A Tabletop Groupware System for Computer-based Music Composition

Mauro C. Pichiliani and Celso M. Hirata
Department of Computer Science
ITA – Instituto Tecnológico de Aeronáutica
São José dos Campos, Brazil
{pichilia,hirata}@ita.br

***Abstract* - Tabletop can be used in groupware systems in order to both provide a rich interface shared workspaces and support work with co-located and remote users. Tabletop-based groupware systems can allow users to work together over a shared horizontal display. Music composition is an application that can exploit the features of tabletops. However, there is no current tabletop groupware approach that allows remote users to collaborate in music composition. This paper presents a discussion on how to promote collaborative music composition with tabletop groupware in a distributed context. In order to support the discussion, we have built a prototype to simulate instruments and to record the notes played into a modified music editing application. The prototype allows remote users to cooperate by playing music and recording the result of the composition. Potential applications of our prototype include music playing and training, computer aided composition, education, and recreational usage.**

***Keywords:* Tabletop, groupware, collaboration, music, composition.**

I. INTRODUCTION

The current advances in technology, the decreasing cost of PCs, and the availability of broadband technology open interesting possibilities for collaboration in many areas including music domain. In the music domain, many researchers [2,10,11] proposed different approaches to seamless integrate the latest computer technology with music in order to explore the possibilities that the technology offers, such as the ability to produce music in a simultaneous collaborative environment.

One of the main steps in music production is the composition. The music composition requires many efforts and practices that can be supported by technological features such as digital and virtual instruments [10], specialized database sounds, musical instrument digital interface (MIDI), and audio/music editing applications [2]. However, the current approaches do not address remote users collaborating simultaneously in the process of music composition when it is necessary to produce music in standard notation that comply with formal music concepts and virtual instruments. Also, the requirement of a specific digital interface between the computer and the instrument is a factor that limits the participation of musicians in simultaneous remote collaborative composition sessions.

In order to address the requirements of remote users collaborating simultaneously in the process of music composition with real and virtual instruments, we propose the use of a tabletop groupware integrated with a collaborative editing system (CES). Tabletop groupware systems allow co-present collaborators to work together over a shared horizontal display [19]. CES are multi-user collaborative editors designed to support users working in a common artifact at the same time from remote sites [3]. Although there are reports of conventional tabletop groupware successfully allowing co-located and remote collaborators to work [11, 24], previous research does not explore the possibilities of employing tabletop groupware as an interface for collaborative work that produces artifacts stored in a CES, such as the result of music composition represented by standard music notation.

This work discusses how virtual instruments and computer-based applications can promote collaboration among musicians connected over a network. In order to support the discussions, we built a tabletop groupware prototype and modified an existing music editing application as an approach to promote collaborative work whereby musicians can play and interact during music composition. The proposed prototype allows co-present and remote musicians to work together over a shared horizontal display at the same collaborative session with virtual and real instruments. It can also be employed to assist the teaching of formal music concepts, melodies, rhythms, musical expressions, and creative thinking by increasing the possibility of interaction between students and instructor.

The rest of the paper is organized as follows. Section II describes the concept of tabletop groupware, multi-touch devices, and some applications using this technology to support co-located and remote work. Section III contains a description of the computer and music integration that focus on virtual instruments, applications that support audio and music composition, and the related work in collaborative computer-based music composition. Section IV presents the observations of work practices, the collaborative and theoretical aspects involved in this work, implementation details of the tabletop groupware prototype developed, and two usage scenarios that illustrate how and where the proposed prototype can be employed. Finally, we conclude the paper and provide directions for future work in Section V.

Digital Object Identifier: 10.4108/ICST.COLLABORATECOM2009.8269
http://dx.doi.org/10.4108/ICST.COLLABORATECOM2009.8269

II. TABLETOP GROUPWARE

This section describes the details of the multi-touch technology and discusses interfaces, devices, and systems that support tabletop interaction. The section also presents co-located and remote groupware applications using tabletop with multi-touch interaction technology when it is employed to support collaboration during group work.

A. Multi-touch Technology

Multi-touch displays are interactive graphics devices equipped with sensing and tactile technologies for direct on-screen manipulation. With multi-touch interactions, multiple input pointers are input to the system in such a way that all of them can be controlled independently. Depending on the size of the display, multi-touch surfaces can allow multiple users to interact with the same display at the same time [14]. The Mitsubishi DiamondTouch [5], the Microsoft Surface [13], and the Rekimoto's SmartSkin system [21] are examples of well-established tabletops with multi-user and multi-touch interaction.

In order to establish a common hardware device for description and discussion, the rest of this paper considers the low-cost FTIR (Frustrated Total Internal Reflection) approach proposed by Han [8] as the base multi-touch interaction technology used in tabletop groupware applications.

B. Tabletop Groupware Applications

Digital tabletops are becoming common in the CSCW (Computer Supported Cooperative Work) research area. One reason for the interest is that tables are a natural setting for collaboration and provide many advantages for group work. In particular, digital tabletops allow for easy verbal and visual communication, simple awareness of presence, location, activity, and a strong shared focus for the group.

When tabletops are combined with interactive systems in order to support group work they are classified as tabletop groupware. According to Pinelle, Nacenta, Gutwin and Stach [20], tabletop groupware is defined as systems that allow co-present collaborators to work together over a shared horizontal display. The authors state that tables are a natural site for group work because of both their ubiquity in the real world and their physical characteristics to support coordination and communication, such as the face-to-face orientation of people around the table, the central location of work artifacts, and the use of direct touch to manipulate objects on the work surface.

Although most research efforts with tabletop groupware concentrate on co-located users sharing the same multi-touch interactive device, there has been a growing interest in systems and approaches that connect two geographically-separated tabletops together to support remote collaboration, as presented by the work of Tuddenham and Robinson [24].

Those remote tabletop groupware systems typically provide a large horizontal shared workspace in which remote collaborators see each other's interactions with virtual task artifacts, such as digital photos, documents, maps, windows, or sketches along with remote representations of the arms, faces, and the environment at the remote sites. In the same way as physical tables, tabletop groupware systems typically enable several collaborators to work concurrently by moving and manipulating virtual task artifacts using touch surfaces or styluses.

However, the main research efforts in co-located and remote tabletop groupware focus on awareness, coordination and concurrency aspects, such as fluid transitioning between individual and group work, coordination based on spatial partitioning, orientation and territory manipulation, and mediation of interactions over digital artifacts. There is few research efforts focusing on tabletop groupware systems that address the integration of the user's interaction with new or existing CES, which limits the type of work that can be produced by customized applications developed specifically to deal with multi-touch interaction.

Examples of remote and co-located applications for tabletop groupware systems include furniture layout [24], document review [23], remote drawing [22] and audio playing [11]. All the examples rely on customized applications designed for multi-touch interaction and they do not contain any integration with existing CES.

III. COLLABORATIVE MUSIC COMPOSITION

This section contains a summarized description of the computer and music approaches that combine real and virtual instruments. The section also presents applications that support audio and music composition followed by a discussion of the related work in collaborative computer-based music composition.

A. Virtual Instruments

Musical instruments are among the most complex and sophisticated machinery humans have managed to design, construct and master, and musical performance may probably be the densest form of human communication [11].

Amateur musicians, as well as professional composers and producers, use a pool of applications which try to melt, more or less seamlessly, the traditional model of the music playing with the ubiquitous and pervasive trends of the last twenty years of human computer interaction, such as traditional WIMP (Windows, Icons, Menus, Point and Click) interfaces, Drag & Drop, and Copy & Paste interactions. The applications that simulate real instruments are known as Virtual Music Instrument (VMI), which aims to provide a way to control parameters of sound notes in an expressive and artistically meaningful way. This requires a degree of synchronicity between a user action and its effect on the sound output with a short response, ultimately converging to real-time.

In traditional instrumental playing, every nuance, every small control variation or modulation (e.g. a vibrato or a tremolo) has to be addressed physically by the performer. Also, acoustic, electric, or electronic instruments force the performer to remain responsible, all the time, for all of the musical actions and details. This type of performance can be considered as the synchronous musical activity per excellence.

With VMI, however, the performer no longer needs to directly control some of the aspects to produce a sound as the

Digital Object Identifier: 10.4108/ICST.COLLABORATECOM2009.8269
http://dx.doi.org/10.4108/ICST.COLLABORATECOM2009.8269

audio now is produced by the computer. For instance, the effort required to blow the air inside some instruments is replaced by some form of interaction with the computer. As a result, the performer no longer needs to concentrate on all the details required to play an instrument, being able instead to direct and manipulate the audio output generated by the computer without worrying about internal details or audio production. On the other hand, playing with VMI can reduce the expressiveness of the song's interpretation from the performer, thus diminishing the artistic aspect of the music and possible turning the act of playing into a mechanical process.

B. *Computer-based Audio and Music Composition*

The process of formal music composition demands many resources and specialized knowledge. Formal musical notation is among the requirements for the composition of a music score, since it is the standard form to represent music. Notation lies at the very heart of teaching, learning, composing and playing music, being employed to many purposes, such as when a music needs to be registered for copyright reasons or when it is used as a supplemental material for teaching basic music concepts.

The industry created notation programs such as Finale, Cubase, and Rosegarden that have many boxes, symbols, tools, menus, and other interfaces widgets used to represent music with scores, clefs, staves, staffs, key and time signatures, and simultaneous tracks for each individual instrument. Those notation programs allow composers and musicians to choose between composition with formal notation, audio composition or a combination of these two modalities.

Aiming to focus the composition process at the music content and to incorporate the notes played by traditional instruments into music composition, the industry proposed a protocol to connect the computer with real instruments. The Musical Instrument Digital Interface (MIDI) is an industry-standard electronic communication protocol that defines each musical note in an electronic instrument such as a synthesizer, a keyboard, or an electronic drum, precisely and concisely, allowing electronic instruments and computers to exchange data with each other. The MIDI is also an open and well-known file format employed to store the formal music notation in music files that can be exchanged and played in compatible synthesizers and computer programs.

In practical terms, when an instrument is connected to a computer through a MIDI interface, via a physical MIDI or USB peripheral port, the computer automatically wrote the notes directly in the corresponding stave or track of the instrument in the music score while the musician is playing.

Although the MIDI interface facilitates the process of music composition, each instrument must be physically connected with a computer equipped with a MIDI interface and a notation program. This setting limits the participation in collaborative composition sessions to co-located musicians that have real instruments with MIDI interfaces.

C. *Related Work*

There are almost no traditional and just a few contemporary digital collaborative instruments available that allow remote and collaborative playing. Some traditional instruments, such as the piano, can be played by four hands although they were not primarily designed for that task. In recent years many musical instrument designers came up with the idea of creating instruments specifically for collaborative music playing [10].

Considering tabletop groupware, there are some tabletop virtual musical instruments that allow a flexible number of users, with no preconfigured roles, and simultaneous performers working, co-located or remotely, on independent audio threads as well as performers sharing threads available. The main example, the reacTable* [11], is a tabletop system that allows performers to share complete access to all the musical threads by moving physical objects on a table surface and constructing different audio topologies in a sort of tangible modular synthesizer or graspable flow controlled programming language.

When two or more reacTables* are interconnected through a network, thus sharing the same virtual space, performers can only move the physical objects on their corresponding surface, which may modify the shared audio threads by provoking interactions between the objects displaced across the tabletop. However, the reacTable* is a modular synthesizer that exchange only audio streams and cannot produce music notes as a physical or virtual instrument.

Bellini, Nesi and Spinu propose the MOODS (Music Object Oriented Distributed System) system [2], which is a synchronous real-time cooperative editor for music scores. The system consists of computer-based lecterns for the cooperative editing, visualization, and execution of music scores, specifically oriented to orchestras, music students and publishers. Although this system does provide a collaborative editor for music scores, its main focus is on music playing and editing and not on music composition with different instruments. MOODS is divided basically in two modules, which were designed for use in two separated electronic lectern positioned in front of the users: (i) DLIOO (Distributed Lectern Interactive Object-oriented), a set of single part lecterns for musicians that allows editing and visualization of single score parts; and (ii) MASE (Main Score Editor), the lectern that displays the main score to directors, conductors, and directors of the chorus in order to show, visualize, and modify the score.

The MOODS systems do not allow the use of VMI or any other instrument that has a MIDI interface, has a tightly coupled user interface, is designed for co-located use and is capable of handling more than 300 classes of music symbols and their abstractions. Our proposed prototype, described in the next section, focus on music composition with real and virtual instruments co-located and/or distributed, has a loose coupled interface and employs all the features of an already existing score music editor. Also, our prototype does not characterize the type of user of the system and does not contain a concurrency control mechanism.

One recent approach to record collaborative musical performance in real-time by exchanging MIDI and audio data from remote composers is presented by the DML (Digital Music Link) plug-in [6], an initiative provided as a service that is integrated with the web site digitalmusician.net, an online community of world-wide musicians and composers. The

Digital Object Identifier: 10.4108/ICST.COLLABORATECOM2009.8269
http://dx.doi.org/10.4108/ICST.COLLABORATECOM2009.8269

system is a plug-in that exchanges specific parts of MIDI protocol captured by commercial products such as Logic Studio, Pro Tools, Digital Performer, Sonar, and others. Whereas this plug-in does allow the remote collaboration during music composition, it does not provide the collaborative remote editing features for the standard music notation elements recorded for each remote composer playing with a VMI or a real physical instrument.

Another recent approach for collaborative computer-based music collaboration is Noteflight [16], an online application that lets internet users display, edit, print, share, and play back music directly in a web browser using formal music notation. Combining the asynchronous style of collaboration and the low requirements needed to use the editor, Noteflight has the potential to create many collaborative compositions. However, the editor provided by the Noteflight web site does not allow the integration of real instruments and VMIs in a synchronous real-time fashion during music composition.

IV. TABLETOP GROUPWARE FOR MUSIC COMPOSITION

This section presents the observations of work practices conducted before the gathering of requirements of the tabletop groupware system for remote music composition. The discussion of the observations is followed by the presentation of the collaborative and theoretical aspects used to guide this work. Next, the section presents the implementation details and the architecture of the prototype tabletop groupware system developed, a list of requirements addressed, the design technique used and the features implemented. Finally, the section discusses two usage scenarios that illustrate how and where the proposed prototype can be employed.

A. Observation of Work Practices

We made interviews with musicians and observed the work practices performed when they are composing with real instruments and notation programs before designing and implementing our prototype. Although we did not conduct any formal experiments, we obtained valuable information from the interviews with the musicians and the observations of the major factors that comprise not only task activities but also processes and behavioral patterns. More specifically, we observed and talked with the musicians when they created music represented by formal notation on a notation program that allows the editing and capture of music scores when a keyboard was connected to a MIDI interface of the computer.

We observed that a musician usually works alone with the notation program and must know very well the editing capabilities of the notation program in order to compose and represent the music in formal notation. Due to this specialized knowledge of the program and the type of interaction that an instrument connected with the computer provides, we note that composition with the computer is, in general, a solo task, in much the same way as when a musician is composing with pen and paper. Nevertheless, we also found that music represented by standard notation is composed in two general scenarios: (i) when the composer does not have a song and must create the musical score from scratch, i.e. when composing a sound track for a movie or a video game; and (ii) when the composer already has the song and must create the musical score, i.e. represent the music in formal notation in order to register it for copyright or teaching reasons.

In the first scenario the composer starts the production of the music alone using his/her creativity and artistic skills and composing the melody for one instrument at time. This means that each set of notes for every instrument in the music is played with the keyboard due to its ability to simulate sounds of many instruments and the MIDI interface. The set of notes played is then stored in a specific track of the notation program as the melody is played. The composer edits one track at time in order to adjust the notes, tempo, timbre and other technical details with formal music notation. Then, the separated tracks are written in a MIDI file and played together so the composer can adjust the details, and listen to the combination of all instruments and the melody composed. Since this type of composition is associated with another artistic work, such as a movie, it is played and recorded latter in performances with real instruments.

In some cases the main composer asks to another musician to come up with the voice for some instrument, or another version of the melody, or even an entire new idea for the music. Although the discussions between composers are characterized by an active face-to-face or remote communication, the tracks are composed separated in an asynchronous fashion. Eventually the main composer receives the different versions of the tracks and then coordinates alone the merging of all the work into a final musical score.

The collaboration is more restricted in the second scenario where the composer already has a recorded song and must create a musical score from it. The composer starts working after a brief period of discussion with the musicians that originally played the song, if some of them are available, in order to obtain general information. Then the composer works alone by listening to the song and applying the same processes described in the first scenario above, with the advantage that now the composer has an existing song to guide himself/herself during the composition. After producing the musical score, the composer compares the MIDI and the recorded version in order to adjust the formal notation elements of each track through the editing features of the notation program. The music score is then printed and is ready to go to the next phase, i.e. the registration office or followed to the original music players or the musical producer.

In either scenario, even when more than one person is working asynchronously on different tracks of the same music and producing several versions of them, eventually one composer is chosen to somehow coordinate alone the merging of all the versions into the final musical score.

Although we observed situations where the composer works alone in the two scenarios described above, we try to understand if the collaborative composition would be valuable. Some of the musicians agreed that the contributions, suggestions and opinions on their work can lead to a better result, while a few others strongly prefer to continue doing their composition alone arguing that they could work better without interruption or that their freedom for artistic expression

Digital Object Identifier: 10.4108/ICST.COLLABORATECOM2009.8269
http://dx.doi.org/10.4108/ICST.COLLABORATECOM2009.8269

could be compromised. In either case, the music undergoes modifications according to the contributions and suggestions of all the involved musicians before going to the next phase.

In order to evaluate and explore remote synchronous collaboration while composing music, we account for the resources and efforts required to schedule, conduct, and document the meetings where the music is being composed. In the situations where more than one composer is working in the same music we conjecture that remote collaboration can both produce results as good as face-to-face and reduce the resources and effort required to conduct physical meetings. The base for this conjecture is the assumption that this setting can lead the participants to be remotely engaged into collaborative composing sessions, thus giving them the ability to interactively explore and refine the original music melody on a remote collaborative environment.

However, we observed many difficulties to coordinate and plan the tasks when the composers are using only the social protocol, probably due the artistic nature of the task. Because of that, we believe that audio/video conferencing features can be helpful to establish the social protocol for remote composers and help them coordinate their actions. Our assumptions is based on that fact that audio/video conference features employs virtual meetings that can create a sense of shared space and support verbal and non verbal forms of communication, thus allowing an easier creation of different versions of the tracks by more than one member at the same time and serving as a valuable option to coordinate the activities of geographically distributed composers. Finally, the production of music in parallel tracks and the further join of them can accelerate the process of music composition, thus reducing the resources and efforts allocated.

B. *Collaborative Theoretical Aspects*

In order to structure the types of collaborative support to be addressed by our approach, we first analyzed the Gutwin's framework for the mechanics of collaboration [7]. The framework describes low level actions and interactions that small groups of people do if they are to complete a task effectively. Basic actions include communication, coordination, planning, monitoring, assistance, and protection. The underlying idea of the framework is that while some usability problems in groupware systems are strongly tied to social or organizational issues in which the system has been deployed, others are a result of poor support for the basic activities of collaborative work in shared spaces. The basic activities are used to guide the elicitation of the collaborative aspects and requirements addressed in our work, although we recognize the importance of the social and organizational issues.

Since our approach suggests the collaboration of co-located and remote composers interacting with local or distributed tabletop groupware system and with real instruments connected to computers by a MIDI interface, we frame the type of collaborative system we propose as tabletop groupware and, in a broader sense, also as Mixed Presence Groupware (MPG). MPG is defined by Tang, Boyle and Greenberg [22] as a synchronous groupware that supports both co-located and distributed participants working over a shared visual workspace in real time. The main characteristics that MPG systems should provide are virtual embodiment, conversational and presence disparity, and consequential communication. All of these characteristics are supported in our prototype by the implementation of the requirements described in the next section.

After studying how people work together over a table Pinelle, Gutwin and Subramanian [19] proposed the concept of highly-integrated collaboration, which covers many tasks that are commonly carried out around tables, such as brainstorming activities, group design tasks, and project planning. In each of these activities there is a high level of interdependence between group members supported by the intense interaction among them.

We believe that in our proposal approach for collaborative composition it is possible to achieve highly-integrated collaboration because our approach allows group members to divide the task and work individually, therefore allowing them to come together to work in a highly integrated fashion for a short period of time when they merge their work. Most importantly, while composing in collaboration, other remote musicians can verify the melody by themselves just after they produce it, which means that the listening-composing-playing loop found on solo compositions scenarios can include collaborative features.

C. *Implementaion of Collaborative Music Composition*

Before implementing the prototype that allows collaborative music composition it is necessary to specify the collaborative requirements addressed. The core requirements usually found in real-time synchronous collaborative applications, and derived from the Gutwin's framework for the mechanics of collaboration, include: (i) communication, used to exchange ideas, discuss, learn, negotiate and to make decisions; (ii) cooperation, which is needed to organize the group, avoids loss of communication and cooperation effort, and prevents the work to be made out of order and out of time; (iii) awareness, which allows a remote participant be aware of the actions of the group and the group itself; and (iv) sharing of information, responsible to avoid repetitive effort and to ensure that all participants of the group have the same information.

The communication, cooperation, awareness and sharing of information requirements usually are not found in single-user application. With the implementation of these requirements we conjecture that the application provides effective means for the user to perform tasks with other users by sharing the work, dividing responsibilities, aggregating new ideas and contributing to the music being composed.

We divide the implementation work into three main steps: (i) the development of a tabletop groupware application that allows the manipulation of the virtual instruments in the multi-touch surface; (ii) the modification of an existing note editing application in order to support multiple users composing; and (iii) the integration of real instruments into the collaboration through the MIDI protocol.

The tabletop groupware application was developed with the C# language of the Microsoft's .NET Framework and received

Digital Object Identifier: 10.4108/ICST.COLLABORATECOM2009.8269
http://dx.doi.org/10.4108/ICST.COLLABORATECOM2009.8269

the name CoComposer. This application was developed with the multi-touch interaction style, thus it does not have common interface widgets such as buttons, menus, list, and others. Since the interaction can be made by multiple input points from different users similar to the manipulations of artifacts on a table, we designed the virtual instruments to appear as they are seen by someone who is looking from the top. Also, the instruments can be moved and resized by the users at any time.

The multi-touch features in the application are implemented by receiving UDP packets via a conventional IP network using the TUIO (Tangible User Interface Objects) [12] protocol, which is based on Open Sound Control (OSC) [25] messages. This is the default mechanism to program applications based on the Touchlib [15], the development kit that allows the development of multi-touch application without focusing on the specific details of the multi-touch hardware. CoComposer also employs the IrrKlang sound library [1] to play the audio files for the notes of the virtual instruments.

In order to address the requirements of collaboration in our prototype we developed interface widgets that represent the local input points, called blobs, and the remote representation of them, named telefingers. Also, CoComposer shows semi-transparent versions of the virtual instruments to remote users. CoComposer has a participants list, contains an access control mechanism to collaborative sessions, and provides audio/video conferencing features. We also create a workspace that allows the sharing of information, so when a user plays a note the corresponding parts of the instrument, and its remote representation, change their color. For instance, when a user touches a key in the virtual keyboard, CoComposer plays the sound of the note and changes its color to the assigned color of the user in the collaboration, thus providing awareness of the user action to other participants.

The second step in the implementation work has the modification of an existing note editing application in order to support multiple users composing at the same time with formal music notation. After reviewing several different tools and applications we choose an existing open source tablature editor named TuxGuitar [9] to receive collaboration features. The choice is due to the availability of the source code, the support of formal and tablature music notation, the organization of components in a specific architectural style, several features to manipulate MIDI files, and its good assessments received from the specialized press.

In order to implement the collaborative requirements on TuxGuitar we use the Mapping of Components, a design technique proposed by Pichiliani and Hirata [17], which is a conceptual design technique that maps the main components of a single-user editor to collaborative components that promote the synchronous editing of a shared document. The mapping aims to help developers in the analysis and design discipline during the elaboration phase of the development of groupware to adapt a single-user application to become a multi-user collaborative editor. This type of editor is classified as a Collaborative Editing System (CES). We chose the Mapping of Components because the approach has low technical requirements, allows high flexibility, and can be used with others approaches [18].

The Mapping of Components requires that the existing application be logically organized into components that behave according to roles of the MVC (Model-View-Controller) architectural style. MVC is a widely used architectural style that separates the data underlying the application (the model) from the input handling code (the controller) and the display maintenance code (the view). The availability of the source code of the application is another requirement for the use of the Mapping of Components, since it generally demands modifications of the source code to change the services provided by the components according to the collaborative requirements.

TuxGuitar was developed in Java and is composed of subsystems that have many components, including: (i) a Song subsystem, which has components that store the internal representation of the music notation, such as notes, measures, tracks, scales, beat, tempo and others; (ii) a subsystem responsible to analyze and validate the music elements inserted in the music score in order to comply with the key and time signatures used for each track; (iii) a subsystem that contains components for the sound engine used to play the MIDI files; and (iv) a subsystem called GUI that contains generic user interface components that handle the user interaction, which include buttons, forms and menus. TuxGuitar also employs third-part external components in order to support different file formats, such as PDF, MusicML, TBL and others. The entire application is composed of 450 source code files containing 390 classes. Using the Mapping of Components design technique, 16 classes were modified and 4 new classes were created.

The Model is implemented by the components of the Song subsystem. The components contain data structures that store all the elements of formal music notation used to compose music in TuxGuitar. The Control and View are implemented through the user interfaces components provided by GUI subsystem, which rely on events to communicate the data from the user input to the Model and to visual user interface controls. After applying the Mapping of Components we name the application CoTuxGuitar.

In order to provide interface between CoComposer, CoTuxGuitar and real instruments it was necessary to implement the MIDI protocol. Since the design technique employed requires a central server that connects all the remote users, the programming required to handle the MIDI protocol messages were implemented in CoComposer in the server, which received the name CoMusic Server. All the software network infra-structure necessary to enable the communication between CoComposer and the CoMusic Server was implemented using the GT Toolkit [4].

The design of the system was created according the technical details and requirements of the approaches employed. The architecture that connects CoComposer, CoTuxGuitar, and CoMusic Server must implement the hybrid model, since this is requirement of Mapping of Components [18] and all the components that behave according to roles of the MVC must remain inside the applications. The design of CoTuxGuitar is equal of the design of TuxGuitar because the minor modifications made on some components do not changed the

Digital Object Identifier: 10.4108/ICST.COLLABORATECOM2009.8269
http://dx.doi.org/10.4108/ICST.COLLABORATECOM2009.8269

design of TuxGuitar. The design of CoComposer is based on the guidelines of the GT Toolkit, the programming model of the Touchlib development kit, and is complemented by the features of the IrrKlang sound library and a generic implementation of the MIDI protocol. The entire programming infrastructure employed to receive and send the messages that handle the distributed communication of notes played, awareness information, sound cues and internal protocol commands were implemented by an event handling approach.

Fig. 1 presents an example of how the applications interact in a remote collaborative composing session that has four users playing simultaneously.

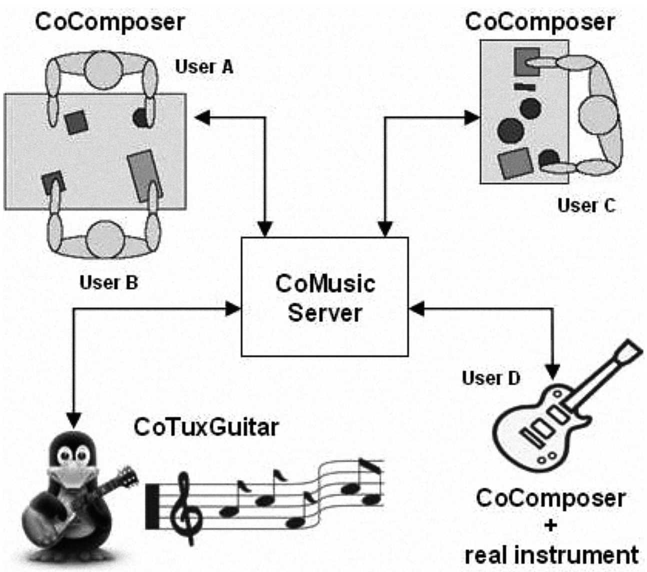

Figure 1. Example of the a remote collaborative composing session: two tabletop interfaces running CoComposer, a real instrument with a MIDI interface, and CoTuxGuitar.

In Fig. 1 users *A* and *B* are interacting with CoComposer in the same tabletop, but with two different virtual instruments. User C is alone at a remote site playing with another virtual instrument in CoComposer and with a second tabletop. User D is at a different remote site playing a real instrument, represented by a guitar in the lower right side of Fig. 1, and it is collaborating with the others by its MIDI interface and an instance of CoComposer. Also, CoTuxGuitar is recording all the notes played by each participant in separated tracks of the same musical score.

From the elements in Fig. 1 it is possible to note that all the communication is bi-directional, even where the MIDI interface and CoComposer are used in conjunction with the real instrument. Although only one instance of CoTuxGuitar is shown in Fig. 1, all the participants can run a copy of this application to see and edit, in real time, the notes stored in all the corresponding tracks. The CoMusic Server is coordinating and transmitting the correct messages between the instances of CoComposer and CoTuxGuitar as they are playing with their own instruments.

When a user wants to start composing with CoComposer she/he has to create a new session or join an existing one. In either case the user must choose one virtual instrument or, alternatively, plug a MIDI interface in the computer to participate in the session by playing with a real instrument. Although our prototype currently only supports virtual keyboards and drum sets, it is possible to choose between a set of pre-configured sounds for the notes of the virtual instruments. In addition to that, it is also possible to connect virtually any real instrument that has a MIDI interface, thus expanding the sound possibilities available to the users.

After establishing a connection with the CoMusic Server each user can open an instance of CoTuxGuitar to see and edit the notes that his/her instrument is producing, which are stored separated in different parallel tracks of the music score. Also, the sound of every note played by any instrument is reproduced to all users and every modification in the notes already played and stored in the track is replicated to all instances of CoTuxGuitar through the server. Any user at any time can disable the audio output of any instrument of the collaborative session if he/she wishes to work separated from the others.

By representing the notes produced with the multi-touch interface of CoComposer and the notes played in a real instrument with the formal music notation supported by CoTuxGuitar, three types of interaction with the system can happen: (i) with the shared multi-touch workspace of CoComposer; (ii) with a physical instrument; and (iii) directly with the music notation. Those three types of interaction with the system and all the possible combinations of them have the potential not only to promote collaboration among remote or co-located users with different skills and abilities but also to integrate the result of the interactions with interfaces and devices that are usually employed in a separated context and do not contain any type of integration during music composition.

In order to address the cooperation and coordination requirements we implemented a virtual metronome, which is a device that produces a regular audio pulse to establish a steady tempo in the performance of music.

We designed CoComposer to support only synchronous collaboration and assign a single track that stores the notes of a single instrument to each participant in the session. Although we implemented the metronome for coordination purposes while playing and limiting each user to edit only the notes of his/her assigned track, we did not design and we did not develop any specific approach to support the versioning of the composition, the asynchronous distribution of the music and the solo work without participating in a collaborative session.

In order to provide awareness information, all the users can see each other finger's positions and the keys pressed in the virtual instruments, which provide embodiment and presence disparity. The interface of CoComposer shows a stylized symbol to represent the participation of real instrument in the session. Communication can occur via the knowledge of the users' actions and through the conversation that the audio/video conferencing features provide. Fig. 2 presents the CoComposer interfaces for two tabletops and one real instrument described in the example presented in Fig. 1, four users are interacting in a collaborative composition session of a music score.

Digital Object Identifier: 10.4108/ICST.COLLABORATECOM2009.8269
http://dx.doi.org/10.4108/ICST.COLLABORATECOM2009.8269

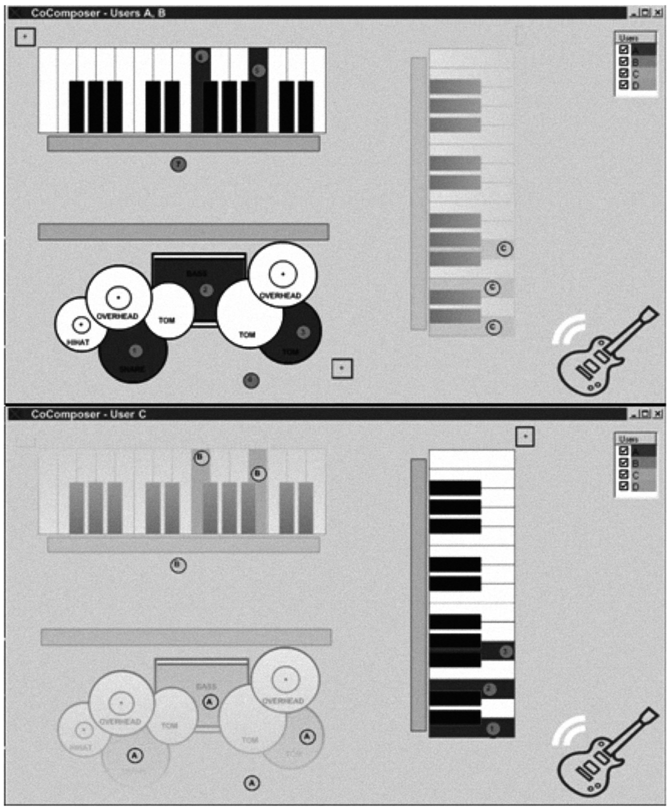

Figure 2. Shared workspaces, finger's representation and telefingers of three composers, *A*, *B* and *C*, on a collaborative composing session of a music score in CoComposer. The upper workspace is the users *A* and *B* surface and the lower workspace is the user *C* surface. User *D* is represented by the guitar.

In the upper shared workspace of Fig. 2 the user *A* sees his/her three fingers representation, known as blobs by the system, positioned upon the snare, the bass and the one of the toms of the virtual drum. The elements of the drum where the user *A's* blobs are located have a different color of the rest of the virtual drum because CoComposer is playing its sounds. The forth blob of user *A* is outside the range of the drum. Likewise, two of the three blobs of the user *B* are upon the keys of the virtual keyboard positioned in the upper left corner of the upper shared workspace.

The blobs of the users *A* and *B* are represented as telefingers, depicted as circumferences with the names of the users inside, in the lower shared workspace of Fig. 2. The telefingers and the instruments have the same location in the two workspaces, however the instruments are semi-transparent when the users who are playing them are remote.

User *C*, whose shared workspace can be seen in the lower part of Fig. 2, has his/her three blobs inside the keys of the virtual keyboard positioned on the right side of the workspace, which are represented by his/her telefingers in the users *A* and *B* workspace. The positions of the drum and the virtual keyboards were changed by the users so that the instruments are positioned in front of them. To change the position of the instrument the user has to drag and drop a rectangular bar positioned on top of the instrument and to change the size, the user has to move the blob upon the button with a plus sign located on the right bottom side of the instrument.

Both shared workspaces show the icon of a guitar on the right lower side of the workspace representing the user that is participating in the collaborative composing session with a real guitar connected via a MIDI interface. The participants list, and the check box that disables the sound of each user, are also visible on both shared workspaces on the right upper side. Fig. 3 presents the four tracks from CoTuxGuitar that contain the notes played by the users during the interactions with the instruments shown in Fig. 2.

The four tracks of Fig. 3 represent the first measure that contains the sequence of the notes played by the virtual drum (user *A*), the top left keyboard (user *B*), the right side keyboard (user *C*), and the guitar (user *D*) in the collaborative session of Fig. 2, respectively. Each track of the music score of Fig. 3, which contains a pentagram, a treble clef, a time signature and different color notes for each user, can be edited by any user that has an instance of CoTuxGuitar connected in the same collaborative session.

For instance, let us assume that the guitar player (user *D*) wishes to change the notes recorded in order to transpose the melody, i.e. change the position of the notes in the pentagram while maintaining the sequence and duration of them. User *D* can select the sequence of notes stored in his/her track and change them by using a command in CoTuxGuitar. The changes are automatically replicated to other users, who may be working in a different measure of the music score.

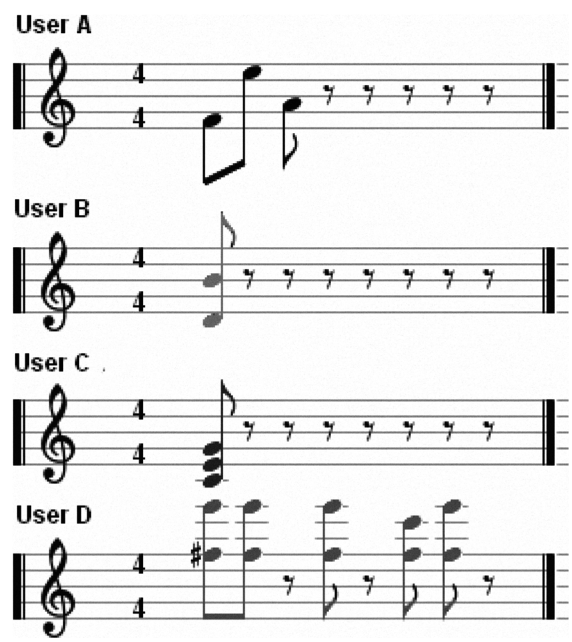

Figure 3. Four tracks representing the notes played in the distributed collaborative composition session with four users.

D. Usage Scenarios

Although the proposed system is not a complete suite of applications for music composition, it is a suitable alternative to create and experiment musical ideas that require formal notation. To illustrate how our proposed prototype can be used to assist the work of composers we consider the following scenario.

Digital Object Identifier: 10.4108/ICST.COLLABORATECOM2009.8269
http://dx.doi.org/10.4108/ICST.COLLABORATECOM2009.8269

A composer needs to create a theme song for a specific scene in a movie. After watching the scene he/she has the idea of using a duo of two instruments that he/she does not have at hand: a harmonica and a banjo. The composer wants to use those instruments because he/she knows that the combination of those instruments' sounds match and enhance the comic feeling that the scene transmits to the audience.

While the composer knows how to play the harmonica, he/she does not know how to play the banjo and how to combine the sounds of the two instruments together in a suitable melody. The composer can start a remote collaborative session with his MIDI keyboard playing the sound of the harmonica and ask another remote musician to join the session by connecting his banjo to a MIDI interface. In this scenario the composer and the banjo player can work together in order to compose a melody for the scene even if the composer is miles away from the banjo player. The banjo player can show to the composer what type of sound, volume, timbre and possible combinations of chords and velocity his instrument can provide. With this information the composer can employ all his/her harmonica skills with the expertise of the banjo player in order to produce the melody for the movie scene with the possibility to edit the musical score as they are playing during the composing session. Besides working together in the composition, the composer and the musician have the opportunity to learn about each other's instruments, thus increasing the music knowledge of both.

The educational domain is other possible use of CoComposer, CoTuxGuitar and CoMusic Server. Let us assume the scenario where a teacher is explaining basic music concepts to students attending a music class. The teacher wants to use different instruments and sounds in order to explain the differences between the instruments and how the formal music notation represents the music played by them. First he/she explains the basic elements of music notation: the clef, tempo, notes, measures and others. Next the teacher can demonstrate how the notes played in the instruments can be represented in formal music notation along with other music concepts, such as melodies and rhythms.

If the teacher has access to physical instruments he/she can play them to the students. Otherwise, the teacher can simulate the instruments on a low cost tabletop, which can help the students to interact on a table while learning about music concepts and the instruments' sounds. If the teacher has more than one tabletop it can divide the students in two or more groups and ask them to compose a melody in collaboration. Or the teacher can propose to the students to start new collaborative sessions by asking them to create and compose their own music directly in the computer, thus stimulating imagination, creativity and artistic skills. If needed, the teacher can assist the students while they are composing the music in the notation program. To complement the lesson presented in the school, the teacher can also suggest that the students use the collaborative composition software at home with their group partners, thus increasing the interaction while doing group homework.

To stimulate the learning process the teacher can promote a festival where the composed songs will be played in a collaborative concert. In this event the teacher can coordinate with other teacher of other school in order to synchronize the presentation of the music scores if the two schools have an internet connection and have access to tabletops and instruments with MIDI interfaces.

V. CONCLUSIONS & FUTURE WORK

In this paper we discuss how computer-based musical applications that support composition can promote the collaboration among musicians connected over a network. We described the implementation of collaborative features in an existing application for formal music notation and the design of a new tabletop groupware for music playing. The combination of collaboration, music playing and composition with formal music notation results in a collaborative system that allows users located in geographically distributed sites cooperate by playing, composing, exploring, listening, and modifying music scores interactively with immediate audio and visual feedback.

The multi-touch technology used to support tabletop interaction was discussed, followed by a description of the main applications that employ this technology to support co-located and remote work. Next, we presented a summarized description of the computer and music integration that focus on virtual instruments, applications that support audio and music composition, and the related work in collaborative computer-based music composition.

We also presented the findings from the observation of work practices and typical scenarios found when musicians are composing, followed by a discussion about the collaboration and theoretical aspects that can arise in this scenario. Then we listed the communication, cooperation, awareness and sharing of information requirements implemented in the new tabletop groupware prototype named CoComposer. We also implemented modifications into an existing music notation application in order to produce a prototype called CoTuxGuitar that allows the collaborative manipulation of the elements of a musical score when used in combination of CoComposer and a central server called CoMusic Server.

The prototype presented is operational. However, we have performed only limited testing of how well the users can employ the application to compose music. Future work includes the evaluation of the prototype with users in a collaborative context. The user study can also provide more information on how the participants communicate and coordinate their activities when they are creating music in collaborative sessions that have virtual and real instruments.

The lack of asynchronous support and a concurrency control mechanism as well as the limited number of users and instruments supported are the main limitations of the prototype developed. In the future work we aim to address these issues considering the context of music composition and previous work with collaborative text and object editing.

Besides presenting a new context for the learning of music notation and composing, this paper discussed the strategy of sharing musical ideas through formal music notation and group interaction, instead of composing and playing music alone. The paper goes some of the way towards proving that interactions

Digital Object Identifier: 10.4108/ICST.COLLABORATECOM2009.8269
http://dx.doi.org/10.4108/ICST.COLLABORATECOM2009.8269

when composing and playing can be more effective than solo interpretation and composition of music. By using groupware systems for the interaction of composers and music learners we argue that it is possible to make the musical education more attractive to students.

The availability of a tabletop system that connects virtual and real musical instruments to the computer can encourage the community of developers to support collaboration requirements in their designs therefore increasing the number of applications that combine multi-user and multi-touch interaction with collaborative features in other areas that support the production of artistic work.

REFERENCES

[1] Ambiera e.U. *irrKlang - an audio library.* <http://www.ambiera.com/irrklang/>, July 2009.

[2] P. Bellini, P. Nesi and M. B. Spinu. "Cooperative visual manipulation of music notation". In *ACM Transactions on Computer-Human Interaction* 9(3): 194-237, September 2002.

[3] D. Chen, and C. Sun. "A Distributed Algorithm for Graphic Objects Replication in Real-time Group Editors". In *Proc. of the International ACM SIGGROUP conference on Supporting Group Work*, pp. 121-130, 1999.

[4] B. de Alwis, C. Gutwin, and S. Greenberg. "GT/SD: performance and simplicity in a groupware toolkit". In *Proc. of the 1st of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, pp. 265-274, 2009.

[5] P. Dietz, and D. Leigh. "Diamondtouch: a multi-user touch technology". In *Proc. of 14th annual ACM symposium on user interface software and technology*, pp. 219-226, 2001.

[6] Digital Musician GmbH. *Digitalmusician.net.* <http://digitalmusician.net>, July 2009.

[7] C. Gutwin, and S. Greenberg. "The Mechanics of Collaboration: Developing Low Cost Usability Evaluation Methods for Shared Workspaces". In *Proc. of the 9th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*, pp. 98-103, 2000.

[8] J. Y. Han. "Low-Cost Multi-Touch Sensing through Frustrated Total Internal Reflection". In *Proc. of 18th annual ACM symposium on user interface software and technology*, pp. 115-118, 2005.

[9] Herac Modern Solutions. *TuxGuitar - open source tablature editor.* <http://www.tuxguitar.com.ar/>, July 2009.

[10] S. Jordà. "Multi-user Instruments: Models, Examples and Promises". In *Proc. of the 2005 conference on New interfaces for musical expression*, pp. 23-26, 2005.

[11] S. Jordà, and M. Kaltenbrunner, G. Geiger, and R. Bencina. "The reacTable*". In *Proc. of the 2005 International Computer Music Conference*, pp. 579-582, 2005.

[12] M. Kaltenbrunner1, T. Bovermann, R. Bencina1 and E. Costanza. "TUIO: A protocol for table-top tangible user interfaces". In *Proc. of the 6th international workshop on gesture in Human-Computer Interaction and Simulation*, 2005.

[13] Microsoft Corporation. *MS surface.* <http://www.microsoft.com/surface/>, July 2009.

[14] L.Y.L. Muller. "Multi-touch displays: design, applications and performance evaluation". *Master's Thesis*, University of Amsterdam, 2008, pp. 96.

[15] The Natural User Interface Group. *Touchlib - A Multi-Touch Ddevelopment Kit.* <http://www.nuigroup.com/touchlib/>, July 2009.

[16] Noteflight LLC. *Noteflight - Online Music Notation.* <http://www.noteflight.com>, July 2009.

[17] M. C. Pichiliani, and C. M. Hirata. "A Guide to Map Application Components to Support Multi-user Real-time Collaboration". In *Proc. of the 2nd International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pp. 1-5, 2006.

[18] M. C. Pichiliani, and C. M. Hirata. "A Technical Comparison of the Existing Approaches to Support Collaboration in Non-Collaborative Applications". In *Proc. of the 10th International Symposium on Collaborative Technologies and Systems*, pp. 314-321, 2009.

[19] D. Pinelle, C. Gutwin, and S. Subramanian. Designing digital tables for highly integrated collaboration. *Technical Report HCI-TR-06-02*, Computer Science Department, University of Saskatchewan, 2006.

[20] D. Pinelle, M. Nacenta, C. Gutwin, and T. Stach. "The effects of co-present embodiments on awareness and collaboration in tabletop groupware". In *Proc. of Graphichs Interface 2008*, pp. 1-8, 2008.

[21] J. Rekimoto. "Smartskin: an infrastructure for freehand manipulation on interactive surfaces". In *Proc. of the SIGCHI conference on Human factors in computing systems*, pp. 113-120, 2002.

[22] A. Tang, M. Boyle, and S. Greenberg. "Display and Presence Disparity in Mixed Presence Groupware". In *Proc. of the 5th conference on Australasian user interface*, pp. 73-82, 2004.

[23] P. Tuddenham, and P. Robinson. "Tabletop Interfaces for Remote Document Review Meetings". In *Proc. of the 2nd international Collaborating over Paper and Digital Documents workshop*, 2007.

[24] P. Tuddenham, and P. Robinson. "Territorial Coordination and Workspace Awareness in Remote Tabletop Collaboration". In *Proc. of the 27th international conference on Human factors in computing systems*, pp. 2139-2148, 2009.

[25] M. Wright. "Open Sound Control: an enabling technology for musical networking". In *Organized Sound* 10(3): 193-200, December 2005.

Digital Object Identifier: 10.4108/ICST.COLLABORATECOM2009.8269
http://dx.doi.org/10.4108/ICST.COLLABORATECOM2009.8269