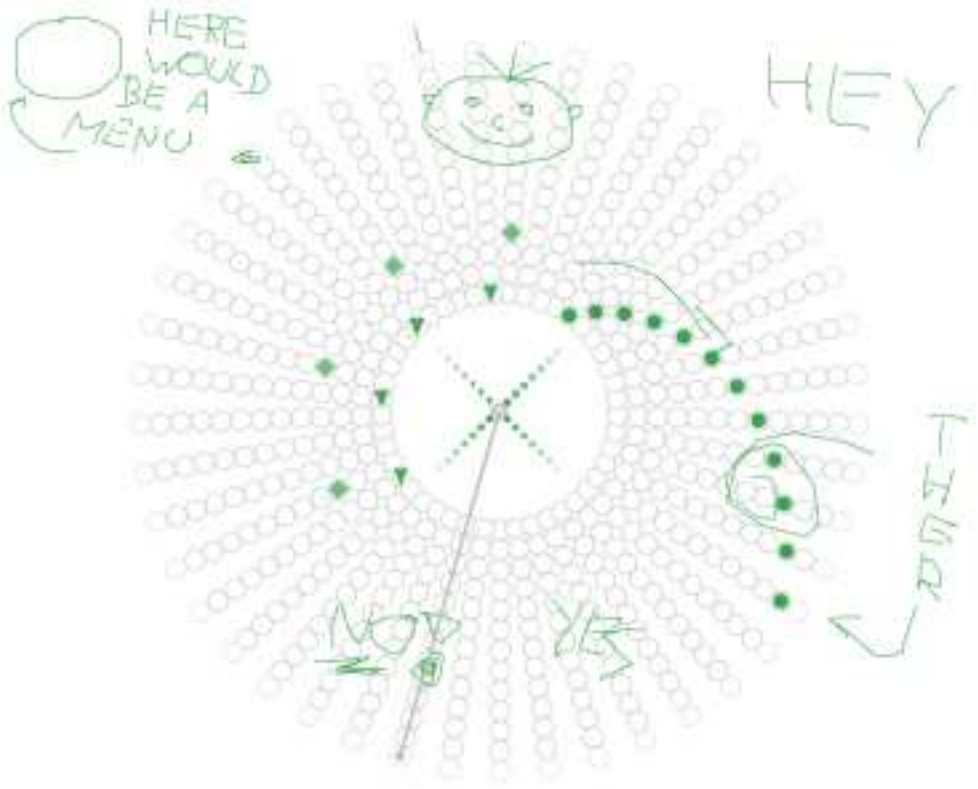
5.3 Daisyphone

The Daisyphone [NBK04] system is the first system we present that actually uses a complete own software and representation of music rather than relying on traditional ideas. It was started 2002 as a research project at Queen Mary, University of London by N. Bryan-Kinns and aims at "remote group improvisation". We will see that this basically fits in what we refer to as common-sense improvisation in this paper.

5.3.1 How does it work?

The basic idea is to play loops of music the user can compose by placing four kinds of graphical symbols on a two dimensional space. The GUI looks like several centered circles of dots with a wandering beat arm. In the middle of this clock-like arragement are four graphical symbols, a dot, a triangle, a square and a diamond. Each symbol is available in different saturations and represents another sound or so to say instrument. After picking one of these symbols with a certain saturation the user can place them onto the circle-dots to make music. By clicking on an already placed symbol it is deleted. Everytime the beat arm crosses a placed symbol the sound is played, where the distance from the center define the pitch, the saturation the volume and the shape the sound. If no symbol is picked (by

clicking on the center of the arrangement a picked symbol is deselected) or the user clicks not on a circle-dot a line can be drawn, allowing the user to draw letters or diagrams to comment on their work. Of course it is a system for collaboration, which means that there has to be the possibility for multiple users to work on the same space. This is possible and in this case each user gets assigned a different color in which he draws and places symbols, so they can distinguish between own work and work of others. Each user has their own wandering beat arm and they can even click on the outer ring to make it jump to a specific place of the loop, but they can not define how quick it moves or stop it from moving. There are no policies given, which means that every user can place symbols where they want and everyone can delete everyones symbols. The picture below illustrates how Daisyphone looks like.



5.3.2 Discussion

This system has a game-like approach to musical improvisation and is not aimed at creating persistent pieces of music in its representation, thus the notes placed fade with time so other, new users find an empty space if nobody else is active or was active for a longer period of time. In case users want to create more than one space, especially since the loop is quite short, they can access other spaces to create new melodies by choosing a space from the ringmenu in the upper left. Considering the use of the system one has to remember that is intentionally made in a game-like way, so it does not surprise that the created music is quite simple. Nevertheless there are some interesting aspects the system shows, good ones and bad ones. Most important is the fact that it has a good method to deal with a short latency. Due to the fact that the loop is quite short and the beat has the same speed for each user they can very quickly react to new contributions done by others and comment and/or add to them. This is a great benefit when compared with

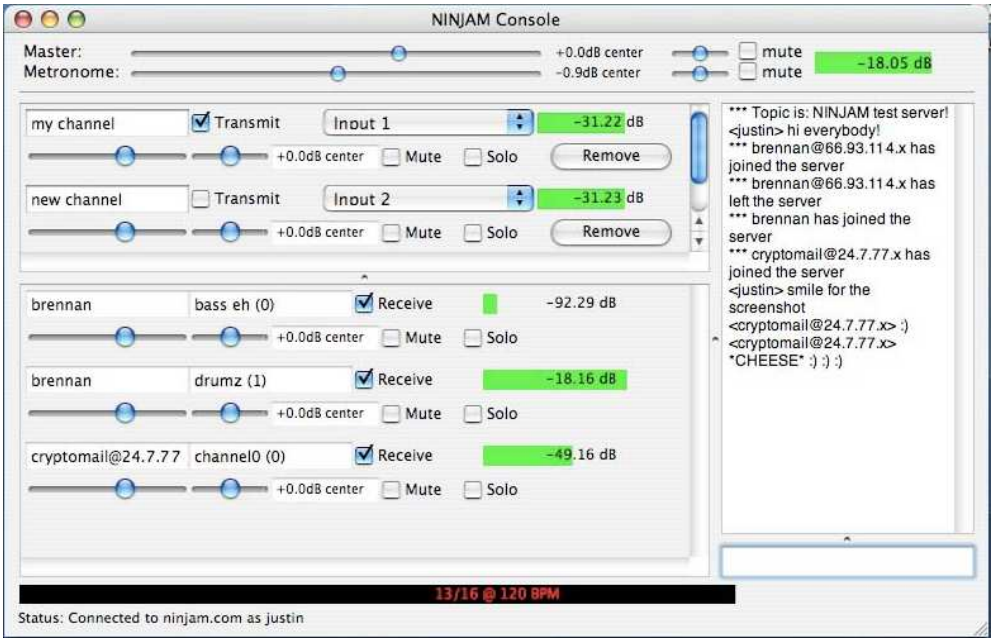
myvirtualband.com in which one user had to wait quite long for feedback from others. Also interesting is that in spite of the new representation of music as a circle rather then a line the learning curve is very low as experiments showed (see [NBK04]). A little weird is the way of commenting on done music, since drawing letters with the mouse seems to be a bit unconvenient compared to typing with a keyboard. A point that can be good as well as bad is the fact that there are no strict policies and everyone can changes everyones work. This enriches creativity but also endangers the final work if someone decides to destroy it. There is no undo provided to prohibit that, but of course users can see who placed which notes by their color. Last, the biggest flaw of daisyphone is that is is very limited in the possible resulting music. The length is predefined and the spectrum of the available sounds is not very large, too. Plus there is no possibility to save ones work without additional software.

5.4 Ninjam

Ninjam has been developed since the year 2005. It aims to support people creating and playing music collaboratively online via the internet [nin]. It also includes some features meant to fit users' needs, e.g. personal stream mixing, text-chat, record and playback data.

5.4.1 How does it work?

A Ninjam client application is required to connect to one Ninjam server, or, in another case, users can set up their own server. Then, Ninjam streams audio data compressed by OGG Vorbis audio technique to the server. Afterwards, the server distributes these audio streams to other participators. The system uses OGG Vorbis for its great low bitrate characteristics and performance [nin]. While retrieving audio streams, users are able to mix them according to their preference, or even remix it later. Furthermore, Ninjam can also save all uncompressed data and let user have a full quality remix after the jam.



5.4.2 Discussion

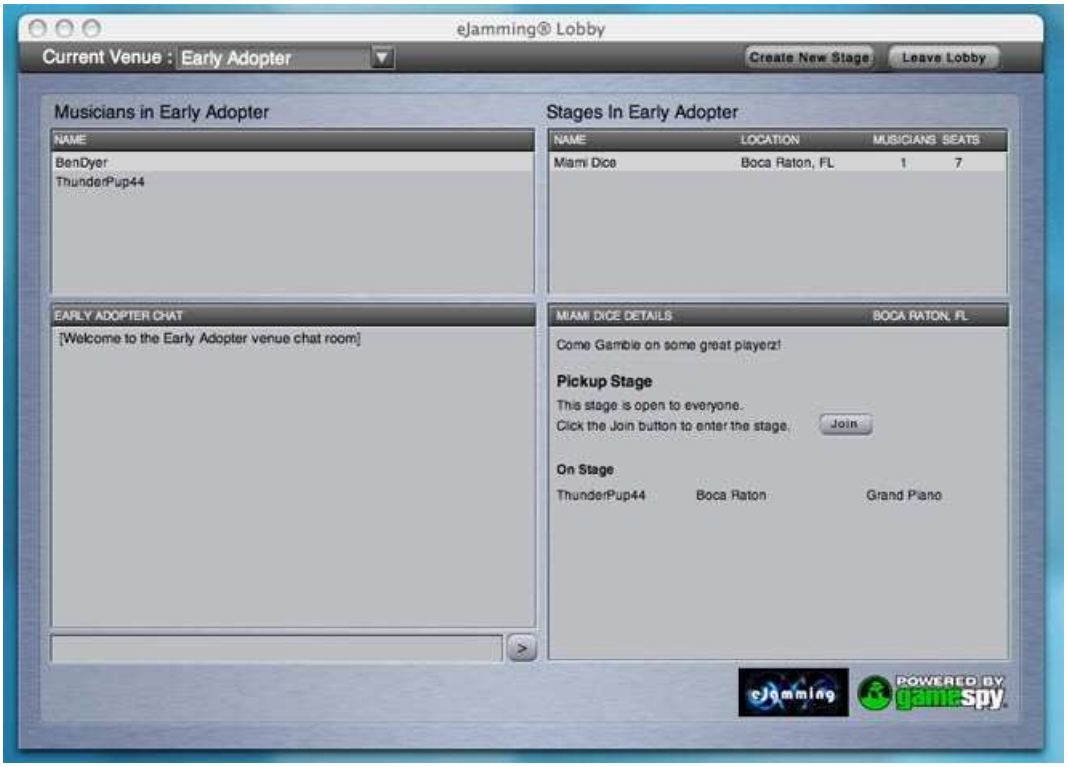
Since Ninjam provides the system supporting online improvisation and performance, real-time communication and synchronization might be a major concern in the system architecture. To fulfill these interactions, Ninjam comes up with a solution called "making latency much longer" [nin]. "The NINJAM client records and streams synchronized intervals of music between participants. Just as the interval finishes recording, it begins playing on everyone else's client. So when you play through an interval, you're playing along with the previous interval of everybody else, and they're playing along with your previous interval."[nin] For other aspects that systems should cope with, Ninjam provides text-chat which makes users able to communicate, evaluate each other's work, or settle improvisation/performance policies. Nevertheless, it misses some professional aspects; for example, musicians should be able to see each other while collaboration, or no mechanism to let users able to find new partners.

5.5 eJamming

eJamming is a commercial product, developed since year 2001. The product contains software technology from GameSpy Industries [gam]. With an "eJamming station", the name of the application itself, users are be able to connect to other musicians, and play together in sync by just plugging a MIDI-capable instrument to a computer, either Mac or PC.

5.5.1 How does it work?

After After installing eJamming, the step-by-step configuration pops up and helps users follow the way to complete setting all necessary configuration values. As users log in to their eJamming station, the station will connect to eJamming server represented to the user via a lobby window.



The lobby window acts as a chatroom, where people can meet to demonstrate their purposes, or just communicate with each other. Also, users can either create or join a stage, as a place to play music together. The stage provides privacy enough as well as it is sharable enough. For each stage, there are up to 16 MIDI channels that let you or other jammers select an instrument to plug into it; yet, the jammers have to get a permission to join from the leader of that stage first as they are called auditioning or reserved seat. Also, there is a copyright function to letting users view a copyright on each stage as well as set it for themselves. In addition, users are allowed to record data while jamming, or later users can playback, add local voice, as well as send this recording to friends.

5.5.2 Discussion

The purpose of the product is to support music collaboration; especially improvisation and performance. Admirably, the UI design is very user-friendly and also creatively represents all software activities as the terms in musicians' conception. The virtual space, the lobby window and the online stage, work similarly to the real world. For real-time communication, voice/text chat let users communicate to each other.
To handle with the latency, eJamming uses its technology and algorithms to delay the sound of users' instrument until receiving the data from other collaborators. Users can set their own latency policy to adjust local delay and tolerable delay limit; four options are possible: tight, regular, loose, and custom. Besides, it provides an advisor function for analyzing all internet conditions users have and give an advice [eja].

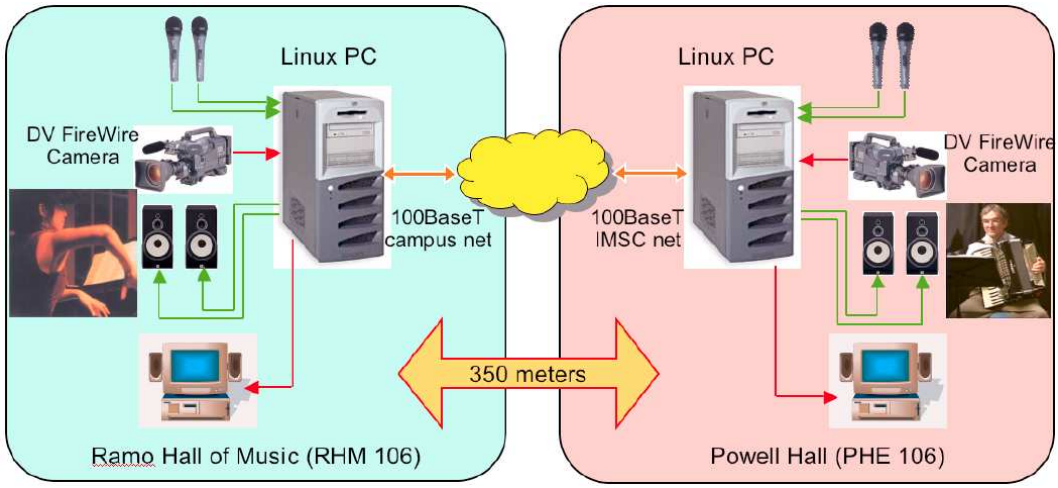
5.6 Distributed Immersive Performance

The last and probably most challenging system we will present is the so called distributed immersive performance. It started in June 2002 and is lead by A. A. Sawchuk, R. Zimmerman, E. Chew and others. Actually it is not a system but a whole research project on how musical performances could be done or presented from remote locations. That means that it also deals with having the performers and the audience in e.g. concerts located remotely and not only, as we look at it in this paper, the performers themselves. For this reason we will limit the representation of that project to one of their experiments that fits into our topic, the so called DIP v1.0 (see [EC04]).

5.6.1 How does it work?

In this setup two performers played a song from remote locations. The audience was co-colacted with one of them and together they could see the other performer on a large screen HD video and listen to her part via 10.2 immersive audio. On the other location

a monitor and an earphone were used to provide feedback about audio and video. The exact experimental setup is best displayed by the following picture. Note that the distance between both performers was not very large, we will discuss this later on.



Picture taken from [EC04]

The average video delay was 110 ms due to compression plus less than 5 ms network delay, the average audio delay was less than 10 ms for processing plus the same network delay. This made video abviously unusable for cueing a synchronous playing, thus the artists had to use audio only for that. They experienced that they had to scale back on spontaneity and could only compensate the delay by anticipating the other atrists actions. Even the co-location of the audience with one artist had impact on the playing, since it caused an imbalance in the artists control over the song.

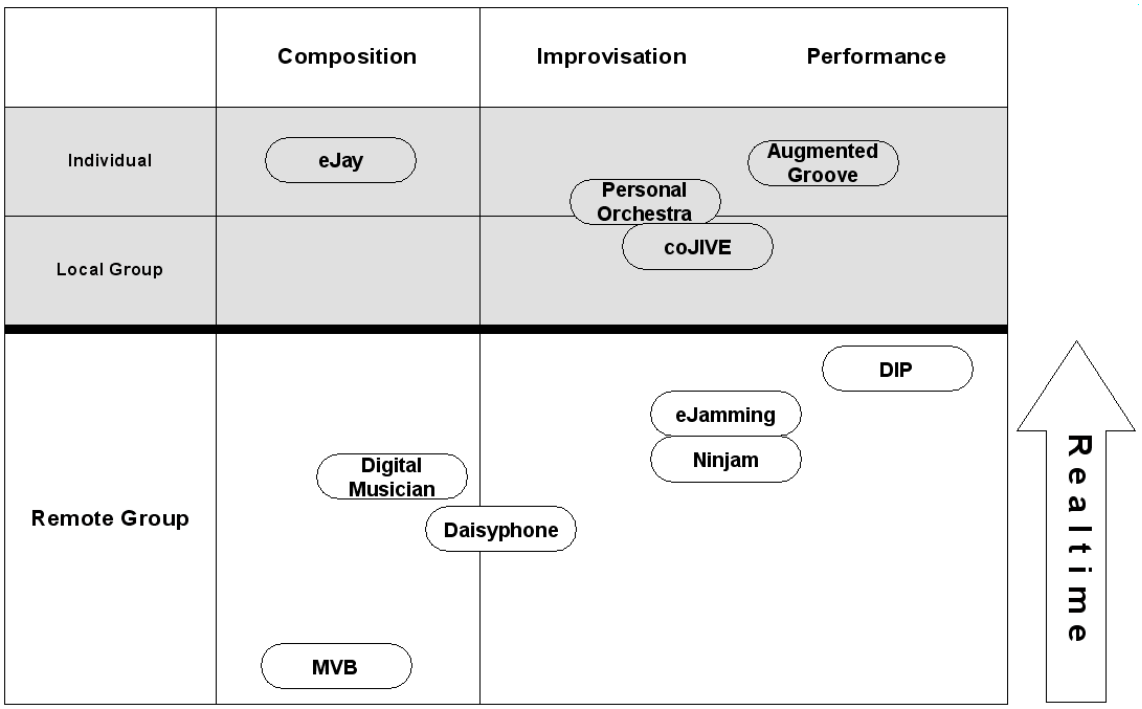
5.6.2 Discussion

For our topic most interesting is what problems this experiment illustrates for systems supporting remotely performing music. The most obvious thing is that of course the social interaction between the performers was possible through audio- and videoconferencing, even if the video could not be used for timecritical communication. Basically all communication problems we faced in the other systems can be solved with this, but the greatest problem actually gets more imminent because of this solution, the latency. Even in this close setup the video delay was quite high, simply because of the needed compression. Though we have seen other methods to deal with the latency, a virtual space that enables collaboration in real-time for people will always have that problem, and not all of the earlier seen solutions fit for all situations. DIP v1.0 shows that for a real remote performance to have the same quality as a traditional one, i.e. a direct *social* feedback from the remote performers any perhaps even the audience, things are needed that can't easily be realized without a high latency. Today these issues can perhaps only be solved with great limitations, like in this experiment the use of a very short distance and an intranet rather then the internet.

6 Conclusion

This paper described the overall concepts of remote collaboration on music and what problems can occur when designing a system for that area. It mentioned several example

systems to illustrate the three different possibilities one can have for collaboration on music and what problems can occur in each of these sections. The perhaps biggest issue almost all of the more complex systems have to deal with is the limitation due to latency. Although some systems only deal with composition most of them try to create some kind of virtual space in which collaborators can work together in real-time, spaces that also enable improvisation or even performances. Esspecially the last mentioned system, distributed immersive performance, shows how big the influence of latency can be. Because of this todays systems are not yet capable of competing with more traditional ways of collaborating on music, but they show interesting approaches and sometimes lead to ideas or projects that are at least usable and convenient now. Perhaps these ideas will grow in the future, when the technical infrastructure is more advanced. For the conclusion of our paper we want to present what we call the "Music-Collaboration Space", a graphic that is inspired from the more common known design space for input devices. Apart from the upper part, where non-remote systems for collaboration on music are listed just for completeness it puts each of the six presented systems in a two-dimensional space. The X-Axis illustrates the in the introduction mentioned linearity of composition, improvisation and performance, the three aspects of music collaboration. The Y-Axis represents the "realtimeness", i.e. the latency the system is designed for.



References

[Cen] Harmony Central. New service to let you make music over the internet. http://messe.harmony-central.com/Musikmesse05/Content/DM_Digital_Musician/PR/Digital-Musician-Net.html.

[dm] Digital musician. http://www.digitalmusician.net.

[EC04] A.A. Sawchuk C. Kyriakakis C. Papadopoulos A.R.J. Francois G. Kim A. Rizzo A. Volk E. Chew, R. Zimmermann. Musical interaction at a distance: Distributed immersive performance. http://www.interactivemusicnetwork.org, Sep 2004.

[eja] ejamming. http://www.ejamming.com.

[gam] Gamespy industries inc. http://www.gamespy.com.

[KS] Scott Mason Kelly Senecal. Myvirtualband.com. http://www.myvirtualband.com.

[MY04] Tatsuhiro Yonekura Masahiro Yoshida, Yuka Obu. A protocol for remote musical session with fluctuated tempo. In *Third International Conference on Cyberworlds (CW'04)*, pages 87–93, 2004. http://doi.ieeecomputersociety.org/10.1109/CW.2004.10.

[NBK04] P.G.T. Healey N. Bryan-Kinns. Daisyphone: The design and impact of a novel environment for remote group music improvisation. pages 164–170, 2004.

[nin] Ninjam. http://www.ninjam.com.

[Wel97] Markus Welzl. Netmusic: Echtzeitfähige konzepte und systeme für den telekooperativen austausch musikalischer informationen. Master's thesis, Dec 1997.