# Designing Mobile Multi-Touch Drum Sequencing Applications

**Mohato Lekena**

**Dissertation Presented for the degree of Master of Science in Computer Science**

**Department of Computer Science Faculty of Science University of Cape Town**

University of Cape Town

**Supervised By Professor Gary Marsden & Hussein Suleman**

**September 2014**

The copyright of this thesis vests in the author. No quotation from it or information derived from it is to be published without full acknowledgement of the source. The thesis is to be used for private study or non-commercial research purposes only.

Published by the University of Cape Town (UCT) in terms of the non-exclusive license granted to UCT by the author.

University of Cape Town

# Table of Contents

## List of Figures

# Abstract

Digital music software can limit the forms of music we create by using interfaces that directly copy those of the analogue instruments that came before. In this study we report on a new multi-touch interface that affords a completely new form of drum sequencing. Based on ideas from Avant-guard music and embodied interaction, a technology probe was created and then evaluated by a wide range of users. We found that for users with no musical training, and for users with a large amount of musical training, the software did allow them to be more creative. However, users with limited training on existing sequencing software found the new interface challenging.

## Definition of Terms

Percussion - musical instruments played by striking with the hand or with a stick or beater, or by shaking, including drums, cymbals, xylophones, gongs, bells, and rattles.

Timbre (in music) - the character or quality of a musical sound or voice as distinct from its pitch and intensity.

Quantization - the procedure of constraining something from a continuous set of values (such as the real numbers) to a relatively small discrete set (such as the integers).

DAW – Digital Audio Workstation

# 1 INTRODUCTION

Music is an activity that has for a long time held special significance within society, both culturally and economically. As a form of recreation it has become ubiquitously adopted by every culture, each with its own local variations. Amongst all the creative arts, it can be argued that it makes the largest impact towards improving one's quality of life. Indeed, one might even go as far as describing biological frameworks for the human capacity for musicality, meaning that music is a necessary and integral part of human development [23]. Beyond this cultural importance, music has also become an economic force. Studies show that recorded music products are now part of a market worth more than US$31billion since 2006 [51], due in large part to the power and impact of the Internet. Indeed, nearly a quarter of sales during that time period were digital [51]. These digital technologies have impacted on more than just the consumption of music though, but also the means of its production.

The content and quality of music over the history of music has, not surprisingly, long been linked with the quality and form of the instruments available to the culture making the music. This has been studied both from the perspective of the gestures that the different instruments afford players [109] and the perspective of the materials available for the creation of musical instruments [112]. The affordance and limitations given by these different instruments affected and moulded how musicians using them played, and thought about the music they were making. It is no surprise then that novel technologies have, more than once, fundamentally changed the way that societies have both consumed and produced music [58]. These were technologies as early as the high-quality piano strings and valved brass instruments borrowed from the resources discovered during the Industrial Revolution [57]. Since then, as evidenced by the invention of early electric instruments in the 1800s, music and instrument design have had a symbiotic relationship with technology. Later, the emergence of the Personal Computer and freely available production software led to the emergence of bedroom producers, home studios and the democratisation of the music creation process. This was all made possible through the widespread adoption of Digital Audio Workstations (DAWs) [44]. This shift towards digital instrumentation was echoed in the worlds of HCI, engineering and formal music academia. From these fields a new research interest was soon established that related to all three broad areas: New Interface for Musical Expression (NIME) [82]. Recently, the emergence of mobile computers and fast, accurate multi-touch screens further increased possibilities in the field of NIME [38]. Not only has this led to an increase in the number of output devices and synthesis techniques that produce unique sounds on multi-touch mobile technology, but has also been used in creating a large number of new input devices and techniques [38]. Typically, the input devices provide mappings between user inputs and the various aspects of the sounds generated, such as volume and pitch [75]. This focus on sound synthesis and control thereof, though, has resulted in fewer interfaces that focus on the control of non-generated (i.e. pre-recorded) percussive sounds. This is somewhat surprising when looking at the global importance of percussion and rhythm in both contemporary and world music [2] [17]. Most of the interfaces that do deal with these sounds strive to recreate digital equivalents of real world instruments. This is often by means such as the physically modelling percussive output (such as in [8]) or by attempting to appropriate gestures and interaction metaphors from real world instruments and the digital device is controlled by the same gestures one would use with the physical instrument [29] [31]. Commercial music systems built for mobile multi-touch hardware, such as Garage Band, are also flawed as they are often the mobile alternatives of popular desktop software. The desktop software itself, though, is often also a digital appropriation of real world interface metaphors, thus limiting expressivity and novelty of expression

[30]. While these types of interfaces are appropriate in some circumstances, by constraining the interface to physical systems one also limits users‘ creativity [74]. This sentiment is echoed by Sergi Jordà who proclaims that, despite constant development, we are still in need of useful, playable, thought-provoking enjoyable instruments, capable of interesting, surprising, enjoyable music [55]. Jorda is not alone in his criticisms of the state of current NIME designs, as Levetin, McAdams and Adams. [65 p.1] argued:

> "Electronically controlled (and this includes computer-controlled) musical instruments need to be emancipated from the keyboard metaphor; although piano-like keyboards are convenient and familiar, they limit the musician's expressiveness (Mathews 1991, Vertegaal and Eaglestone 1996, Paradiso 1997, Levitin and Adams 1998). This is especially true in the domain of computer music, in which timbres can be created that go far beyond the physical constraints of traditional acoustic instruments."

While keyboards and timbres might be foreign to the idea of the representation and performance of percussive music, they can stand as easy analogues for percussive equivalents. By moving past considering only previously popular physical gestures we might be able to represent a far wider array of rhythms, beyond the constraints of traditional acoustic instruments. An over-reliance on these traditional instruments can lead to a limit of expressiveness and also a waste of the new technology these interfaces are being designed for.

Another trend in existing systems research and experimental music is the high number of interfaces dedicated to performance only [60], despite the fact that current music trends favour loop-based production [117]. This is possibly another consequence of the above-mentioned trends in NIME design, as there are no traditional acoustic elements that handle percussive pattern sequencing. This means that these tools that focus more on pattern specification than pattern performance (or sequencing) are often not focussed upon in the world of NIME. As a consequence, most recent developments in sequencing have happened in commercial systems, where there is often less incentive to innovate [52]. Thus, many of the drum sequencers available commercially today, whether on desktop or multi touch systems, are still based on the gridded structures. These structures rely on having musical notes or events constrained to a discrete timeline as opposed to a continuous one, and were introduced by the first of such technologies in the 1970s. These are approaches and designs that do not have to be replicated on multi-touch displays, which have entirely different sets of affordances than the original technologies that the structures came from. This project therefore aims to explore the design of a gestural drum sequencer for multi-touch mobile phones that is based not on any traditional instruments or interfaces. Instead the design will begin from the standpoint of the research on multi-touch gestures. It will implement the design in a way that leverages this information to create a new musical interface that should engender creativity in users.

## 1.1 Motivation

### 1.1.1 Drum Sequencing and Gestures

Percussion forms the basis of most contemporary music, being literally the beat upon which the music is built. Rhythm and percussion has also been long known to be one of the chief differences between the musical styles of various different cultural groups, such as between Western and

African music [2]. Despite this, the rigid structures and interface analogies borrowed from traditional electronic drum sequencing tend to allow for the easy interpretation of only a certain type of rhythm. This is one which is quantised perfectly, linear, and amiable to only certain time signatures (see Related Work chapter). These interfaces, which often imposed a compositional framework on users, have been argued to be holding users back both in terms of the creativity and expressiveness that they can operate with, placing limits on their output potential [74]. These limits are also further exacerbated when inappropriate gestures, as mentioned above, are chosen for these interfaces when they are designed for multi-touch displays [74]. Interfaces for percussive music performance and specification are often based on striking gestures ([59] is an example). This is despite the fact that continuous or stroking gestures have been shown in research to have cognitive advantages [125] [20] [48]. These discrete striking gestures are also in contrast to recommendations for direct manipulation in interfaces (touch screens devices support direct manipulation) when one considers that the first feature of direct manipulation is the continuous representation of actions [97].

### 1.1.2 Mobile Music

Mobile, multi-touch musical interfaces provide unique advantages that have been previously not available to designers of new instruments. Previous hardware-based systems for specific tasks such as drum sequencing were designed at a time when the limits in technology were entirely different to today. Disregarding these limits to design new interaction techniques should only benefit the interactions and the users [74]. This is especially true for mobile phone systems amongst the various available multi-touch devices. This is as their proliferation in modern society allows for easy access and familiarity, while recently the ever growing power of these devices allows for complex, meaningful systems to be built on them [11][116]. Focussing on mobile phones carries the added advantage of being able to reach those users left behind by the personal computer and DAW revolution. Research shows that cell-phone adoption, especially in the developing world, now eclipses PC adoption [56]. Modern mobile phones are advantageous for use when exploring gestures not only because of their popularity though, as they do also present a unique form factor. Smart phones today have screens that are both large and sensitive enough to support meaningful multi-touch gestures. This is all while still being portable enough to allow rapid, easy access to the entire screen estate – even when operated single handed. This makes modern mobile devices perfect for exploration of multi touch musical interactions such as the kind we are interested in.

Thus an opportunity exists here in the research of musical interfaces. This is an opportunity that can be exploited through the implementation of a drum sequencer that has an open, non-linear interface. This interface should also capitalise on the true capabilities of multi-touch mobile technology through the use of continuous gestures as opposed to discrete ones.

## 1.2 Aims

This project presents a new gesture-based mobile drum sequencing interface for multi-touch screens that was created to engender creativity with its open, unquantized structure. Named Xen, it was designed to disregard traditional drumming gestures and interfaces borrowed from both physical and digital drum machines. This started by even removing even the basic assumption of striking a surface in order to create percussive sound. This project aims to explore new forms of percussive interfaces to empower creativity in users who might struggle with existing interfaces and their digital analogues. Xen is therefore based not on striking (discrete, poking) gestures – which are appropriate for working with physical instruments – but on continuous (stroking) gestures.

Gestural control of computer generated sound can be seen as a highly specialized branch of the field of Human Computer Interactions (HCI) [82]. Thus, principles from HCI and also User Experience (UX) were used in the initial evaluation and design of the system. The system was then launched as a technology probe. After these initial evaluations and probing, the prototype was finally evaluated using a methodology for assessing the creativity and expressivity that musical interfaces afford, using Discourse Analysis [105].

Therefore, considering these motivations, the research questions that this project aims to answer are as follows:

1) To investigate whether employing stroking gestures in the design of a drum sequencing application will engender creativity in users
2) To investigate the use and viability of an open, non-linear and unquantized mobile sequencing system

## 1.3 Organization of Dissertation

These issues, as well as other related issues, are explored and further expanded upon in the remainder of this dissertation. The organisation of the thesis and its chapters is as follows:

*Chapter 2* presents the literature and works related to the formulation of this project. This review will look to both review the state of the art in the fields of mobile NIMEs, drum sequencers, and HCI, but will also present sources for design inspiration in the decisions taken. This chapter will give further motivation for the project, contextualise its place within current research, and lay a foundation for the chapter to follow.

*Chapter 3* presents the design goals that the designs of the final probe are drawn from. These design goals are drawn from the world of experimental classical music. They are the goals that the project will aim for in order to make sure that the probe is backed on strong and legitimate musical principles.

*Chapter 4* gives both a detailed description of the probe's final design and also the steps that led to this final design and the providence for the design decisions. The chapter also looks into inspirations from the field of experimental electronica for some design decisions taken in executing the probe.

*Chapter 5* lays out the details for the evaluation sessions run in order to evaluate the system. This begins with describing the chosen experimental methodology and why it was chosen, explaining what factors will be looked for during evaluation and finally giving the results and discussion thereof.

*Chapter 6* concludes the research and presents future work.

# 2 Related Work

This chapter will look to review the research drawn on by this project, listing work that it has taken inspiration from and looked to improve upon. The aim will first be to build the contextual background of research in NIMEs, focusing both on mobile phone NIMEs and general mobile NIME research. This information will illuminate exactly where in the world of current mobile music interfaces this project lies, and why, at this point in time, the research here is appropriate and relevant. The chapter will then look at the current state of drum sequencing interfaces, and the salient points in music technology, and music theory, that have led here. At this point focus will be given to projects that have also attempted to tackle the problem of providing a mobile drumming interface, looking both at each system's advantages and areas of possible improvement. Lastly, HCI literature on gestures, and work from various fields focusing on describing and assessing creativity will be synthesized. This is as both of these topics were considered extensively, both in the design and evaluation stages of this project.

## 2.1 NIMEs

New Interfaces for Musical Expression is an ever growing research area encompassing multiple disciplines that has seen increased prevalence in research of late. The term NIME first surfaced at a workshop the CHI conference in 2001, where a document was issued stating the goals of NIME, which were [86]:

1) To survey and discuss the current state of control interfaces for musical performance, identify current and promising directions of research and unsolved problems.

2) To identify major issues involved in the interplay between technological change and changes in musical forms.

3) To identify the ways in which alternate controllers affect the overall creative process from composition to performance and determine what impact this has on musical expression.

4) To put together the collective working experience and wisdom of the participants in some tangible form, such as strategies for success and a list of the 10 most difficult problems in musical controls.

Researchers involved in producing NIMEs often also came from fields more technologically oriented, using methods from their fields of interest to create new interfaces and instruments. This is a trend that can be traced back through the history of musical instruments, as Mulder shows [75] - advances in technology changed both the way people interacted and created with music. He goes on to state that acoustic instruments transduced performer movements into sound in a one to one manner, but had a limited gesture set and timbral control because of their purely physical nature. The instrument's user interfaces (so to speak) were linked to the physical creation of the sounds they produced, thus limiting the range of gestures possible. Electro acoustic instruments, which came into being when electronics were first introduced into Western compositional music, increased the range of potential sounds dramatically. They, though, did not allow for more gestures of adaptivity, and Mulder contends that this was due to the cumbersome and complex nature of the technology then. Any composers who wanted to use synthesisers and computers at the time would also have to be technically proficient when it came to the electronics used to generate the sound. Recording devices were not yet portable or accurate enough, and manipulating recordings meant hand editing tape. Computers at the time were slow and prohibitively large. This was nevertheless a step that Mulder saw as being a salient one as it led eventually to electronic music instruments, such as keyboards which used the Musical Instrument Digital Interface (MIDI). These allowed for a vastly expanded number of possible sounds and also sound parameters to modulate, except with more accessible means with which to interact with the instruments. The gestures that were initially used to control the synthesis were borrowed from existing physical instruments. This though was at the cost of expressivity due to the limited resolution, accuracy and responsiveness of gestures, and the inadequacy of the popular MIDI standard for sending musical information. None of this was helped by the fact that the gestural interfaces did not afford users the real-time control during performances of all of these parameters. This was despite the fact that the sound synthesis capabilities of these instruments meant almost infinite variability and sound choice options. This was due, again, to technical constraints at the time, and the fact that sounds were pre-generated and gestures set in stone once designed and implemented.

The latest revelations in digital, portable technologies, and also in the number of sensors that are easily accessible, has led to the next step in new instrumentation being what Mulder describes as Virtual Music Instruments (VMI) [75]. The two main aspects that he described as being definitive of VMIs were firstly the fact that any gesture or body movement could be used to control the instrument, and secondly the fact that these gestures could be programmed (or mapped) to any aspect of the sound being generated. This idea of mapping - decoupling the process of interaction and sound generation for arbitrary links between actions and sound effects - was a major step. It allowed the idea of instrument design to reach beyond any boundaries. It also allowed those who specialize in the algorithms responsible for sound synthesis and those specializing in gesture and interface design to focus only on their respective fields, knowing that, through mapping, links could easily be created between both worlds. This all leads directly to the current state of NIME, where experts in various fields involved with design (and specifically interaction design) have lent their expertise to the problem of musical interaction. This can be seen in recent moves in the NIME community firstly to document the implementation of techniques from fields such as HCI [40] [88]. Then, later, to use these techniques as inspiration for the synthesis of entirely new methodologies with which to design and evaluate new VMIs [118] [104]. The work in these areas that is pertinent to this project will be discussed in further detail in the next section.

Mapping, as a concept, was also important as it represented the decoupling of the gestural interface and the sound creation. This meant that designers interested in gestures could essentially use any sensor currently available as an input device, and later have this communicate via a mapping to a synthesis module. Hence, in today's world where sensors for a wide variety of different inputs are becoming more readily available, designers of new interfaces are left with a wide scope for any new potential instruments. This fact was alluded to by Fels in his Siggraph 2011 course on advances in NIMEs [32]. He stated that the first step to building a NIME is (1) picking a control space, followed then by (2) picking a sound space, (3) picking a mapping, (4) connecting the software and then, finally, (5) composing [32]. With this in mind, it is not surprising that many designers, when looking for a control space today, look towards the sensor-rich smart phones. Commodity high end devices more often than not have enough input sensors for the control of a NIME [30]. Therefore, as their proliferation into the mass market continues, so do efforts by designers to exploit them. The following sections in this chapter will look at just this: mobile efforts within NIME.

### 2.1.1 Mobile NIME Research

In [52] John gives an overview and survey is done of all the mobile music papers presented at NIME in the decade since the very first was published in the conference proceedings. Through looking at these 98 papers and classifying the mobile projects in terms of various categories, a high level view of trends in mobile NIME research can be achieved. The survey also used a broader definition of mobile music than just interfaces and systems for music making. Rather, it used a definition proposed by Gaye, Holmquist, Behrendt, and Tanaka [38] which sees mobile music is a field concerned with any musical interaction in mobile settings, using portable technology. This definition then warranted the inclusion of systems for the sharing and consuming of music as well, for example.

The first categorization offered in the paper is that of mobile music activity. While the authors admit that an exhaustive listing of every activity type would yield a much larger list, they settled on 5 activities that they saw as representing the largest cross sections of published work. These categories were: Exploring the influence of location; file sharing and Collaborative Composition; Interaction with Wearable devices; Using mobile phones as a performance device and Examining HCI design issues. As is to be expected, some projects would span the lines among the various categories here. Of these chosen categories, the one best represented was that of examining HCI issues, with 48% of the papers surveyed making reference to design issues influencing interaction. Using mobile Phones for interaction was specifically mentioned in 27% of all the papers, the second largest segment, which was closely tailed by File sharing and collaboration at 26%. Lastly, 16% of all the projects included some form of context awareness (although 22% ascribed significance to the geographical location of the system), while only 10% of projects have designed clothes that contain sensors for interaction. These results are to be expected, as NIME design is fundamentally involved with designing usable interfaces, as is evidenced by the fact that nearly half of the projects surveyed being about HCI. Beyond that, accessibility of technology seems to be another important factor in deciding the direction that projects take. This is evidenced by the fact that mobile phone projects (easily accessible technology) form the second largest section while wearable sensors (a less accessible technology) came in last. Mobile phones also imply geographic mobility, even if it might not be explicitly stated that projects are context aware or make use of location.

The next categorisations offered by the authors were comparisons by project type and comparison by author type [52]. The project types chosen for comparison were those first posited by Behrendt [13], which are technological, geographic and socially based projects. The author type categorisation split the papers according to the author's background and main interests. These interests were ascertained by studying the type of publication they were involved with, their biographical information (as listed in the paper), and information found on the authors' personal sites. The three most prominent types here were: Musicians - including composers and performers; Academics; Others – including commercial entities, private companies and authors with unlisted affiliations.

The results of the project type categorisations showed that out of the 98 papers, an overwhelming majority of 67 (68%) fell within the technological category. 46 of the papers surveyed were about studying social implications, while only 22 were concerned with interactions depending on geographic location. These Figures (tabulated below) cannot be considered in isolation though, as further investigation will show there are large areas of overlap among these three categories. For example, nearly half of all the papers primarily investigating a social issue were also concerned with the technical aspects of implementation.

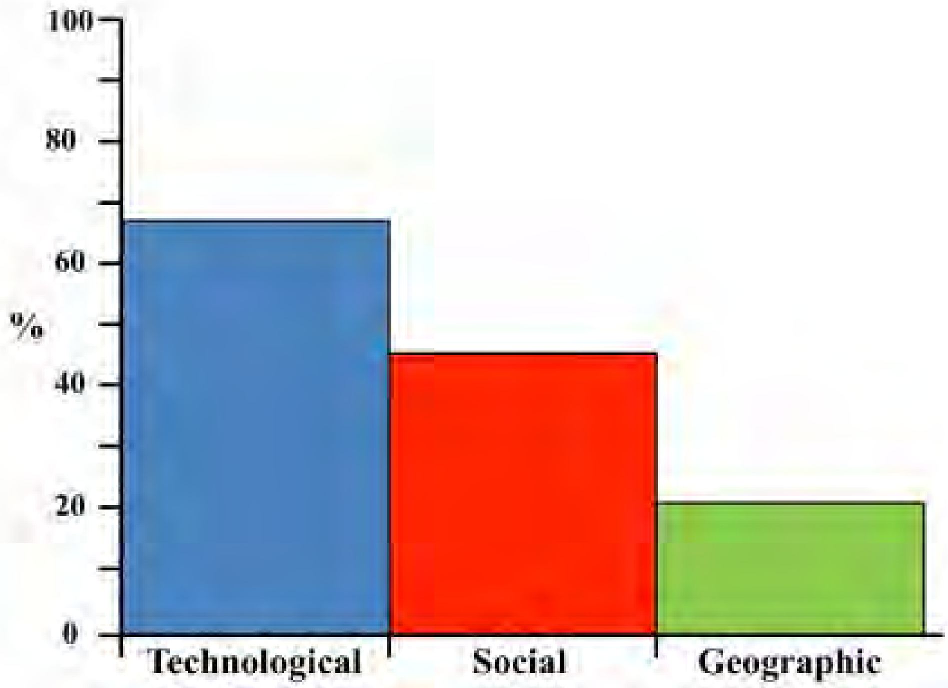


**Figure 1 Percentage of papers published at NIME as categorised by project type**

A more interesting observation can be made when looking at the number of papers published in each category over the past ten years (Figure 1 above), using the numbers from the same survey. While initially in 2003 all papers described both technical and social issues, with half of these also having a geographic aspect, how these distributions changed over the years is indicative of global trends in technology in general. After 2006, where the number of papers published as a percentage of the whole dropped, there was a steady increase in the number of technologically orientated papers. This continued until between 2010 and 2011 papers published fell almost exclusively under the technological category. The growth of that category within that time period coincides directly with an explosion of the number of mobile phone-based applications. As the hardware specifications of mobile phones grew, development platforms became easier to use, and the number of sensors on commodity devices increased. One might conclude that the fervour within the NIME community to develop for these new devices and test their capabilities (a fervour also possibly borne from the realization that quick and relatively painless publications were to be had) stunted the growth of the other two categorisations. This phenomenon then leaving only technical descriptions of products as the predominant type of research published at NIME.

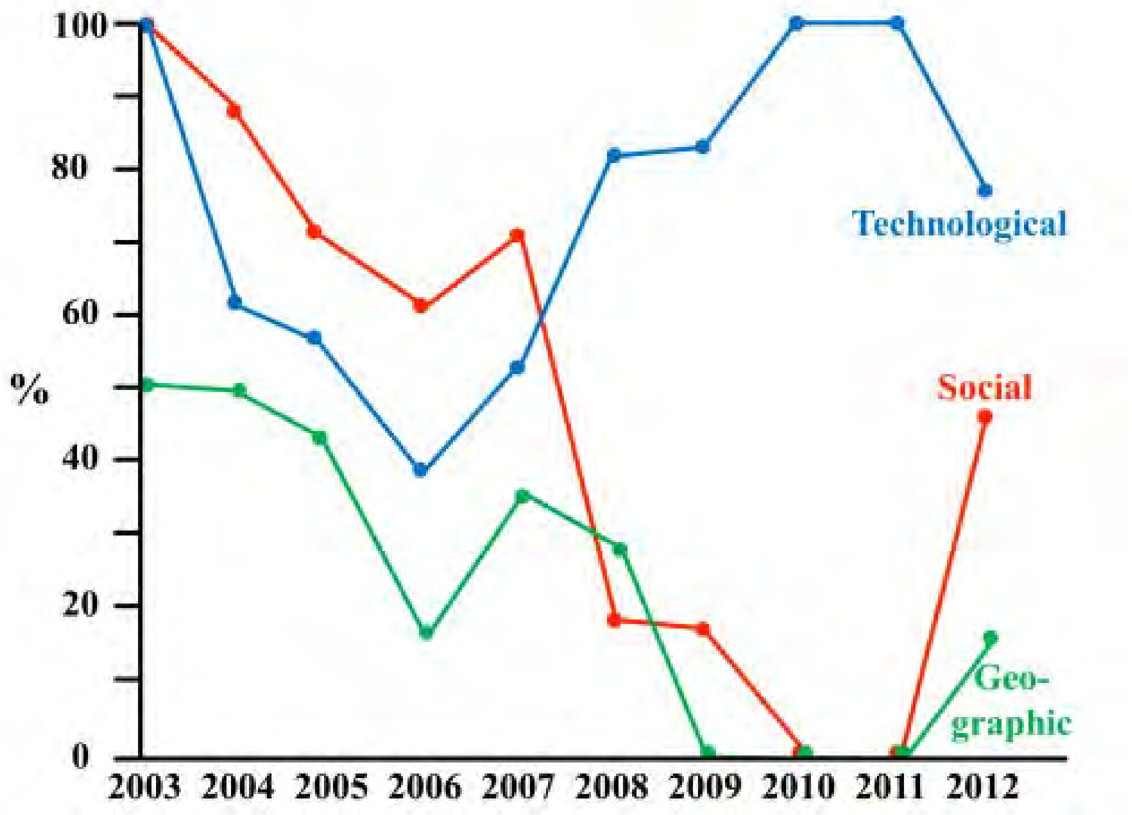


**Figure 2 Percentage of papers in each category published at NIME over time**

Another type of research that saw a decrease in the number of publications, according to the survey numbers [52], was projects where the researcher had developed their own unique hardware. The percentage of papers with unique hardware is seen in Figure 3 below, plotted over the same ten year period stretching from 2003. As can be seen in the Figure, projects were initially all based on custom hardware, as the sensors and processing power needed to operate a mobile NIME were not easily accessible. Therefore as the sophistication of mobile phones increased, they attracted more and more researchers and the number of projects using unmodified commodity devices grew steadily. As would be expected, this growth corresponded with a drop in the number of projects with unique hardware requirements.

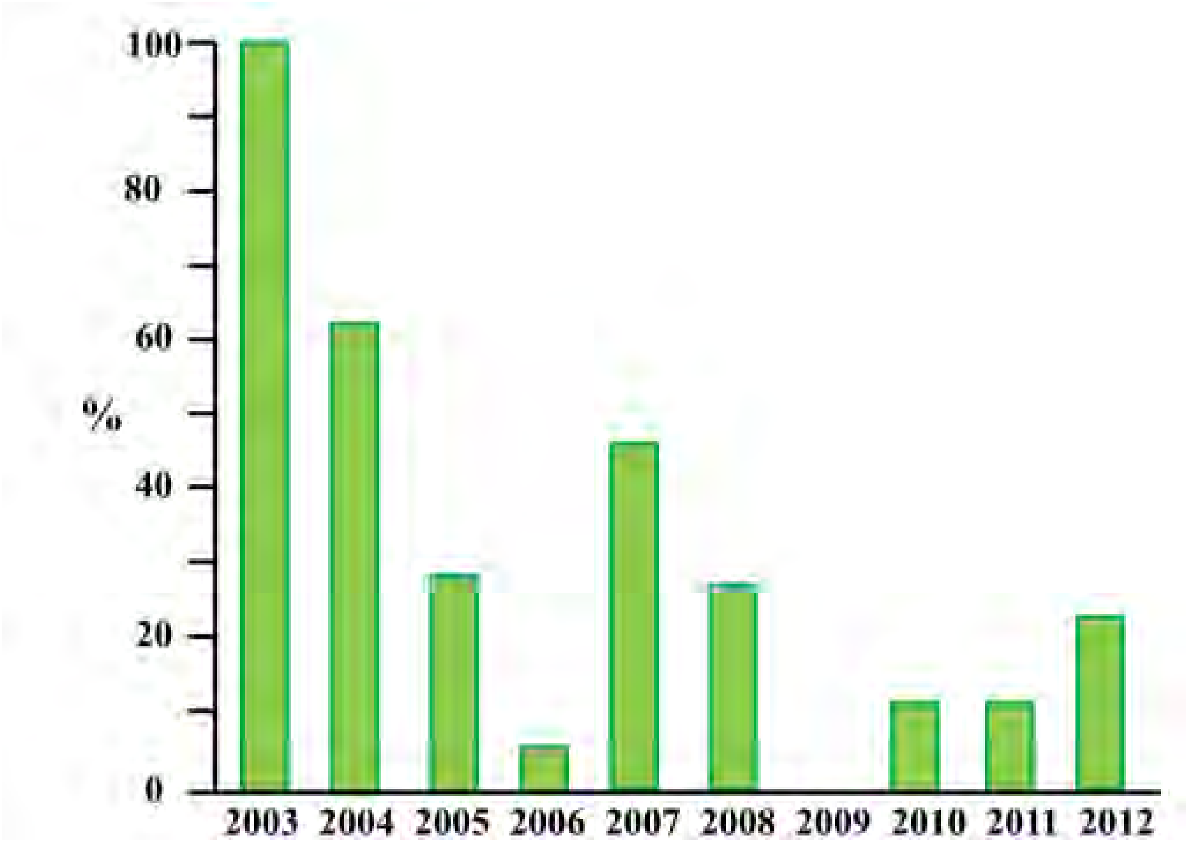


**Figure 3: The number of mobile based papers published at NIME that introduced unique hardware, plotted over time**

What both of these graphs from [52] also show though is a slight change over the last two years, where the trends noted above have begun to reverse. In 2011-2012 the number of projects that were technological descriptions of systems dropped significantly, with projects studying social implications jumping in number, as did as those with an inherently geographic nature. This hints towards a possible decline in the novelty of new mobile devices, with the capabilities of the latest generation of smart phones being firmly established. Therefore less innovation is associated with simply building a mobile app, and now researchers are again searching for novel and alternative ways of doing things. One can also posit that there is a limit to the number of purely technological items of research that can be produced using new technology. At some point the academic community must take stock or what works, codify this as rules, methodologies and guidelines, and focus on social implications or products as well as how they are used. This can possibly be seen as the reason for the trends in the types of authors publishing papers.

In Figure 4 the number of authors who were artists, academics or other stakeholders is plotted over the ten year period beginning in 2003. In the 98 papers published since 2003, there were 160 individual authors involved. Overall the number of Academics in the entirety of the papers published sits at 56.25%, with artists making up just under a third (32.5%) [52]. If one looks at these numbers over time though, it can be seen that the number of academics publishing papers saw a peak at the same time that the number of technological papers published did. One can also see that with the resurgence of papers studying social impacts more actual artists are getting involved. This could speak to the idea of moving the efforts of research back towards people, and servicing the parties most likely to be interested in using the artefacts developed as part of NIME research.

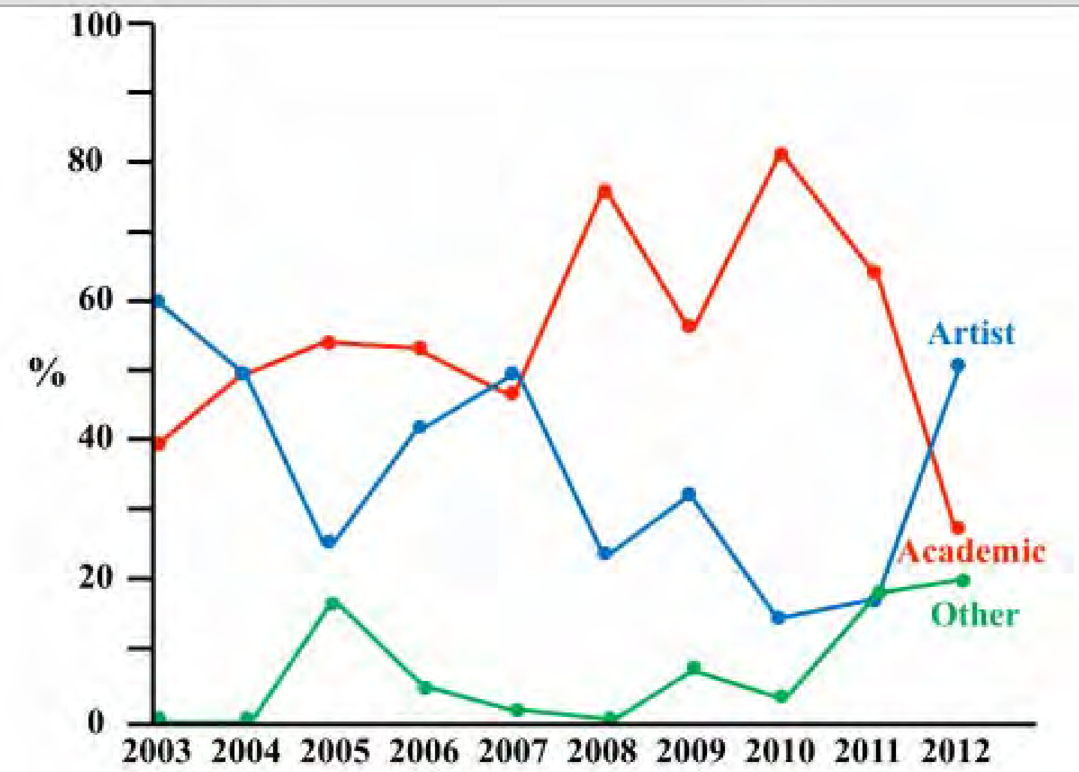


**Figure 4: The number of people involved in publishing papers who were artists, academics, or other stakeholders over time**

### 2.1.2 Beginnings of Mobile Phone Music

The idea of using mobile computing devices for artistic representation through music is not a new one, and is one that has seen a recent rise in popularity as both mobile adoption around the world and the NIME community grew [38]. An early pioneer in this regard was Geiger [40], who designed a touch-screen based sound synthesiser paradigm using a port of the PureData (PD) [73] open source visual sound programming platform. PD was, and still is, a major tool for Digital Signal Processing (DSP) in contemporary classical and experimental music and also for sound related research. The visual language works by allowing users to place elements (represented usually as squares with text values) on a scrollable, zoomable two dimensional plane. These elements either create or transform data, and users can connect these elements to organize the flow of data between them, and thus write patches (executable PD files). What Geiger did was to port the PD DSP engine and interface to the PocketPC Personal Digital Assistant (PDA) without much revision or change. This essentially made the project one about the implementation of new technology (or proof of concept) rather than a new interface. It was found that writing patches on the smaller screen was prohibitively difficult – a fact that wasn't helped by the not very sensitive touch screen input of the PocketPC [40]. Users of the system ended up having to write patches on a desktop computer and then transfer them to the PDA, where they would be further edited and executed. Geiger himself noted when writing about the project that in future work would could be done to ameliorate interface issues and provide an easier usage experience [73].

Regardless of these problems, though, this project still represented a major step forward. This is mainly because at the time, having on-phone synthesis was simply not plausible in many cases, as the computational power of mobile devices was not adequate [110]. This led to the development of more systems that were simply controllers of PD (or MaxMSP, a similar language) patches running on dedicated servers – essentially having mobile phones acting only as the gesture interfaces. A seminal example in this regard was Tanaka and Gemeinboeck's system [107]. It was based on a custom-made PDA and used accelerometer and GPS capabilities to control streaming music coming from a PD patch running on a server. In this way, it aimed to use the social dynamic of mobility as the inputs for a music recomposition system. Although conceived initially as more of a multimedia installation than software with a long life span, it did explore interesting themes of using two-dimensional topography as a musical interface [107]. After this initial success, other similar GPS-based systems were introduced [106] [108], many of which also relied on server-based musical computation. Many of these tools, though, were mainly aimed at musical re-composition, as opposed to the creation of original pieces, controlling pieces or performing.

After these initial successes moves to turn commodity mobile devices into physical instruments were also made, and several notable systems have been both proposed and built. Schiemer and Havryliv's PocketGamelan [94] instrument is seen by some as a system that paved the way in this regard [123]. Its main aim was to allow non-musicians to play and experiment with advanced (and not often used) musical tunings using handheld instruments. The specifics of these instruments would be coded, once again, using the PD language on a desktop, then cross-compiled to the then popular Java 2 Micro Edition (J2ME) for mobile phones using the custom Pd2J2me application. The fact that the system was borne from the purely musical need to provide alternatives to interfaces designed around 12 equal divisions of the octave meant that the Pocket Gamelan was of immediate interest to some groups outside of the world of technology. This could be seen by the fact that the system was soon used in real performances of experimental classical pieces Mandala 3-6 at various conferences and venues around the world [94]. This idea of creating interfaces to break away from computer music's most persistent legacies is one that this project takes direct inspiration from, as explained throughout the remainder of this chapter.

While the J2me technology used then has eventually fallen away, the legacy of the ideas introduced by the system still remains. Currently there are several institutions that boast mobile orchestras – maybe the ultimate realisation of the concepts introduced by the Pocketgamelan. The foremost of these orchestras is arguably Stanford's Mobile Phone Orchestra (MoPho) – a repertoire-based ensemble using mobile phones as the primary musical instrument [122]. It defines a "platform of hardware and software configuration and players, enabling composers to craft mobile instruments and write music tailored to such an ensemble" [122 p.1] and was the first ensemble of its kind. The orchestra exists as an exploratory body, moving towards goals both within a research and music making capacity, investigating the fusion of technological artefacts with human musicianship. At the core of the MoPho is a well-defined platform of hardware and software configurations with which anyone could build mobile instruments and compose music suited for the ensemble [122]. In many ways it can be seen as a logical step forward from the various laptop orchestras that are active globally [113].

All the instruments created for the iPhone based MoPho are written in the mobile music (MoMu) toolkit [18]. MoMu is an open source collection of application programming interfaces (API) and utilities geared specifically towards mobile music development. The toolkit allows for real time audio synthesis and control – and in this regards it follows in the lineage of Essl and Rohs' Mobile Synthesis Toolkit (STK) [28]. Subsequent to this, many similar toolkits have been released which, along with the release of several papers detailing aspects of design that relate specifically to mobile music, have made the creation of new mobile music systems more viable than ever [82] [118]

### 2.1.3 Modern Mobile Phone NIME projects

Sound synthesis was now possible on proprietary devices – and at not much cost to developers due to the various synthesis toolkits and libraries listed above. Designers of new instruments could now focus solely on creating unique gestural interactions using mobile phones. Possibly the most practical way of chronicling the development of new mobile interfaces for musical expression is by looking at the sensors that a standard device gives you for potential input and examples of how they have been used. In this way one can gain a general overview of both developments in the world of mobile NIMEs and a look at the tools available to their designers (interfaces for the representation of drumming and percussion will be discussed later in section 2.2). Essl et al. [30] provide a theoretical analysis of the sensor capabilities available to commodity mobile phones through a design space. He goes on to define this design space define through a taxonomy that suggests early on what mappings might be possible given the characteristics of a technology. In this case the technology refers to sensors available for input on mobile devices. This was all done with one goal in mind: to explore and ascertain what impact these technological choices would have on emerging musical practice. In this section each sensor in this list will be looked at, with an example of a mobile NIME developed to take advantage of it. The paper by Essl et al. differs from others giving summaries of sensors for NIME design as it focuses specifically on those available to new modern mobiles. The nine sensors listed by Essl et al. are:

- Accelerometer
- Magnetometer
- Gyroscope
- Camera
- Touch Screen
- Multi-touch Screen
- Touch Pad
- Capacitive proximity sensor
- Microphone

A prominent example of both multi-touch and microphone sensor input for music is the Smule Ocarina [123], a stand-alone mobile musical instrument with a custom audio engine for real-time sound synthesis on the iPhone. The main model of interaction is based around having users blow into the phone's microphone. Then, the amplitude of the noise generated by this blowing is used as the velocity (a measure of how rapidly and forcefully a note is played) of the sound generated by the Ocarina. This is an oft used example of a technically inventive new input methodology, even if the interface metaphor is taken directly from a real world instrument. Multi-touch input is used on the Ocarina to finger chords and modulate the qualities of the sound produced, much in the same way as one would on a flute. Another interesting use of mic input is the ACE system [98], which records the user making rhythmic mouth noises (or "beat-boxing") and converts this information into MIDI (Musical Instrument Digital Interface) data. The MIDI data can then later be replaced, on a fully fledged DAW, with any drum or percussion sounds the user might want. This project was also more focused on the technical details of audio recognition than by user experiences or interface design.

Systems which analyse audio information in a similar way to the Ace system have also been proposed for the desktop.

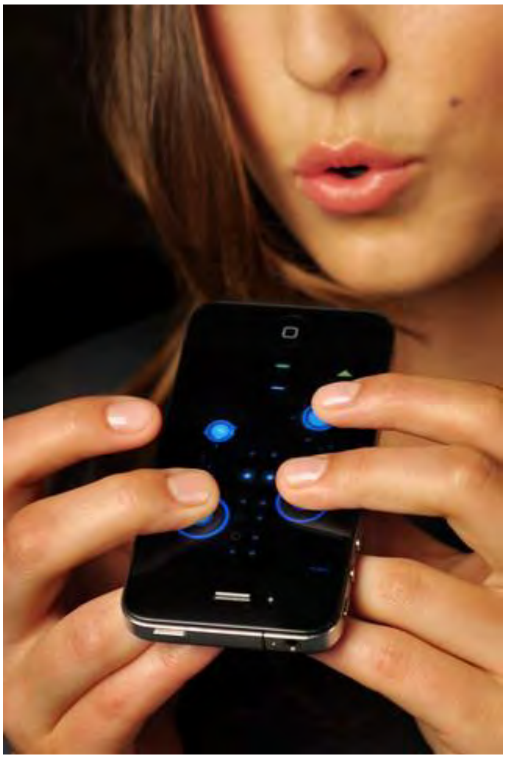

**Figure 5 The Smule Ocarina being played**

Musical accelerometer and gyroscope use has been often discussed in literature, with an oft cited example being the ShaMus system [29]. ShaMus allowed for the mapping of accelerometer data to synthesis algorithms on the phone in real-time, offering a wide range of possible interaction metaphors and gestures through this data. The method chosen in this system was to have the user interact with it by mimicking the striking or shaking gestures associated with the use of various percussion instruments, such as drums or maracas. One reason stated by the creators of ShaMus for focusing mainly on percussive elements is the lack of an overall frame of reference when it comes to accelerometer data. Accelerometer data only gives you changes in position, but not absolute position. This makes using it unsuitable for interactions that rely on a fixed reference, such as pitched sounds [29]. ShaMus also uses magnetometer readings to segment the virtual space into angular sectors, each with distinct interaction attributes. In this way, the system would imitate a virtual drum kit, with different “virtual” instruments being placed in different physical locations. A more interesting example of magnetometer use, though, was the MagiMusic system [59], which has the user performing touchless gestures while holding a physical magnet for musical interactions. This is similar to how many capacitive proximity sensors would be used, though magnetometers hold the advantage of being more readily available in commodity devices and having higher degrees of accuracy.

The final sensor is the phone camera. In order to derive input data from the camera there are two main methods: using two dimensional marker recognition, or exploiting optical flow theories [30]. The first of these methods relies on having the interactions taking place on a grid of virtual markers derived from codes that the creator of the system defines. The grid represents a large workspace, and by using image recognition algorithms, a user's phone can work out where on the grid it is and how fast it is moving in any direction relative to the grid itself. This would be achieved simply by having the phone over the grid, with the camera facing the grid. In this way the camera's screen acts as a window onto the virtual workspace. The main drawback of this method is that it requires the printed grid in order to function, which is a limitation that using optical flow could potentially eliminate entirely. Optical flow methods work by implementing general optical movement detection algorithms similar to those found on optical mice [30]. Using these, a mobile phone could, through analysing camera input, detect relative linear movement in the display as well as relative rotational movement around the optical axis, giving three degrees of freedom. This method can be less accurate that using 2d markers, and will also not give absolute positions - only relative ones. All of these methods can also be enhanced by introducing colour detection to further enhance the richness of the potential input data. All of the input data generated by these methods can be used in a variety of ways. An example would be using the movement data to follow a pre-composed score set up as markers on a two dimensional space, or using the data to modify various parameters of already playing sound. This potential grows even more when one looks at systems such as SoundCodr [103], which allow users to draw multi coloured blobs, create rules for how these blobs interact with each other and a stream of sound. This essentially means specifying a graphical score grammar, and then reading over these scores using image recognition on a mobile phone. The system was created primarily for use in live situations, and so it allows users to edit the action-condition based graphic score language in real time on the mobile phone's screen [103]. The sound synthesis in this system is handled remotely on a desktop computer.

**Figure 6 The Live Codr system in operation**

The idea of drawing, or using continuous touch gestures, is one that seems much more suited to multi-touch screens than camera input though, as it gives more immediate feedback. Indeed, multi touch screens, with their incredibly rich set of potential interaction types, have become as popular in the world of mobile NIMEs as they have been in general mobile computing. An example of how many designers have approached multi-touch interfaces can be seen with the Gliss [114] system. It works by allowing users to draw images (on a touch-screen interface) to control musical parameters.

In the end the design space designed by Essl et al. was based around the following factors: the types of movements that can be detected (either linear or rotational); the maximum velocity that can be sensed; and the maximum interaction range. This range of interaction is strongly related to the frame of reference in which the interaction occurs, which can be either fixed to an absolute outer frame, or one that is relative only to previous sensor readings. From a musical standpoint this difference is that absolute measures can be both stored and repeated more easily and therefore relate to musical parameters that are themselves absolute (such as pitch or loudness) [30]. Lastly, the space also separates the sensors on the basis of their reach. This dimension denotes the maximum physical space afforded by any particular sensor for interaction. The reach of a touchscreen device, for example, is limited by the size of the screen itself. Taking all this into account, the final design space is described by Figure 7 below. Looking at this graph one can see that, for instance, Multi Touch screens (item 8 on the list) capture absolute data (which is not dependent on the data that came directly before it). Also, they capture linear input, as opposed to rotational, and with unlimited velocity. The shape of the entry in the design space, which in this case was a square, shows the type of data (position, velocity or acceleration) that the sensor captures.

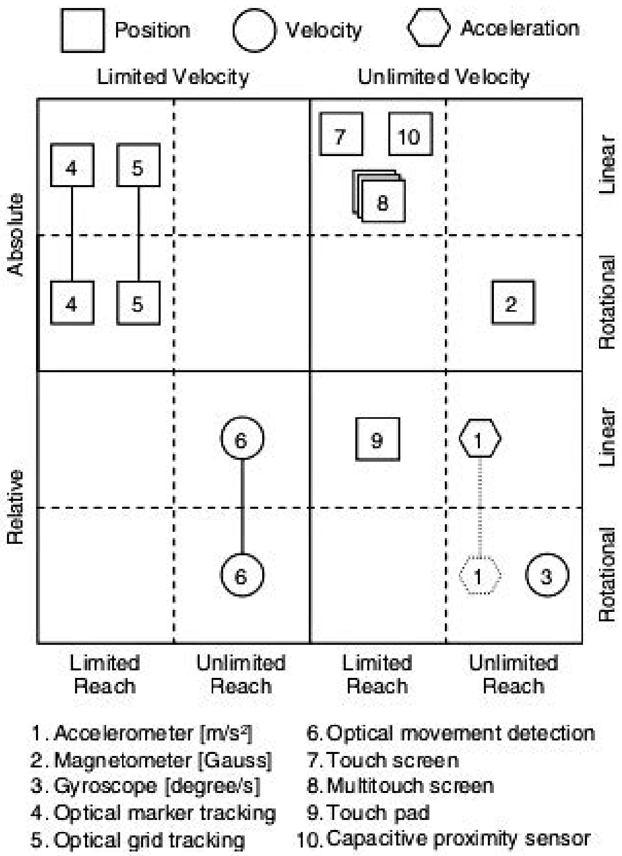


**Figure 7: Graphical representation of Essl's design Space [30]**

### 2.1.4 Commercial systems

In order to get a grasp of the state of the art in the area of commercially available mobile music making interfaces, two prominent proprietary systems will be discussed. Both are from major musical software developers and are also the mobile equivalents of popular DAWs: Fruityloops and Garageband. Although both systems can operate as stand-alone apps, they can both be seen as the mobile extensions of their respective makers' flagship desktop DAWs. As such, concepts and interaction metaphors, such as click and double click drum sequencing and linear piano rolls and loop arrangements, are carried from the desktop apps to the mobile ones. This means that one may generally start a project by tapping out beats either manually using the drum pads, or achieve more precise timing by using a step sequencer. Beyond this, GarageBand specifically adds two notable features, namely Jam Sessions and Smart Strings [7]. Jam Sessions is the name given to GarageBand's collaborative component, which allows for people to wirelessly connect their iOS instruments and record live music together. Smart Strings is an alternative to the drum-pad/step sequencer input methods mentioned above. It is a novel interaction metaphor that allows users to play instruments with only one finger, and leverages user control for mediated playing. This mediation is done by the Smart instruments themselves, which offer a range of chords, arpeggios and scales already in the pre-ordained key of the song so that anything played sounds in tune.

## 2.2 Drumming and Percussion

Now that an overview of the history and current status of both general NIMEs and the more specialized subset of mobile phone music has been given, the next section will provide a historical and theoretical background on percussion and drum sequencing interfaces. These interfaces differ in a few key ways to those identified in the chapter thus far. Firstly, percussive music in the context of drum sequencing consists only of non-pitched events, so they have one less dimension (namely the notes played) when represented. Secondly percussive elements tend to be more discrete than continuous, representing single points in time at which a sound source is played. It is for these reasons that percussive music is well suited to being sample based – i.e. based on playing back pre-recorded instrument sounds rather than synthesising sounds every time they are needed. This project does not focus at all on the process of sound generation. The last difference to note is that of pattern specification (sequencing) as opposed to performing. This project focused on the former rather than the latter, making the problem one of how to best represent rhythms (or patterns) based on these discrete elements.

### 2.2.1 Drum Sequencers

The earliest predecessors to drum sequencers as we know them today were the early drum machines from the 1930s. Drum Machines differ from sequencers as they were mainly focused on synthesis of drum and percussion-like sounds, not rhythm and sequencing. In this regard, the very first was realized by famed inventor Leon Theremin (who invented the Theremin instrument) at the behest of composer Henry Cowell. Cowell wanted a system with which to play multiple rhythm patterns that were too difficult to perform by hand. It was called the Rhythmicon, and while it was seen as highly innovative when it was released in 1930, it was limited to playing only 16 pre-programmed rhythms and was so cumbersome that it was eventually discarded even by Cowell [3]. After this initial system, some further experiments were done using tape loops with pre-programmed drum loops (such as the Chamberlin Rhythmate in 1957). It took until 1959, though (some 29 years after the Rhythmicon), for the first commercial drum machine to be released – The Wurlitzer Sideman [9]. The Sideman was an electro-mechanical drum machine that allowed users a choice between 12 electronically generated but pre-defined percussion patterns at variable tempos. Sequencing was achieved using a rotating steel disk with contacts protruding from the face of the disk spaced in the specific patterns to be played. In terms of an interface the Sideman only featured a rotary knob for rhythm selection, a slider tempo control, and ten buttons for manual launching of rhythm patterns. Despite this, it still managed to be a large device due to technical constraints at the time. This was problem that was later ameliorated by drum machines that were fully transistorized and no longer relying on large vacuum tube to produce drum sounds. This reduced drum machines to desktop size, as opposed to the previous floor-standing models [9].

At this point companies such as Ace Tone, founded by Ikutaro Kakehashi (who would later go on to found the Roland Corporation), were producing commercially viable drum machines that were beginning to reach critical mass [9]. A key characteristic of these older drum machines as opposed to their modern alternatives was the fact that they used sound synthesis to create the drum sounds, as opposed to digital sampling. Bursts of white noise, clicky sine waves and other basic sound waves, were used to imitate real percussion sounds [47]. What this meant was that while often the sounds of these machines weren't realistic, each machine had a unique character and distinct sound. The sounds would also often be pushed to levels beyond what designers of the machines would have intended. This has led to specific machines having cult status amongst producers, and their specific sounds creating trends in popular music of the time, as opposed to simply being used to replace human drummers during live performance [17]. This, though, was a trend that was held back initially by the fact that drum machines only rendered pre-programmed rhythms, a state of affairs that began changing in the early 70s.

In 1972, Italian electric guitar manufacturer Eko released what is regarded as being the world's first programmable drum machine[1] : the ComputeRhythm. Due to technological constraints at the time it was bulky, expensive and still considered difficult to use. As it was the first such device, it introduced a physical interaction method that would be highly influential. The ComputeRhythm had at the centre of its interface a 6 by 16 grid of buttons. On the horizontal axis of this grid each button represented an equidistant point on a musical timeline, which was one measure in length. If any of these buttons was "active" (or pressed), then it would trigger a percussive note, but if not then it would only play 1/16th length's silence. The machine would then play the 16 buttons sequentially (i.e. either playing a percussive note or the requisite length of silence) in an order going linearly from left to right. Horizontally, each row of buttons had the same functionality, but with different percussion sounds for the active buttons in the grid. Figure 8 below demonstrates what this layout looked like.

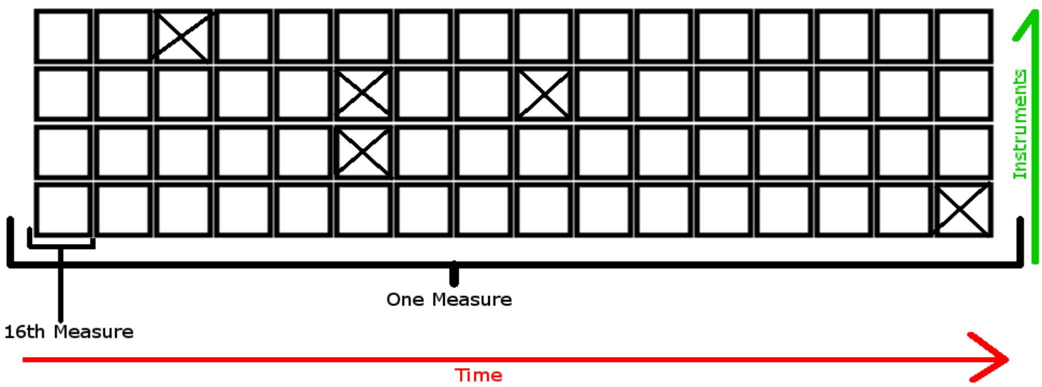


**Figure 8 A diagrammatic overview of 16 step sequencing**

This initial physical design became somewhat of a de-facto standard and was copied by many famous 16-step sequencers from the 1980s onwards, such as the early Linn LM-1 Drum Computer

[1] http://www.synthtopia.com/content/2009/08/25/the-eko-computerhythm-jean-michel-jarresdrum-machine/

and the Akai Tr-808[2] . These all streamlined the button matrix idea introduced by the ComputeRhythm by cutting down the matrix to a single row of 16 buttons with which to program changeable percussion sounds. While there were numerous technological innovations such as the move to digital sampling, which was introduced by the LM-1 (released in 1980 at an expensive retail price of $4 995) [12], the ComputerRhythm's interface design remained influential.

The popularity and pervasiveness of these hardware devices even led to the implementation of software equivalents of drum sequencers, which were included in most major Digital Audio Workstations [1]. This could be seen as early as in 1980, when the Fairlight CMI Series II was released [9]; this combined traditional step sequencing with sample playback, using the interface metaphors established by drum machines to play samples beyond percussion. These software sequencers also kept the age old tradition of representing music in a linear, left to right manner. While this was also done by early drum sequencers, the trend reaches further back than them as it stems from the traditions of Western music and writing [15]. Thus it is not a design decision that necessarily accommodates music from non-Western cultures, and some have questioned whether this representation is well suited for the purposes of musical interfaces [111].

With the switch to software, and the resulting freedom from physical device constraints, sequencers could now offer more than 16 steps and allow micro-timing and gridless playing (on a continuous musical timeline, rather than a discrete one). Despite this, it is still a common practice for sequencers to "quantize" music (automatically adjust rhythm to fit the 16 or 32 step grid), and then have the software add intentional error (or "swing") for a more "human" sound – which indicates expressivity [14]. Another issue with quantization is that in order to work properly it implies that a tempo and time signature have already been decided when the music is being played, something that is often not the case during the exploratory initial stages of song production.

[2] http://www.theguardian.com/music/2014/mar/06/roland-tr-808-drum-machine-revolutionisedmusic


**Figure 9 The Akai MPC 5000**

In the 90's, while software sequencers were on the rise, there was also a resurgence in physical samplers and home studios. These modules, of which the legendary Akai MPC 5000 groove machine is a canonical example (see Figure 9) [92], allowed users to load sound samples onto the 16 plush pads on the module's main interface. Users would then specify loops by striking these pads with their fingers while having the module record the sequence on its internal memory [102]. What this setup meant was that any music made with these modules would avoid the problem of having robotic sounding music with no natural human error. In fact, MPC was so influential that popular DAWs such as Ableton live now have automatic swing patches to emulate its recognizably unquantized timing error sound. A visual history of some of the important steps taken in the design of modern drum sequencers is shown in Figure 10.

These errors, or deviations in time, are so salient in music that it has been stated by Bilmes that "deviations are most important for percussive and non-Western music, and that they are indispensable study for any drum-machine architect wishing to create an expressive sounding product" [14 p.1]. He later goes on to give the design recommendation that drum machines and sequencers should start providing facilities for experimenting with deviations. This is as opposed to leaving it to the numerous software swing inducing solutions available (a list of which is given by Blimes in [14]). The goal of this project is therefore to explore how mobile multi-touch technology can be used to create non-quantized percussive music and, in so doing, allow users to explore their musical creativity.

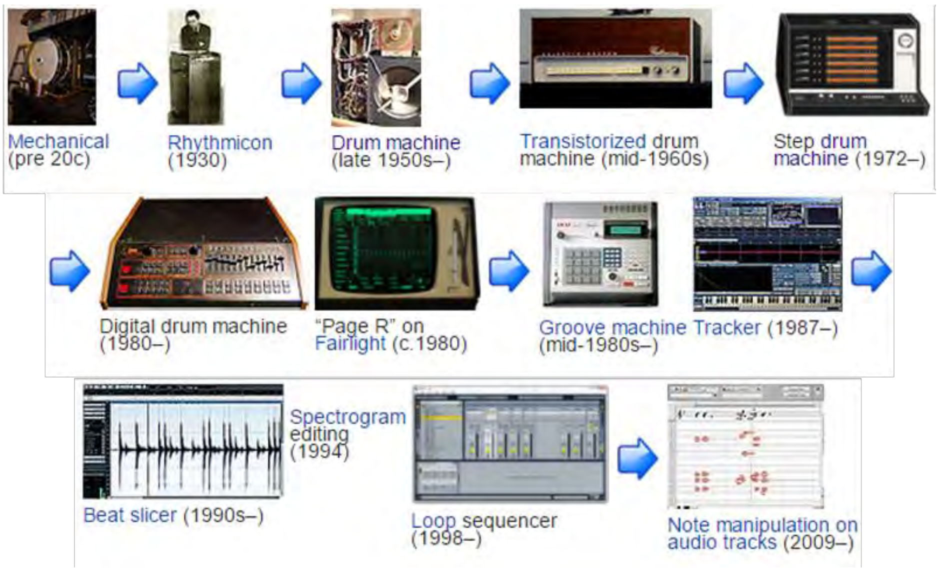


**Figure 10 History of drum sequencers**

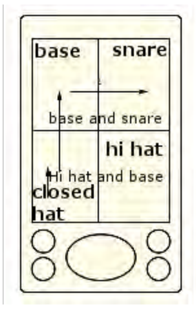


**Figure 11: Geiger Mobile drumming screen layout**

### 2.2.2 Mobile Phone Drumming and Design

Numerous mobile phone based musical instruments have already been proposed, both in the NIME and HCI communities [39] [73] [106]. A fair number of these also contain a percussive element. A report on the emergent mobile music community, being written in 2006, saw that the new technology "encompasses specificities that encourage us to reconsider the basic tenets of musical interaction" [38 p.2]. It was this thinking that led to the publication of several papers detailing guidelines, methodologies and strategies for the design of new instruments for mobile phones and gesturally controlled music [55] [120] [34]. However, a sizeable number of these influential papers [119] [59] [83] advocate for the stance of innovating new instruments and interaction methods through the study of existing gestures and sounds. Evidence of this stance can be seen when one takes a look at many of the available mobile interfaces for percussive performance. The gesture of drumming on real drums has been replicated in one way through requiring users to strike or poke areas on mobile devices in order to produce sounds – essentially using the phone as the drum [40]. An example of the screen layout for one such application can be seen in Figure 11. What this specific interface also represents, besides a recreation of the striking gestures found on physical drums, is a lingering ode to the pad based MPC style series of physical controllers. This is another direction of interface design taken by many mobile designers (including Akai themselves, who have released mobile MPC equivalents). Another popular interaction method is having the device be the "drum stick" in the interaction metaphor. This implies using its sensors (accelerometer, gyroscope, and compass) to allow users to strike virtual drum areas in order to interact with percussive sounds [68], as mentioned earlier in this chapter. This style of design thinking is taken to its logical extreme by interfaces that seek either to augment existing physical instruments with new sensors for input (such as the augmented djembe drum interface described in [67]). Another example is projects that take the approach of designing virtual equivalents of drums using the virtual modelling of their physical acoustic workings [76]. When it comes to sequencing, even less innovation can be found in research and contemporary interfaces. The desktop versions of famous step sequencers are often just used as the basis for new mobile applications. Commercially, the makers of DAWs have little incentive to stray from established designs and the emphasis of product development has been the creation of handheld versions of desktop products [52]. This style of design often leads to unwieldy and impractical designs, not suited for small screens or leveraging any of the advantages of touchscreen interfaces, with Figure 12 below showing an example of this.


**Figure 12 ReBirth iPad app**[3]

One notable exception in this regard comes from the company Proppellerhead, who recently launched their Figure iPad app with its compelling new interface for percussion programming [87]. Figure uses pattern based sequencing for percussion programming. The interaction works by having users select a pre-made pattern for each of the 4 chosen percussion instruments. When a user touches (taps or holds) a finger in the blue bar below the chosen instrument, the instrument will play the pattern at the chosen tempo. By swiping left/right or up/down, though, users can change either the pattern playing or the instrument sound dynamically during performance. Sequencing is achieved by letting users record a performance. This interface design is far from being the most expressive, but it does show that new interfaces for percussion programming can be successfully.

[3] https://itunes.apple.com/za/app/rebirth-for-ipad/id401704148?mt=8

.built around multi touch devices without simply copying previous interfaces designed for other devices.

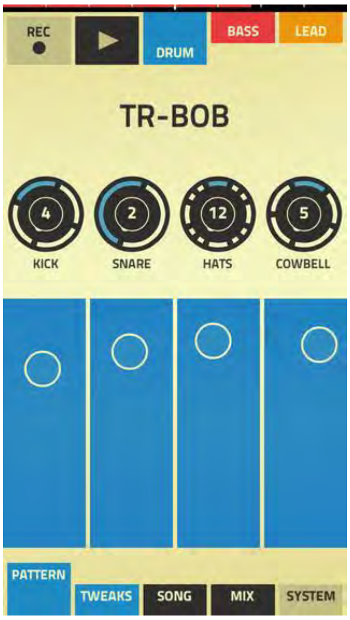


**Figure 13 Figure iPad app interface**

This method of working (taking inspiration from physical interfaces and instruments) may indeed produce interfaces that are familiar to users. The issue, though, is that the digital nature of these instruments means that designers need not adhere to the design constraints which informed interactions on physical devices, as shown by the Figure application. In this way truly novel interfaces may be built. With the design of truly novel interfaces, designers will be mindfully changing the instrument's conceptual capabilities, allowing users new ways of establishing relationships with, interacting with and playing music [55].

## 2.3 Gesture Based Interfaces and Direct Manipulation

The now universally accepted concept of direct manipulation has a long history spanning multiple decades [35]. As technology has continued to progress, new input devices, such as multi-touch screens, afford newer forms of direct manipulation. This has allowed designers to bypass the need for a mouse and allowing users to touch graphical interfaces directly. When considering that direct manipulation depends heavily on physical action and the visual representation and rapid reversible actions [97], it is easy to see how multi touch screens might best suite it. Indeed, it was theorized nearly 20 years ago that the link between gesture-based interfaces and those offering direct manipulation is one worth exploring [91]. This led also to researchers proposing so called "post-WIMP" interfaces as they realized that WIMP was a paradigm that was flawed when it came to use on touch screen interfaces. An example of one such paradigm was PWIG (Page/Frame, Widget, Icon/Button and Gesture), which placed gestures squarely in the centre of all proposed designs within the paradigm. What made researchers at that time see gestures as an attractive interface metaphor in this context was the fact that an action as well as its operands and parameters could all be communicated simultaneously with singular, easily drawn gestures [91]. This step towards gesture based interfaces from direct manipulation becomes even more of a logical progression when considering what Ben Shneiderman himself considered the three features of direct manipulation which best described it, namely [97]:

1) The Continuous representation of both objects and actions of interest
2) Physical actions or presses instead of complex syntax
3) Rapid Incremental reversible operations whose effect on the object of interest is immediately visible

The first two of these features are directly accounted for by gestures when defining them as handmade marks used to give a command to a computer [91]. With gestures, the continuous representation of actions and the use of physical interactions are more easily supported.

Still, many sequencer interfaces don't take of advantage of multi-touch gestures when they either implement switch state matrix button style interfaces or interfaces on which the user strikes to create sounds. This shortfall becomes more pronounced when one considers work in embodiment. This work suggests that interfaces that constrain users' gestural abilities are likely to hinder their thinking and communication [61]. This idea is centred on the idea of embodiment, which can be described as the premise that the particular bodies we have influence how we think [49]. With enaction being the idea that organisms create their experiences through actions [49], one can go on to conclude, when looking at these two ideas, that bodily action is not only an expression of preformed ideas; bodily action, including gestures, are actually a part of the formation of these ideas and concepts [49]. Even in conversation, gesturing has been shown to help lighten cognitive load in adults and children, further showing its importance [41]. As Gallagher concludes [93], even though bodies naturally gesture, gesture is never only a motor phenomenon; it draws the body into psychological and communicative orders defined by their own pragmatic rules. These findings become particularly salient when one considers interfaces for musical expression. There, gestures and interactions are less often to convey and manipulate information (cut/copy/paste) but to express new creative ideas and inspirations. This observation has led some researchers within the field of NIMEs to state that embodiment is key to the development new interfaces. This is especially if the interfaces are to reach any level of intimacy, with intimacy here being described in this case as a user's perceived match between the behaviour of a device and the control of that device [33].

The implications of this work in embodiment for interface design are tackled by Klemmer, Hartman and Takayama in [61] in his paper on Why Bodies Matter. Here the authors give five themes from research in embodiment that apply directly to HCI and design. Some of these apply directly to this project and its aim of designing a new musical instrument. Of particular interest is the idea that since people learn through doing, interactions should be designed to encourage embodiment. To this end, the authors offer the concept of Reflective Practice: the finding that design and evaluation tasks are worked through by people, rather than just thought through. This points to the fact that physical action and cognition are interconnected [70].

When we gesture, especially with continuous gestures, we are not simply acting, but also creating meaning while doing so. This is often how we experiment and interact with tangible objects in the real world. We begin these gestures often not with an end goal in mind, but rather working through the possible endings that the action might have. Thus, through these gestures we explore, and therefore learn as well, while acting. In this way there becomes an added element of vagueness when operating with continuous gestures. This is a vagueness that encourages exploration and leads users in unexpected directions, which should engender creativity. All of this rich information and vagueness is lost with discrete gestures, though, that allow no time for introspection during the enaction of the gesture [61]. By moving away from these discrete gestures, such as those that require striking a multi-touch display, towards those that are continuous (such as a stroking, or drawing gesture), one should be able to leverage some of the cognitive advantages of gestures stated above..

Stroking gestures have been explored before within the HCI community [125] [20] [48]. Some have been pen-based [48]; others based on marking menus [62] [63], and others [6] in which stroking gestures were used to replace keyboard shortcuts. Many of these projects (especially [6]) reported clear cognitive benefits, in line with what we would expect with regards to the findings above. In music, a few multi-touch interfaces exist that employ stroking gestures but these are mainly for the control and synthesis of continuous, pitched sounds – much like digital Theremins. Our goal, then, is to explore if similar cognitive benefits can be afforded by abandoning discrete gestures for the control of drum sequencers

## 2.4 Technology Probes

In order to best explore the benefits of continuous gestures and the effect they have on creativity, we had to embody these ideas in a touch-based system for music creation. Specifically we settled on Technology Probes, which would allow us to explore the design space afforded by new forms of input [50]. A technology probe is defined as a technical artefact deployed to find out information about a design space. They were deployed with the combined goals of collecting useful information, field testing, and inspiring users and designers to think of new kinds of technology to meets their needs [50]. Technology probes are flexible, lightweight and often intentionally open-ended artefacts. They differ from prototypes in the goal they are deployed to achieve – that of illuminating a design space by opening up a dialog with the user [50]. The flexibility of technology probes allows users to impose their own desire and usage through the technology, freeing designers from having to consider all (or even any) usage scenarios [71]. This flexibility, though, does not mean the technological artefact should be one that is of a low standard, as probes are often best implemented as high fidelity artefacts that resemble late prototypes. The main difference is that probes are deployed in a manner that accepts uncertainty of use and unanticipated behaviours. The design of the probe itself followed User Centred Design (UCD) guidelines, in an iterative nature, gathering user feedback at several stages of development. The details of this development will be given in section 4.1 of the thesis. The process followed here was similar to that carried out by other recent digital instrument design projects, such as the Thummer Mapping project [83].

## 2.5 Concluding remarks

In this chapter we have taken a look at the broad history of mobile music making applications and interfaces suited for percussive programming. This history was drawn out as various threads that led together in the end to the current state of the art. First the history of virtual musical instruments and NIMES was laid out, from its early conceptual beginnings underpinned by researchers such as Alex Mulder to the formation of the NIME conference – including work in the modern mobile NIME community. A summary of the development of drum machines and interfaces for percussive programming was also given. Through this it was shown that despite developments in sensor and input technologies (such as the advent of multi-touch screens) these interfaces were not being updated to accommodate new platforms.

It was also shown that while designers of NIMES were often found to be recycling tried and tested design ideas and metaphors, new research into areas such as embodiment and direct manipulation showed the potential for more usable yet less conventional interfaces. The usability in this instance often meant lowering users' cognitive loads, allowing for learning through doing and encouraging exploration. These are all attributes that one would want in an interface for musical expression. By employing stroking, for example, as opposed to only discrete gestures, one might be able to create percussive interfaces with previously unexplored advantages, and therein lays the potential discovered through this research.

Technology probes were then identified as the most appropriate method of exploring this new design space by leveraging users' own experiences. Though technology probes differ from more traditional prototypes in how open ended they are by design, it has been shown that they can still be completed to a high enough level of fidelity while remaining flexible to different interpretations of use.

# 3 Design Goals

In the previous chapter a summary was given of the work that inspired the project, both in terms of projects with elements that influenced the project in terms of work to follow and in terms of gaps in previous systems that can be filled. This chapter looks at inspiration from the field of experimental music and its influential composers, drawing from their works and views on music in order to establish a set of goals from which the design of Xen was based. These design goals serve as the providence for the design decisions taken in the next chapter, which will give a detailed description of the final design of the probe used for the evaluation.

## 3.1 Probe Design

The inspiration for the probe application is a drawing metaphor, implemented through stroking gestures on the multi-touch screen. The advantages, from an ergonomic standpoint, of implementing both gesture based interfaces and, more specifically, stroking gestures (as opposed to striking or poking gestures) were described in section 2.3 in the previous chapter. The initial design of the Xen probe proceeded by combining these as a drawing gesture with which users could specify musical commands on a graphical interface. The idea of drawing not only fits the chosen goal of having stroking gestures, but is also a metaphor with a real world equivalent that is accessible and should be well understood by users [111]. Translating the act of drawing into a musical metaphor is not a trivial or simple task though, but it is one for which there is already precedent in the world of music. Taking example from the writings, works and designs of influential individuals within the realm of experimental music, a set of design goals were drawn up. These goals were arrived at through a review and synthesis of relevant work from the realm of experimental music, and the work that inspired them will form the reminder of this chapter. The goals are as follows:

1) Implement composition as a drawing metaphor
2) Have the drawn lines representing musical events on a discrete time-line
3) Create an open and free interface which imposes as little compositional style on its users as possible
4) Allow for non-linear representations of time
5) Find an appropriate analogue for all of structures represented by traditional scores

What follows in this chapter is the theoretical founding for each of these goals. This will be given by means of a synthesis of the important work by composers and theorists associated with the general field of musical timing, which is what percussive sequencing is concerned with. Of particular importance is the final goal and the findings that lead directly from considering it. This is as the analogues for the score structures lead directly to what the design of the final probe looked like. Therefore, after the rational of each goal is given, more writing on the link between these structures and goals and the actual design of the probe are given.

### 3.1.1 Implementing composition as drawing metaphor

The idea of using drawn or graphic interpretations to both describe and play music has a long tradition in the world of music, specifically in the world of avant-garde classical music. One notable composer in this regard was the late Iannis Xenakis, after whom the system has been named. Xenakis was a Greek composer, architect and music theorist is generally considered as being one of the most important and influential post-war avant-garde composers[4]. Xenakis had received formal mathematical training and a degree in civil engineering before training himself in architecture under the influential designer Le Corbusier. An example of his work in this role can be seen in his design of the Philips Pavilion in Figure 14 below.

**Figure 14 The Philips Pavilion, Designed by Xenakis**

Xenakis composed pieces based on mathematical and statistical models, architectural concepts and an exploration of spatial concepts. He initially explored these concepts in the space of traditional instrumentation. Soon, though, he began researching electronic and computer-aided composition schemes [69]. It was at this point that he developed the UPIC (Unité Polyagogique Informatique CEMAMu), a computer system that could translate graphical images into musical results. UPIC is the main inspiration that this project draws from Xenakis' work, both in its interface and interaction models as well as in the philosophy behind some of its core aspects.

#### *3.1.1.1 UPIC*

The idea for the UPIC was born when Xenakis was working on the score for the piece Metastasis, where he used graphical curves, similar to the architectural designs he was learning to create, in order to represent gradual changes in pitch [69]. At the time, though, Xenakis had to then convert these drawings manually into scores for each of the 61 players in the orchestra. This was an arduous task as graphically, the pitch changes were represented by straight lines on a time vs. musical scale graph. Thus the idea of creating a computer system to allow the creation and playing of graphical scores was born.

The interface for UPIC is based on a set of notation based graphical objects (Figure 15 and 16), all of which in turn can be made up of one or several graphs, depending on the object type. Each of these object types plays a specific role in the sound synthesis, with no other hidden objects or parameters involved, in an effort to have users fully in control. The main object is the *page*, on which users can specify pitch vs. time graphs. The qualities of the sound generating the pitch and of the timing with which the graphs are played are controlled from the other objects. Examples of these are the *wavetable* object which defines fundamental aspects of the sounds generated, *envelope* object which defines how this sound changes over time, and the *sequence* object which was concerned with rhythmic timing. By implementing the interface in this open manner, Xenakis allowed users to control all aspects of both the sound generation and sequencing.

[4] http://www.theguardian.com/music/tomserviceblog/2013/apr/23/contemporary-music-guidexenakis

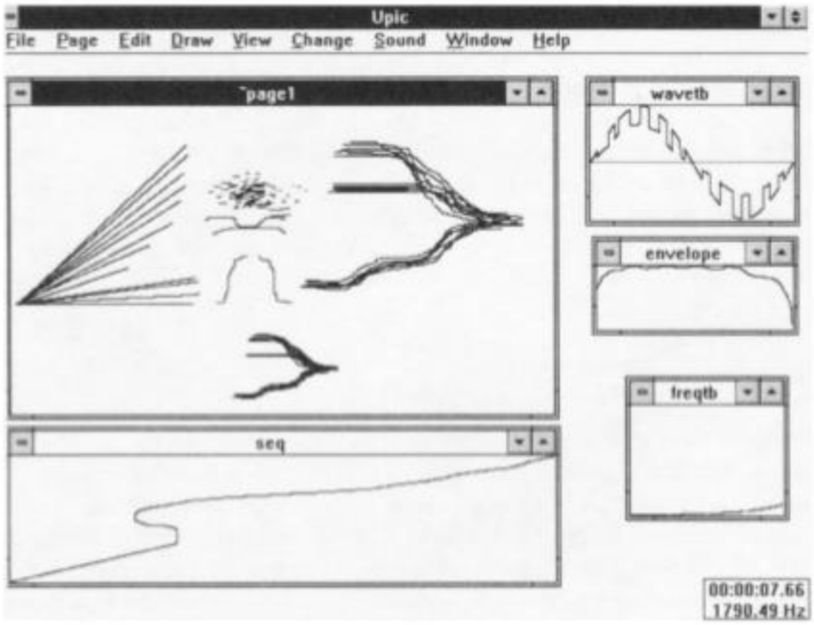


**Figure 15 The UPIC's objects**

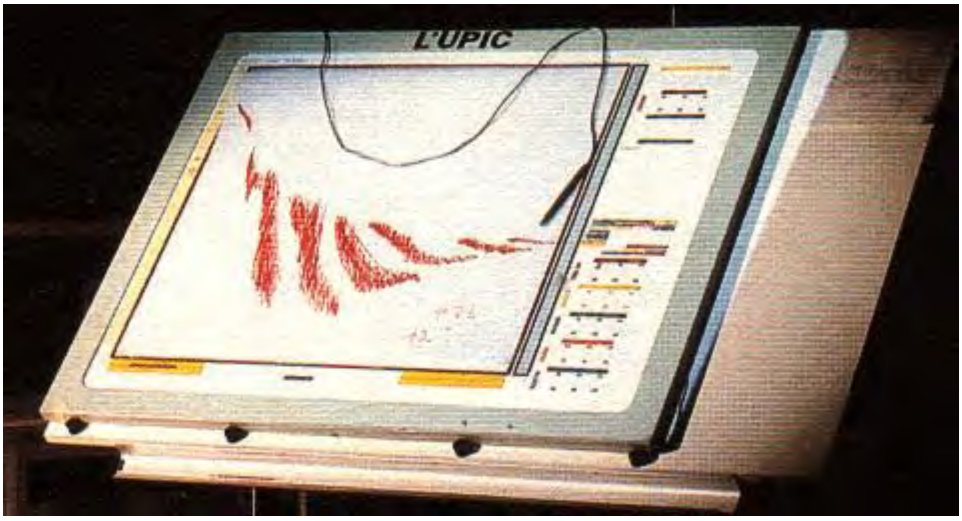


**Figure 16 UPIC in Xenakis' studio**

Of utmost importance was this idea of the composer's autonomy, as well as having fast real time interactions between the system and composer. This allows the composer to get the results of her work directly, so that the exchange between thought and ear is made easy and immediate [69]. This was a sentiment and aim, as discussed in the previous chapter (in section 2.3), which is shared by many designers and prominent thinkers within the world of HCI, Direct Manipulation and Embodiment. UPIC was an early proof that all these concepts, as mentioned before, have pertinent use cases within music that go beyond metrics as simple as the quick completion of tasks. Xenakis himself hinted at similar advantages to using drawing interfaces, saying that when composers used his system their hands and actions are not only guided by their eyes, but also their ears. He asserted at the time that the main advantages of the drawing gesture, and real time execution, were not to do with speed of execution, but rather the possibilities of auto-regulation that they bring [69].

### 3.1.2 Have the drawn lines representing musical events on a discrete time-line

In traditional music, and the scores that represent it, pitch and time are discontinuous in nature, and are structured differently from culture to culture [69]. These dimensions – the pitch (or scale) and the time - are so important to all the music represented by the score that they are often represented separately [19]. The same approach is taken by the UPIC system, which specifies these in the frequency and sequence tables [69]. Of particular importance to this project was the way in which the time was represented via the sequence table. UPIC allows two methods for the playing of graphs. The first is a linear, traditional manner with the read index beginning at time zero and moving forward at a constant tempo, adding a constant value to the current read position every six seconds. The second mode allows for more variation by adding the current value within the sequence table to the read index, as opposed to a constant value. This mode allows for more complex views of time, with reversed playing, tempo variations and jumps to non-continuous places in the page all possible.

Xenakis' thinking on time which led to this specific feature of UPIC are highlighted in his writing concerning time [126]. He asserts that time for the musician is only perceivable through reference events, on condition that they are inscribed somewhere – even if this inscription is only in memory. Thus, due to the principle of chronology, this time is equipped with a structure of total order, in a mathematical sense. This image of time as a chain of reference points can be represented in a one-to-one manner with a set of integers (or real numbers), and can thus be counted. This counting is represented in music by the metronome, which can be placed in one-to-one correspondence with the points of a line and can thus be drawn. This is done in the sciences, but also in music. Xenakis asserted then that it would thus be possible to design rhythms as temporal architectures [126].

A tentative axiomization of this thinking about time was given by Xenakis as [126]:

1. We perceive separable events.
2. Thanks to separability, these events can be joined to points of reference in the flow of time. These points are these points can be seen as leaving a trace in memory, which exists outside of musical time.
3. The comparison of the points of reference allows us to assign to them distances, intervals and durations. A distance, translated spatially, can be considered as the displacement from one point to another. It becomes a not a temporal jump, but a spatial distance.
4. It is possible to repeat, or to link together these steps in a chain.
5. There are two possible orientations in iteration: one by an accumulation of steps, the other by a re-accumulation. From here, we can construct an object that can be represented by points on a line.

This gives a framework from which time and rhythm - the main musical attributes with which percussive music and sequencing are concerned - can be represented in a drawn manner. This shows that from a musical standpoint the idea of using drawn representations for timing based tasks is a valid one. This thinking was also adopted in the design phase of the Xen prototype – that musical events can be references linked by a timeline of integer values that can be represented by a drawn line. Xenakis himself saw the benefits of drawing and sketching, which is why he applied a drawing analogy to his UPIC system. Important evidence exists that some of the advantages of drawing and sketching derive directly from their vagueness, which has been observed in architectural sketches [96] [42]. This is a result that not only validates the earlier reviewed research in embodiment and direct manipulation, but also one that has pertinence in musical interface design. Surveys have shown that vagueness is integral to the ways composers get started with music composition (and often through sketching). When looking at the goal of engendering creativity in users, a drawing metaphor also holds advantages, as argued by Amitani and Hori [111]. They concluded that externalization of music into a two dimensional space supports musical creativity in composition by fostering changes in the way information is represented [111].

### 3.1.3 Create an open and free interface which imposes as little compositional style on its users as possible

Another core idea for the UPIC system was to allow composers and users of the system to define and control all aspects of their compositions: from the sounds and symbols used to the syntax and so on. What this meant was that the system would not impose any predefined sounds, compositional processes, structures or musical ideals. In essence, it was seen as "essential for the creative mind that ideas not go through theories or limitations that might not suit the composer" [126 p.85]. This sentiment was of utmost importance when first conceptualizing what the Xen system would eventually be. In order to properly engender creativity in users, an interface should ideally not impose any concepts or musical standards, but rather let users roam free and explore. While, much in the same way as it is nearly impossible to write from an entirely objective perspective, it might not be possible to have an interface entirely neutral to all musical use cases [36]. Even so, interfaces that impose less than those interfaces such as the one found in Ableton, with its default tempos and time signatures (for example). An example of how the UPIC system circumvented this problem can be seen in the various levels at which the interface allows users to control the music being created. It allows users to specify, firstly, the waveforms that form the fundamentals of the sounds being created, secondly to specify the notes being played by these waveforms, and lastly to specify the entire composition and how it flows through time. Traditionally, only the second of these two layers is supported in sequencing applications such as those mentioned in the previously in section 2.2. Hence, allowing for this level of musical control also became a goal when initially designing Xen.

An analogous way of looking at the system is comparing drawings created on it as notes on the score, with the score being represented by the page. The control objects that define the attributes of the sound are then the properties of the instrument playing the note. Being able to create a conceptual mapping such as this shows that the UPIC, while novel in its time and imposing little compositional style, could still encompass tasks demanded by older methods of composition as a subset of the wider range of uses it offers.

### 3.1.4 Allow for non-linear representations of time

This project is not the first to take the timing concepts founded by Xenakis and his UPIC system as inspiration for the development of a new system. Iannix is a desktop computer interface inspired by the UPIC, developed by La Kitchen Company and concerned with answering questions of time. It had been argued before by Vaggione [115] that representations of time should follow the various time scales that composers operate. It has also been argued, relatedly, that the linear, horizontal representations of time prevalent in most musical interfaces today do not allow for the construction of parallel dynamic events that evolve at their own rates [111]. In order to address these shortcomings, La Kitchen developed a system that allowed users to specify two or three dimensional abstract shapes. These shapes exist individually, run concurrently and have their own behaviours.

This implementation successfully addresses the goal of moving away from linear, horizontal representations of time, but the interface fails in that it only allows users to create shapes derived from a predefined set of lines, circles and curves [111]. These shapes are modified using JavaScript, and the system itself must be re-wired to a sound-producing source in order to actually create musical output. Recently, it has been used more for facilitation of digital arts projects than as a sequencer [111]. The design of Xen also looked to take a similar approach to circumventing the problem of linear representations by allowing for separate objects with their own behaviours. The interface, though, also would have to offer interaction methods far more usable and suitable for mobile devices than those offered by Iannix.

### 3.1.5 Finding appropriate analogues for structures represented by traditional scores

Xenakis' work was followed by others such as Kristian Vester[5] (aka Goodiepal/Gæoudjiparl) who also showed an active interest in the strong dependence on linear time representation in modern notation and computer music [15]. His approach, though, advocated removing time altogether in representation, which is an extreme step. Regardless, though, in deconstructing the idea of a linear score to its fundamental parts, his work does show a realization that that having a score implies the mapping of musical events occurring in one or more spatially different regions onto a single temporal axis [15]. Considering this, when first looking to design new concepts upon which the interface would be based, the definition of these musical events was inspected. Buxton, Reeves, Baecker, and Mezei [19] looked at this, and defined musical events in this context as events that occur during the course of a composition that have a start-time and an end. Thus an entire composition could be seen as an event, as could a single note, or an ordering of notes. This is an inherently hierarchical view of musical compositions which has the advantage of, by its very definition, addressing every possible aspect of a piece of music. When looking at percussive sequencing though, not all events are immediately important (such as chord progressions). Since one of the most important steps of designing a new musical interface is deciding which data structures will be used [19], this step was taken early on in this project, before any interface elements were decided on. In the end, the following parameters were seen as salient:

1) Instruments: the items that produce specific sounds
2) Notes: A single instance (or hit) of a percussive instrument
3) Sequence: a collection of notes
4) Score : a collection of sequences, representing a body of work
5) Time/tempo : the passage of time

Thus, these five data attributes formed the score structures that would need to be represented within the interface. In order for the probe to have proper musical significance (as well as being founded in sound design and HCI principles) proper analogues for these attributes would need to be founded.

## 3.2 Moving from Goals to Design

It was with all this in mind that the concepts of **nodes**, **paths** and a **pendulum** were arrived at. Nodes represent a single note instance of a specific instrument and paths a collection of notes linked on a temporal axis by the pendulum, which represents the movement of time. In this way sequences of

[5] Former head of Electronic Music Department at Danish Institute for Electro-acoustic Music (DIEM)

notes can be connected by paths, on which the pendulum moves in order to represent the movement of time. Thus the goal is to have independent objects (paths) each running with their own behaviour and time scale (defined by the pendulum). In this way paths could exist solitarily on their own temporal axes, and interact with nodes in a manner more natural than the score, as described above. Once these overarching simple concepts were reached, we started to explore if it was possible to create a system based on them.

The interface was designed in an iterative manner, in the UCD tradition. Early prototypes of the probe were developed with reference to the theories and literature listed above. These ideas were further distilled and refined through quick and dirty user evaluation sessions. As soon as it was possible, ideas were embodied in software on a device. As the goal was to explore how music could be expressed on a multi-touch device, it was important to start developing software early in the process. The other goal was then to use the probe to gather as much feedback. Therefore an agile development process was adopted, wherein the most recent bug-free version of the code was installed on the developer's handset which was then used to show potential users and gather ideas at every opportunity. In the next chapter is a description of the final probe and the reasoning behind the design choices taken.

# 4 Design and Implementation

This chapter is both a thorough description of the interface of the application that served as the technology probe in the final evaluations, and also an elaboration of the design choices that went into it. It will begin with detailing the design process and the steps taken, before giving a thorough overview of the final probe design used for the evaluation carried out in the proceeding chapter. At the end of this chapter a brief look into some of the details of the back end and code design that went into the production of the probe will be given, though this will be constrained to only the most important details

## 4.1 Design Stages

The initial design of the system was based on the body of existing literature in the subject of mobile drum sequencing. Having taken a look at the current state of the art in the field in chapter 2 and identified potential gaps in research, the design began with synthesis of work from within the fields of music and interface design in order to inspire the beginning of a new system. This first step was detailed above, where the work synthesised was detailed, as well as how this work inspired and was fed into the initial design goals. From these goals the user interface was constructed (with code being written directly for the devices after a failed experiment with prototyping first on a Web based platform).

Once an initial prototype was completed, opinions on it were gathered in very informal sessions with various users. The first group of users was two musicians whose opinions on the interface were gathered after a demonstration of the interface's features and an explanation of the project's research goals. These two users were both 24 year old, experienced musicians (drums, bass guitar) who also had experience with DAWs (Ableton, Logic). The next set of users were volunteer users at a graduate demo evening on campus (7 in total). There, a poster was put up explaining the goals of the research and asking willing participants to use the software and give feedback. During the development of the probe, the decision was taken to always push the latest version of the code to an available device as soon as possible. This allowed for the interface to be seen and critiqued at several stages by the other researchers in the laboratory where development took place. This group of peers was the final group, and also the group with which the interactions were the most frequent. While this was very informal, data from these sessions was extremely useful in helping to refine the initial prototype. At this early stage, the questions being asked of the prototype were to do with usability issues and the design choices made, not yet trying to prove the broader concepts that led to the development of the project. The aim here was to assess the implementation of the ideas established in the chosen literature and to discuss possible alternatives.

All of the suggestions and comments received during this early stage of informal assessments were reviewed to see which would be implemented. These decisions were made on the basis of whether the suggestions:

- Could be completed in a timeous manner (considering the project's time constraints)
- Whether the changes were cosmetic or to do with usability
- Whether implementing the change would undermine one of the project's goals

After all the suggestions were noted, and a set of these was chosen as the ones that would have a positive impact on the project if implemented, the project went back into a design phase. Some suggestions were simple to implement, from a design perspective, and for those with less trivial solutions, HCI and design literature was consulted in order to find a suitable solution. After these were implemented the system was considered to be complete enough for formal usability testing sessions detailed in the next chapter. The description of the system below will be of the final state of the probe and, where necessary, design changes inspired by users during the preliminary informal evaluation will be pointed out. These sessions, and the methodology used to conduct them, will be described in further detail in the next chapter's section 5.1.

## 4.2 System Hardware

Xen is a multi-touch mobile drum sequencing application that was developed on the Windows Phone 7.8 operating system and deployed on the Nokia Lumia 900 phone. The Lumia 900 is a mono-block, multi-touch device housing a large WVGA (800 * 480 pixel) screen; with a diagonal measuring at 109.22mm that dominates most of its 127.8mm by 68.5mm size. It can be comfortably used singlehandedly, although reach on the touchscreen is limited by the reach of the holding hand's thumb. Two handed operation with the phone in landscape orientation allows for easier access to more of its screen estate. In terms of other input sensors besides the multi-touch screen the device also houses an accelerometer, gyroscope, proximity sensor and magnetometer. Access to all of these is given to users via the Windows Phone 7.8 SDK, along with access to pre-built UI tools for quickly creating applications. The SDK also comes with support for a mobile port of Microsoft's XNA framework, which is what was used for this project. XNA is an environment designed to facilitate video game development on the .NET framework supported by Windows PCs and X-box consoles, and was later ported for Windows Phone. Being an IDE created for both 2-d and 3-d game development, it comes pre-packaged with classes and libraries for creating unique graphics and custom GUIs, as well as for facilitating sound and collision detection tasks. The port to Windows phone allows access to the phone sensors for use within the XNA game loop, giving the programmer raw data as opposed to data that is filtered in any way (for example raw 2-d touch co-ordinates). Finally, content management and audio support are more sophisticated within XNA than when only using the standard SDK. The main drawback is that since there are very few standard widgets within XNA for Windows Phone, most graphical assets have to be sourced from elsewhere. Thus all graphical elements used in the interface were created and edited in the GIMP open source image processing application, and all audio resources were taken from free online repositories.

## 4.3 Probe Design

### 4.3.1 Initial Design and Drawing Metaphor

The first decision to be made when looking at the design of the system was where, and how, to incorporate the drawing element via the stroking gesture. When looking at the goals listed in section 3.1 of the previous chapter, as well as the taxonomy of possible musical events that can be represented, having the path (which represents the passage of time) being the element that is drawn makes the most sense. This is especially true when considering the earlier quoted findings by Xenakis on time, and also because the paths, or sequences, do form the main part of the operation of sequencing. Thus the user could draw paths, which show visually how time will flow via how the pendulum moves across it. This design decision also assumes that the user has a two dimensional space to work in, and in order to be able to traverse the space in an efficient manner, pinch zooming and drag scrolling were to be implemented. This makes the phone display like a window through which the user can see a much larger *canvas* area. This would allow users to have enough space to potentially have many path drawings concurrently running without worrying about running out of space, although limits to the zooming levels and maximum scrolling amounts were placed in order to prevent users from losing their place in the interface. Pinch zooming and drag scrolling were chosen mainly for their universal acceptance in modern devices as an interaction model. This did lead to a conflict though, as it makes it difficult to differentiate between drawing gestures and those intended to be dragging ones.

### 4.3.2 Assigning Gestures

One potential way to accommodate two different interactions, such as drawing and scrolling, that have similar gestures, is by the implementation of a multi-modal interface. In this case there would be a canvas mode in which users could scroll and traverse the 2d plane, and a drawing mode on which they could specify new paths using stroking gestures. This approach, though, was one that was not advised for implementation by users during the preliminary evaluations stage of the probe's design. This was firstly because users wanted to act as soon as inspiration hit, and having to first enter the drawing mode would somewhat stifle the exploratory nature the interface tries to foster. Secondly, when in drawing mode all the loops that had already been specified would stop and no longer be accessible, meaning that users would have to specify loops while not listening to any of the progress they had already made.

This issue quickly became a complicated one to try and overcome, as any solution would require shuffling of the meanings of various gestures, as well as a change in the potential visual display of the interface. In terms of efficiency, it was easy to see why choosing not to implement multiple modes would be advantageous. Research into mobile interaction design has also shown that having modes in interactive systems is known to cause users difficulties [54]. There is one legitimate advantage of a modal implementation, though - that of having a greater amount of screen estate available for drawing. This is as drawing would have a dedicated mode where more potential estate, and gestures, could be dedicated only to drawing. Thus, when designing this aspect of the interface, and finding the balance between the advantages and disadvantages of any one approach, the two main goals of this particular part of the design became:

- Needing to differentiate between drawing and scrolling.
- Having as much screen estate available as possible for the drawing of new loops.

Of the HCI literature and other projects found that faced similar problems of screen estate and supporting multiple gestures, the Bezel Swipe gesture system was found [90]. It was developed in order to solve the exact same problems faced by Xen: having conflict-free scrolling and zooming gestures in an interface that relies on recognizing a variety of other gestures. A Bezel Swipe is a swiping gesture that starts on the phone's bezel - the physical touch insensitive area that surrounds a smart phone's screen. As these gestures move onto the touch screen they are recognized as being different gestures from those which originated from inside of the screen. This way the designer has one more set of gestures to use when working out the fundamentals of their systems and, importantly, normal swipes operate differently from Bezel Swipes. The researchers, when conducting usability tests using their system for image selection tasks, found it to be a viable alternative. It is for this reason that it was decided to take inspiration from this project for the probe and introduce a small scrolling area on the prototype where users can begin scrolling using a dragging gesture. This way the rest of the screen can be a permanent drawing area on which users can specify new loops.

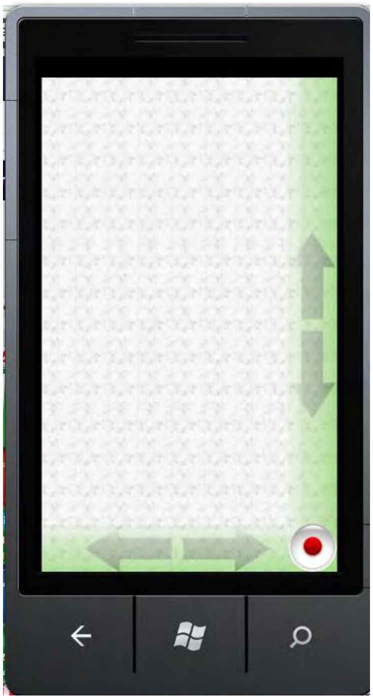

**Figure 17: The drawing and scrolling areas**

These areas on the right and bottom edges of the device are shaded in green with directional arrows intended to show in which directions the users can scroll. The visualisation of the area is shown above in Figure 17. It was decided that it would make more sense to sacrifice a minimal amount of screen estate rather than using the actual bezel of the device. This was because doing an up and down dragging gesture, for example, would require the user to only begin their swipe from the bottom of the screen and not the side, which can be inconvenient. Instead of using all of the screen edges, only two were used in order to maximize the space in which users can draw loops. Thus the entire non shaded area in Figure 17 is the drawing area on which users can specify loops simply by placing their fingers anywhere and initiating a stroking motion. Although this implementation shares some similarities with traditional desktop scroll bars, it differs in small yet consequential ways. The main difference is that while these scrolling gestures begin in the predefined area they are not confined to them. This one could begin scrolling left or right (from the bottom area, for example), have their gesture move onto the main drawing area, then also scroll up and down in the same gesture. Thus the gesture is still a flexible one that allows for a full set of two dimensional moves, with the only restriction being where the gesture begins.

The left and bottom edges of the screen were chosen because these are the closest edges to the thumb when a phone is used in a single handed manner (a similar rationalization was used in the design of the Fat Thumb system in [16]). Unfortunately this rationalisation does not hold of left handed users, although in landscape mode (i.e with users sung both hands) there should be minimal difference in usability. This would allow quick access to all sets of functionality through a set of easily distinguishable gestures:

- Zooming via multi touch pinch zooming
- Drawing by using a regular stroking gesture
- Scrolling using a Bezel style swiping gesture.

An effect of choosing this interaction style was that there was now an unused space on the phone's bottom right corner, where users would not be able to start a directional swipe or begin a drawing. One of the most requested features mentioned by users during the preliminary evaluations was the addition of recording functionality. This functionality was facilitated by the use of a recording icon. Since, with the implementation of the new swiping mechanism, there was now unused screen estate in the bottom corner of the screen (where no swipes would originate from) it was decided that this space should be where the record button should be placed. The icon chosen was one which is visually similar to the universal record sign found on physical camcorders and other devices. When recording, this icon flashes and, upon the completion of recording, the user is given the option, via a simple dialog box, to either save, playback or delete the recording of all the loops playing.

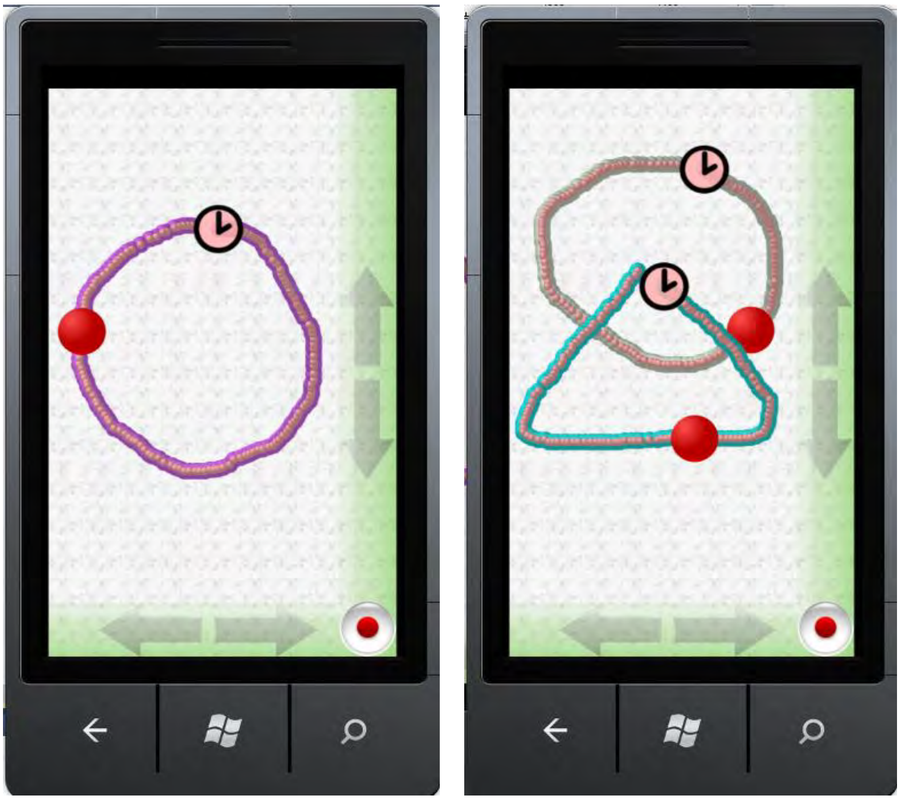

**Figure 18: Completed loops**

### 4.3.3 Loops

Once drawn, loops are represented by a continuous trail of circular points that follow the user's strokes when in drawing mode. Once a path is closed (i.e. the user's end touch position equals the beginning position for the same gesture) the red-ball shaped pendulum item flows across the path This can be seen in the first screen in Figure 18 above – which shows a single completed loop with the red pendulum. Since paths can potentially overlap and run across each other, each is given a colour at random upon completion, to help users differentiate among them. The second screen in Figure 18 shows two overlapping loops with areas that are clearly distinguishable from each other because of the randomly chosen colours. At the start/end position of the loop a clock icon is placed in order to give the user a substantial area to press when selecting a path. Even though the interface had a faster way of specifying these loops now, users were still observed to be struggling with synchronizing start times of the different loops. Even when they drew loops that they were happy with, they would often not be happy with how it sounded with other loops, often asking if there was some way to make sure they all start at the same time. This was not a suggestion that would require the quantization of any loops, or using a grid, so it would not undermine any of the fundamental elements of the probe as a research project. Luckily, as a result of having employed the bezel swiping technique, there was still an unused gesture which could be mapped to an activity: the gesture of double tapping. It was decided to have the double tap activate a pausing function, stopping all the loops running in the system. What this allows users to do is to stop all the loops, draw a new one, and then double tap the screen to have them all playing in unison. In order to further help with synchronization and to allow for more complicated timings than to simply have all loops beginning at the same time, pendulums are also individually stoppable by touching and holding them in place. In this way users can dictate when they start moving again by simply letting go at the appropriate time

**Figure 19: An open radial menu**

### 4.3.4 Note Instances and Instruments

The instruments and note instances in the system are represented by singular items: nodes. New nodes are created by long pressing anywhere on the screen while not in drawing mode. What this does is to create a new circular icon on the screen and give the user a radial menu from which the user can choose which instrument this single note instance will represent. Superimposed onto the image of the newly created node is an image of the instrument that the node represents an instance of – the same image as seen in the menu used to choose instruments. This implementation should allow for the quick identification of which node is which within a user's composition without having to first play the note instance to hear what it represents. Radial, or pie, menus are menus that offer users the options available to them via items placed at equal radial distances along the circumference of a circle, as shown in Figure 19, which is a screen shot of a new node with the selection menu still open. The centre of this circle in our application is usually also the centre of the new node that has been created, but is also offset whenever the node is placed near the edges of the device, in order to make sure that all options are visible. The options themselves are represented by icons that show the various instruments available. The use of menus is already advantageous as it facilitates recognition rather than recall. The use of images instead of text here further helps with that as humans are much more efficient at recognizing images rather than text [111]. Since there is no writing on the icons, there was therefore no logical way to order the icons (alphabetical ordering would be impossible, for example). It is generally accepted, though, that users' performances (i.e. time to seek a target) with different placement styles (even a random ordering) converges with practice [10]. These menus hold advantages over linear menus in that the distance from the centre position that the menu pops up at (often the position where the user has last pressed) to each item in the menu is equal. It also allows for a larger target area for each item – both features of good HCI design as described by Fitts Law [21}. When a user touches an instrument's icon, the sound that the instrument represents is played, and the icon is highlighted in blue. Once the user has selected the instrument they would like, they must then press the close button in the centre of the menu and their choice is recorded. If at a later point the user wishes to change the instrument represented by a node then they simply long press the node and the same radial menu appears, allowing them to make another choice. Figure 20 shows both what the nodes look like when places on the interface, and also a closer look at the icons which represent the various instrument choices available.

**Figure 20: A group of nodes, with the icons representing each instrument shown below.**

While this method works well when a user is placing their first few nodes, as they are likely to want to listen to the various options available to them in the radial menu and place the correct ones, the interaction can take long once this decision has been made. This was something that was specifically spoken about in the preliminary sessions held at the university poster evening. More than one user wished that they could create nodes more quickly, especially once they had decided exactly which node they wanted to place. Thus, it was decided to add a shortcut to avoid the need to open the radial menu, and to rather quickly create a known node. This is an action that might not benefit new users at first, but which will increase the efficiency of expert users of the system – an approach advocated for within the HCI community [78]. In order to speed up the creation of new nodes, functionality was added to copy already created nodes in a much quicker manner. This is achieved by employing a double tap and drag gesture which the users can use to copy the node they want to create and then placing it where they want on the screen. This saves users from having to constantly bring up the radial menu or long pressing in order to create a new node.

### 4.3.5 Achieving Sequencing

Once nodes are created they can be moved around the two-dimensional space by clicking and dragging them into new positions. The representation of note instances for specific instruments represented is similar to the approach taken by physical samplers such as the MPC5000 and, like those samplers, when a user presses a node it plays a single instance of the sound it represents. In this way users could, if they so wish, simply place nodes where they please on the screen and play out rhythms by tapping on the screen. This, though, is not intended to be the main compositional interaction that users engage in. If a user places a node anywhere on a path, once the pendulum reaches this position it will “strike” the node, playing the instrument, and thus sequencing can be achieved in this manner. Visual feedback, via a green splash visualization, is also given to the user when this striking action happens in order to further solidify the metaphor being used. Users could, importantly in no order, place nodes that represent instrument instances on the screen, draw shapes through them, and have the pendulum play the sequence of the nodes placed on it. Nodes can be played by more than one path’s pendulum, unlike with traditional linear sequencers with one unified timeline. Another way in which the implemented form of sequencing differs from the traditional method and leans further towards non-linearity is by allowing for nodes to be struck more than once by the pendulum during its cycle. This was functionality that was, encouragingly, suggested by users during the preliminary evaluations of the system. The implementation of this particular feature demanded the refactoring of the collision detection algorithms used (which will be described in greater detail later in this chapter’s section 4.4). It also helped move the interface closer to the theoretical frameworks on time laid out by Xenakis in the section 3.1.1. Even with all the differences from traditional sequencing that this application has, similar results can be obtained when sequencing with it. Figures 21 and 22 demonstrate exactly how this is possible. In Figure 21, a simple repetitive four to the floor rhythm has been programmed using Ableton’s piano roll sequencer loaded with drum samples. The four to the floor is a staple rhythm in much of contemporary Western music. In Figure 22, the same rhythm is approximated, using the probe, by a square shaped path with drum nodes at each corner, recreating the four to the floor pattern. It should be noted, though, that this is only one potential rendering of the pattern in Xen, and that different people might see the rhythm as different shapes that would, in the end, produce the same patterns, but within the user’s own definition of how the music should be represented.


Figure 21: A "four to the four" rhythm in Ableton

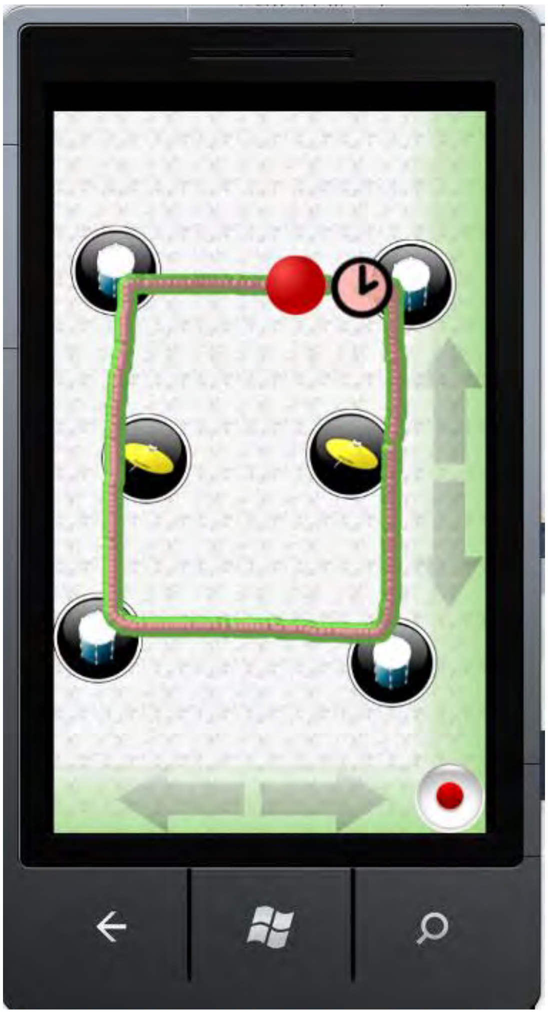

Figure 22 "four to the four" approximation in Xen

**Figure 23: The interface zoomed out, with multiple paths**

As many nodes and paths as one wishes may be created and, through scrolling and zooming, systems of paths and nodes can be placed in spatially different regions of the interface's two dimensional area. Figure 23 shows just how this could be achieved, with a number of different paths being clustered in spatially separate areas. Thus users would be able to zoom in to any of these areas, tweak a particular loop, and then zoom out to keep track of the entire composition. While pinch zooming did work appropriately for zooming, and limits were placed on how far users could zoom in our out of the system. These limits were chosen as reasonable ones by the researcher and later refined. The initial simple zooming algorithm had to be changed, though, as users found the visibility of certain interface items worse than others when zoomed to the extremes. The main culprit in this regard was the radial menu, which became inoperable at both high and low levels of zooming in the system. It would becoming either too big for all the menu options to be seen or too small to operate. The way this issue was solved was by having the interface zoom different elements within the interface at different rates. This way, even when the nodes and paths appear small because the user is zoomed out, or appear larger because they have zoomed out, the radial menu will be at a comfortable, less drastically altered. This way all the options within the menu will always be visible in all use cases. The effects of this change can be seen in Figure 24, which shows the interface in a zoomed out state, with the nodes and loops appearing smaller, but with the radial menu still at a relatively larger size. Thus the complexity of the systems that a user can construct should only be limited by their own intentions in using the application.

**Figure 24: Adjusted zooming of elements**

### 4.3.6 Deletion

The deletion of any nodes or paths is done by long pressing either the node or the path's clock icon. At that point a red deletion area appears on the bottom and right hand edges of the device when in portrait orientation, replacing the green dragging area. When a user drags the item over this area it becomes a darker shade of red and if the user lifts their finger from the device at this point the selected item will be deleted. This process that is demonstrated in the two screen shots that form Figure 25. Paths are deleted - in a similar manner, except users click and drag the metronome icon that is visualized whenever a path is created and drag that to the red area that appears. This method was inspired by contemporary mobile interfaces, such as Android, which employ a similar method.

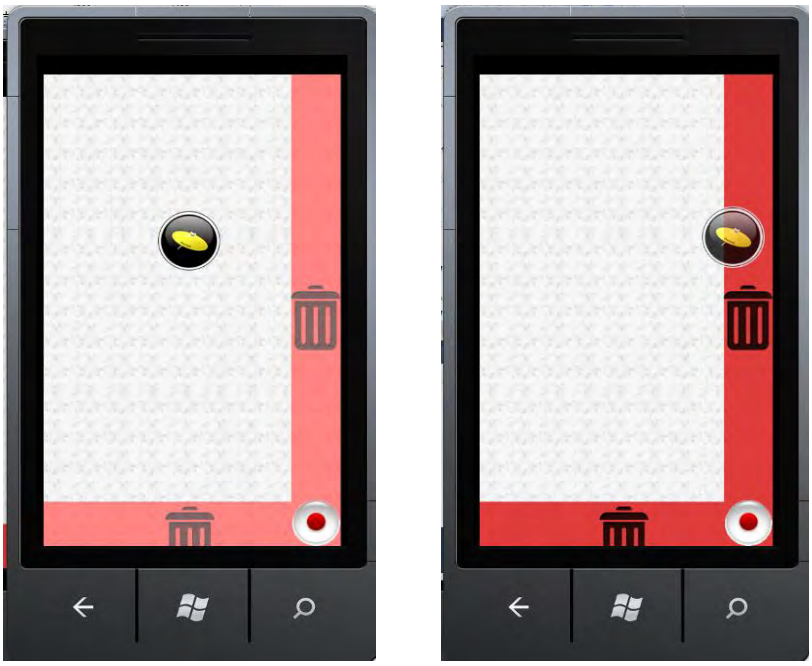

**Figure 25 Deletion of nodes**

### 4.3.7 Closing Loops and Timing

Any path that is drawn must first be closed before it becomes operational, and this was done for several reasons. The first was in order to have the interface be consistent in the interface metaphor it uses (i.e. having loops). Secondly, having incomplete loops would present a difficulty in establishing when users have completed drawing a path that users are happy with. There would have to be a more complicated method of ensuring that drawings are completed at the correct time. Lastly, requiring loops to be closed was decided upon in order to ensure that users don't simply draw straight lines with nodes placed on them, emulating the style of sequencing that the project aims to move away from. This interface therefore has users converting from time to geometry and distances when using the interface – facilitating the change of information in the way that Amitani and Hori had postulated would engender creativity [111]. This idea, of drawing out loops, also fits well within the goals of direct manipulation from Ben Shneiderman that were reviewed in section 2.3: having continuous representations of objects of interest; favouring physical action over complex syntax; and having rapid, incremental, reversible operations whose effect is immediately visible. The loops that you draw - the passages of time - are the objects of interest and can be manipulated directly, bringing them to the fore as opposed to having them only be linear and hidden from the user in a manner that doesn't allow them to be changed. The rhythms created are also directly manipulated through the use of stroking gestures as users either have to place nodes first then draw through them, or draw paths, then drag nodes into the correct positions later.

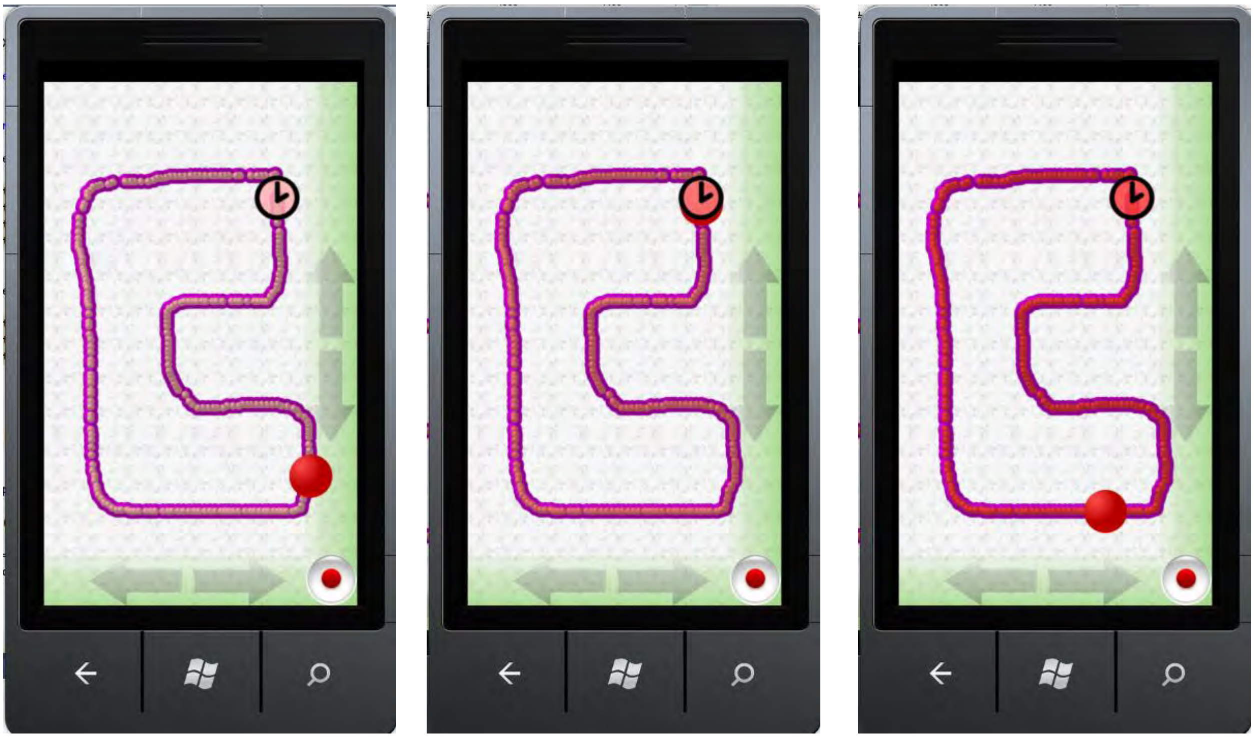

**Figure 26: Visualisation of a loop being made faster**

It should be noted, though, that while users can make quicker rhythms by drawing smaller loops, provision was also made for users to alter the tempo at which individual paths run at. This interaction is achieved by pressing and holding the path's clock icon and then either tilting the device forwards or backwards to increase or decrease the speed. In order to communicate this change to users, each path has a white, smaller, path visualization running through it. This area remains white when the path is running at its original pace, but becomes a darker and darker shade of red when sped up. In this way users can quickly see how fast a number of separate paths are running by simply glancing at their colour. Figure 26 shows, over three screen shots, what it would look like if a node were to be gradually sped up in the system, with the first image being the loop at its slowest, and the last being at its fastest.

### 4.3.8 Final Probe

The final prototype was left strategically incomplete, in some senses, as an open-ended design. When the application is first run there are no default tempo configurations or templates. There is also no way to quantize the music or have it artificially correct rhythms to any sort of grid. This was done for the reasons stated in section 3.1.3, relating to allowing music to be more expressive by having room for deviations, but also in order to fulfil some of the goals of employing a probe. With launching this prototype as a probe, it was hoped that the way in which users adopted and appropriated such an open-ended design, bending it to their will and making work best for them, would shed light on the design space that this project inhabits [50].

## 4.4 Implementation

In this section an overview of the technical implementation of the probe will be given, with further detail being given on both the overall structure and of interest and complexity, and also issues and solutions to issues found during implementation. As part of this summary, a necessary overview will be given of the XNA framework and how choosing it has helped shape the way in which the solution was coded.

XNA is a cross-platform set of libraries developed for the production of games. As with most other game development platforms, it employs a polling loop method, which entails that games using the platform do not fire events when the game receives input or needs to be redrawn to the screen. Rather, XNA games check each of the input devices when the game is updated, and then acts on this input accordingly. Thus, this process of updating – handled in the Update method that is always called before the Draw method – is an extremely important one in XNA. This method is called firstly from the main Game class in the project (which always inherits from the XNA Game base class). This Game class handles the entire logic and rendering of the game, often not directly, but through the instantiation and handling of several smaller classes. These classes will also all have Update and Draw methods which are called within the Game class' own Update and Draw procedures. Another important concept in both game development and, more specifically, with the development of musical applications, is that of timing. In XNA, games will attempt to run the Update-Draw loop a fixed number of times per second for the entire duration of the game's operation. This means that operations will run smoothly, as long as the game's Update-Draw loop is quicker to run than X milliseconds, where X is in this case 1000 / the number of Update-Draw loops per second. With the relative computational simplicity of the Xen probe, and the processing power of the Lumia 900 smartphones, this is almost always the case. Although, in scenarios where this isn't the case the default behaviour within XNA is to switch to isRunningSlowlyMode, which means skipping draw loops until the problem is solved. What this would mean for the operation of the Xen probe is that a user would still be able to hear all the audio output (produced during the Update phase) even if the image lags for a few frames – which is an acceptable compromise. This is just one example in which having fundamental functionality made for game development has benefitted the operation of Xen.

This overall framework is implemented as follows. There is a main Xen Game class, from which only background elements are drawn and very little logic is handled. Instantiated within this class is the PadView class, which handles almost all of the logic pertaining to the overall structure of the prototype, such as updating positions, collision detection, handling input, creating and deletion of nodes and paths (which are both represented by their own eponymous classes). All Node and Path instances created during operation are stored in one of two ArrayLists, with there being one for each. All touch events are sent to this PadView for processing as well, with the events being sent from here to all other relevant classes within the framework from this point too after their positions have been processed.

The Node class is a simple one, which houses nothing more than its position on the canvas, an integer representing its instrument type and a link to the audio file that this instrument represents. The Path class is slightly more complicated, in that it needs to hold its own overall position, as well as all the positions in the loop (held in an array), and the index of the position that it's pendulum occupies on this loop. The Path class also holds the necessary methods for its representation, such as that for picking its random colour on inception and linearly interpolating its shade of red depending on the velocity of the pendulum (when users increase their speed). The Collision detection between the Pendulums and the Nodes is handled by the PadView class that contains all the objects. In order to keep track of these collisions a third two-dimensional array is used, in which each position represents the possible interaction between any Node and Path. The array therefore has dimensions of the number of Nodes * the number of Paths. By implementing this third array, Xen is able to avoid having a collision between two objects registering each of the several times in a second that the update methods are called while still ensuring that nodes can be interacted with more than once in quick succession by the same pendulum. More specifically, this helps to keep the pendulums on each path from registering more than one strike of a node simply because the update method was called again.

All touch events within the system have to be translated back and forth between the screen's x-y coordinates to the world co-ordinates used in the system because of the way that the pinch zooming is handled. Zooming and movement around the system are handled by a 2-d Camera class, in another method borrowed from the world of gaming. What this means is none of the positions of the objects within the code are ever being manipulated when the user scrolls or zooms. Rather, the view from which they are being observed is being manipulated – moved back and forth around the x-y plane. What this necessitates is the conversions of the user's touch positions from the screen position (between 0 and 800 pixels on the y-axis and 0-400 on the x) to the much larger world position that the objects within Xen inhabit. Having different items zooming at different rates is achieved by drawing items at different times during the main Draw method, applying different scaling at each level.

## 4.5 Summary

This chapter focussed mainly on giving a thorough description of the interface, the process by which this interface was created and also how the designs chosen address the goals stated in chapter 3 as being salient ones to reach. Finally, a brief section on the hardware and implementation details was given. The chapter began with the procedural detains of the interface's design, in detailing the stages that the design took. These stages were namely synthesising the design goals, coding the initial prototype, gathering feedback from informal sessions, designing improvements based on the feedback and finally implementing the changes to the final probe.

The probe is based on a drawing metaphor, with the canvas area that users draw on dominating most of the interface's screen estate. Instances of percussive instruments are represented by nodes which can be places anywhere on the canvas area and moved around using a dragging gesture. The main method of musical interaction in the probe is drawing paths through nodes which create a timeline of sorts. This timeline is traversed by an interface element called a pendulum which plays the sound of any node it interacts with. Scrolling, pinch zoom, and tilting gestures are all supported. In order to further visualise how the system might be used, examples of popular rhythms transcribed both in the system and on a popular DAW were shown

Now that a clear description of the full final system and all its features and some imagined use cases has been given, the remainder of this thesis will focus on the evaluation of this final probe. The very next chapter will go on to detail how the usability sessions carried out to evaluate the system were executed. It will also give the results of the sessions and the analysis thereof before the final chapter concludes the entire thesis and its findings.

# 5 EVALUATION

In the previous two chapters a new approach using stroking gestures was put forward in the form of a technology probe. During the development of the probe, suggestions from a set of users on the usability of the system were taken into account. The goal of this project, though, was to create an interface that allowed for creative usage by those sequencing using it. Hence the goal of this evaluation is primarily to see if users can use the software to be more musically creative - if strokes, nodes and loops led to users making more expressive music. The challenge then is designing a study to measure that outcome. Analysis of data concerning musical performance, though, has always been a difficult issue, with there being many potential factors to look out for when attempting to evaluate a new instrument or interface. Despite the existence of these factors, many traditional evaluation methods from HCI that are used in the world of NIMEs rely solely on task completion based methodologies. These aim to measure completion rates and times, which is inadequate for richer explorations that take into account the qualitative nature of musicality. This approach, along with evaluations using design heuristics [121], has become popular with designers of digital instruments. This is despite the fact that musical actions inherently have creative and affective traits, meaning that they cannot easily be distilled into tasks for which completion rates can be reliably measured [105]. This conflict between the need for qualitative veracity of results and intrinsically personal and even intimate nature of music performance might be the reason that many of the papers published on novel musical interfaces seem to forgo evaluation altogether [105]. Typical of the reasoning for this are statements such as, “control is not equal to expressiveness” [27 p.2].

Critical to technology probing is the placing of technology into a real use context, observing its use and then reflecting on this use in order to inspire further designs [50]. This implies a mostly qualitative framework from which to view the evaluations. With this in mind, this project chose to work within a methodology based on that proposed by Stowell, Robertson, Bryan-Kinns, and Plumbley in [105] in order to run evaluations of the system with users. The methodology is based around using the concept of Discourse Analysis (DA) – where user sessions are designed to encourage the user to explore the new instrument, speaking out loud. It is a method designed to have users operating without disturbing their flow, which has been shown to be integral to creative performance [25]. The DA method proposed by Stowell et al. is recommended for use by the authors when the system being evaluated is

- Not meant to be emulating interaction provided by a human
- Designed for complex musical interactions as opposed to separable musical tasks
- Will be evaluated from the performer’s perspective only (as opposed to an audience)

Each of these points applied to the chosen methodology of technology probing DA is a methodology that aims to illuminate the mental and conceptual structures that people create when applying an interface to their social context, as opposed to giving a simple binary usable/unusable score.

This chapter will focus on the issue of evaluation, beginning with the issues behind the defining and measuring creativity and expressivity, approaches taken by other researchers, and the finally the methodology chosen will be detailed. After this, the details and results of the chosen evaluation methodology will be given, as well as a discussion of these results.

## 5.1 Evaluating Expressive Musical Interfaces

Despite the recent surge in interest from fields such as HCI and experimental music in both creating interfaces and designing methodologies in order to create them more efficiently, work in the evaluation of these interfaces has remained sparse [104]. This is despite the importance of evaluation in research. Interactive computer music composition and performance poses a different problem to what the fields of HCI and Engineering have become specialized in solving. This is not only because of the issues stated in the introduction to this chapter, but also how different the perspectives of the various stakeholders involved are. Evaluations could be centred on the perspective of the user (or performer) – who might view effective performance and reproduction of existing patterns as vital. Evaluations could also, though, take the perspective of the composer, who might likely be interested in being able to create original music. Lastly, one could decide to have evaluations centred on the audience members as well, for whom the performance might have a visual aesthetic. Since this particular project focuses specifically on the task of pattern specification, the performance aspect is less important and therefore a composer-centric approach is the one that was focussed on.

Within the framing of composer and performer-based evaluation methodologies, some work that has been highly influential in research and to this project is the work done by Wanderley and Orio [118]. They proposed a user centred approach to evaluation based on using Likert Scale feedback for their experiences using the interface. Their usage of the interface during the usability session would be broken down into what they define as maximally simple tasks, and having users report back on their experiences in trying these. This is as opposed to having any objective measure, was highly advantageous as any objective measure (such as error rate) is more likely to miss on the inherently musical qualities of the gestures used. This approach leverages the user's own sense of musicality in looking to evaluate the interface. While the use of tasks can be seen as a draw back as it equates controllability with expressivity, the overall framework is one that is helpful in moving forward. It is also one that was expanded upon by Stowell, Plumbley and Bryan-Kinns [104], who detailed the DA methodology chosen for this project. DA is an analytic methodology that gives researchers a structured way to analyse the construction and reification of these social structures in people's discourse [81]. The source data for this analysis is text, which can be appropriately transcribed from recorded conversations

A large part of the technology probing methodology lies in finding the affordances of a given system. The word affordance was originally defined as being the actionable properties between the world (or an item in the world) and an actor (often seen as being the user). When the term was introduced into the field of design by Norman [80], he used the slightly different idea of perceived affordances. He used this term to define what designers should be looking to explore in their systems. This exploration would involve not only what the object or interface affords the users, but also the users' goals, abilities and past experiences [81]. All of this information that the user brings into the interaction with a new interface shapes the way that they see and use the interface. In fact, one could say that an interface is only really born or constructed when it comes into use [104] in this manner. This is especially the case when one looks at affective, musical interfaces.

The observational analysis of users' interaction with a system is common, interviews and free text comments are often used when evaluating interfaces (as they were also used in this project). DA, though, has the advantage over mere observation of being a structured practice that takes apart speech to show the themes, connections and implications within [5]. This gives a view into the mental construction that takes place while the interface is being used. To this end, the framework proposed by Stowell et al. looked to use DA as part of a qualitative and formal methodology with which one could explore issues related specifically to musical interfaces, such as expressivity and affordances. What this method also provides is the chance to explore higher level factors such as the users' conceptualisation of the interface: whether they see it, for example, as an instrument, toy, computer, as an extension of themselves, or (tellingly) it could also not be mentioned in the discourse.

### 5.1.1 Chosen Methodology

The methodology chosen is designed to elicit as many useful comments from the user as possible, in order to have a meaningful amount of discourse to analyse later. Thus users were encouraged to speak in an unconstrained manner while using the interface at hand. Thus study sessions were designed in such a manner that users could explore the system freely, while recording their speech and actions. The methodology, as described by Stowell et al., consists of three main phases: a solo session, a group session and finally the analysis phase. For the purposes of this particular project, though, the group phase, which encouraged participation and users playing music together with the new interface, has been omitted. This is as with pattern sequencing (as opposed to pattern performance) there is no real musical sense of collaborating or playing together on two separate sequencers. The only way this is achieved with traditional sequencers is using a timed MIDIsynchronisation – which implies having a linear and gridded interface, exactly the type we are trying to avoid emulating with this project. Collaboration on a single device also becomes difficult with mobile devices. Thus, semi-structured interview phase was also added, as described in the Thummer Mapping Project [83] (which also used DA based evaluation of the system). This addition was in order to account for the loss of analysable dialog from removing the group sessions.

Thus, the solo sessions form the crux of the user sessions, and were be recorded by video camera. The purpose of these sessions was to explore each user's personal response to the interface. This was where users are invited to try the interface for the first time, and the session was split into three separate phases:

**Free Exploration Phase**

The first part of the session consists of a period of free play, where the user was encouraged to try out the interface for themselves, in their own way. At this point users were encouraged, when asking any questions, to try to find solutions themselves.

**Guided Exploration Phase**

In this phase of the solo session, users were given audio examples created using the interface and asked to use these as inspiration for creating pieces of their own. This was done to give the users an idea of the range of possibilities that are achievable with the interface, and was not treated as a task where the completion rates are taken account of. Users were explicitly told that this was not a precision-of-replication task, but rather one where they are given audio ideas from which to build upon.

**Semi-structured Interview Phase**

Semi structured interviews are a well-used method of teasing out any ideas or behaviours that an interviewer might be interested in. In this case it was used to encourage the user to discuss their experiences with the interface in the previous two phases of the solo session. The interviewer at this point also used some of the recorded footage in order to tease out and ask about particular behaviours. Paine [83] stated that using semi-structured interviews also helps in accounting for the differences in musical ability and maturity levels between users in testing. This was another reason interviews were used in the Thummer project and here as well.

At this stage the data produced from the session was the video from which the transcriptions were made, and also observational notes taken by the interviewer running the session. When attempting to complete a qualitative analysis, contextual information such as this will always be valuable – even when using as structured a practice of analysis as DA. Observational analyses of user performances are not uncommon when looking at attempts by researchers to evaluate their interfaces. An example is the approach chosen by Johnstone, Candy and Edmonds [53]. Here users were allowed free play with the system while observational notes were taken as the main data output from the sessions. In this project, though, these notes from observations were used only to augment the findings of the DA. Further details on how this chosen methodology was implemented will be given in the next section of this chapter.

### 5.1.2 Procedure Followed

The latest version of the Xen system, as described in the design chapter, was evaluated using the method described above. While the main aim was to see how participants would use the system, and if it would allow for expressive and creative play as described as Jorda [55], the fact that the software was implemented as a technology probe meant being open to any potential results. 20 participants were recruited though poster ads placed on notice boards throughout the UCT campus and informal music groups on Facebook. The group consisted of an equal split of people who would describe themselves as musicians and those who would not. The researcher's own position as a member of the local music making community made it easier to be able to find groups of appropriate members to advertise the evaluation sessions to. Before running any of these sessions in Ernest, though, a pilot study with a single university student (recruited in the same manner as the others) was run. This was in order to ensure that the logistics of the sessions ran smoothly.

The original work on technology probes [50] gives no guidelines for sample sizes, but gave examples of technology probe implementations with two families of about 12 people each. This number, though, was partly dictated by the settings of the project as a collaboration with an already existing, international project. In fact, many of the projects which later followed the original work on Technology Probes stated that they also used smaller sample sizes in order to be able to have a "more in-depth view" of the qualitative nature of the data output, without giving further rationalisation. This might have been expected, as probing is a methodology more aligned with design than evaluation, so a closer look into the chosen evaluation methodology was taken when trying to find an appropriate answer. In this case, the chosen DA being used to analyse data collected using informal interviews and structured exploration was inspired originally by Banister's guidelines for qualitative research [12]. Banister's original work, which was adapted by Stowell et al., was written from the perspective of applying qualitative work to the field of psychology, where inductive research techniques, such as that found in this project, are far more common. The fact that the research is inductive also means that many of the guidelines which have become commonplace in the world of HCI don't apply very well here. For example, it has been well established that having 5 users is the minimum number for sufficiently carrying out usability evaluations [77]. The problem here, though, is that the usability evaluations in that context are performed simply to find errors, and after 5 users the number of new errors found begins to plateau. This project and many technology probes in general, are not attempts to define usability in new systems, and so this definition of evaluation is not the most appropriate. Unfortunately, in the context of qualitative and interview based research there is not an agreed-upon number to use as a sample size. A well-founded concept, though, is that of the data saturation point. This is defined as the point at which no significant new concepts appear in the analysis. When doing interviews, because of the fact that they give rich and in depth data, this point is usually reached quickly (within a low number of users evaluated). In fact, some have argued for the importance and scientific reasoning for having small sample sizes [24]. In fact, while increasing the sample size could improve reliability of results, it does not significantly improve the generalisations of a sample to its population [64]. Looking at other research projects and how the evaluations were run, the survey done in [4] shows an average of 16 users for lab based interview evaluations. In [45], one of the papers that attempts to give a finite number for when the saturation point is reached, Guest, Bunce and Johnson [65] stated that 6 users may be enough to develop meaningful high level themes. Doubling this number allowed them to reach saturation. Even though the researchers themselves stated that it is difficult to say just how generalizable their findings are, it is still a useful yardstick to have. Thus this project proceeded in enlisting 10 users from each of the groups of self-described musicians and non-musicians, giving a sample size of 20 users.

Each participant was remunerated to the amount of R40 for their time after the session was completed. Once recruited, the interested participants arrived either in an experiment room at the University of Cape Town, or a music studio located centrally Cape Town, depending on which was more convenient for the participant. Both locations were free of distractions and quiet. Prototypes, as research artefacts, are intended to be used in situ in order to accurately describe the design space that they are intended to illuminate. In this regard, the setting of the experiment room or the studio is appropriate for this user study as it is likely that anyone making music on a mobile device might seek a similar setting when working on new music. While this does not represent every use scenario possible, it is an accurate enough representation.

Users were then given an informed consent form and informed about the goals and procedure of the experiment. Once they agreed at the beginning of the session, information was collected about the users, including information about their musical experience and listening habits. This information was used to infer their general influences and skill with respect to music, and how they approach it. It was felt that this information would be useful when making qualitative judgments and analyses based on any information captured during the sessions. The musical questions were:

- Whether or not users could play any instruments, or had experience with digital audio workstations, and if so for how long
- Whether users could sight read or had any experience with experimental, Avant-garde or modern classical music
- What music they listened to, how they found it and how much time they spent listening to it.

The average age of the participants was 24.88 years, with the oldest being 31 and the youngest 21. There was an equal split of users who played a musical instrument and those who didn't, with experience ranging from two to 15 years. Nine of the participants indicated that they had experience with digital music software, with the most popular applications being FruityLoops and Ableton live. Lastly, only three of the participants indicated that they could sight read music, while only four had experience with Avant-garde classical music.

The sessions varied from 17-45 minutes in length and were conducted as described in the previous section. All of the sessions were video recorded in their entirety, and the audio from these recordings was later transcribed into written text. Extensive notes were also taken by the researcher, and were later transcribed digitally and reviewed with the help of the recorded video footage.

In summary, the user studies were conducted as follows:

1) Users were given informed consent forms and informed of the goals of the research
2) Users were asked to fill out the demographic information forms
3) Users were allowed a period of free play with the system, in which they could ask questions
4) Users were given audio examples and asked to use these an inspiration for new works
5) A semi-formal interview was conducted with the user
6) After all the sessions were conducted, audio from the sessions was transcribed
7) All notes taken from the sessions were transcribed
8) Data was analysed

Before the sessions, users were given all the information that has become standard to convey to users before usability evaluations. This included: an emphasis on the fact that the evaluation would be of the system and not of either them or the music they created using the system; that, since the sessions are voluntary, they can be stopped at any time, and any questions are welcomed; where the results of this study went; how the system is not intended to be a complete, commercially ready application; and lastly, that any recorded information would anonymised if made public in any way. After running the pilot study with another university student successfully, the rest of the sessions were conducted over a two week period. Below are the results of these sessions.

## 5.2 Data Analysis

The DA of text can be an extremely time consuming and arduous task when done by hand. Several software solutions exist to help researchers extract meaning from transcribed texts. One of the foremost of such solutions is the Leximancer tool, which is a software system used to do conceptual analyses on bodies of text in a language independent manner [101]. There are several reasons for preferring an automated approach to DA, such as the fact that it has been long known that it is possible for the results to be influenced by the decisions made by a human reporter without their knowledge [79]. Beyond this it can also become a prohibitively difficult and expensive exercise trying to remove any subjectivity from the content analysis process [100]. The Leximancer tool is a software solution used for the analysis of the content of collections of textual documents with the aim of displaying the extracted information visually. This visual display comes in the form of a two dimensional map that gives an overview of the main concepts represented in the text as well as how they are related to one another. This visualises the results of two separate forms of analysis: **Conceptual analysis** and **Relational analysis**.

Conceptual analysis comprises of analysing documents by measuring the presence of concepts and their frequency. The definition of these concepts can take various forms during analysis, and can be derived from words, phrases, collections of words or even more abstractly defined ideas defined by groups of phrases. The definition of concepts is usually one of the more difficult aspects of manual DA, and one of Leximancer's main features is the automatic extraction of concepts from text. Its algorithm (which will be detailed later in this section) infers concepts contained in bodies of text, creating a *thesaurus* of terms for each one and relieving the user from having to do this by hand.

Relational analysis is the analysis of how these concepts relate to one another once they have been identified. While there are several ways in which one might define a relation between two concepts, Leximancer uses the approach of measuring the co-occurrence of the concepts in the text. The occurrences, though, must happen within a set resolution (or distance, measured in the number of sentences that defines a co-occurrence). The information created by Leximancer while running both of these analyses is represented in the concept map.

The algorithm that the Leximancer software uses for the analysis works in the following manner:

- Concept definition in Leximancer revolves around the notion that a word's context (i.e. the words that surround it) indicate its meaning. Concepts begin with a set of Seed Words, which are words identified as the most frequently occurring in the text. Seed Words therefore form the first stage of the definition of a concept.
- Once potential Concepts are found in the text, seed words are weighted according to the frequency of their occurrence in sentences near this concept as compared to elsewhere.
- A sentence or phrase is only tagged if the evidence accumulated (or the sum of the weight of its terms) is above a set threshold. Thus, a concept can be coded with evidence words, even if the seed word is not present. This means that terms that are more appropriate to the concept may become the concept's new central term, replacing the original seed word. All the words associated to the concept form its thesaurus.
- Concept definitions are then used to calculate the frequency of co-occurrence between concepts (the relational analysis) in order to generate the concept map.
- On the concept map, concepts are then clustered into higher level Themes – which are comprised of concepts that frequently co-occur in the text. Themes take the name of the most prevalent concept within them.
- Concepts that co-occur in the text will attract each other strongly during map building and will settle near each other on its 2d space. Themes aid interpretation of the map by grouping these concepts.
- Straight lines called paths are drawn into the concept map, connecting certain concepts. Two concepts being linked by a path in this manner means that the two concepts were often mentioned one after the other in close succession in the text. Following the paths between two concepts gives you the most likely relationship chain between two concepts
- Lastly, the Themes are heat-mapped to indicate their importance to the text. This means that the most important Theme is shaded red, the next hottest being orange and so forth following the colour wheel.

One of the main goals of the Leximancer system and the algorithm it employs is to give users a global view of how various concepts are linked. This helps to make one aware of the concept's context and significance in the text, helping avoid the human tendency of becoming fixated on specific anecdotal evidence. In fact, when compared to human transcription and analysis of tests, it was found that Leximancer's unsupervised analyses generated well-defined, meaningful concepts [101]. It has also been proven to provide reproducible results in tests that analysed the same text in different languages, further proving its applicability in research. Because of this, the Leximancer tool has been used several times in research already within the Social Sciences and beyond [43] [85]. Leximancer also greatly reduces both the time and expense needed in order to complete this form of analysis on a body of text as large as was expected for the user evaluations.

In terms of analysing the other data collected, namely the actual video recordings of users' interactions and any notes made by the researcher during the sessions, extensive use of Jorda's work was used [55]. Many of the terms introduced by his work help anchor many of the discussions of the users' behaviour in section 5.3, so it is salient that these terms are discussed here beforehand. In his work on creating and evaluating affective interfaces Jorda devises metrics with which one could use to look at how users interact with systems. To this end he describes the *Efficiency* of a musical instrument as being what's most important to its evaluation. This efficiency is defined as the *MusicOutputComplexity * Performerfreedom / ControllerInputComplexity.* In this context the *OutputComplexity* of the system speaks to an instrument's potential and can be thought of as its musical range. An instrument with a high range would be able to create both the micro-sonic variations of a violin and the mid-sonic notes of the piano [55]. The *InputComplexity* speaks to how much control the user has over the system, and the degrees of freedom allowed to them when operating it, or the number of gestures and gestural nuances one can use in the system. This term is also closely linked with the precision of operation that the instrument offers. The final term in his *efficiency* equation is the *performerFreedomOfMovement*, which is a direct measure of the degrees of freedom available to the performer in using the instrument and their freedom of choice as well. This term was introduced to penalise cases such as having a CD player (which offers no freedom of choice or movement to its users) considered as an excellent instrument because with is one can produce amazing music.

While the dynamic relation between a player and the instrument of their choice is a complex one that has proven itself difficult to quantify in the past, the factors mentioned above by are good points from which one can begin to look at evaluating a new instrument. Thus, when looking to analyse data from the experiment sessions, results will be framed in the context of the above factors when discussed further.

## 5.3 Results and Discussion

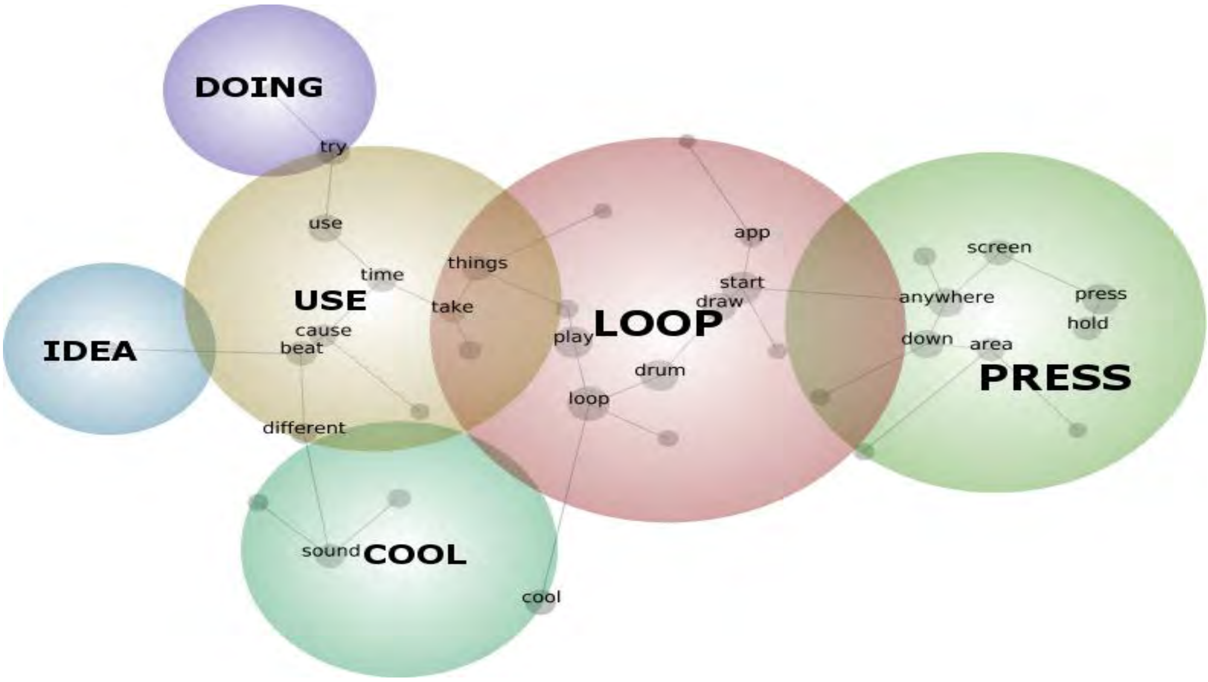


**Figure 27 Leximancer concept map overview**

### 5.3.1 Concept Map Findings: Exploration, Output Complexity

The technology probe methodology aims to explore a design space, using a probe to inspire insights and design directions. As such, one intentionally sets out with no firm hypotheses in mind for what results might look like. Having said this though, one might expect that if there are any major differences in how users interact with the system, the differences would be between the users who were musicians and those who were not. This, though, was not the case here.

When looking at an analysis of the transcripts done using the Leximancer, the concept maps created using both the musician's and non-musician's individual data did not differ significantly. Both virtually mirrored the joint concept map shown in Figure 27 above, rendered using the entire data set. The top 50% of concepts are visible on the map in order to avoid clutter. These are the concepts that appear most frequently in the text and those that are most-connected to other concepts on the map [99]. The number of concepts shown does not affect the formation and size of the themes present.

In the map produced using the entire data set the 6 main themes are Loop, Press, Use, Idea, Doing and Cool. Looking at their colouration and comparing that to the standard colour wheel, the Themes in an order of importance are as follows:

1) Loop
2) Use
3) Press
4) Cool
5) Idea
6) Doing

Both sets of users had Loop and Press as two large and highly related, overlapping themes. The Loop theme encompasses the concepts of *loop* and *play*, showing that this is how users mainly described their interaction with the system. This is as opposed to using other music related terminology such as "sequences", or "rhythms". The loop concept within the theme is highly connected, with direct paths running between it and several nodes. The concepts of drum and play are included here, being directly related to the loop concept through statements such as "[in order for] the drums, to play together, we need something like a loop to join those". A cursory look at the quotes associated to these concepts also shows that users often used the term "drum-loop" as opposed to just loop, which would skew the relationship between them positively. Another concept that gives further insight into how the users thought of the interface was the appearance of the term app as a concept as opposed to any other descriptors such as "instrument". An example of this from the transcripts can be seen when one user spoke of the system as being visually not comparable to other "app[s] or any sort of electronic music software". Within the Loop theme the term *draw* also makes an appearance, interestingly being more related to the Loop theme than any of the other concepts which were more directly related to physical interaction. It is also directly linked to the concept of drums. The link between *draw* and *loop* might be expected, as *draw* is a much more grammatically appropriate verb for the act of creating a *loop* in the system - one does usually speak of "drawing" drums. Regardless of this, though, examples in the text which showed these terms' connectedness can be found. An example of this is when one user stated that "[they] would like you to try… draw something that goes through all four [nodes]". This is an encouraging connection between the concepts. It means that users were speaking of the system in a holistic manner in which the drawing was less of a physical act. Rather, it was inherently connected to the concepts of *playing*, *loops* and even of *drum* (to which the *draw* concept is connected by a path). This hints at the users having a grasp on the interaction metaphor used within the system and its conceptual model and how exactly to control it. The links in dialog between the physical acts that facilitate gestures on the system and the musical effects thereof can further be seen in how vast the overlap area is between the concepts of Loop and Press, and also between Loop and Use.

This makes sense if one thinks of how the Loop is created in the system by the physical act of pressing. The Press theme, unsurprisingly, encompasses other physical and gesture based concepts, such as *press* and *hold*, which are both terms that describe gestures used in the system to create. *Screen* is another concept that one would expect to be in the cluster of Press themes. *Screen* is linked to the theme of *anywhere*. Taking a look at the quotes from the transcripts that contain the word, most occurrences came from users asking if they could place nodes anywhere on the screen area (*area* being another concept appearing in this cluster). The *down* concept is also one that isn't out of place with the other concepts in the theme. This becomes clear when one considers how often the term was used when talking about the various gestures applicable to a mobile device (such as to "hold down").

The other clustering that intersects with the central theme of Loop was that of Use, which encompasses other verbs and action words such as take and the concept of *use*. The concept *cause* is also found within the theme, but an inspection of the text reveals that this was a colloquial shortening of the word *because*. The term *beat* also becomes a concept and makes it into this theme. While the terms *beat* and *loop* were sometimes used synonymously, the fact that they appear as separate concepts, in separate themes, shows that this wasn't always the case. Inspecting the text it can be seen that the term beat was often how users spoke of completed musical pieces, as opposed to just a single loop, which forms a part of a beat. A canonical example of this lexical difference can be seen in how users would refer to the audio examples used during the evaluation as, for example, "[a] simple repetitive house drum beat" while still referring to "drawing one of those loops". Users were already speaking in hierarchical terms when talking about how they use the system, with *loops* being on a different level to completed *beats*. This further indicates a level of understanding of the system by these users. While this alone is not enough to conclude that this speaks well of the interface's input complexity (a term introduced by Sergei Jorda which is related to user's control of the system, from their perspective), it is a piece of evidence which will be built upon later in this evaluation (section 5.3.2).

A few of the Use theme's concepts contained in the cluster, such as *time*, *take* and *things*, are actually more suited to being on the stop list of words to exclude. This is as they are often mentioned within the context of discussion not directly related to the interface's operation. The other concepts within the cluster are concepts that the Use theme shares with two of the smaller themes in the map - Doing and Cool. The Doing and Use themes are connected by the concept of *try*, which also has paths running from it to both *doing* and *use* concepts. This makes sense as a verb used when talking about the act of operating the interface to make music. The *try* concept is shared almost equally between the Doing and Use themes, showing just how much they are linked in dialog. The concept shared between Use and Cool is that of *different*. Cool, in the contexts of this analysis, is the colloquial term describing something in positive terms, such as when one user exclaimed that "That's cool that you can do them independently… I get a simple loop going". The fact that this is a theme means that, at the most basic level of analysis, terms that were in the thesaurus for the *cool* concept (such as "cool" itself) were often mentioned by users, which is a positive sign. The Cool theme also encompasses the concepts of *cool*, *sound*, and, as mentioned before, part of the concept *different*. This hints towards the positive sentiment with which users were speaking about these terms, talking about "how cool the sounds are" for example. The fact that the concept of *different* makes it into the list at first seemed like an indicator of the system's novelty. Upon analysing the raw text though, there were many more instances of users describing their own work with the system as different. This was either in saying it sounded different or that they were going to attempt something different. This was another important piece to the analysis which will be explored in more depth section 5.3.2 when looking at the complexity of some of the music created by the users.

More is revealed in the map when taking a wider look at the placement of various themes and concepts. Two smallest themes - Doing and Idea - are the least connected in the map. Doing, a performative action word, and Idea, a conceptual and thought level word, were connected in the dialog by the theme of Use. They were also placed on the opposite end of the conceptual map as the theme Press, which contained many of the concepts one would associate with the physical screen. This shows that the dialog surrounding these themes was often quite disparate, with there being very little co-occurrence of both their concepts and the terms that these concepts represent. The Idea theme is also connected directly to the *beat* concept (which is extremely close to the concept of *cause*) in the Use theme. Use is quite central, and connects to both the main theme of Loop and the theme of Doing. What this seems to indicate is that the Theme of Use, which encompasses the idea of embodied action (and linked to the idea of Doing), is connected to the users' speech concerning their ideas (as opposed to the more physical action orientated Press theme), and words concerning approval or being impressed (such as those related to the Cool concept).

These were broadly the results found both when looking at the data set in its entirety and when looking at the data sets of the musicians and non-musicians individually.

### 5.3.2 Observations

#### *5.3.2.1 Overview*

When looking at the users' behaviours in interacting with the system (from notes taken during the session and video recording), the expected split between musically experienced users and novices wasn't found. Instead, it was found that about two thirds (14 of the 20) of the users freely explored the system and experimented with creating unique and complex loops. This group, which shall be referred to as Group A, consisted mainly of users who could be either described as being novices musically, or on the extreme opposite end of the spectrum. That is, novices and users who had experience with both creating music and with consuming various different genres of music. What brought these users together enough to consider them as a single group, though, was their open and explorative attitude towards using this new system. This group was not only willing to learn by trying, but often actively looked to push the boundaries of the system. One user, for example, explicitly stated that they were "going for chaos now" during the session. It did take longer, though, for four of the users in this group to reach a point where they were comfortable enough with the system. Half of these instances (two of the cases) were due to the users having very little prior touchscreen experience. The rest of this group had the confidence from early on to be exploring the system and experimenting, and most of the users in Group A were freely improvising. They also created systems more complicated than the bare minimum needed to replicate the audio examples given to them. This was their defining attribute (and as such the groups were formed only when taking stock of the usage data of the probe).

#### *5.3.2.2 Group A Behaviour*

In this regard, one user from group A recreated the audio example using three different arrangements of nodes while one of the slower adopters stated that they would have "mad fun" with this if given more time to acclimatize to the system. This sentiment was not an isolated one, and was indicative of the attitude with which the users in this group approached the audio examples. This group also showed a greater number of different drawing strategies in order to achieve the goal of working towards the audio samples given to them. A quarter of these users soon figured that symmetric, geometric shapes "makes sense" in terms of representing rigid structures. Many of them, though, also soon abandoned these shapes for more interesting configurations such as "zigzag kind of patterns" or clusters of micro-loops that only contain a single node each. One user who was only creating relatively simplistic loops stated that he could still "actually have a lot fun with this".

A feature of having an interface such as this, which relies on a drawing metaphor, is that users will have to translate rhythmic ideas from being time based to being based in a geometric, two dimensional interface. While the entire set of users showed varying levels of success with coming to grips with making this translation, each member of Group A was more willing to draw non-symmetrical and non-regular loops. This was even when attempting to take inspiration from the given audio examples. This lead to a wider variety of sounds coming from what these users created. This fact can be seen as a large reason that the term Different made it onto the concept map in the context that it did (that of users describing their own work as such). That users, even ones with experience with music creation, were self-reporting the music as being different is encouraging.

Another interesting behaviour found in a fifth of the members of Group A, both in the musical amateurs and those with more experience, was an interest in exploring the generative nature of some of the rhythms. These are rhythms created by chance, and where users are looking for one-off moments of concord in complex arrangements of loops. This hints at the same style of experimental music that the likes of Xenakis had helped spearhead, and a style of music normally seen as demanding to the point of being beyond most casual listeners. The fact that half of this group consisted of novices should not be taken as proof that they were simply fooling around. Research has shown that even novice users have a level of inherent musical competence drawn from familiarity to musical form [26]. Analysing the text, this subset of Group A – the amateurs who experimented – contributed highly to the Cool theme and different concepts discussed earlier. That this group had indicated that they had had no prior experience with this type of experimental music yet were able to, with time, arrive at a point of creating this also tells us something about the interface in terms of the factors that Jorda gave [55]. Specifically, the terms *OutputComplexity, which* spoke to a system's musical range, and the *InputComplexity, which* was the control that users had over the system. The output in this case ranged from users being able to create exact replicas and works inspired by simple loops examples to much more complex systems with generative aspects. This is a wide range of possible outputs, and the fact that this range was reached by a large group of amateur users speaks well of the control that they were able to exact over they work they were producing. This control was hinted at during the Lexical analysis, but these observations confirm those findings.

That the musical experts were, once they came to grips with the interface, able to experiment and explore in this manner is also encouraging, and not something to be taken for granted. The fact that, in a short period of time, users were confident enough to try to achieve this level of experimentalism is extremely encouraging. This idea was explicitly stated by a few users, with one saying that exploiting the conflict between "random rhythms vs. structure" is quite fun. A quote from a user that encompasses this well was "OK, let me take a completely different approach now, just experiment, instead of trying to create something that I know what it's going to sound like, I'm just going to go crazy."

#### *5.3.2.3 Group B Behaviour*

Group B, which consisted of all the users not in Group A, was far less prone to experiment with the system and was often much more frustrated at what they perceived as the imprecision of the system. This group consisted mainly of users who could be described as having some limited musical production experience, and often on a desktop based DAW. This DAW experience could be seen in how a number of the users in Group B sought to recreate a linear step sequencer environment by attempting to draw straight lines with nodes placed equidistantly on these lines. This, though, would fail as users would then have to still complete the loop, so it could not simply be a straight line, but rather a straight line that then loops into an ellipsoid shape in order to close the loop.

*"*Oh, OK, but the fact that it's just on a blank screen or on a white screen, it's without any, what do you call it, bars, you see Fruity Loops and all of that… but now it's going to be hard for you to keep a rhythm here.*"*

As the quote above alludes to, a half of these users showed some knowledge of other DAWs to compare the system to, such as Fruity Loops. The fact that they were unable to bend the probe in order to make its functionality more like them often frustrated them. This frustration that two of these users felt was palpable as one user exclaimed “What I’m making, it sounds like a beginner” while trying to play towards one of the audio examples. Even users with a slightly higher musical vocabulary intimated a similar sense of frustration at the system. One users stated, for example, that the system would benefit from having a “grid… or some kind of quantisation feature” added to it. This is essentially the same feature set that was missed by the less musically experienced members of Group B. Three participants even arrived at a point where they expressed this desire for a gridded structure via the drawing metaphor, saying that the interface would be better if was less free and only allowed people to draw “perfect geometric shapes”. The biggest defining factor for this group though was their unwillingness (or inability) to look beyond their initial apprehensions and explore the system, experimenting with what was possible with its open ended design. None of the members of this group created anything that was beyond their control or much more complicated than the audio examples, when asked to take inspiration from them.

Two users in this group explicitly stopped the free exploration stage of the session, saying that they had seen enough. The sessions with these users were also slightly shorter, on average, as they didn’t strive to explore the interface, which was an activity that was supposed to take up a large part of the session’s time. Another two gave up on using the drawn looping functionality entirely. These users, instead, arranged the instrument nodes ergonomically around the screen and then exclusively tapped out rhythms. In this way they were no longer specifying rhythms (sequencing), but rather solely performing using striking gestures. While this behaviour was unexpected, and this project strove to investigate the merits of moving away from striking gestures in mobile musical interfaces, the technology probe is chiefly used to explore a design space. Thus users were never discouraged, but rather left to their own devices when exploring the interface and working from the audio examples, even if they were using the interface in ways that were unintended. The fact that these users were still accommodated by the interface, even if in a way that the research was not expecting, is also encouraging and shows the flexibility and *explorability* that the system has. One user in particular was still impressed by the system, even though they weren’t sequencing at all This was because no interface they had used before allowed them to move drum pads into any position they wanted on the screen. Indeed, this level of configurability would be impossible on physical drum machines and samplers such as the MPC5000 and many of the digital interfaces based on these. This might have been an unintended discovery, but therein lays one of the main advantages of employing technology probes for a study.

In discussing these two groups, it is useful to look back towards the concept of *balance* introduced earlier, which related to the balance between triviality and difficulty within an interface. From this split in behaviours, it seems that users in Group B found that the mental effort that would need to be spent in trying to reach some sort of mastering point of the system would be too high. They therefore would stop exploring it entirely. While this speaks to the balance of the system not being perfect, the fact is that a majority of the users in group A felt that exploring the system was both worthwhile and encouraging. They were still trying to attain goals such as being competent and different, which means that the balance was still in a positive place. The different behaviours found here define two separate groups of people, with two different sets of experiences with the system.

Therefore, in reporting our results further, we will make a distinction between group A and Group B as they had very different reactions to the system.

### 5.3.3 Practice, Improvisation

Another interesting behaviour was that of practice and performance in the system. As mentioned above, the interface requires the translation of information from being temporal to being spatial. What this meant was that users, when first using the system, would need to *practice* or tap out rhythms before specifying those rhythms using loops. Interestingly, it seemed these behaviours would slow as users became more accustomed to the system. At that point, they would first try specifying a loop and then modifying or deleting it if it strayed too far from what they wanted. Once users did create a loop they were happy with, they would often then leave it running and tap other complementary rhythms over it – not necessarily drawing them out. Even though the interface was built to be based on stroking gestures, users showed the need for this even more immediate form of interaction. This was mainly for the purpose of practicing rhythms to be specified later and for improvisational performance. Even though the interface was made for pattern specification, this practicing and improvisational behaviour showed that having more support for pattern performance would be fruitful. This would especially help users mentally navigate having to translate ideas into actual music using the interface. Similarly to the tapping behaviour seen with members of Group B, this was unexpected but still entirely valid information. What it suggests about the system is that while the stroking gestures implemented in the interface worked in allowing users to specify loops and giving the control needed to be confident in experimenting with the music, they should not be implemented alone. In regards to the more immediate, and possibly less cognitively challenging and more instinctive task of pattern performance, the striking gestures that this project sought to move away from served a purpose. While the drawing gestures allowed the users a level of *Reflective Practice* when using the interface, this advantage is not one that could be transferred to the act of pattern performance, where no drawing is done at all. One might argue, though, that this is due to the inherent difference between the activities of sequencing and performing, and that pattern performance wouldn't necessarily improve with added reflective practice.

### 5.3.4 User Thoughts on Stroking

During the semis structured interview phase at the end of the sessions, 5 of the users were asked directly what they thought of the stroking gestures. This number was because of the semi-structured nature of the interview (the discussion that occurred in the interview varied between sessions). The resulting answers were varied, although there were some commonalities.

The first user asked was outspokenly negative on the experience of using the drawing interface, stating that it was "quite tricky". His reasoning for saying this was that he was unable to get used to visually recognising how one of his drawings corresponded to "some set time signature". This left the user wishing that there was some way of seeing, once you've drawn a loop, what that "[actually] means". This need for having a greater sense of control within the system was echoed by two other users, although their outlook was more positive.

The first of these two stated at first that "drawing the loops [was] the easiest" and that they thought it was a "good idea". Later, though, they stated that they had had some difficulty with the issue of timing. They found that the main problem was that "adjusting the time on each note" was difficult. They would rather have adjusted the global speed of all the loops running to make them "maybe run faster”. This issue of speed came up again with another user stating that they enjoyed drawing, especially when combined with the zooming possibilities, but also felt that they “need the standard speed of everything to be much higher and then that could solve a lot of my problems”.

The final two sets of users’ comments were more plainly positive. The first of these, encouragingly, came from a user who had “never had a touch [screen] phone” before but had found that by “the third time [they] made the third tune… it was easy” to use the stroking gestures. The final user made similar remarks about finding the gestures easier to use after some time with the system.

### 5.3.5 Usability Issues

Even though the technology probe and DA methodologies are employed to evaluate higher level concepts of creativity and its conceptualisation, some usability and technical issues were still found during the sessions. While these issues were not big enough to derail the sessions, they are worth noting. Chief amongst the issues was the fact that the Nokia Lumia 900s, which were recently discontinued after the sessions, had a bug that shut them down permanently when they ran out of battery life. This was an issue uncovered during the pilot session, and in order to mitigate for it each session was ran with a spare phone running the software. What this meant though is that 4 sessions had interruptions of approximately 2 minutes while the researchers switched the test to another device. This, though, would not affect the results.

The usability issue mentioned the most by the users was the zooming algorithm used. More than one user noted that it would make more sense to have the zooming gesture affect the distance between items, rather than making everything bigger, although this would be more difficult in implementation. Often users would lose track of place after having zoomed in/out too far while working on the interface. An easy way to get around this would be to have a single button or gesture that allowed users to return to the default zooming level. Another usability issue with the system was the fact that the mechanism by which the speed that a loop runs at (using the accelerometer) does not accommodate landscape orientation. Thus when users who were working in that orientation wanted to increase the speed of one of the loops, they would first have to turn the phone back into portrait mode in order to do so. Another comment made by one user concerning this functionality was that they wished that more extreme tempo changes could be achieved.

The rest of the issues that were noted by users were different in that they didn’t come from the users spotting unwanted or unexpected behaviours caused by errors in the system. Rather, some of the design decisions made when creating the system that could have gone another way were noted. None of these were unanimously disliked by users, with the overall consensus being ambivalent when considering the whole group. These are still issues that are worth being mentioned though, as they highlight areas for potential improvement and future work. This is especially if ways are found to ameliorate the downfalls of the said design decisions without detracting from their advantages. The first of these was the fact that loops, when first being created, cannot begin within the surface area of a node. This is because clicking and dragging a node is the gesture used to move it in the two dimensional space, and there is no way of distinguishing between this click and drag motion and that used to draw a new loop if it began on a node. A small minority of three users noted this as a disadvantage to the system. While it can become one when operating the interface in that way, most users were able to either work around this or not notice it at all in the first place. Again, this is less a usability issue, and more a consequence of the intended interaction methodology. Another is the fact that in order to represent multiple instrument hits one needs to place multiple nodes, which is something that two users argued was space and time inefficient. The alternative to this, though, which is having a single instance represent multiple hits, leads itself to the type of linearity that this project strove to move away from. The number of users who mentioned this issue was also within the minority though, so making a change to remove this interaction style would not be a worthwhile amendment to the system.

The only other areas for future work suggested by participants were based around the idea of synchronization. While some suggestions, such as having the interface automatically making rhythms fit, would defeat the purpose of this exploratory probe, others have more merit. As an example, just under half of the users asked about having the ability to copy and paste exact duplicates of already created loops. This is a valid option for potential future development, and a possible middle of the road compromise for offering more *synchronizability* without losing any of the freedom that the interface is built around.

## 5.4 Discussion

The results show that a percussion interface based on continuous stroking gestures does hold advantages – both for novices and the experienced. Looking at the terms given by Jorda [55], most users seemed to show greater *MusicOutputComplexity* as the usability session went on. A large set of users (namely Group A's 14 of 20) seemed to have been encouraged by the interface to further experiment with unusual rhythmic timings, a major goal of the project. This can be evidenced both from the quality of the work they produced and by the appearance of Concepts on the concept map which indicate approval. The Complexity of the output found during the sessions was far greater than what one might expect from the non-musicians in particular, with users often going beyond the simple examples given to them for replication. The unquantized and open nature of the probe hinted at this, but only when the users themselves operated the system with almost no upper bounds on what they allowed themselves to try could we conclude anything about its output complexity. The fact that, when looking at the audio examples, users found a variety of ways to achieve the goal of creating a similar piece speaks to the high levels of freedom afforded to the performer. As Jorda had stated, interfaces that allowed users a larger number of ways to communicate with the system and affect output, the better the general expressiveness and *efficiency* of the system will be. This *efficiency* was experienced equally by both novices and users who would consider themselves experts (which is also a sign of there being a good balance in the interface between accessibility and complexity). Although a more longitudinal study over a greater time period would be needed to see if any users would pass the barrier into becoming virtuosi. Nevertheless, it is encouraging that members of group A (a majority of the users) seemed to be taking the first steps towards this eventual goal.

The fact that a large number of the people who did not resonate with the interface were those with some DAW experience is also telling, and linked directly to the *ControlInputComplexity* that the interface offered. The potential complexity that the system offered is (deliberately) higher than that of many grid based, linear music interfaces. When members of Group B, who have experience only in these types of interfaces, were faced with this increased complexity, they found it an insurmountable barrier between them and their enjoyment of the system. The input complexity was a term added by Jorda to define the musical range of the application. While having the system set up as it was (non-linear, ungridded) shows a high complexity in an effort to be non-impositional, having a lower input complexity is useful for other systems with different goals. In systems like Ableton, where the goal is not to be non-impositional but rather to allow users to create commercial ready music, having a lower output complexity is of advantage. Thus it can be postulated that if users are only familiar with this style of production, and have not yet reached the point of looking beyond its constraints, then having an open system such as the probe presented here would not be something they take well to. This can be seen in comments by users bemoaning the lack of absolute accuracy within the system amidst fears of sounding amateurish. This can be seen as a weakness of the chosen interaction metaphor.

Group B's seeming need for precision was not catered for by the interface, and this could possibly be seen as a result of the current state of popular music. Today, mechanical and perfect percussion and instrumentation is vaunted, and the loss of precision can be seen as a deterrent. Thus, the interface's lack for the support of this more contemporary sound can be seen as a gap in its feature set. Group A, on the other hand, has shown a desire to explore the creation of systems beyond their control, which may or may not provide once off moments of concord. This style of compositional thinking has existed for some time in the world of Avant-garde classical music and Post-Digital [22] electronic music; it was thus encouraging to see this behaviour amongst people who have had no exposure to such music before. This speaks directly to the user's high level of freedom of movement and freedom of choice.

This is a fact that is also indicated by the close linking of the ideas of doing, creating, and forming in the concept maps created from the analysis of discourse from phase three of the evaluation. This concept map result from the Leximancer analysis seems to align with the ideas around embodiment, gestures and the creation of new ideas through action. The clustering of the Themes seems to be a positive justification of the concepts mentioned in previous the chapters' sections 3.1 and 2.3, where it was shown that doing, creating and developing ideas when using an interface are all related. Abstract terms such as Ideas and Loops were, on the concept map, entwined with more physical ones in a way that showed their interchange-ability, co-occurrence and connectedness in the user's dialog. This is all in a way that seems to confirm that gesturing, especially through continuous gestures, forms an essential part of creating meaning. The level of the music created, the balance of users who engaged and those who did not, and the fact that, despite being able to create complex systems, users were also able to satisfactorily re-create and go beyond simple example loops leads to the conclusion that having a stroking based drum sequencer did indeed engender creativity in users. While this was not the case universally, the 14 of 20 users for who this was true form the majority.

While striking gestures might have been appropriate for more immediate tasks such as pattern performance, when it came to sequencing and creating music that is sequenced, the drawing gestures were successful. This is not to advocate for the removal of all striking gestures though, as users did also show the potential for the two to be used together in an interface. A prominent example of this being users improvising by tapping a rhythm over work created by the drawing sequencer. Also, while some users found the open ended and unquantized nature of the application to large a hurdle to work around, a larger group of the users took the challenge as an opportunity to explore and create something they thought was different. For those who did not, though, the interface was frustratingly unusable, and made two particular feel amateurish.

The goal of technology probing, though, is to aide in the design process by helping researchers actualise their ideas through use in-situ. Thus they are not often launched with the goal of proving a particular methodology correct, but rather to gather all use cases as potential for further reflection and development. In trying to present to users an open ended design that didn't impose any use cases, one cannot write off any behaviours as being incorrect, even if they were unintended. Thus Group B were not wrong for approaching the system in the manner they did, and the fact that members of this group often resorted to simply tapping out rhythms when faced with the feeling of the inadequacy of drawing loops is a valuable asset to the research. One can see that, from the fact that users had a rewarding time placing pads in an order they found to be most ergonomic and simply tapping rhythms out, that there is a gap here for further design. This gap could also be seen in the behaviour of users in Group A, who embraced the drawing metaphor whole-heartedly. Even they found a place for the poking gestures, even though much of the research presented in earlier chapters suggested that they could be inadequate.

The behaviour discussed above of creating loops to play bass rhythms and then improvising over these rhythms with patterns played out using striking gestures was an unexpected yet welcomed one. It showed that there is a gap for hybrid systems to be built that leverage not only the advantages of both embodied stroking gestures and discrete striking gestures, but is also a hybrid of pattern sequencing and pattern performance orientated interfaces. In trying to move away from interfaces that impose one rigid style of interaction, designers should not then arrogantly try to impose freer systems that seemingly impose nothing (which is actually not possible). A balance must be struck, and in this balance lay the key to having users from both Group A and B using a system satisfactorily. Although a balance was somewhat stumbled upon in this probe, it would be better suited to all groups of users to have this new goal be one which the initial designs rely heavily upon.

## 5.5 Summary

After having given an extensive description of the final interface used for the probe, this chapter sought to give the details of evaluation process and a discussion of the results thereof. The chapter began with a brief overview of work done in the field of evaluating musical and affective interfaces, introducing the concept of DA, which was used in analysing the transcripts from the usability sessions. 20 participants were recruited for the usability sessions, which consisted of three phases. Two of these phases involved having the user exploring and creating with the interface, and the last phase being a semi-structured interview.

There was an equal split within the 20 participants between those who would describe themselves as musicians and those who would not. Analysing the data produced within the sessions, one could see that 70% of the entire set of participants (referred to here as Group A) used the interface in a free, exploratory and unhindered manner. This group had an easier time adjusting to using drawing as a musical metaphor, unlike the remainder of the users (referred to as Group B). Besides their behaviours when interacting with the system, the users in Group B also shared the trait that they had all had intermittent prior experiences with digital music making. This was in contrast to the Group A users, who were exclusively either users who had had a lot of experience with music making or extreme novices. While these results do hint at the fact that the interface falls short on providing for users looking for a controlled, quantized experience, the fact that the majority of users fell into Group A validates that a percussion interface based on continuous stroking gestures does hold advantages is viable.

# 6 CONCLUSION

We have presented the results of a new multi-touch mobile interface for drum pattern specification which was implemented as a technology probe. Based on continuous stroking gestures, as opposed to discrete striking or poking gestures, the probe was designed to explore the merits of employing these stroking gestures for pattern specification. Along with the continuous gestures, the probe also presented rhythm in continuous and unquantized time-steps as opposed to the discrete ones common on most traditional and physical drum sequencers. It was theorized that stroking gestures would allow for greater levels of creative expression because of findings from work in embodiment. This work found that continuous gestures presented a number of advantages over discrete ones – such as lowering cognitive load. Despite these findings from embodiment, many multi-touch interfaces that already existed for drum sequencing were still based on interaction metaphors borrowed from physical devices, which meant employing discrete gestures. Another interface characteristic borrowed by digital sequencers from their older physical counterparts is that of having a limited, discrete number of time steps within which to specify rhythms. The reasoning behind employing a continuous time scale for the sequencing in this project lies within work from the world of experimental music and NIME. In both these realms it has been shown that by limiting the scope of potential inputs in a musical interface, one also hinders the users of this interface in terms of creativity. Thus the main question that the research sought to answer was whether or not employing the stroking gestures would indeed engender creativity in users. It also sought to investigate the viability of employing an unquantized and open drum sequencing application. This chapter will summarise how this project proceeded to investigate these problems, answer the research questions, and highlight the contributions made by this work. Finally recommendations for future work will be suggested.

In order to be able to explore the validity of all the concepts that led the research, they were embedded in a technology probe. The probe was designed to be both flexible and open ended, yet still adhere to a set of research defined design goals. These design goals encompassed all the ideas that the project hypothesised would bring improvements to a multi-touch sequencer, such as the need to embrace embodiment, having the interface impose as little structure as possible and allowing for non-linear representations of time. The stroking gestures were implemented in the system as a drawing metaphor, with which users could specify drum loops. Using drawn representations of compositions is an activity with a long history in experimental music, and it is after one such experimental artist, Iannis Xenakis, that the final probe was named Xen. Once the probe was completed, the methodology laid out by Stowell et al. [104] for evaluating musical interfaces was chosen. This methodology relies on conducting usability sessions with semi structured interviews and analysing participant's speech using DA. A group of 20 participants were recruited for the sessions, with there being an equal split of users who would describe themselves as musicians and those who would not. The sessions were conducted either at a silent music studio or an experiment room used for evaluations such as this. While technology probes are meant to be used in real world situations in order to give honest reflections, most producers and creators of music do tend to work in secluded spaces such as these. Therefore, the locations worked well as in-situ examples of usage. Technology probes are by design somewhat open ended, and one would expect users not only to adapt to the new technology, but also to do so in creative ways [50]. Therefore, both in the observations of the users' behaviour and the analysis of their discourse used we shall explore the musically creative use of the software, as described by Jorda in [55]. There, the researchers give a set of guidelines by which one might be able to give judgements of whether new instruments are successful or not.

Conducting this study required a careful blending of methods that would allow participants to be creative (implying an open-endedness) but allowed us to glean and evaluate design ideas (implying a focused structure). In the end this led us to adopt a technology probe approach for creating the system, but relying on open methods, such as DA, to allow us to interpret the unstructured responses of participants. Tools such as Leximancer helped to bring structure to our evaluation, but we borrowed heavily from domain-based literature to help us in our evaluations. In our case, the research on avant-garde music provided a useful structure and dimensions for us to investigate the system's efficacy.

The following section will give an overview of the results obtained from the evaluation sessions held. It will then be going into further detail as to how these results answer the research questions that drove the research.

## 6.1 Results

The results of the user evaluation sessions were positive in terms of a majority of the participants' creative engagement with the technology probe presented to them. The only exception to this being users who were familiar with existing sequencer software and seemingly unwilling to abandon the mental models they engender. Out of a total of 20 users, this applied only to a small group of six users. When looking at a concept map created from the entire set of user's text transcripts, it was also possible to see that users conceptually had linked ideas behind doing, forming ideas and using the app. It would seem, in as much as we are able to measure such behaviour, that the literature on how embodied gestures help users to think through their actions and explore ideas is directly applicable to the field of musical interface design. While the users themselves still showed a propensity for using tapping gestures to play or practice drum rhythms, these were blended successfully with stroking gestures. Thus, they were realizing the benefits and playing to the strengths of each modality.

### *Does employing stroking gestures in the design of drum sequencing application engender creativity in users?*

In the larger group of users described above, who are referred to in the research as Group A, there were signs of the interface having engendered and enabled creativity in users. This group of users freely explored and pushed the boundaries of the probe application. When given simple example drum loops from which to draw inspiration, these users showed the capabilities to recreate the examples using a number of different configurations of loops. They then, even more encouragingly, would often go on to create even more complex loops than those given. These users also often spoke of being eager to explore and push the system, saying out loud that they would like to see what it can do. This was especially encouraging since not all of these users were musicians who would already be confident enough to experiment in this manner.

The most surprising and complex behaviour coming from this group consisted of users using the system in a generative manner, similar to some of the experimental classical music that Xenakis and his peers were known for. This was despite the fact that a majority of the users had indicated that they had had no prior exposure to the world of avant-garde classical music. Regardless of this, though, they felt confident enough to be able to pursue this line of composition. This fact alone, though, can only tell one so much about an interface. When this is considered along with the fact that these same users were able to recreate and even surpass simple example loops showed that they had the control to be able to make music usually within the scope of more traditional and modest beats. Users seemed to feel both the need to push themselves and the beats they created further, while also feeling that they had both the ability and the tools required to achieve this.

While members of this group were not equally successful in their implementation, each member of this group was willing to try employ non-conventional and irregular shapes when drawing out the loops. This, though, was not the case with members of Group B. A subset of these users in Group A also adopted the practice of first drawing loops that would serve as a *base*, and then tapping out complementary rhythms using striking gestures.

### *Investigate the use and viability of an open, non-linear and unquantized mobile sequencing system*

Apart from the creative behaviour which the users in Group A displayed when exploring the possibilities that the interface allowed them, all members of this group were also able to satisfactorily either recreate of draw inspiration from the audio examples given to them. This shows that, despite the openness of the system, users were still allowed a level of control and reproducibility that enables them to satisfactorily realise their musical ideas. The fact that the interface was unquantized, and therefore rhythms created with it would not have mechanical precision but rather have slight timing errors, did not deter most users. This confirms earlier quoted research which stated that these deviations in timing are in fact advantageous in creating expressive sounding music [14]. Users were creating music that they saw as being different, and this self-reporting aspect is important as the quality and acceptability of the music is defined by what the user finds as acceptable. It seems that implementing an open interface with no bounds placed on the types of input acceptable by quantizing has led to a wider variety of outputs from users. Examples of this could be seen in users finding multiple variations of the drawings that can yield a similar result or moving forward in experimenting with the more generative aspect of the interface. The fact that users also incorporated the unquantised loops in a usage pattern, where the loops would act as a base rhythm that they improvised over, further shows the applicability of the unquantized nature of the system for users. Overall this majority group showed encouraging signs of being able to both cope with and thrive using such an open system.

As mentioned in the introduction to this section, though, members of group B did not actively engage or seek to explore the system any further than they absolutely had to. Two users even cut the time taken to run the evaluation shorter by insisting that they no longer wanted to continue exploring the system and had seen all that they wanted to. In terms of experience level within this group, it was a mixed group consisting of some users who described themselves as musicians and some who do not. One of the biggest commonalities of users in this group was the fact that a majority of them had had some limited experience with music production, mainly on desktop DAWs. Indeed this seemed to be part of the reason that they could not engage with the probe. The unquantized nature of the probe stood against the interfaces that they had used before, which used discrete grids and emphasised quantisation. These users would often try to recreate the gridded sequencers that they were accustomed to by trying to draw straight lines, but would more often than not fail. This is when users would often revert to simply tapping out rhythms on the interface, forgoing the sequencing functionality altogether. Thus, for them, the unquantized and non-linear sequencing that the interface presented was far too much of an obstacle to get around for them to reach the point where using the interface became a pleasurable experience. These users would often explicitly say that the interface would benefit from having a "grid… or some kind of quantisation feature". While this group was in the minority, their experiences do tell us a lot about the interface and the role of unquantized interfaces, especially for users who have had similar previous experiences with DAWs

Technology probes are meant to allow researchers to elucidate a design space by highlighting users' behaviours in-situ and unexpected behaviours. Hence this minority group's behaviours are still important, and speak to a balance that must be found within systems such as this one. While some interfaces impose more of a framework on users than others, it is virtually impossible to be non-impositional in designing an interface. In trying to achieve the goal of imposing as little as possible, this interface lost some users. The fact that they could not engage with the system means that they felt that whatever music they were trying to create or had in mind could not be reached. While employing a grid structure within the probe might undermine some of its key principles and potentially alienate the other users, a compromise could be reached by. For example, we could allow for the copying of loops as well as nodes. This would allow for a slightly higher level of synchronisation and repetitiveness without forgoing any of the theorised advantages that having an open structure would give. This is because users would still have to draw loops on the unquantized 2d plane, but would then have the ability to use these loops as the basic building blocks for larger structures. While, again, this group was in the minority, their position is one to note and still draw inference from.

## 6.2 Summary

From these results, one is able to conclude that the probe was successful overall. Not only were most users able to create using the app, but a set of users were able to create with a complexity beyond what one might have expected. This was a complexity that the drawing gestures helped users to achieve, in a way that they might not have been able to if sequencing in any other way. Indeed, for some users who had already looked into music production, it was an approach that they hadn't tried before. The reflective practice that was theorised would play a large role when the activity of sequencing represented by a continuous gesture was indeed happening. Allowing users to actively draw time loops, as opposed to being "given" these loops in perfect time sequences forced the users to re-think the music they were creating, and move beyond these bounds. The unconventionality of this probe led to unconventional music, but the interface still offered users enough control to be able to replicate simple loops. Thus, it showed its applicability beyond simply creating *unique* music. This, though, seems only to stand to further emphasise the importance of the fact that the majority of users didn't make simple loops when using the interface, but rather chose to aspire for something more.

The closing keynote at the CHI 2005 conference, Michel Waisvisz, challenged the community to stop tinkering with interfaces and allow people time to learn to use and access the underlying functionality of the computer. This paper is not so much about the interface to existing music, but creating an interface to afford entirely new forms of music to exist. Our hope is that, by starting with notions of embodiment and optimizing the interface for creativity, we can begin to address that challenge – if we get the initial direction correct at the start, then creativity can increase and the interface can be left to be interpreted by users.

## 6.3 Future work

The work and results presented in this thesis do show a scope for future work which further explores the area of gesture based drum sequencing. The work here might be a definitive step, but it can be but the first towards a better understanding of both gestural interfaces and drum sequencing. The first area involves the musical range of the interface presented here, which only deals with what could be described as a macro level view of music production. A continuation of this research could involve looking at the macro levels of production too that involve not only the creation of beats and loops, but also their successful arrangement over a longer period of time. While this would fundamentally shift in focus away from sequencing, it would also represent a step towards the commercial applicability of the results.

Further work could also be done in investigating the balance between synchronise-ability and openness in the application. While nothing in the research shows that implementing a completely gridded structure would be advantageous, work can still be done in investigating other strategies. Such strategies could involve finding ways to allow for the copying and manipulation of the sizes of already drawn loops.

Lastly a longitudinal study based on these findings could also bring more to the research area by seeing how people's behaviours change over a longer period of time. While it is impossible to predict how long it would take for a user to become a virtuoso, or whether that would happen at all, a longitudinal study could help to further map out the learning curve that users follow over a given period of time. This work has shown that unquantized, drawn loops are a viable method for drum sequencing. A longitudinal study could take the next step and see how users who are already musicians integrate the system into their already established work cycles, and what kind of music they produce.

# 7 References


1. Ableton Live. https://www.ableton.com/
2. Agawu, Kofi. "The Invention of" African Rhythm"." *Journal of the American Musicological Society* (1995): 380-395
3. Albert Glinsky, "*Theremin: ether music and espionage*." Urbana and Chicago: University of Illinois Press, 2000 p.136.
4. Alt, F. Schneegaß, S., Schmidt, S., Muller, J., Memarovic, N., "How to evaluate public displays." *Proceedings of the 2012 International Symposium on Pervasive Displays*. ACM, 2012.
5. Antaki, C., Billig, M., Edwards, J., and Potter, J. Discourse analysis means doing analysis: a critique of six analytic shortcomings. Discourse Analysis Online, 1(1), 2003.
6. Appert, C., & Zhai, S. (2009). Using strokes as command shortcuts: cognitive benefits and toolkit support. Proc. international conference on Human factors in computing systems (pp. 2289-2298).
7. Garage Band apple.com/ios/garageband
8. Aramaki, M., Kronland-Martinet, R., Voinier, T., & Ystad, S. (2006). A percussive sound synthesizer based on physical and perceptual attributes.Computer Music Journal, 30(2), 32-41.
9. Arar, Raphael, and Ajay Kapur. "A HISTORY OF SEQUENCERS: INTERFACES FOR ORGANIZING PATTERN-BASED MUSIC." Proceedings of the Sound and Music Computing Conference 2013, SMC 2013, Stockholm, Sweden
10. Assayag G., Baboni J. and Haddad K., "Open- music 4.0 user's manual reference", IRCAM, Paris, 2001. www.ircam.fr/openmusic][ Assayag G., Baboni J. and Haddad K., "Open-music 4.0 user's manual reference", IRCAM, Paris, 2001. www.ircam.fr/openmusic
11. Ballagas, R., Borchers, J., Rohs, M., Sheridan, J.G., "The smart phone: a ubiquitous input device." *Pervasive Computing, IEEE* 5.1 (2006): 70-77
12. Banister, P., "Qualitative Methods In Psychology: A Research Guide" McGraw-Hill International, 2011.
13. Behrendt, F.: 'From calling a cloud to finding the missing track: Artistic approaches to mobile music', in: '2nd International Workshop on Mobile Music Technology' Vancouver, Canada (2005)
14. Bilmes, J. A. (1993). Techniques to foster drum machine expressivity. Proc.International Computer Music Conference (pp. 276-276).
15. Blake,T., Sharp, S., "CONTINUUM, EXPANDED" online http://rwm.macba.cat/uploads/twitter/continuum-expanded.pdf
16. Boring, S., Ledo, D., Chen, X., Marquardt, N., Tang, A., Greenberg, S., "The fat thumb: using the thumb's contact size for single-handed mobile interaction." Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services. ACM, 2012])
17. Boys, Beastie, and Super Disco Breakin., *From Kitsch to Kanye: The Meteoric Rise of the TR-808*. Jillian Reddy 21M. 380 11/23/2009
18. Bryan, N.J., Herrera, J., Oh, J., and Wang, G.: 'MoMu: A mobile music toolkit', in: 'Proceedings of the International Conference on New Interfaces for Musical Expression', Sydney, Australia (2010) pp. 174-177
19. Buxton, W., W. Reeves, R. Baecker, and L. Mezei, "The use of hierarchy and instance in a data structure for computer music." Computer Music Journal, 1978. 2(4):p. 10-20.
20. Cao. X. and Zhai. S., (2007). "Modeling human performance of pen stroke gestures." Proc. ACM Conference on Human factors in computing systems, 1495–1504.
21. Card, S.K., Moran, T.P., and Newell, A. The Psychology of Human-Computer Interaction, Lawrence Elrlbaum, London, 1983
22. Cascone, Kim. "The aesthetics of failure:"Post-digital" tendencies in contemporary

computer music." *Computer Music Journal* 24.4 (2000): 12-18

23. Cross, Ian. "Music, cognition, culture, and evolution." Annals of the New York Academy of sciences 930.1 (2001): 28-42
24. Crouch, Mira, and Heather McKenzie. "The logic of small samples in interview-based qualitative research." *Social science information* 45.4 (2006): 483-499.
25. Csikszentmihalyi, M. (2009). Creativity: Flow and the Psychology of Discovery and Invention. HarperCollins.
26. David Smith, J., Nelson, D., Grohskopf, L., Appleton, T., "What child is this? What interval was that? Familiar tunes and music perception in novice listeners." *Cognition* 52.1 (1994): 23-54.
27. Dobrian, C., & Koppelman, D. (2006, June). The 'E' in NIME: musical expression with new computer interfaces. Proc. New interfaces for musical expression (pp. 277-282).
28. Essl, G. and Rohs, M., 2006, Mobile STK for Symbian OS, Proceedings of the International Computer Music Conference, pages 278 -281
29. Essl, G., & Rohs, M. (2007). Shamus-a sensor based integrated mobile phone instrument. In Proceedings of the International Computer Music Conference ICMC (2007) (Vol. 40, p. 50).
30. Essl, G., & Rohs, M. (2009). Interactivity for mobile music-making. Organised Sound, 14(02), 197-207.
31. Essl, G., Rohs, M., & Kratz, S. (2010, June). Use the Force (or something)-Pressure and Pressure-Like Input for Mobile Music Performance. In NIME 2010 Conf. on New Interfaces for Musical Expression.
32. Fels, S., Lyons, M., "Advances in new interfaces for musical expression." *ACM SIGGRAPH 2011 Courses*. ACM, 2011
33. Fels, Sidney. "Designing for intimacy: Creating new interfaces for musical expression." *Proceedings of the IEEE* 92.4 (2004): 672-685.
34. Flores, L. V., Pimenta, M. S., Miranda, E. R., Radanovitsck, E. A., & Keller, D. (2010, October). Patterns for the design of musical interaction with everyday mobile devices.Proc. Symposium on Human Factors in Computing Systems (pp. 121-128).
35. Frohlich, D. M. (1993). The history and future of direct manipulation. Behaviour & Information Technology, 12(6), 315-329.
36. Gadamer, Hans-Georg. "Truth and Method, trans." *W. Glen-Dopel, London: Sheed and Ward* (1975)
37. Gaver, W., Dunne, T., and Pacenti, E., Cultural probes. Interactions, 6(1):21–29, 1999
38. Gaye, L., Holmquist, L. E., Behrendt, F., & Tanaka, A. (2006, June). Mobile music technology: report on an emerging community. In Proc. New interfaces for musical expression (pp. 22-25).
39. Geiger, G.,2003, PDa: Real Time Signal Processing and Sound Generation on Handheld Devices, Proc. International Computer Music Conference
40. Geiger. G. Using the Touch Screen as a Controller for Portable Computer Music Instruments. In Proceedings of the NIME, Paris, France, 2006.
41. Goldin-Meadow, S., H. Nusbaum, S. D. Delly, and S. Wagner. Explaining math: Gesturing lightens the load. Psychological Science 12(6). pp. 516-22, 1991
42. Goldschmidt G., "The dialectics of sketching", Creativity Research Journal, 4(2):123–143, 1991
43. Grech, Michelle R., Tim Horberry, and Andrew Smith. "Human error in maritime operations: Analyses of accident reports using the leximancer tool."*Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. Vol. 46. No. 19. SAGE Publications, 2002
44. Grogan, A. "Today's producers." *Engineering & Technology* 6.11 (2011): 32-35.
45. Guest, G., Bunce, A., and Johnson, L., "How many interviews are enough? An experiment with data saturation and variability." *Field methods* 18.1 (2006): 59-82.

46. Hassenzahl, Marc. "User experience (UX): towards an experiential perspective on product quality." *Proceedings of the 20th International Conference of the Association Francophone d'Interaction Homme-Machine*. ACM, 2008.
47. Hearne, Herb. "Electronic Production of Percussive Sounds." *Audio Engineering Society Convention 12*. Audio Engineering Society, 1960
48. Hong. J., and Landay. J., (2000). Satin: a toolkit for informal ink-based applications: Proc. ACM Symposium on User interface software and technology, 63–72
49. Hutchins, E. (2010). Enaction, imagination, and insight. Enaction: Towards a New Paradigm for Cognitive Science, 425-450.
50. Hutchinson, H., Mackay, W., Westerlund, B., Bederson, B. B., Druin, A., Plaisant, C., ... & Eiderbäck, B. (2003). Technology probes: inspiring design for and with families. Proc.SIGCHI conference on Human factors in computing systems (pp. 17-24).
51. IFPI Digital Music Report 2014 ifpi.org/downloads/Digital-Music-Report-2014.pdf
52. John, D. "Updating the Classifications of Mobile Music Projects."
53. Johnstone, A., Candy, L., Edmonds, E. (2008). Designing and Evaluating Virtual Musical Instruments: Facilitating Conversational Interaction. Elsevier Ltd. Design Studies, Vol. 29, No. 6
54. Jones, M., and Marsden, G., “Mobile interaction design.” John Wiley & Sons, 2006
55. Jordà, S. (2004, June). Digital instruments and players: part I---efficiency and apprenticeship. In Proceedings of the 2004 conference on New interfaces for musical expression (pp. 59-63).
56. Kalba, Kas. "The adoption of mobile phones in emerging markets: Global diffusion and the rural challenge." *International Journal of Communication* 2 (2008): 31
57. Kamien, Roger, and Anita Kamien. *Music: an appreciation*. McGraw-Hill, 1988
58. Katz, Mark. Capturing sound: how technology has changed music. Univ of California Press, 2010
59. Ketabdar, H., Jahanbekam, A., Yuksel, K. A., Hirsch, T., & Haji Abolhassani, A. (2011, January). MagiMusic.. Proc International conference on Tangible, embedded, and embodied interaction (pp. 241-244).
60. Kirke, Alexis, and Eduardo Reck Miranda. "A survey of computer systems for expressive music performance." *ACM Computing Surveys (CSUR)* 42.1 (2009): 3
61. Klemmer, S. R., Hartmann, B., & Takayama, L. (2006, June). How bodies matter: five themes for interaction design. Proc. Designing Interactive systems (pp. 140-149).
62. Kurtenbach. G. and Buxton. W. (1994). User learning and performance with marking menus. Proc. ACM Conference on Human factors in computing systems, 258–264
63. Kurtenbach. G., Fitzmaurice. G., Owen. R. and Baudel. T. (1999). The hotbox: efficient access to a large number of menu-items. Proc. CHI ’99, 231–237.
64. Lee, Allen S., and Richard L. Baskerville. "Generalizing generalizability in information systems research." *Information systems research* 14.3 (2003): 221-243.
65. Levitin D.J., McAdams, S., Adams, R., (2002) “Control parameters for musical instruments: a foundation for new mappings of gesture to sound. organised Sound”, 7(2), 171-189
66. Macleod, M., Bowden, R. and Bevan, N. (1997) The MUSiC Performance Measurement Method. Behaviour and Information Technology, 16
67. Maki-Patola, Teemu, Perttu Hämäläinen, and Aki Kanerva. "The augmented djembe drum: sculpting rhythms." *Proceedings of the 2006 conference on New interfaces for musical expression*. IRCAM—Centre Pompidou, 2006.
68. Mäki-Patola, Teemu. "User interface comparison for virtual drums." *Proceedings of the 2005 conference on New interfaces for musical expression*. National University of Singapore, 2005.
69. Marino, Gérard, Marie-Hélène Serra, and Jean-Michel Raczinski. "The UPIC system: Origins and innovations." *Perspectives of New Music* (1993): 258-269

70. Markosian, N. (1995). The open past. Philosophical Studies, 79(1), 95-105.
71. Marsden, G. (2009), «Electronic tablecloths and the developing world», interactions, vol.16 (2), p. 67, URL http://doi.acm.org/10.1145/1487632.1487648
72. Melucci, Massimo, and Nicola Orio. "Musical information retrieval using melodic surface." *Proceedings of the fourth ACM conference on Digital libraries*. ACM, 1999.
73. Miller, Smit, Puckette, Pure Data: another integrated computer music environment., Proc. Second Inter-college Computer Music Concerts (1996), 37–41.
74. Morreale, F., De Angeli, A., & Masu, R. (2012). Interfaces for Musical Creativity for Non-Experts. DIS 2012
75. Mulder, Axel. "Virtual musical instruments: Accessing the sound synthesis universe as a performer." *Proceedings of the First Brazilian Symposium on Computer Music*. 1994.
76. Murphy, Damian, Kelloniemi, A., Mullen, J., Shelley, S. "Acoustic modeling using the digital waveguide mesh." *Signal Processing Magazine, IEEE* 24.2 (2007): 55-66.
77. Nielsen, J., "Why You Only Need to Test With 5 Users", Alertbox March 19, 2000, at http://www.useit.com/alertbox/20000319.html
78. Nielsen, Jakob. "Iterative user-interface design." *Computer* 26.11 (1993): 32-41
79. Nisbett, R. E., & Wilson, T. D. (1977). Telling more than we can know: Verbal reports on mental processes. Psychological Review, 84, 231-259
80. Norman, D., (1988). The psychology of everyday things. New York: Basic Books
81. Norman, D., "Affordances and design." Unpublished article, available online at: http://www. jnd. org/dn. mss/affordances-and-design. html (2004)
82. Orio. N, Schnell. N, and Wanderley. M. (2001) Input devices for musical expression: borrowing tools from HCI. Proc. Workshop New Interfaces for Musical Expression—ACM CHI01 Online. Available: http://www.nime.org/
83. Paine, G., Stevenson, I., & Pearce, A. (2007, June). The thummer mapping project (ThuMP). Proc. New interfaces for musical expression (pp. 70-77).
84. Paine, G., Stevenson, I., Pearce, A.,. "The thummer mapping project" *Proceedings of the 7th international conference on New interfaces for musical expression*. ACM, 2007
85. Penn-Edwards, Sorrel. "Computer Aided Phenomenography: The Role of Leximancer Computer Software in Phenomenographic Investigation." *Qualitative Report* 15.2 (2010): 252-267
86. Poupyrev, Ivan, Michael J. Lyons, and Sidney Fels. "New interfaces for musical expression." *CHI'01 Extended Abstracts on Human Factors in Computing Systems*. ACM, 2001
87. Propeller head Figure propellerheads.se/products/figure/
88. Ren, Z., Mehra, R., Coposky, J., & Lin, M. (2012, May). Designing virtual instruments with touch-enabled interface. In CHI'12 Extended Abstracts on Human Factors in Computing Systems (pp. 433-436). ACM.
89. Rokeach, M. The nature of human values. New York: Free Press, 1973
90. Roth, Volker, and Thea Turner. "Bezel swipe: conflict-free scrolling and multiple selections on mobile touch screen devices." Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2009
91. Rubine, D. (1992). Combining gestures and direct manipulation. Proc. SIGCHI conference on Human factors in computing systems (pp. 659-660)
92. Rule, Greg. "DJ Shadow+ Akai MPC= History." *Keyboard, October* (1997)
93. Schacher, Jan C. "The Body in Electronic Music Performance." (2012).
94. Schiemer, Greg, and Mark Havryliv. "Pocket Gamelan: Tuneable trajectories for flying sources in Mandala 3 and Mandala 4." *Proceedings of the 2006 conference on New Interfaces for Musical Expression*. IRCAM—Centre Pompidou, 2006
95. Schlei, K. TC-11: A Programmable Multi-Touch Synthesizer for the iPad. https://itunes.apple.com/za/app/tc-11/id488577050?mt=8

96. Schwarz D., Beller G., Verbrugghe B. and Britton S.,“Real-Time Corpus Based Concatenative Synthesis with CataRT” Intl. Conf on DAFx,2007
97. Shneiderman, Ben. "Direct manipulation for comprehensible, predictable and controllable user interfaces." *Proceedings of the 2nd international conference on Intelligent user interfaces*. ACM, 1997.
98. Sinyor, E., McKay, C., Fiebrink, R., McEnnis, D., and Fujinaga, I., 2005, Beatbox classification using ACE., Proceedings of the International Conference on Music Information Retrieval
99. Smith, A. E. "Leximancer manual (version 2.2)." *University of Queensland: Australia* (2005).
100. Smith, A. E., & Humphreys, M. S. (2006). Evaluation of unsupervised semantic mapping of natural language with Leximancer concept mapping. Behavior Research Methods, 38(2)
101. Smith, Andrew E. "Automatic extraction of semantic networks from text using Leximancer." Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: Demonstrations-Volume 4. Association for Computational Linguistics, 2003
102. Snoman, Rick. The dance music manual: tools, toys and techniques. Taylor & Francis, 2004
103. Stead, Alistair G., Alan F. Blackwell, and Samuel Aaron. “Graphic Score Grammars for End-Users.” Proceedings of the International Conference on New Interfaces for Musical Expression (NIME). 2012.
104. Stowell, D., Plumbley, M. D., & Bryan-Kinns, N. (2008). Discourse analysis evaluation method for expressive musical interfaces.Proc. New Interfaces for Musical Expression (NIME)
105. Stowell, D., Robertson, A., Bryan-Kinns, N., & Plumbley, M. D. (2009). Evaluation of live human–computer music-making: Quantitative and qualitative approaches. International Journal of Human-Computer Studies, 67(11), 960-975.
106. Strachan, S., Eslambolchilar, P., Murray-Smith, R., Hughes, S., O’Modhrain, S., (2005), ”GpsTunes: Controlling Navigation via Audio Feedback”, Proc. International Conference on Human Computer Interaction with Mobile Devices & Services
107. Tanaka and P. Gemeinboeck., 2006, A framework for spatial interaction in locative media, In NIME ’06: Proceedings of the 2006 conference on New Interfaces for Musical Expression, pages 26–30
108. Tanaka, A., Valadon, G. and Berger, C., 2007, Social Mobile Music Navigation using the Compass, Proceedings of the International Mobile Music Workshop
109. Tanaka, Atau. "Mapping out instruments, affordances, and mobiles." NIME, 2010
110. Tanaka, Atau. "Mobile music making." *Proceedings of the 2004 conference on New interfaces for musical expression*. National University of Singapore, 2004
111. Thiebaut, J., Healey, P., Kinns, N., Mary, Q., "Drawing electroacoustic music." *Proc. ICMC*. Vol. 8. 2008.
112. tms.org/pubs/journals/jom/0708/roncone-0708.html
113. Trueman, D. “Why a laptop orchestra?” Organised Sound, 12(2):171–179, 2007
114. Trützschler von Falkenstein., J., 2011, “Gliss: An Intuitive Sequencer for the iPhone and iPad”, Proceedings of the International Conference on New Interfaces for Musical Expression
115. Vaggione H., “Some ontological remarks about music composition processes”, Com- puter Music Journal, pp 54-61, 2001
116. Van de Walle, Guy, Phil Turner, and Elisabeth Davenport. "A study of familiarity." *Human-Computer Interaction-INTERACT*. Vol. 3. SVB, 2003
117. Venderosa, Tony. The techno primer: the essential reference for loop-based music styles. Ed. Rick Mattingly. Hal Leonard Corporation, 2002.
118. Wanderley, M. and Orio, N. (2002). Evaluation of Input Devices for Musical Expression:

Borrowing Tools from HCI. Computer Music Journal, Vol. 26, No.3. pp 62-76.
119. Wanderley, M. M. (2001, September). Gestural control of music. International Workshop Human Supervision and Control in Engineering and Music (pp. 632-644).
120. Wanderley, M. M., & Depalle, P. (2004). “Gestural control of sound synthesis” .Proc.the IEEE
121. Wanderley, M. M., & Orio, N. (2002). “Evaluation of input devices for musical expression: Borrowing tools from hci”. Computer Music Journal, 26(3), 62-76.
122. Wang, Ge, Georg Essl, and Henri Penttinen. "MoPho–A Suite for a Mobile Phone Orchestra." *8 th International Conference on New Interfaces for Musical Expression NIME08*. 2008
123. Wang, Ge. "Designing Smule’s iphone ocarina." Proceedings of the International Conference on New Interfaces for Musical Expression. Pittsburgh. 2009
124. Williams, Paul. A research project exploring selected post-punk influences and their relevance to my contemporary compositional work. Diss. University of Huddersfield, 2010
125. Wolf. C., Rhyne. J. and Ellozy. H. (1989). The paper-like interface.Proc. International conference on human-computer interaction on Designing and using human-computer interfaces and knowledge based systems, 494–501.
126. Xenakis, I; Brown, R., “Concerning Time” Perspectives of New Music, Vol. 27, No. 1. (Winter, 1989), pp. 84-92