



The  
University  
Of  
Sheffield.

Laptop Performance in Electroacoustic Music:  
*The current state of play*

MMus

I. J. Algie  
2012



Laptop Performance in Electroacoustic Music:  
*The current state of play*

Ian Algie

MMus (Reg No. 090256507)

Department of Music

April 2012

Abstract

This work explores both the tools and techniques employed by a range of contemporary electroacoustic composers in the live realisation of their work through a number of case studies. It also documents the design and continued development of a laptop based composition and performance instrument for use in the authors own live performance work.



## Contents

<b>TABLE OF FIGURES .....</b>	<b>6</b>
<b>PART 1 – WHO IS DOING WHAT WITH WHAT? .....</b>	<b>7</b>
INTRODUCTION.....	8
SCANNER.....	13
HELENA GOUGH.....	15
LAWRENCE CASSERLEY .....	19
PAULINE OLIVEROS .....	22
SEBASTIAN LEXER .....	25
LEAFCUTTER JOHN .....	28
ALEX MCLEAN .....	30
DAN STOWELL .....	34
SCOTT HEWITT.....	36
JEFF KAISER.....	38
BRIAN CRABTREE.....	41
GREGORY TAYLOR.....	45
FOUR TET .....	48
BEARDYMAN .....	50
CHRISTOPHER WILLITS .....	53
HANS TAMMEN .....	55
CONCLUSIONS .....	57
<b>PART 2 – THE INSTRUMENT .....</b>	<b>59</b>
WHAT AM I TRYING TO ACHIEVE? .....	60
WHY? .....	62
HOW? .....	64
IN PRACTICE.....	67
PLASTICITY.....	70
LITTLE MACHINES .....	74
NTWICM .....	77
MT02.....	80
HOLD THAT THOUGHT .....	83
COOKING WITH KIDS.....	84
SET IN STONE.....	87
CONCLUSION .....	89
<b>REFERENCES .....</b>	<b>92</b>
<b>APPENDIX 1 – PORTFOLIO SUBMISSION AND PROGRAMME NOTES .....</b>	<b>96</b>
<b>APPENDIX 2 – DVD CONTENTS .....</b>	<b>98</b>

## Table of Figures

Figure 1 Monome 128.....	43
Figure 2 Plasticity Overview .....	70
Figure 3 Plasticity OSC to Live bridge .....	71
Figure 4 Plasticity iPad UI.....	72
Figure 5 Little Machines interface.....	75
Figure 6 Little Machines iPhone UI .....	76
Figure 7 NTWICM patch .....	79
Figure 8 mt02 patch in presentation mode.....	82
Figure 9 Hold That Thought patch in presentation mode.....	83
Figure 10 Cooking With Kids patch.....	86
Figure 11 Set in Stone patch .....	88
Figure 12 Set in Stone iPad UI.....	89

## Part 1 – Who is doing what with what?

## Introduction

Artists have long been interested in the applications of new technology and in one way or another computers have had some hand in the composition process since Cage and Hiller's early experiments with the Illiac suite.

Conversely technologists have long had an interest in both the creation of sound and the capture of sound. (Roads, C, 1996) More recently though, the evolution of the personal computer<sup>1</sup> has meant that developments in power, performance, size and price have conspired to create the perfect storm for using and abusing this technology in a live performance context. It is unlikely that any contemporary composer's work is not touched by technology at some point in its life cycle from creation through notation and onto the recording or live realisation. (Dubois, L, 2007) The further possibilities offered by actively engaging with the computer as a creative partner in the composition and performance processes are potentially limitless.

The laptop has now become a ubiquitous tool for the performance of electroacoustic music across a wide range of styles and genres, from the relatively simple task of being a sound source for a performer via the use of soft synths and samplers, to being the sole tool used in a multimedia audio visual real time performance set and pretty much everything in between.

---

<sup>1</sup> Whilst using the term computer generically here, I think in this instance it makes sense to include perhaps the dedicated digital signal processing (DSP) chips found in embedded computer systems such as FX units



The first part of this dissertation text explores some of the artists that are currently exploiting developments in technology in the creation and performance of their work. The list is by no means meant to be exhaustive and is really an indicative exploration of what artists are doing and perhaps how and why they are doing it with a particular set of tools in a live environment.

It would be futile to attempt to list all of the possible permutations of creativity, hardware and software and all of the variables therein, but this section aims to explore why the artists decided to go in a particular creative direction and how and why their chosen toolset facilitated, or sometimes even informed this movement. Is the creative element being driven by the technological possibilities or is the technology serving the creativity of the artist?

To try and keep as up to date as possible in this incredibly fast paced area, a substantial quantity of the research has come from the internet, the blogosphere and wherever possible personal conversations with the artists involved and personal experience of their live performance work. There are obviously some key texts covering improvisation, composition and performance theory and practice and a great deal of the underpinning ideas will still stand, however the last twenty years have seen developments that have allowed artists to expand their practice in a number of ways, sometimes quite subtly and at others more radical. For some practitioners it

has simply made what they do a little easier to achieve and allowed their workflow to be simplified. For others it has meant that they are able to gain access to and utilise technologies that would have been unthinkable in a real time context until relatively recently.

There are of course a huge range of creative tools available to the contemporary music technologist. Some of the tools discussed are commercially available, some are freely available through open source channels and some are built and/or programmed by the artists themselves. Sometimes these tools are in a constant state of flux and development and sometimes the tools have been refined over a number of years to provide a clearly defined set of compositional or performative strategies and processes.

In talking to artists I have uncovered a number of personal, financial, technical and philosophical reasons for them to utilise different hardware and software tools: I will be exploring these together with the creative decision making process. These approaches reflect the notion of the changing roles of composer and performer and the concept of the finished work versus the improvised set that the technology has now made possible.

This research arose from a personal desire to develop my own individual environment. I have tried to illustrate the artists' overall strategies for both composition and performance and not to promote any specific tool or

paradigm of working. Consequently my findings have prompted me to explore an extensive range of tools in my own practice which has definitely proved useful in stretching my own ideas about what constitutes composition and performance and where the increasingly blurred line might now lie (as discussed in part two).

It is interesting to note the similarities and differences in practice especially where different methods and technologies can sometimes lead to similar results. For example a painstaking process of micro editing and montage (Helena Gough or Carsten Nicolai) could be similar in feel to the real time improvised performance work of another artist (Oval or Scanner). This is not meant to be a criticism of either approach, but merely an observation of how on occasion very different approaches can yield similar results whilst slight differences in the process can lead to wildly differing output, depending completely on how the particular technology is used in any given context. For some a process might seem long-winded and cumbersome, whereas for others this attention to detail is perhaps part of a meditative mind-set absolutely necessary for the accurate interpretation of their creative thoughts and processes.

Many of the artists highlight the fact that improvisation has played some part in their creative process, but they go on to say that often this improvisation has been restricted to the compositional or even pre-compositional elements of the overall strategy, experimenting with various

treatments and manipulations of their original source material before committing it to a more permanent form, or perhaps fine tuning a mix or automation through performance, but again at the end of the process the work is committed to a concrete form for dissemination or archiving.

The continued development and evolution of the personal computer, and especially laptop based tools has given artists the freedom to re-purpose their work into a live context and given them the ability to extend any improvisatory elements in to their live performance work; for example the recent Max For Live product from Ableton and Cycling74 essentially combines Max/MSP and Live into the one piece of software that offers the advantages of both, the traditional sequencing timeline, the non linear arrangement possibilities and the ability to create audio and MIDI elements from scratch.

For dissemination and recording purposes the final product still tends to be some kind of audio or video file (as is necessary), but in a live context the artist is much freer to engage with their audience interactively should they wish to do so and use this interaction to inform the work as it evolves.

## **Scanner**

Robin Rimbaud, aka Scanner, came to prominence in the early 90's through his performance and composition work using an analogue radio scanner as the primary source of compositional material. Listening in on the airwaves of the then burgeoning analogue mobile telephone networks, he would sample whatever happened to be going on at the time of his performances and then use these conversations as a basis for electronic improvisation using the signal path of the scanner itself along with a sampler, an effects unit and a mixer, as the basis for live performances. (Rimbaud, R) The live input would be effected, processed and often augmented with a collection of pre-recorded elements played back from the sampler.

Essentially a very similar setup and process was used for his composition work, however there was obviously much more control over both the material that was being treated and the type of manipulation and strategies being used due to the non real-time nature of the process. The treatments were often more severe due to the fact that any processing could happen in non real-time and then be placed in a timeline with a great deal of precision leading to a much more structured recorded output. The use of a basic four track tape machine along with a sampling delay facilitated a great deal of flexibility in the mixing of loops and ambiences. Elements were assembled in an avant-garde fashion after Rimbaud's long-term fascination with Cage and his methods. (Prendergast, M, 1995)

Scanner's first three self-produced albums (Scanner 1, Scanner 2 and Mass Observation) were all created with this setup. The output however gradually evolved to include more soundscape and ambient music production, using a much wider variety of source material as he moved away from the scanner as the main source of input into the live setup.

More recently his live performances have continued to evolve and are now run predominantly from a laptop using Ableton Live<sup>2</sup> to trigger a library of pre-recorded samples and loops. (2004) Tracks can be performed using live triggering from a preselected group of audio files, allowing for a previously recorded track to be deconstructed into its constituent components and then essentially rearranged and remixed on the fly. Alternatively a track can be built from scratch, selecting loops, audio files and even incorporating live input and combining them in an improvisatory manner. In this way the laptop and Live software has replaced and extended the notion of the sampler in both studio and live contexts, giving very flexible access to a library of sounds and live mixing possibilities.

Further sonic manipulation, and an element of visual performance, is added to the live work through the use of an Alesis AirFX<sup>3</sup> unit to effect and selectively loop the audio output from the laptop. This live system has

---

<sup>2</sup> <http://www.ableton.com> Live is a digital audio workstation with some interesting nonlinear possibilities.

<sup>3</sup> <http://www.alesis.com/airfx> Now discontinued, the AirFX allowed for the control of effects parameters through the movement of body parts through a sphere above the unit without physical contact.

continued to be refined and has become increasingly reliant on Live as its core.

### **Helena Gough**

Helena Gough is an English sound artist now based in Berlin. Her work is primarily studio based and the compositional strategies involved in her work are fairly traditional in the way that pieces are often produced by using a variety of processes and tools to create the source material before finally arranging them in a multi track digital audio workstation environment in non real-time.

Her work is often categorised by the use of micro editing and micro montage, using very small elements of original source recordings as the raw material for longer compositional applications. The works are focussed upon the “collection and manipulation of real world sound material and the exploration of its abstract properties.” (Gough, H, 2012) This material is used in improvisations with the technology, in the first instance without a pre-planned structure. The composition process generally works from the bottom up, from the material and toward a structure, rather than through the application of a pre-imposed structural idea.

There is deliberately no attempt to keep track of this original material (unless it is the product of an instrumental player’s input and credit needs to

be given as such.) or how it has been manipulated, and some elements of source material may be recycled repeatedly throughout different pieces in a cycle of editing, layering and mixing.

Her main studio based composition work uses Reaper<sup>4</sup> as the digital audio workstation tool to arrange and assemble the source material, along with a range of fairly standard plug-ins. Having previously worked with both Nuendo and Pro Tools the move to Reaper was a pragmatic one based around the continued cost of the tools. Reaper offers many, if not all, of the facilities of the other software systems at a fraction of the cost and it was becoming increasingly difficult to justify the cost of the ‘industry standard’ tools when the use of more cost effective tools had no appreciable impact on the quality of the material being produced.

Ableton Live is also used from time to time during the early compositional stages to enhance the real time nature of the improvisatory framework:

“...it allowed me to work in a more spontaneous way. When I am stuck, I set Wiretap recording and play around with multi-tracked blocks of material and long chains of plug-ins. Usually 90% of what I get is crap, but 10% yields things that are unexpected, or that I couldn’t generate simply by editing or mixing.” (Gough, H, 2012)

This toolset of relatively inexpensive and off the shelf software tools allows Gough to create a combined approach utilising both more traditional non

---

<sup>4</sup> <http://www.reaper.fm/>



real-time structural ideas and a much freer real-time improvisatory framework often within the development of the same pieces. The non-linear elements of Live are really what set it apart from the majority of the rest of the contemporary digital audio workstations, being particularly useful when incorporating improvisational elements in to composition and performance. Many alternatives tie the composer strictly to the linear timeline, whereas Live allows the user to play with structural elements easily and experiment with arrangement and textural ideas in a free environment. Physical controls can be mapped freely to any parameter of the software and the follow actions element of the clip view allows for clips to trigger other clips upon completion, with a certain degree of randomness if required. This relatively simple section of the application means that some quite interesting generative arrangements can be created with the raw material.

Her current live setup is quite minimal and based essentially around the same set of tools. A laptop running Live is combined with a multi channel soundcard (MOTU Ultralite) and a commercially available MIDI controller (evolution uc33). A bank of source material, which has been organised into categories, can then be selected, effected and combined constitutes the raw material for a performance. Studio based rehearsal allows decisions to be made on a loose structure for any performance, and then small segments of compositions are used as a framework. This gives the flexibility to improvise transitions, shift timings and add layers to textures reinterpreting the recorded material into a live performance variant. (Gough, H, 2012)

In this particular case Live is used both as, essentially a very large sampler, and a host for a selection of digital signal processing tools through the Virtual Studio Technology (VST)<sup>5</sup> system of plugins. Live allows the artist to deconstruct their studio creations and repurpose the elements into a real-time context with hands on control of both when and how audio events are sequenced and their path through the various effect chains. Whilst in many ways this approach is very simplistic, it is also a very effective way of removing the artists from a 'space bar to play' approach, something that live laptop music is often accused of, and allows real engagement with the elements at hand and potentially more with the audience. It appears that in some ways this is only really one small step removed from the traditional playing of tape-based pieces, although it offers a number of possibilities in expanding and building on this history and in this respect is fundamental to the evolution of laptop based performance. The key here is the interaction beyond pressing play.

In this way the performance is able to keep old material alive (Gough, H, 2012) and becomes part of a cycle of development. The technology allows for the works to be performed with more of an element of chance and edginess, much more like the experience of a traditional instrumentalist, by building improvisation and risk into the work.

---

<sup>5</sup> A recognized standard for the development of software based virtual instruments and effects processors. <http://www.steinberg.net/en/home.html>

## **Lawrence Casserley**

Lawrence Casserley has been performing live electronic works since the late 60's in both solo and ensemble contexts. This live work often contains large elements of improvisation and needed the development of systems that were, and are, flexible enough to facilitate the level of interaction that this generally requires.

As the technology has continued to evolve, so too has the equipment that has been used in the various performance and composition systems that Casserley has developed: from the early adoption of analogue synthesis, through the use of digital effects, eventually settling on the use of the laptop as a musical tool. One thread running through all of this is Casserley's concern that the computer system should be an instrument in its own right and not just an adjunct to something else. This is a subtle but fundamentally different viewpoint from how a number of other laptop artists view the use of the computer as a tool.

This evolution has effectively focussed on the development of the Signal Processing Instrument. This system was developed from early ideas and experiments at IRCAM<sup>6</sup> and STEIM<sup>7</sup> using at the time cutting edge DSP processing hardware. Initially the early system used the ISPW system, which used a personal computer to control a separate hardware DSP module in real time. The software used to control this system was an early version of

---

<sup>6</sup> <http://www.ircam.fr/?&L=1>

<sup>7</sup> <http://www.steim.org/steim/>

Max (originally developed by Miller Puckette at IRCAM). This working method covered both the work Casserley produced during the 1980s designing digital signal processing machines (essentially trying to do something very similar to Max plus signal processing) and the early studio experiences of the 60s and 70s that used a physical patch cord paradigm. (Casserley, L, 2011) Whilst flexible, these early systems were not really portable and live performance was still time consuming and difficult to organise, requiring extensive setup.

A breakthrough came as the price of the necessary technology continued to fall and it became practical to use the same development system for audio as for control:- Max extended with the MSP (Max Signal Processing) real-time audio objects, on a consumer laptop was, and to a certain extent is currently the de-facto standard. For Casserley the instruments that had been dreamed about since the 70's could now be fully realised. (Casserley, L, 2011) Using a laptop and a variety of controllers allowed Casserley much more freedom for performance in terms of both interaction with the system and how and where this could take place.

His system is essentially based around various delays, filters and ring modulators that can be combined in a number of ways. Early on a conscious decision was made to use delays as opposed to loops. In terms of their existence in computer memory the difference is negligible, however it does force the performer to think differently about how they are interacting with

this audio. An explicit recording mode might have been distracting from controlling the performance element of the system. There is an immediacy and inevitability to the way that you are working with a delay as opposed to a loop. Moreover, Casserley was keen to point out that he grew up in an era of electronic music before sampling existed. (Casserley, L, 2011)

In a solo performance context the system is often used with acoustic input via microphones, for voice, monochords or selections from a collection of self-built percussion instruments. With ensemble work the input is the instrumental material of the other performers. In both settings the parameters of the system are altered through the use of various controllers. Over time Casserley has made use of a MIDI exoskeleton<sup>8</sup>, DrumKat<sup>9</sup>, Wacom tablet<sup>10</sup>, JazzMutant Lemur<sup>11</sup> and various keyboard and foot controllers. These controllers allow for the system to be controlled in a very gestural way which enhances its potential in a real time improvisation setting.

The same system, or a variation of it, is also used for the development of installation-based work. Much of Casserley's work focuses on the concept of networks and journeys, in both a metaphoric sense and also in terms of the audio in the pathways through the systems.

---

<sup>8</sup> <http://www.soundonsound.com/sos/oct06/articles/sonalog.htm>

<sup>9</sup> <http://www.alternatemode.com/drumkat.shtml>

<sup>10</sup> <http://www.wacom.eu/index2.asp?lang=en>

<sup>11</sup> [http://www.jazzmutant.com/lemur\\_overview.php](http://www.jazzmutant.com/lemur_overview.php)

This features both in performance where Casserley might for instance send audio into a signal chain and then allow it to take its natural course, much like the ideas behind some of the process music of the early minimalists. The audio feedback is then used to guide the improvisation as the performer plays with and against the resultant sounds. This is extended in an installation setting where the process becomes more automated to cater for the altered listening experience.

### **Pauline Oliveros**

Pauline Oliveros came to prominence in the late 60's as an early exponent of tape music. Whilst technology has evolved dramatically since she started composing, the central thinking behind her work has continued along a thread of developing sounds through the layering of textures and alteration of timbre over time. Early work utilised quite minimal tools, originally being based around a small number of variable oscillators, which could be combined through a small patching matrix and then fed into loops of tape for delay purposes. Quite early on the meditative nature of this music and the potential links with therapeutic work led to Oliveros' theories of deep listening techniques. (Oliveros, P, 2005)

Although very much based around performance in the sense that the artist had to control all elements of the system in real-time, this early work was necessarily studio grounded due to the size and sensitive nature of the various components involved. The move from using pure sine tones as input

to the system to the use of instrumental textures, primarily from Oliveros' own accordion playing, meant that real time live performance had become a real possibility. Whilst it worked sonically this period of performance was still difficult in terms of travelling and set up with the cumbersome combination of reel to reel tape machines required for the tape looping element of the system.

A major milestone in the evolution of what came to be known as the Expanded Instrument System (EIS) (Oliveros, P) was the release of the PCM42 digital delay system from Lexicon. This unit meant that the tape machines could be replaced giving the system both a much enhanced flexibility and a far smaller footprint and weight for touring purposes. This setup was the core of the system for some time. As it grew it became useful or possibly even necessary for a computer to become involved to allow for the control and synchronisation of the various elements along with the storage of pre-sets on a system wide basis. A Macintosh running Max (by this time owned and maintained by Opcode) was used for this purpose as MIDI was a useful standard across the effects and processors used and Max offered a convenient way of interfacing with the system through the development of custom patches.

The current state of the system is still based around these conceptual parts but the technological development has meant that it can now all hosted effectively in a laptop. The core of the Max based control system is there,

however the introduction of the MSP objects meant that most of the outboard equipment could be replaced with software. The only element that remained in hardware form for quite some time were the Lexicon units. Oliveros felt that their warmth and particular sound could not be effectively replicated otherwise. Even that though went soft when PSP developed a VST plugin recreation of the PCM42 unit<sup>12</sup>. (Oliveros, P) It is now included in the EIS as a plugin and so consequently the entire system is now software based.

In Oliveros' own work the system is predominantly used with the accordion as the input via contact microphones (however the system is input agnostic and has been used by others with a variety of instrumental, vocal and textural inputs) (2007). These pick up not just the relatively simple melodic and harmonic material but also the clicks, scratches and noises that can also be used as textures. The core of the system is a matrix switching section that allows for the patching of audio signal chains. Input can be sent to a number of multi tap delay lines, to the PCM delay units or to a number of looper units. Either an in-built low CPU usage and grainy reverb section can be patched in or the rather more taxing but much smoother Altiverb<sup>13</sup> convolution reverb can be used. Spatialisation is based around the VBAP<sup>14</sup> external for Max, although this requires a minimum of four speakers to be set up.

---

<sup>12</sup> [http://www.pspaudioware.com/plugins/delays/lexicon\\_psp\\_42/](http://www.pspaudioware.com/plugins/delays/lexicon_psp_42/)

<sup>13</sup> <http://www.audioease.com/Pages/Altiverb/>

<sup>14</sup> <http://www.acoustics.hut.fi/~ville/>



The system has also been used in the Telematic performances with which Oliveros has been involved. This extends the concept of the system being used to dislocate sound not only in time but also in space, extending the focus of her work.

### **Sebastian Lexer**

Sebastian Lexer currently works as a freelance recording engineer and programmer for interactive music and media software primarily developing systems to facilitate other artists' digital output. He began his own performing and composing career in a more traditional pianist role but the discovery of the potential coupling of instrumental textures and technology led to the long term development of his piano+ (Lexer, S, 2010) system. Essentially microphones capture the acoustic sounds of the piano and this is then analysed by software to report pitch, loudness and density information for use throughout the rest of the system. This control data is then used to manipulate and further treat the incoming audio to create a feedback loop in a sense.

Max and its digital audio extensions, MSP, are used as the main development tool for this system. This environment facilitates the visual development of an acoustic analysis system through the combination of core objects. It encourages the development of a modular system in which individual

elements can be developed and tested independently before being combined into a complete instrument.

The focus of the piano+ system is with the creation of a system that allows the player to extend their instrument and techniques and interact with the technology in a more organic fashion. Instead of using only the direct controls such as MIDI faders and pedal boards to influence and change the system's parameters, the system makes use of analysis data from the instrumental input. This real time input is captured continuously and the resultant data (e.g. pitch, loudness, density etc.) is used to further control real-time processes within the system itself. This feedback loop leads to quite a flexible and adaptive system that is ideally suited to free improvisation. The player can perform with the system from their own instrument interacting and reacting to the audio produced in response to their instrumental playing.

Again part of the thinking behind the development of this system is that the laptop and associated software becomes part of the instrument. It is not just an instrument being played through some effects, but it is a dynamic part of the timbre and texture. (Lexer, S, 2010) The distinction may be subtle, as if the performer was playing through effects, these could still be operated in real time by adjustment of the various control parameters via knobs, faders, sliders etc., but the idea of processes being triggered by the same movements and impulses that are being used to excite the acoustic

instrument feels slightly different in approach and I am sure that this subtle difference will have some mileage in the psychology of the performer.

The extensible nature of the system means that it is continually being refined and that newer alternative controllers can be utilised where appropriate. Open hardware platforms like the Arduino<sup>15</sup> have been used to further facilitate a range of real world sensor data, such as accelerometer input, so that the player might further influence the electronic side of the system by virtue of his physical movement either extra to the standard instrumental technique or simply as a consequence of playing their own instrument. This relatively simple system can be battery operated and can use Bluetooth transmission technologies, so is therefore flexible and transparent to the player. (Lexer, S, 2010)

The mapping of the instrumental and sensor input is open and flexible, allowing the performer to be both subtle and dramatic in their linking of instrumental playing to the acoustic output of the system. Ultimately this permits the performer to focus on their instrumental improvisation and have the dynamics of their playing directly inform the software, which in turn offers a further level of material with which to improvise. This directness of approach facilitated by the continual analysis of the incoming audio makes for a very flexible live performance framework.

---

<sup>15</sup> <http://www.arduino.cc/>

## **Leafcutter John**

Leafcutter John, aka John Burton, is an artist, songwriter and electronic musician who works across composition, recording, performing and installation areas using a variety of instrumental and electronic textures.

(Burton, J)

Much of his performance work centres on the use of digital signal processing (DSP) systems with a range of acoustic and electric instruments as the input source. The original input sounds are effected, twisted and mangled through a range of granular and spectral techniques to extend the sound in time.

These often make use of random and chaotic elements.

Initially compositions were studio-based creations and the bulk of Burton's recorded output still is. A standard DAW is used (often either Apple's Logic Pro, or Avid's ProTools) to assemble, process and arrange the source recordings. When looking for a way to take this material out live and move beyond the push space to play kind of mentality (Sellars, P, 2002) Leafcutter came across Max/MSP. Some brief exploration demonstrated that the original source recordings and associated effects chains and processes could be quite easily recombined using this software allowing for the live sound to be 'live'. Improvisation is very important and even though the audio files used were essentially the master files from his recorded output Burton is able to dissect them and produce something sounding recognisable whilst being able to react with more immediacy to the audience and their mood.

This potential glitch style gave rise to his first small range of purpose built applications that were released freely. These allowed the user to take a folder of audio files or a compact disc and have the tracks algorithmically rearranged broadly in the style of Leafcutter. Basically the tracks are loaded into buffers that are then traversed with a certain amount of chance controlling the speed, direction and loop lengths of audio snippets. The results are then piped through a series of time based effects chains.

The ideas behind this experiment were refined in the moderately successful Forrester<sup>16</sup> application. Here similar principles are at work, but are extended through the use of the visual metaphor of trees in a forest. A folder of audio files is loaded in and a random selection of audio from these files is loaded into a series of buffers. A simple button is pressed to create an approximation of a top down view of a forest of trees and play is pressed. It is possible to define some parameters, how many trees, how densely packed they are for instance, but the final process is essentially random within these parameters. An avatar meanders through this forest with its position affecting a number of the processes at work (delay time, reverb depth, granular parameters etc.). The movement can be guided by clicking and or dragging around the forest or the sound can be left to follow its own path.

---

<sup>16</sup> [http://leafcutterjohn.com/?page\\_id=14](http://leafcutterjohn.com/?page_id=14)

Group work takes elements from these systems and allows Leafcutter to both create sound in an ensemble context and also to treat the sounds of the other members. This can vary from folk duos (2009) on to experimental free jazz ensembles with the treatments extending from ambient washes of sound through to spiky click and cut glitch orientated audio.

Similar systems are used in his installation work, such as SoundTrapII<sup>17</sup>, in which contact microphones are placed around a specific space and used as acoustic input to a system of granular tools, delays, reverb and spatialisation before being released back into the space. In all these systems random, chaotic and chance elements are present to a greater or lesser degree with the installation systems utilising a lot of the work from the automated software applications to produce a generative variant of the Leafcutter sound.

### **Alex McLean**

Alex McLean is one of a quite new breed of laptop performers, taking part in the rather macho pursuit of live coding. Whilst some of the systems I have investigated so far have allowed the composer to develop their own compositional and digital signal processing systems it is a relatively recent phenomenon for the artists to do this development live (Historical Performances - Toplap), in front of an audience. This is pretty much improvisation with code, on the edge.

---

<sup>17</sup> [http://leafcutterjohn.com/?page\\_id=35](http://leafcutterjohn.com/?page_id=35)

McLean is a member of the vanguard of this off shoot of laptop performers, himself being a member of both Slub, a trio of live coders, and half of Silicone Bake, a spam-pop band and a regular performer and educator in this field.

In terms of the tools used for live coding, McLean is an (McLean, A, 2004) advocate of open source tools and primarily runs software based on the Linux operating system. A long time user of Ubuntu, often seen as the friendly face of Linux, he switched to a specialist audio visual distribution know as Pure Dyne as Ubuntu became more consumer oriented and problems with audio drivers began to manifest themselves. This particular distribution comes as standard with a range of audio and visual environments (SuperCollider, CSound, Processing etc.) and can be installed as a live system from a CD/DVD or USB memory stick making it a useful variant for workshops and performances if a machine should malfunction, as well as a dedicated operating system install.

McLean's early experiments were based in the Perl language, a general purpose interpreted programming language with over twenty years of development. There are now a great number of variations in the tools that live coders use with some preferring the already established languages with a music or audio slant such as SuperCollider or Pure Data, but more and more coders are moving towards much more general purpose computer

languages and coupling those with audio and MIDI libraries to achieve their goals. These early Perl based pieces, whilst live, were primarily based around a collection of pre built scripts that could be launched sequentially and combined to produce musical phrases and rhythms. (McLean, A, 2004) Whilst this obviously showed the direction of the artists it wasn't quite yet live coding.

The performance system continued to evolve, and in fact still does and a landmark of the system becoming truly live coding was the artist's switch to the Haskell programming language. Although a general purpose computer language some of its features were obviously a draw to McLean and it became the primary front-end for his live coding exploits communicating with SuperCollider for the audio generation under the hood via Open Sound Control (OSC). This was necessary as most general purpose programming languages that may have any number of features relevant to music creation and the pattern elements required of live coding often have very convoluted methods of accessing libraries for creating the actual audio part of the process, whereas SuperCollider was designed from the ground up for just this purpose with the audio generation and programming elements separated. In this way any programming language can make use of the audio engine by sending properly formatted OSC messages to it.

In terms of performance it has become commonplace for the artists to project their screen in the venue, the audience can then follow along with



the action. There continues to be some uncertainty as to whether this is a good course of action. With it not there you are really just watching a performer watch a laptop, which is probably not all that engaging. Some members of the audience may be proficient in code and so therefore enjoy watching the program unfold and listening to the effective sonification of the text. For others who don't understand the code it is proof if any were needed that the artist is actually doing something other than perhaps simply checking their email!

McLean's approach continues to be refined and more recently he has developed a purpose-built live coding environment using Haskell<sup>18</sup> called Tidal (McLean, A, 2010a) that allows the performer more fine grained control and a tighter focus on the purely musical elements of any code. This is still a command line based application, as are many of the live coding tools, but the text used is more accessible and readily understood by musicians and potentially the audience as well as being quicker and more efficient to develop in a live context for the artist.

This continuing development of the tools has also seen experimentation with a visual overlay for the Tidal<sup>19</sup> system called Texture (often simply Text). This environment calls to mind the visual object and connection paradigm used in Pure Data and Max/MSP although it is approached from a

---

<sup>18</sup> <http://www.haskell.org/haskellwiki/Haskell>

<sup>19</sup> <http://yaxu.org/tidal/>

slightly different angle, in that the proximity of the elements used takes on a significance within the system.

In the spirit of the open source community all of the tools that McLean has developed are released for other artists and live coders to use freely in their own work.

### **Dan Stowell**

Dan Stowell is another member of the rapidly growing live coding community. Another user of primarily open sourced tools, the main environment of choice for his live coding performances is SuperCollider. SuperCollider was itself developed from a closed source application by one individual who then decided to give it away for free when he was no longer able to maintain its development effectively and open sourced the code to allow others to grow and evolve the system. This has really seen the growth and adoption of SuperCollider as an environment for composition and performance amongst composers and sonic artists. It has also become a well known tool in live coding circles.

Stowell's approach to live coding is subtly different from that of the previous artists discussed. The coding is there, the projection of his laptop screen is there but the actual performance sees him providing audio material to his systems through human beat boxing. This vocal input is used as loops and single hits in generative sequences that are guided by the live code. Whilst in

some ways this difference is quite subtle in the performance I witnessed this had much more impact than looking at just the code. I found this simple act much more engaging for the audience, much more of a performance from a personal point of view.

The use of beat boxing has another impact on the performance alongside the theatrical. The use of vocal sounds as input to the system gives an altogether different sound to the combination of textures and tones one often hears in live coding shows. The organisation of the sounds remains similar though. A lot of the environments used seem to encourage or facilitate a very sequenced sound. There is nothing inherently wrong with this of course and it is merely an observation that much of the live coding I have personally witnessed has been along the lines of minimal 'techno'.

One of the main draws of Stowell to SuperCollider is the extensibility offered by the system and indeed he has written a chapter of the recent Supercollider book on doing just that and extending the base SuperCollider by developing your own UGens. (Stowell, D, 2011) This facility allows the artist, should they desire and be capable of doing, to create modules for the main program. These could be a replication of a drum machine for instance, or just an element of that, say a crash cymbal. Smaller units are probably going to be more effective here as they can more easily be combined. It would be difficult for instance get at those individual drum sounds if they

had been hard coded into a full drum machine, whereas any number of sequencing elements could address the individual sounds much more easily.

### **Scott Hewitt**

Scott Hewitt is the last of the live coders I will be discussing His general approach is similar; however there a couple of variations that make Hewitt an interesting case study.

Whilst a great many programming languages have been explored by live coders, some general purpose and some with specialisms leaning towards live coding, Hewitt is making use of a system quite recently developed and with a particular focus in the modification of code on the fly making it an ideal contender for live coding work.

Chuck<sup>20</sup> was developed (and is still developing) at Princeton as a new audio programming language for real time composition and performance. Hewitt arrived at Chuck after working for some years with Max/MSP and now uses it as the primary tool for live coding performances. (Hewitt, S, 2011) For Hewitt the switch from the graphical patching paradigm of Max/MSP to a purely code based environment offered a much clearer programming method. Whilst patching allows for rapid prototyping and it is often much quicker to actually build something usable, there reaches a point where

---

<sup>20</sup> <http://chuck.cs.princeton.edu/>

systems can become unwieldy and difficult to extend or develop further without a great deal of work, whereas a text based environment such as Chuck can make this much easier. For example making a Max/MSP patch function polyphonically can be quite cumbersome. Within Chuck it is greatly simplified by simply calling a function repeatedly.

Hewitt is also part of another laptop performance development, the laptop orchestra. The collective performance of laptop artists is gaining some momentum in certain circles. A couple of the more public ensembles are the Stanford Laptop Orchestra (SLORK)<sup>21</sup> and the Princeton Laptop Orchestra (PLORK)<sup>22</sup> and for these institutions it is proving a useful teaching aid as well as a performance outlet. (Wang, G et al., 2008) Chuck was developed alongside the PLORK ensemble and is used as the main software platform. Possibly because of the finances they currently enjoy, some of these American academic based orchestras seem to have access to better resources and are often quite prescriptive about the hardware and software tools to be used, with some developing and manufacturing their own speaker systems<sup>23</sup> to allow the laptop to radiate sound as a traditional musical instrument might. How necessary this is could be questioned (are they trying to recreate some of the old with new technology?) however it is one of their aesthetic goals to achieve this. Here in the UK, in my experience, these ensembles tend to be a little bit more DIY and 'Heath Robinson' in their approach, although no less committed to what they are doing. The

---

<sup>21</sup> <http://slork.stanford.edu/>

<sup>22</sup> <http://plork.cs.princeton.edu/>

<sup>23</sup> <http://silvertone.princeton.edu/~skot/plork/delorean/>

Huddersfield Laptop Orchestra<sup>24</sup> (established in 2008 and currently directed by Hewitt) is deliberately lo-fi in contrast. The members are not tied to any specific software or hardware (although Hewitt is somewhat of an ambassador for Chuck) and performers use whatever they feel can best represent their goals.

These ensembles vary in size and whilst works created for them are not necessarily coded live there is often an element of real time interaction with the code that is creating the eventual audio output through the use of various physical and non physical controllers that are available for mapping purposes.

### **Jeff Kaiser**

Jeff Kaiser is a composer and improviser who uses technology to extend, manipulate and treat the instrumental sound of the trumpet throughout his work. For quite some time the system of sound modifiers used was based completely on hardware devices. A number of rack units and guitar effect type boxes comprising delays, distortions, ring modulation etc. were linked through a traditional mixer and where possible tweaked in real time to effect the trumpet tones. Whilst this setup was effective and offered a great deal of sonic manipulation possibilities, it was cumbersome to set up and meant that touring was as much an exercise in physical fitness as it was artistic expression. After seeing other performer's setups Kaiser decided to

---

<sup>24</sup> <http://helo.ablelemon.co.uk/doku.php>

move away from dedicated hardware and towards the continued flexibility of a software based system. Max/MSP became the tool of choice for this, primarily because friends and performance partners were already using this system and so a source of education was readily on tap.

The evolution of the system began by using Max/MSP primarily as a host for a range of VST plugins that replaced or emulated their hardware counterparts. The hardware components were essentially now just the laptop and a soundcard with the signal chain being simply trumpet into microphone into the computer. In one fell swoop a number of pedals, effects, patch cords and power supplies had been replaced. However whilst the simplicity of setup and operation had been greatly simplified there were now the complexities of software development to contend with.

As the artist became more proficient with the software, the development of a more individual performance instrument began. The VSTs that had replaced the hardware units were now in turn replaced by dedicated modules that offered increased flexibility and control. The primary element of Kaiser's performance system is the Kaiser Looper. Four recordable buffers take in audio and allow for the control of speed and direction during playback with the facility for constrained random elements to scrub through the loop. This scrubbing facilitates the introduction of a rhythmic element to act as a counterpoint to a texture of ambient layers that can easily be built up using the looper system. Kaiser's live work is primarily based around

improvisation and this looper system facilitates the build up of layers in combination and contrast quite straightforwardly.

In terms of hardware control Kaiser has been a long time advocate of using cheap off-the-shelf MIDI or USB controllers wherever possible, rationalising that in his view equipment like this needs to be easily replaceable when on tour. To further promote the 'small is beautiful' touring aesthetic Kaiser also uses an interesting doubling up idea for implementing a continuous foot controller through the use of a spare input output pair on a soundcard.

(Kaiser, J, 2007)

Like many of these tools the system is in a constant state of evolution.

Although this slows as the performer settles on a working method, there will always be new technological development that might be worthy of investigation. In terms of Kaiser's system two of these are the introduction of Max for Live and the Macmillan SoftStep controller. M4L has meant that the Kaiser Looper system can be integrated into Ableton Live, which opens up the doors for the easy inclusion of more traditional sequencing techniques and a very flexible software mixer. The SoftStep looks very much like any other foot controller at first glance, however as well as acting as simple switches the buttons are both pressure and direction sensitive. This means that any button can output continuous control changes (up to five at a time) as well as simple digital selections. The MacMillan company who produce the unit also provide a Max development kit. As a standalone unit it



outputs serial information over USB which can be converted to the MIDI control protocol through a secondary unit. The Max development kit though provides a Max object which reads the serial information directly and allows the programmer to convert the raw serial information to MIDI, OSC or directly control other audio visual Max objects offering a very flexible system.

### **Brian Crabtree**

Brian Crabtree composes and performs under the alias of Tehn. His work is primarily directed towards the glitchy end of ambient electronica consisting essentially of minimal techno and clicks and cuts using loops and micro loops. He uses exclusively a number of self-designed and built hardware and software systems for both the generation and real time control of sound. The integration of the hardware and software elements is very tight and whilst they can be used separately they often make much more sense and in some cases only function fully when in combination.

Again Max/MSP is the main development environment for the software side of things. The artist has developed a number of patches dealing mainly with sequencing and sample mangling (flin, 64step). However for live performance it is the MLR<sup>25</sup> system that appears to offer the most flexibility and be the go to application. A number of buffers are filled with either precomposed looped material or from live input from a variety of

---

<sup>25</sup> <http://docs.monome.org/doku.php?id=app:mlr>

instrumental sources. This selection of loops can then be chopped, sliced and recombined in real time in a very flexible manner. The loops can be sliced against a rhythmic grid to retain their relative timing or in a more ad-hoc fashion. This deceptively simple system is capable of producing obviously rhythmic grid based work, more glitch pieces right through to less obviously rhythmic almost ambient textures.

Whilst the software side of the performance system is flexible and useful the hardware side is of particular interest in that whilst it was developed by a single individual for his own particular working method it has proved to be quite popular amongst a range of artists and has become a commercial product in itself with Crabtree utilising an interesting business model.

The Monome is a very minimal eight by eight (there are now also eight by sixteen and sixteen by sixteen variants, see Fig 1) grid of buttons and LEDs that out of the box is unable to do anything and needs a software system with which to integrate. By default the basic driver and communication software provides output in the OSC protocol, however this can be quite easily converted into MIDI information should it be needed. The buttons and LEDs are decoupled, meaning that a button press does not necessarily mean that current will flow through an LED, and communication is two way to allow for flexibility in the design of user interfaces. One area for development that makes itself quite obvious is that the grid lends itself well to step sequencing and there are a number of patches available, from Tehn

and other developers, that highlight this fact. The Roland x0x step sequencing paradigm is used most often, as this has become somewhat of a standard, particularly in rhythmic work, since the introduction of the 808 and 909 drum machines.

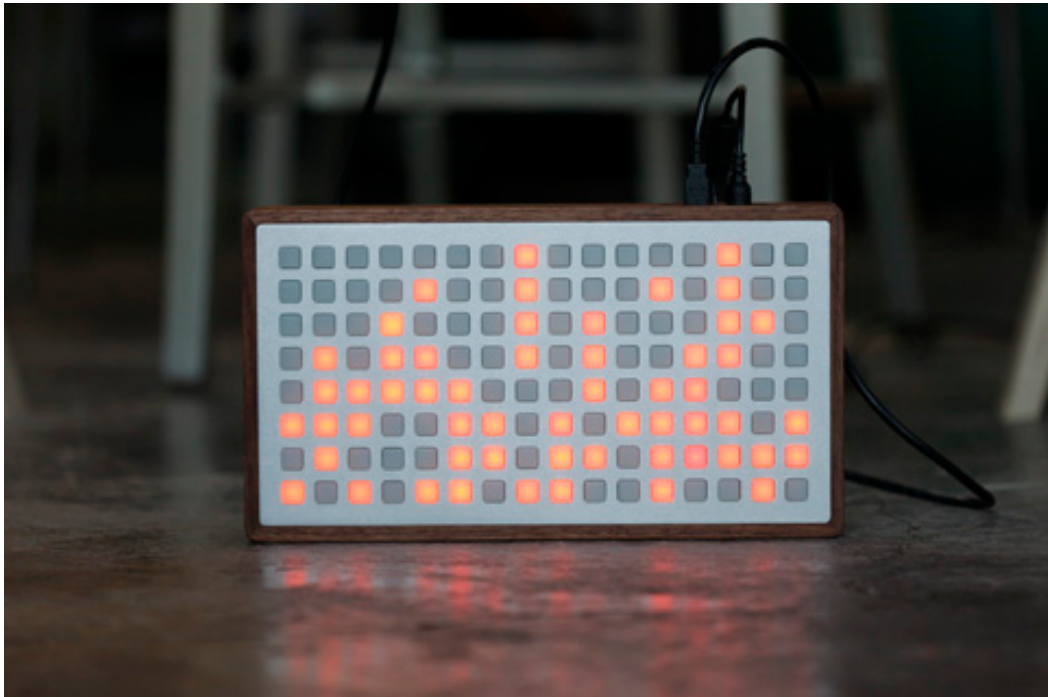


Figure 1 Monome 128

Further to the grid, the Monome also provides a continuous control output through a tilt sensor, which can be mapped through OSC or MIDI to further manipulate audio.

The grid of lights gives a visual focus to the live work of the artist often allowing for the laptop to be tucked away. In the pieces there is often a clear link between button presses and sonic results for the audience to catch. As an approach to live performance, this can be subtle or fundamental and is

something that a number of Monome users do. Some audience members have a perception that a laptop on stage somehow means what they are seeing and hearing might not actually be live in the sense that it is being performed.

Furthering the minimal simplistic user interface of the Monome, Crabtree has more recently developed the Arc. Whilst obviously designed with the same aesthetic in mind, it both complements and contrasts with the Monome's strict grid layout by offering either two or four rotary encoders which allow for the fine grained and continuous control of mapped parameters. Again the unit does nothing out of the box and its function depends on the development of applications that make use of the continuous rotary encoders, which offer rotation, click or rotating while clicked. One patch that really takes advantage of this control is the tml (Tehn micro looper<sup>26</sup>), which again is based on a number of audio buffers. This time the focus is live input, which can then be manipulated via the Monome and Arc in combination. Essentially, as the name suggests, small fragments of recorded material are looped with their speed, direction and loop start and end points being altered in real time via the Arc.

Whilst initially developed by Crabtree and his partner for a very specific set of personal performance goals, both the Monome and the Arc have been underground hits being used by a number of other artists across a range of

---

<sup>26</sup> <http://vimeo.com/19039646>

electronic music genres. The business model used by the Monome company is interesting in that whilst both units are sold as finished products, with green ideals and sustainability being high in the agenda through the local sourcing of parts and materials etc., the Monome can also be purchased as a kit for self assembly. This in itself is nothing particularly new, however the schematics for both units are freely provided under an open source creative commons licence on the website so that anyone with the time, money, knowledge and inclination can produce their own fully compatible Monome or Arc clone. All of the software applications and patches used by the artist are also provided for free, although they do require a full Max/MSP license to be edited.

### **Gregory Taylor**

Gregory Taylor has been a composer and performer of electroacoustic music, both as a soloist and ensemble player (PGT<sup>27</sup>) since the 80's. As an employee of Cycling74, Max was the obvious tool of choice for his electroacoustic composition and performance works. The main tool for live laptop performance work is the application Radial (now discontinued) from Cycling74, a system which Taylor shared in some development duties. (Murphy, B, 2011) Originally developed by jhno (John Eichenseer) in Max/MSP, it is a stand-alone application that allows for the recording, looping, combining, processing and mixing of audio streams. Initially these streams were pre-recorded audio loops that could be manipulated in terms

---

<sup>27</sup> <http://www.rtqe.net/pendergartontaylor.html>

of speed and direction, but also be chopped, sliced and looped on the fly using an interesting circular payback visual interface, hence the name. The extensive mapping of MIDI control is also available. Taylor's involvement in the iterative development of Radial was initially facilitating the ability to utilise VST plugins within the system itself, adding to the already present mixing matrix and the built in filtering and delay elements, and secondly to allow live audio to be piped into the looping system. These two advances moved Radial on from a relatively simplistic loop player system to a more fully functioning real-time composition and performance system.

For quite some time Radial was the only tool used in live performance with Taylor often taking to the stage without any pre-recorded material at all, just the laptop, the software and some noise making devices. Often this was likely to be some sort of instrumental or vocal texture provided by a second performer. Pieces then evolved through improvisation with the laptop and its performer being more fully integrated into the ensemble.

The setup varies slightly depending on the nature of the performance. After many years using the tool certain aspects have proved to be more or less useful in either a solo or ensemble context. An ensemble performance might utilise fewer looper units but make use of a more complex effects chain, whereas a solo gig might benefit from more looper units being combined with less real time control of the effects to take care of. These individual setups can be saved as pre-sets and then recalled and edited easily.

In terms of the real time control of the system, Taylor's approach is deliberately minimal. The only external controller used is the MotorMix<sup>28</sup>, a bank of motorised faders that can be mapped to any element of the system. The fact that the controls can be mapped and remapped according to the needs of the show means that this is an effective and small footprint controller for use in combination with the software. All other control is from the laptop's keyboard and track pad. The keyboard provides a useful set of digital switches that can be used for the simple triggering of loops. The track pad on the other hand provides a continuous xy controller that can be mapped to effect modules and used very much like a Kaoss Pad. The utilisation of the standard laptop interface is perhaps a small point, but one that potentially helps to redefine the computer as a self contained instrument in some ways.

Whilst Radial is really very much Max/MSP based underneath it is a polished front end for a very specific set of tools. Taylor has made much use of this over the years but more recently has returned to developing in raw Max/MSP as it were for some more focussed work. He is responsible for the Buffer Shuffler effect in the recent Max for Live collaboration between Cycling74 and Ableton and has developed some spectral tools directly in Max/MSP for use in both composition and performance work, for instance the release 'Two Maps of Danaraja'. (Taylor, G, 2008)

---

<sup>28</sup> <http://www.cmlabs.net/motormix.html>

## **Four Tet**

Probably the most commercial artists in this discussion, Kieran Hebden aka Four Tet is a producer of music that whilst it is obviously related to dance, often has more experimental instrumentation and rhythmic leanings.

Essentially the performances are club based affairs and split between DJ sets and live real-time interpretation of his recorded output. Many artists of his ilk would probably just play back pre-recorded files and concentrate on looking the part whereas Hebden is more concerned with taking chances with the live shows, treating them very much as DJ sets in that the performance can follow the crowd momentum. If a track or section of a track is going down well then it can be extended and developed on the fly. Interestingly again for an artist in this genre Hebden doesn't perform with visuals. Many, if not all, club based performances feature either a VJ performing or some kind of audio reactive visual as an accompaniment to the sonic performer. Hebden sees this as a distraction though and prefers the audience to see him working. I can see his point here: in some respects it is possible that additional visual elements in a live show may detract from the audio component, however even though the performance aesthetic of laptop performing is obviously evolving currently we are still watching a person with a computer.

In terms of software and hardware tools, Hebden's live setup is based around two laptops (although these are replaced by two decks for DJ



performances) a DJ mixer and a BOSS Dr Sample sampling unit. There is nothing particularly esoteric about the individual units and everything is deliberately kept relatively cheap and replaceable just in case of any problems whilst touring. The computers used are never cutting edge and often the software version being used is not the latest release, Hebden preferring to get to know an application intimately before moving onto the newer and better features - something that many of us are guilty of and something that developers to some extent need.

On the software side Hebden has been a long time user of the Audiomulch<sup>29</sup> application. (Inglis, S, 2003) Audiomulch is again a commercial composition tool and allows the user to develop systems along similar lines to the patching functionality of Max/MSP. Audiomulch differs however in that the elements that can be patched together are of a higher level. For example in Max/MSP to play an audio loop one would need to combine a number of individual lower level elements, whereas in Audiomulch one would just drag out an instance of a loop player. In essence this means that whilst the user loses access to some low level functionality, the process of developing custom systems is simplified.

The software is used in the composition stage as an improvisational tool before the resulting audio is arranged and mixed in a more traditional digital audio workstation environment (Pro Tools), but also in the live

---

<sup>29</sup> <http://www.audiomulch.com/>

performance elements of his work. Traditionally one of the laptops has run Audiomulch live to cover all of the rhythmically oriented loops whilst the other ran Cool Edit to trigger one shot samples and ambiences. This is evolving however as Ableton Live has been introduced into both the composition and performance setups. The audio is then piped through a simple DJ mixer with individual channel faders, a cross fader and simple filtering and EQ. A feed from this mixer is sent to the Dr Sample unit for the live capture of loops. This is something that could easily be done in the software being used, but as Hebden likes to stick to what he knows until compelled to move on, the hardware sampler with its simple interface is the most straightforward way to sample loops on the fly.

### **Beardyman**

Beardyman is a recent convert to the live use of the laptop, in the past preferring to use the dedicated and well proven hardware sampling and effects route. When I first came across him he was working with Sebastian Lexer (mentioned elsewhere) in the development of a Max/MSP based system to replace the hardware-based setup discussed later. However after a couple of years this has not materialised and it looks as though the setup is evolving in a different direction.

Beardyman is a very theatrical performer whose act is based around beat boxing. As an award winning beat boxer (UK beat boxing champion in 2006 and 2007) he was looking for a way to extend the range of what might be

achieved as a performer. The relatively simple procedure of sound on sound looping was a straightforward way of exploring a range of textural approaches and the Korg Kaoss Pad sampling and effects unit and the BOSS range of pedal loopers offered the promise of a simple but effective performance setup.

The Kaoss Pad offers some straightforward sampling facilities along with a range of timbre and time based effects algorithms all accessible and controllable in a real time context through the intuitive xy pad. This allows the performer to effect their vocalisations in a very immediate fashion. The BOSS range of loopers is again very simple in operation, stomp to start and stomp to stop and begin playback of the loops. This kind of immediacy is extremely useful for the solo performer in terms of keeping the momentum of a piece going.

Although this entire setup could be recreated in software, and as we shall see it is in the process of being so, there is a lot to be said for this hardware-based approach. As above the immediacy can be extremely important to a performer of rhythmically based material, missing a beat at any point can really upset the flow of a show and it can be difficult to regain the momentum. Reliability can be an important factor.

Some feel that the timing of more dedicated machines is better. I have never really experienced this myself and it could really be psychological and part

of the studio voodoo that abounds; either way despite modern computers being able to do all that they do there is still a market for much older machines such as the Atari ST and the AKAI MPC range of hardware samplers and sequencers that people swear by in terms of rock solid timing.

The user interface provided by a piece of hardware or software can really have an impact on the approach of a performer and the efficiency of their working methods. This can manifest itself both in terms of safety in a live environment, knowing what you use and using what you know allows the performer to concentrate on the performance rather than worrying about operation of machinery and also in terms of flexibility of use and being able to react to elements of the performance more directly.

Whilst underneath both the units in question here are really just computers, the fact that they are dedicated to just one task inevitably means that the risk of glitches or even crashes is something that doesn't really need to be thought about whereas when using a more general purpose computer for audio tasks this is a reality that needs to be faced and dealt with both in terms of preparation and having a backup plan.

Finally it is difficult to extend the capabilities of dedicated units in any meaningful way. They do what they do really. This ultimately means that the user is compelled to concentrate on the feature that makes the unit useful.

This can have a useful side effect in that quite often these kinds of constraints can force creativity. (Magnusson, T, 2010a)

Having said all of that regarding the positives and negatives of the hardware approach, Beardyman recently put out a call via his Facebook and Twitter pages for a C++ software developer to work on some custom software. It would appear that although the Max/MSP variant of the software replication of his hardware setup didn't come to anything it did serve as a useful prototyping process and point the way for future developments. Currently Beardyman is working with both Sugarbytes software and DMG audio on the development of a live looping, processing and mixing system, which will ultimately be headed for a commercial release. As an artist you have to be in a certain position commercially speaking to make this kind of development viable, it is really an extension of the artist developing their own systems in an environment such as Max/MSP and as artists become more familiar with code it highlights the direction in which electroacoustic composers and performers might move.

### **Christopher Willits**

Willits is a multimedia artist who combines elements of ambient, drone, looping and sequencing with visuals in his performances. Willits studied with Pauline Oliveros at Mills College and so much of his thinking is along the lines of the computer as a tool for extending some more traditional

instrumental resources. In this case the sound source is often Willits' own guitar playing.

The physical setup<sup>30</sup> begins with an electric guitar, most usually a standard Fender Stratocaster modified with custom pickups and a Roland guitar to MIDI pickup. This allows both audio and MIDI information to be used as either source or control elements throughout the rest of the processing system. The guitar audio output is fed into a Line 6 Pod modelling preamp, which is in turn patched into the soundcard. The output from the MIDI pickup is sent directly to the MIDI inputs of the soundcard.

Real time control of the system is mapped through a range of hardware controllers including a Behringer FCB1010 foot controller, Livid Instruments Block grid controller, and MAudio Trigger Finger, a Doepfer Fader Box and numerous other items. The choice depends on where and when the performance is and on whether it is a solo or ensemble show.

The software side of the system originally consisted of a range of Max/MSP patches based around the real time manipulation of buffers and delay lines. A common thread of Willits' work is his folding technique, in which textures are built up by scrubbing through multiple audio buffers at different speeds. The elements of this technique can be controlled manually from the audio

---

<sup>30</sup> <http://cwillits.wordpress.com/2009/10/01/willits-guitar-setup/>

and MIDI inputs to the system or generatively seeded from other internal or external sources.

More recently however the system has evolved to take advantage of the Max for Live bridge between Max and Ableton Live. This has allowed for further flexibility in a live performance context and simplified the use of MIDI mapping and any timeline based processes.

Often Willits' live shows incorporate the projection of visual elements, as he is also a visual artist. Interestingly these are also far from static and various elements of the visual patches are controlled from the audio or MIDI parts of the musical side of the performance setup.

### **Hans Tammen**

Tammen's work is based primarily around free improvisation, using the guitar more as a noise toolbox than an instrumental resource very much in the vein of Keith Rowe. Similar to Rowe's work, the guitar is played flat on the table top and the strings are excited into motion using any number of objects as well as the performer's fingers. The guitar pickups are also often used as crude lo-fi microphones to introduce other sonic sources into the performance system.

The experimental guitar sound source is physically treated in a number of ways and a number of extended playing techniques are utilised in the

production of source sounds. In addition to the more traditional guitar technique, various mallets, crocodile clips, mobile phones and electronic gadgets are used to provide impetus to the system. The E-bow is also used to provide sustaining grounds and beds where necessary.

The signal path travels from guitar, through various effect pedal type units before arriving at the computer's soundcard input and on into the software part of the system. Here again Max/MSP has been chosen as the tool with which to process the incoming audio. In many ways Tammen's 'Endangered Guitar' system as it is known, is conceptually similar to Lexer's Piano+ system discussed earlier. As well as the performer creating the original sound sources themselves, the system uses pitch recognition and amplitude tracking for the mapping and control of effects and processes throughout the system and ultimately the audio output. This provides for an interesting feedback loop of generative processes through the system.

A buffer is constantly being filled to provide material for use alongside the real time guitar input. The system records and then analyses pitch and velocity to control and change the settings of various modules throughout the system. In this way an element of randomness, or even chaos, can be introduced into improvisational performances to which the player can choose to react in a number of ways either fighting to regain control or going with the flow on a sonic journey.



This randomness or fuzziness is needed in an improvisational tool to provide constant surprises for the performer and ultimately the audience.

## **Conclusions**

The artists discussed in these case studies range from the commercially successful to the niche within a niche. In many ways the sound worlds they create have a lot in common all making use of drones, glitch elements and sequenced rhythmic elements to a greater or lesser extent. However one obvious aesthetic difference is the increased use of regular beats as the music reaches towards the commercial end. It would appear that in some ways the simple inclusion of some kind of obviously repetitive rhythmic element often makes the most challenging music more accessible.

Until quite recently the choices for selecting an environment for laptop performance were simply between something that represented the virtual studio paradigm or developing your own environment from patches or code. The patching paradigm often seemed to appeal to artists that wanted to dip a toe in the water whilst the coding applications offered performers with a technical bias a way to further extend their creativity. As art and technology continue to blend I don't think that this is the case anymore and tools are becoming more accessible for all.

Ableton Live is a good example of this kind of development. The application found its niche in offering a tool that allowed for the live reinterpretation of audio material in a way that the traditional DAW could not. The continued development of Live and the company's relationship with Cycling74 gave birth to the Max for Live crossover application. With this grouping the artists can combine elements from both ends of the spectrum and go as deeply as they like into the technology on a per project basis.

None of the tools discussed presents any kind of stylistic bias as such and choices are being made on a very individual basis. Some artists will always prefer the immediacy of a commercial product and the support that it offers, whereas others will feel the need to get under the hood and have control over every aspect of the development of their unique performance systems. In addition for some artists the choice to go open source is almost a lifestyle choice and necessarily dictates the tools that they have at their disposal.

## Part 2 – The instrument

## **What am I trying to achieve?**

I am exploring the development of the laptop as an instrument, or at least the basis of a combined hardware and software instrument. There are some initial questions that this research requires. Which element(s) are actually the instrument, is it the hardware, the software or the combination? Can it really ever be the same as a traditional instrument? How and what is to be controlled? How are choices made regarding the mapping of physical controls to digital processes within the system?

One of my goals in developing an instrument is to take to the stage using nothing but the designed system in performance, no premade loops or pre-recorded audio files and still sound like me. To what extent is it possible to use technology live and still have an obvious identity as an artist and composer? I have experienced this as an instrumentalist with the guitar and also as a noisemaker with found object created instruments. Free improvisation and improvisation within a framework requires some kind of vocabulary of rhythmic, melodic and harmonic objects and it could take some time to develop this using technology.

This instrument requires the development of a tool set that allows for elements of improvisation in both performance and composition, real-time composition perhaps. My work often involves improvisation at various stages: most obviously source sounds and recordings are experimented with in various digital signal processing systems, stretching, warping, mangling

sometimes with a relatively clear goal in mind, other times not. These sound objects are then placed in time or collaged in a digital audio workstation, again often in an improvisatory manner. An arrangement will be tried, if it works it stays, if not then another combination is experimented with and this continues until a decision is made regarding the completion of a piece.

Whilst there is no overarching working method or strategy across the entirety of my work, it is this notion of experimentation and improvisation that permeates everything to some extent. My aim is to capture some of this and combine the acts of composition and performance using this underlying improvisational concept. The vocabulary for this will develop as a combination of fragments of my work being repurposed in a live context and the remodeling of some of the DSP systems I use into a real-time controllable instrument.

Currently my creative output is based around a kind of minimal ambience, encapsulating controlled randomness and chaotic elements whilst exploring temporal and timbral components through stasis, texture and process. I am aiming to create a toolset that will allow me to use some of my current compositional strategies in a real-time performative context. The system needs to some extent to be modular as I don't imagine that necessarily all elements will be used at the same time, but it might be useful to be able to pipe data from one to another. I do not wish to preclude other compositional ideas though and am certain that the system will evolve over time to

encompass newer strategies. Using Max/MSP as the main development environment, as discussed later, will allow me to have a core element that then facilitates extension both through Max/MSP itself and the use of external objects, but also through a number of other environments, for example SuperCollider, Chuck, CSound etc. This will allow me to develop elements that play to the strengths of these text-based languages and incorporate them relatively straightforwardly into the overall system.

(Garton, B, 2007)

### **Why?**

To ultimately be more involved in the performance. I have experience of fixed media pieces in the electroacoustic arena and would like to combine and extend this to include my experiences from other genres (as a rock guitar and vocal instrumentalist and free improviser using again the guitar - but also anything else that might be at hand).

There is an element of reactivity and interactivity with the audience that is sometimes absent in a tape music concert. The use of space and the malleability of audio material is perhaps something that is taken for granted as an improvising instrumentalist and I feel that this kind of flexibility is lacking from my electronic work. I am aiming to transfer the immediacy that I feel with a guitar in my hands into an electronic context.

Whilst there is nothing inherently wrong with the long established tradition of tape based music and I have myself enjoyed many 'playback' concerts, I think that technology has evolved to a point where work can be reimaged in a live setting and I personally feel that this offers a great deal of potential for audience interaction, in that their reactions can both directly and indirectly influence the course of the work.

Outside the walled garden of electroacoustic music, often the audience require or desire some element of theatricality to a performance as well: they go to 'see' a band or artist. This inevitably means some kind of visual element to the performance and can run the gamut from the sensory overload of an arena dance act such as the Chemical Brothers and their wall of LCD video screens screaming out eye candy in sync with the four on the floor beats to the sensory deprivation of Francesco Lopez and Helena Gough who both often perform in as near complete darkness as can be achieved to both focus the listener's attention and no doubt introduce a performative element beyond the music and sound itself to draw in the audience. There are of course any number of performance tricks and tags on this sliding scale, but I think that being able to take some control over the audiovisual representation of my work in a real time context is where I am currently aiming to be.

## How?

Develop a series of software systems and modules with associated hardware controllers. There are a number of development environments available both as commercial items and open source frameworks.

I am aiming to make these systems as modular as possible so that they can be reconfigured and recombined easily allowing the overall instrument to evolve without requiring rewrites and rebuilds too often and hopefully without suffering from feature creep and becoming too bloated. Also the separation of technologies for the Human Computer Interaction (HCI) element and the sound producing part should future proof the instrument to a certain degree in that as new technologies become available or the current ones are incrementally improved, again the system can be updated piecemeal as opposed to starting again from scratch in another programming environment.

For the software side of things I settled on Max/MSP from Cycling74 for a number of reasons. Firstly I already owned it and am quite comfortable with it as a development environment. The flexibility it offers, coupled with the graphical programming environment simply just sits better with my working method and thought processes than some of the other text based systems. I did explore SuperCollider, CSound and Chuck, however whilst I was able to get some interesting sounds out of them I really didn't feel that for me this was the way to go for the development of an entire instrument.



The fact that Max can be easily extended by the compilation of new modules also meant that, should I need it, I could include elements of the other programming languages within my patches. Each language and programming environment offers its own set of pros and cons. Each will force the user to think differently conceptually and therefore hopefully develop new compositional ideas that staying with one specific language or environment would not. Because of this I will no doubt continue to experiment with the other possibilities.

I found Pure Data interesting and am drawn to the open source initiative for a number of reasons and I can see that at some point I will probably move in that direction, however for now the more polished application interface of Max/MSP meant that the current system will be developed in that environment. The Max/MSP presentation mode is essential when creating something complex but useable and Pure Data doesn't currently have this flexibility.

In terms of control I have explored a number of possibilities. Initially I used a small USB MIDI controller with a number of assorted keys, wheels and buttons. This was perfectly functional and straightforward to map within Max/MSP, but I found that hit the limits of what I could control at any given moment pretty quickly. This was then coupled with a Novation Launchpad for a sixty four button grid of switches. This effectively added to the flexibility of the system but added significantly to the physical footprint,

which is something that I wanted to avoid. Eventually I came to the Apple iPad and the TouchOSC<sup>31</sup> app. This combination allowed me to have everything I needed in terms of control in a very compact form. Any layouts can be developed as a series of pages utilising dials, faders, grids etc. this setup is not perfect: for example I miss the tactile feedback of tweaking a control when just smearing my finger across a piece of glass, however the flexibility available within this form factor far outweigh the negative points. The Jazzmutant Lemur since being discontinued has become available as an iPad app and this is definitely an area that I will be exploring.

To keep the design and use of the instrument as flexible as possible whilst trying to avoid the bloat and feature creep that is potentially a problem with the development of a system such as this I am visualising the instrument as a series of abstracted layers that can be developed and extended separately without requiring the complete redesign of the entire instrument. These layers essentially consist of sound source (input, sound files, instruments, piezo mics, vocals, environmental etc.), mixing (panning, diffusion, FX, dynamics) and control (iPad, OSC etc.).

The use of OSC seems to be a reasonable way to facilitate effective and timely communication between the interface and audio generation layers, but also as a way to future proof the system. The OSC protocol is well established and a great number of both commercial and open source

---

<sup>31</sup> <http://hexler.net/software/touchosc>

environments include it as either a default library or an integral part of the application. The only real alternative would be to use MIDI (Musical Instrument Digital Interface). This particular protocol has indeed stood the test of time, being prominent since its inception in the mid 1980s. It is perhaps now showing its age, though, and whilst it is straightforward to implement on a basic level it can become difficult to extend it in any meaningful way. It is straight jacketed in the note, velocity message paradigm that can either restrict some musical mappings or limit the range of parameters that might be available or more accurate.

### **In practice**

Overall the aim is to develop a system capable of facilitating some of my approaches to composition and improvisation and follow this to its natural conclusion and allow the real time performance of my music.

In fairly basic initial terms the system needs to be capable of live looping as well as sound file playing and the manipulation of live input from a variety of sources and finally the live mixing and processing of the sound sources.

Currently the sample playback need not be synchronized, as there is not much of a focus on obviously rhythmic elements in my music. This also has the added benefit of considerably lessening the complexity in building a looper section. Loops of the same length could facilitate some rhythmic work should I need it in a fairly basic way, but there would be no need for the automated pitch shifting and time stretching available in some other

systems to ensure that all loops are in the right key and tempo. This is something that could be added later on if it becomes useful to me anyway.

On occasions when the rhythmic triggering of samples or loops might be useful I have decided to include a more traditional and simple system of step sequencers available for either sample playback or effects purposes. I have found a simple grid based interface most useful and immediate so far. It will also be useful to accommodate a number of LFO's and make them available throughout the system for processing and automation purposes.

The instrument is to be composed of various modules so that elements can be redefined and rebuilt without having to start from scratch every time. Also this would make it easier to add extra elements as and when they might be needed with a minimum of disruption to the entire system. The simpler modules will include sound file players to facilitate the playback of prerecorded material in a more concrete fashion. These will need basic control over speed and direction. The primary purpose is for accessing longer files direct from a hard disk.

The inclusion of a number of buffers will allow for the looping and manipulation of both live input and sounds from elsewhere in the system. The fact that the sound is stored in RAM also makes it much more straightforward to create rhythmic scrubbing effects and a more glitch aesthetic.

The various sounds will be piped through the system via a patching matrix allowing a pretty much anything into anything else network before ending up in a more traditional fader based mixer. This flexible routing will mean that any module will be able to send its output to any other module for processing or output. This needs to happen in an immediate way to allow for the real time building of compositional performances.

There will also obviously be a number of processing and manipulation tools based around time based effects such as delay and reverb and the filtering of sound. For many of these I have decided to use VST effects. With the system being modular these can be refined or replaced with self built units should I decide, but in the mean time utilizing these processors is a very effective way of leveraging a lot of sound processing power easily. The standard allows for easy access to parameters and sits in with the Max/MSP base very well.

While developing these units a number of exploratory pieces were created to test and assess the effectiveness and usefulness of the combined components. The following is a discussion of these pieces in terms of how the compositional idea informed the technology and how it sits into the development of the overall system. Really highlighting the compositional ideas and their technical execution as a test bed for elements of the larger framework.

## Plasticity

This piece plays with the minimalist Reich-like phase ideas and is based around the idea of a step sequencer but introduces some random and chaotic elements into the mix, to contrast with the potentially very rigid sequencer engine.

Essentially multiple step sequencers are all synchronized to the same rhythmic pulse, however all with different step lengths so any repeated patterns will go in and out of phase with each other. Random elements are introduced into the filtering and selection of the sequencers on triggering by the performer via simple on/off switches (see Fig 4).

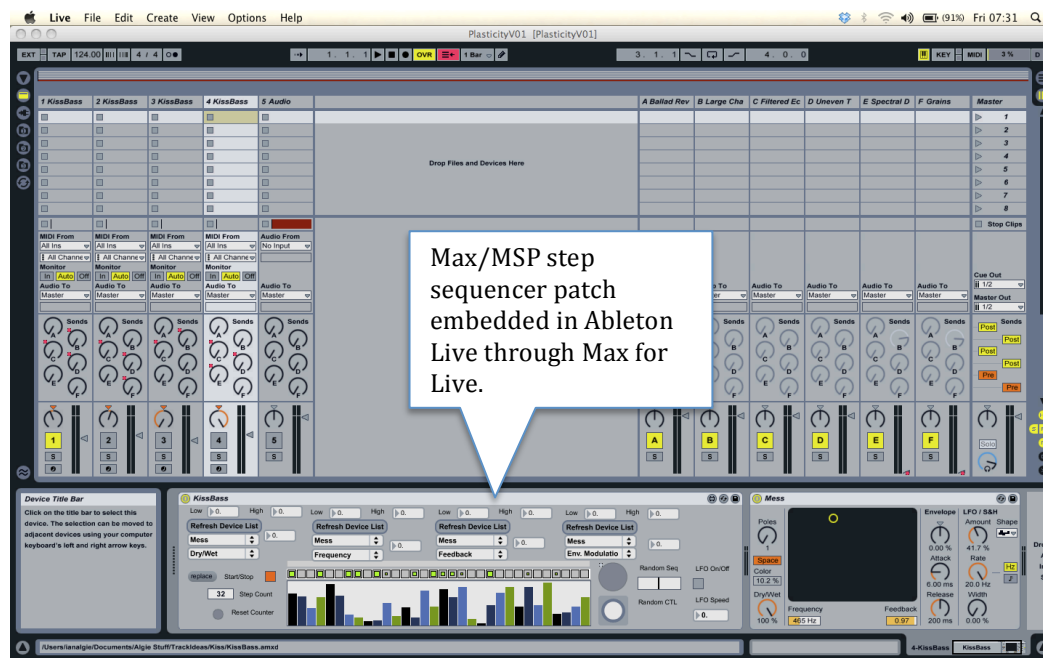


Figure 2 Plasticity Overview

This was also an experiment in the use of Ableton Live alongside Max/MSP via the Max for Live bridge (See Fig 2). The chaotic and controlled random

elements are much easier to create and control in Max and the use of Live means that I do not have to reinvent the wheel regarding a mixing and processing engine. Any mapping of MIDI controllers is also much more straightforward in Live and states are saved with the project as presets in a simple way. However this is a pull back to the commercial end of the software offerings, something that I am keen to avoid and as it will be necessary eventually I will be recreating a mixing engine in Max/MSP for final inclusion in the eventual instrument.

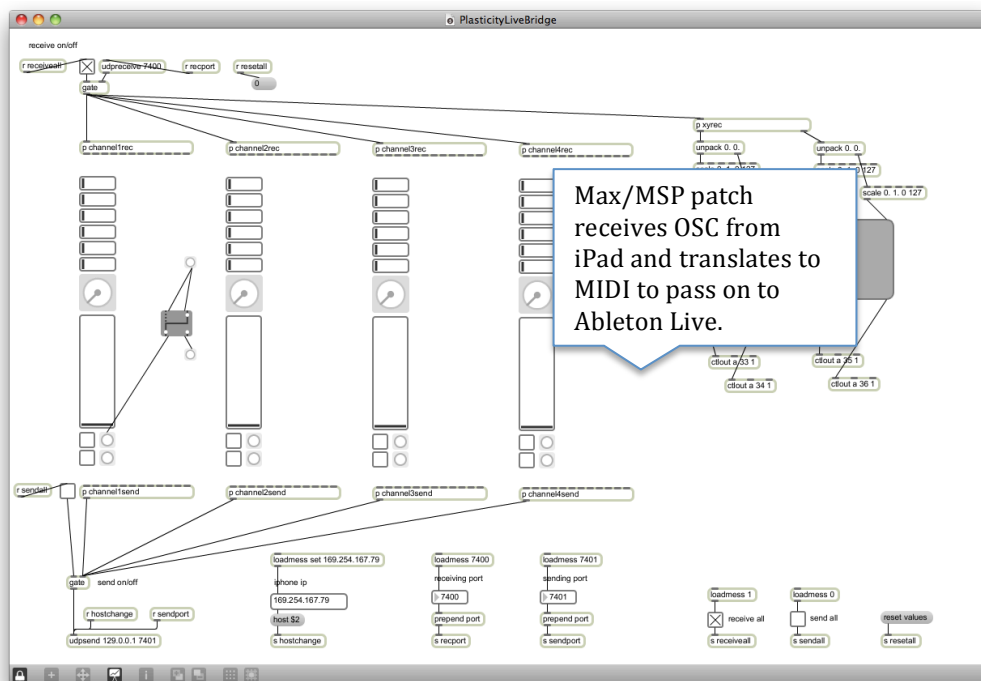


Figure 3 Plasticity OSC to Live bridge

Some elements of the ideas here were used in the production of ‘The Long Kiss Goodnight’ track in the portfolio. This sequencing engine was used for the live elements of the piece that were then arranged after the fact in a DAW. This example is an attempt to work with the engine purely in real time.

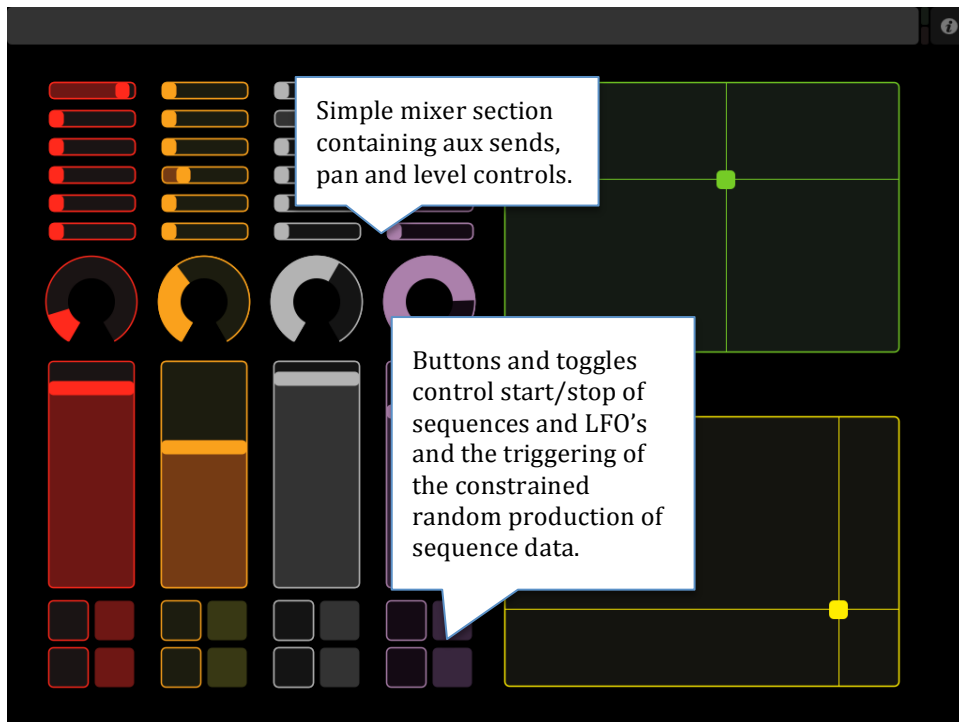


Figure 4 Plasticity iPad UI

During performance the artist has control over the population of the four individual step sequencers, the speed of the LFO's that control the density of the step population and the send levels, panning positions and track levels of both the sequencers and the auxiliary tracks. (See Fig 4). Each sequencer can also be stopped, started and reset individually during performance.

The source of basic movement and evolution throughout the piece is the simple saw tooth waveform behind the LFO's and it ultimately even controls the structure of the overall piece. As the LFO ramps up any triggering of the individual step sequencers will become much more dense. As this continues



through the piece the overall texture will move from a sparse pointillist texture on to a much fuller construction.

The shared pulse gives a strict rhythmic feel to the piece, but the phasing nature of the combined sequences means that any repeated rhythmic phrase is unlikely to hang around long, thus highlighting the intensity as the piece progresses.

There are six auxiliary sends used with four of them allowing the performer to briefly add various reverbs and delays as accents to the sequencers at various points. The other two are pre fade sends and used as feeds to two granular and spectral processors that take the rhythmic input and produce thematically related drones as output.

LFO speed can be set individually or collectively and it is this setting that has an impact in the overall timing of the piece. The increase in intensity could signify the end of a section or the end of the piece depending on the performers preferences. A shorter LFO time will introduce increased intensity quicker while conversely a longer LFO time will take longer to achieve a fuller sound.

Max/MSP was also used as the bridge between TouchOSC and Live (see Fig 3). The incoming OSC messages needed to be converted to MIDI controller messages for mapping in live.

## Little Machines

This piece explores the use of a granular engine, which allows both extreme ranges of pitch shifting and time stretching. These can both have a really interesting effect on the timbre of the source sounds. Speed and pitch can both be controlled in real time independently of each other. When the speed is set to zero the source sound is frozen. Little Machines uses multiple copies of the same source sound file. Progressing through the sounds at different rates with harmonically related speeds leads to a smeared sound.

With this slow scrubbing through the sounds the dynamics of the piece are led by the choice of source file or combinations.

The fundamental object behind this piece is the ElasticX Max/MSP object from ElasticMax<sup>32</sup>. Essentially this object is a direct replacement for the groove~ object within Max/MSP in terms of patching with the advantage that it encapsulates the pitch shifting and time stretching technology from Zplane<sup>33</sup> development who also license this technology to Ableton, Steinberg etc. Being able to use this technology directly in Max offers up many opportunities. There are some free alternatives that I did explore, notably freeelastic~<sup>34</sup> and some FFT based solutions, but nothing else seemed quite as smooth.

---

<sup>32</sup> <http://www.elasticmax.co.uk/elasticx~details.php>

<sup>33</sup> <http://www.zplane.de/index.php?page=music-creation>

<sup>34</sup> [http://devinkerr.com/2008/10/30/free\\_elastic-independent-pitchspeed-control-in-max/](http://devinkerr.com/2008/10/30/free_elastic-independent-pitchspeed-control-in-max/)

I would also like to be able to use live input to the piece and whilst it works with an individual player, subject to the obvious caveat that the play head cannot pass the buffer position, any use of multiple players currently leads to a large number of glitches. I hope that the developer can sort this out as it really would be great to use this with live input.

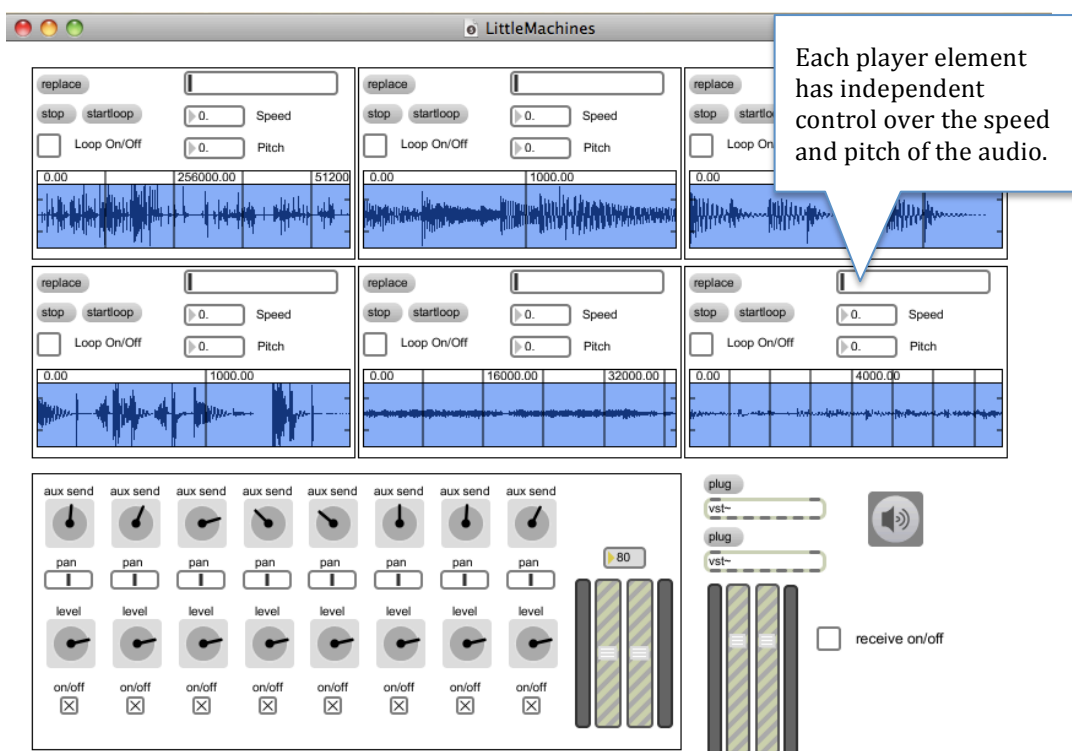


Figure 5 Little Machines interface

In terms of control the performer has access to the speed of the individual loops, their level and pan positioning and the send level to a reverb VST (see Fig 5). Whilst these can all be controlled individually and directly, I chose to use a mapping that means that as the sound file becomes slower so the send

to the reverb becomes higher. This relatively simple mapping allows the performer to build interesting spatial textures with simplicity of control.

(see Fig 6)

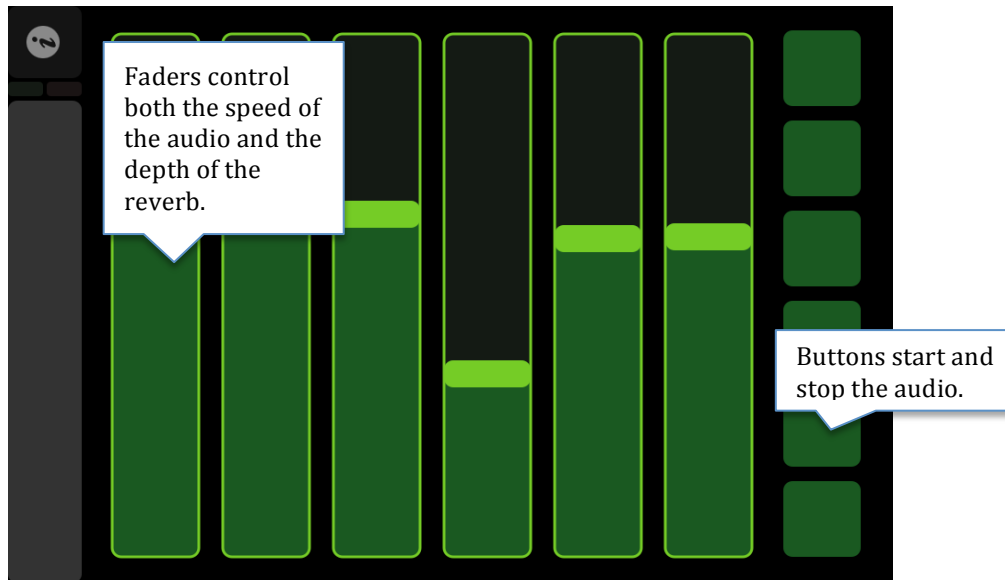


Figure 6 Little Machines iPhone UI

The simple uncluttered iPhone based user interface was again developed using TouchOSC.

Overall this quite simple process allows the performer to build some very rich textures with minimal material. As in this piece, using multiple copies of the same source sound works very well as depending on the nature of the source material melody and counter melody drifts and harmony seems to hang in the air then disappear with the sounds becoming ephemeral.

## NTWICM

This piece further explores the use of constrained randomness alongside commenting on the sound bite, short burst attention span of a lot of contemporary popular music listeners and in particular the 'Now That's What I Call Music series'.

The technical side of the piece is an extension of the Between The Lines engine built for inclusion in an on line installation piece for the 40<sup>th</sup> anniversary of Get Carter.<sup>35</sup> The iconic Luder designed Trinity Square car park was recreated in the online world of Second Life<sup>36</sup> and this audio engine was used to stream the piece live to players as they moved around the structure.

Basically a number of sound file players randomly select files from a playlist and then play them at various speeds. These speeds all produce sounds that are harmonically related at different octaves (0.12, 0.25, 0.5 and 2). When each file is played through completely another file is selected at random from the playlist. There is also a user definable delay between play messages and this can be used to shape the density of the texture. (see Fig 7) Because the files are being played at these speeds and in this case only tiny

---

<sup>35</sup> <https://www.tynesidecinema.co.uk/whats-on/carter-is-40/carter-is-40-second-life-party>

<sup>36</sup> <http://www.secondlife.com>

fragments of the source sounds are used the overall sound is unrecognizable from the sources.

The sound file trigger also selects the speed randomly from the possibilities, although this can be overridden by the performer should they wish to do so.

It is quite an open system and the choice of source files really does have a dramatic effect on the overall sound of the final texture. The Get Carter piece used any non-dialogue extracts from the film's soundtrack, whereas in this version, a collection of sounds from my own iTunes library provides the source sounds. Small elements from tracks were used, snare drum hit, a chord strum or a short phrase for example. All file players have access to the same playlist but due to the chaotic nature of the choices being made repetition is not easily recognized.

I like the thinking behind plunderphonics and the fact that a totally unrecognizable sound collage is being created in real time from appropriated materials. The dynamics come directly from the choice of materials and the texture from their combination.



## **Mt02**

I have long been a fan of the simple sound on sound looping of instrumental and vocal textures and this piece is an experiment in the combination of that kind of sound with some glitch elements. The actual looping parts of the patch are based on the Kaiser Looper used by Jeff Kaiser and elements of Oliveros' EIS, mentioned elsewhere in this dissertation. This particular patch allows for the juxtaposition of obviously rhythmic scrubbing against ambient washes of sound. It is also quite an effective way for one musician to create a lot of noise with fairly minimal input to the system.

In this instance for this piece the sound source is the electric guitar. Once inside the system there are two separate signal paths that can be selected, one going straight to the mixer and the second going via the loopers. In this way I can select which material will be looped and which will be played alongside any already looping material. Both signal paths utilize delays and reverbs as a way to extend the guitar's relatively short sustain period.

The performer is producing mostly tonal and modal material and although this isn't a necessity this system does lend itself more to this kind of material as any harmonic material established is going to be around for quite a while and so it is a little bit easier on the listener. The use of volume swells combined with the delay effects allow for the build up of ambient washes of sound quite easily from the guitar.



I find that the system also influences the structure of pieces played through it in an interesting way. With the one performer creating the piece through the layering of sound it is difficult to start off with a high degree of intensity and so the initial dynamic state is often quite soft. Once material is in the system, a piece can start to be built, this makes a kind of arch structure or at least a ramp up to an inevitable crescendo to some extent. Either the piece can be built up and then broken down or simply halted more dramatically at its peak. In this way the system is dictating the direction of the piece.

Whilst I do enjoy the simple kind of ambient drones that are easy to create with this kind of system the looper does facilitate the introduction of a rhythmic counterpoint to these sounds with the ability to scrub through the looping buffers at timed intervals. These intervals and the length of the audio to be scrubbed can be altered with random elements, within constrained parameters, if so wished. In this way some rhythmic elements can be established without worrying about the synchronization of the individual loops with each other and the sounds based around the same source material. There is no crossing zero checking or crossfading deliberately in this scrubbing process as I found the inevitable glitches quite appealing in highlighting any rhythmic accents.

The repetition and minimalist aesthetic of the sound production is quite sonically pleasing to my ears. Some sort of sync facility between the glitch elements might be worth exploring and is something that could be added

relatively easily in the future, with all looper elements taking their cue from a single synchronized pulse.

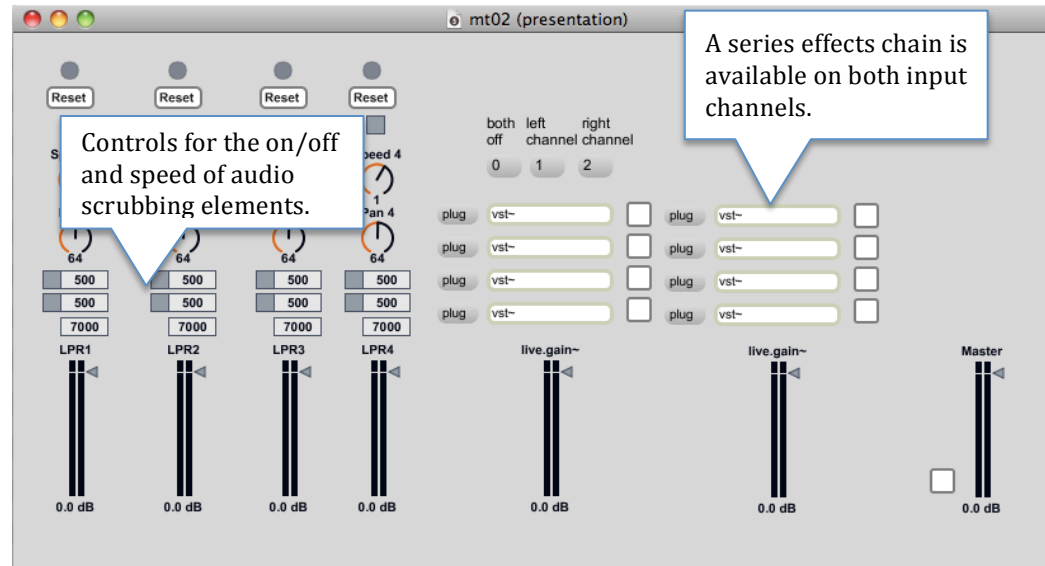


Figure 8 mt02 patch in presentation mode

Also in terms of control, currently any parameter can be mapped to incoming MIDI or OSC information. (see Fig 8) Whilst this is flexible I think that this needs to be more accessible to the performer and I need to explore foot controllers rather than the keyboard control currently used. This would free up performer hands and allow for enhanced concentration on the instrumental element of the piece. This would in turn encourage different structural approaches as having to remove hands from the instrument, as is currently the case really limits the user to the long form ambiances displayed here.

## Hold That Thought

This is again quite a simple piece technically that is capable of producing huge washes of sound. Essentially the basic technique at work is the spectral freezing of audio - either live input or from prerecorded sound sources.

(Charles, JF, 2008) In terms of performer control it is again quite straightforward. The performer has control of when to freeze either of the two freezer elements, the speed of the crossfade between the currently frozen and the next freeze and the levels to be sent to the systems effects chain. (see Fig 9)

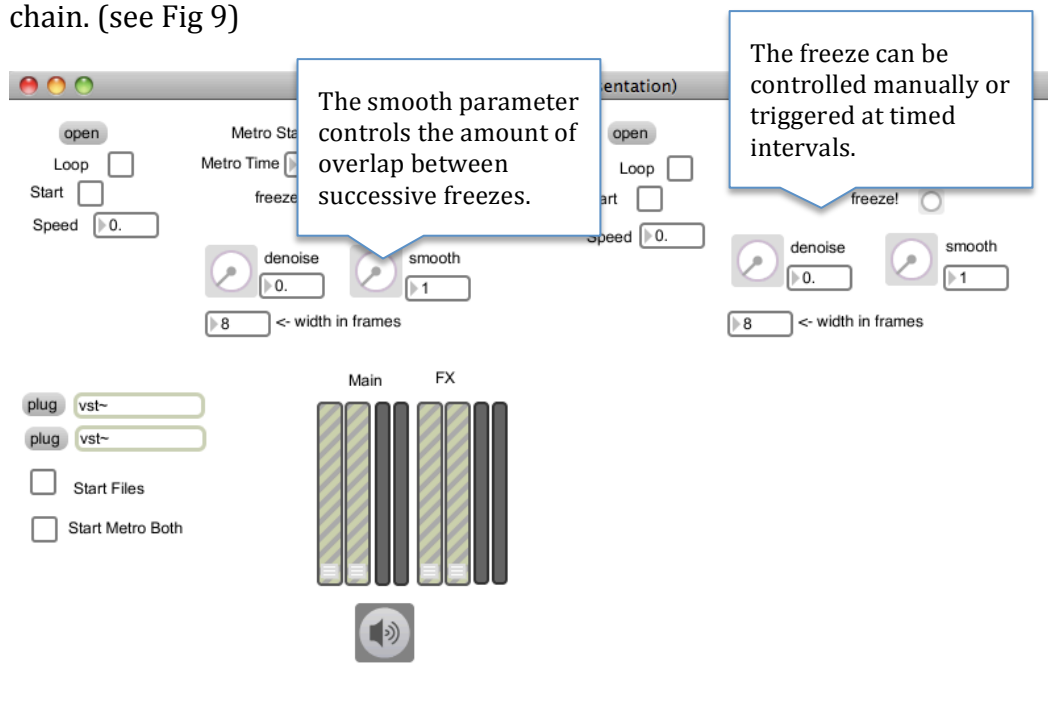


Figure 9 Hold That Thought patch in presentation mode

Although this seems deceptively simple it can lead to some really quite interesting combinations of sounds and again is quite useful when manipulating multiple copies of the same source sounds which are played at different speeds and triggered and frozen at different points. In this way some really dense textures can be created.

The original input source material can be mixed in if required but I think the system works best for me when the listener is unaware of the source sounds. Any input material becomes purely textural and loses any rhythmic sense, if there was any, as it is smeared and smudged against itself.

A number of delay and reverb effects are offered on auxiliary sends to further blur and extend the transitions between freezes if necessary.

Even though the controls are simple, I have included a facility to automate the key parameters for use in installation situations. The source sounds can be set to loop and timings can be set and automated for the triggering of freeze points. This kind of automation is also useful for setting up a bed of sound for the performer to improvise either with or against by way of further extension.

### **Cooking with Kids**

Cooking with Kids began as an exercise in using Max/MSP as a tool to combine elements of other programming environments to create generative elements for performance. In this instance a Max/MSP patch is used to combine an RTCmix script, some elements of JavaScript and a software emulator that allows the Novation Launchpad to operate as a Monome.

The RTCmix script is quite a short piece of code that simply plays sine tones at pitches randomly selected from a one octave C Lydian scale. These tones are enveloped with a slow attack and a long release and then further delayed when piped back into the Max/MSP audio system. These modal generative drones are then used as a bed for the rest of the musical material produced in the system.

The ability to emulate the Monome protocol opens up quite a range of software developed for that interface discussed elsewhere and the Nonome<sup>37</sup> patch enables the Launchpad to do just that. Apart from sharing the same eight by eight grid it is not really the same thing at all missing out on the hand built craftsmanship and green criteria, however the one thing it does have going for it is the fact that it is less than 20% of the cost. Whilst the grid isn't the answer to all control problems in this instance I am using some JavaScript code originally produced by Pixelmechanic<sup>38</sup> to drive a generative sequencer. Emulation of the Monome allowed me to simply drop the required modules into my patch to control the sequencer in this way. (see Fig 10) The sequencer is only 8 note polyphonic but facilitates the creation of some quite dense melodic material through the introduction of phasing elements.

---

<sup>37</sup> <http://post.monome.org/comments.php?DiscussionID=6245>

<sup>38</sup> <http://www.pixelmechanics.com/> - Boiingg

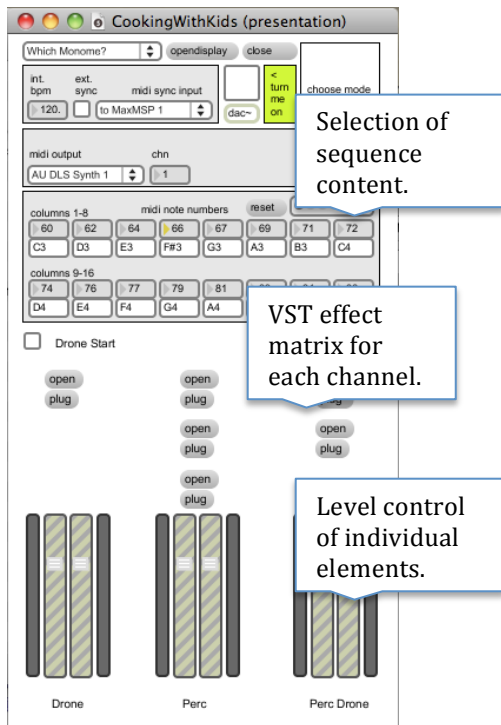


Figure 10 Cooking With Kids patch

Originally the sequencer sent notes via the IAC<sup>39</sup> buss to trigger virtual instruments in the Logic Pro application. Whilst this served its purpose, it seemed a little bit clumsy to have the two applications open and also dragged me further back towards commercial software. After prototyping the system in this way, I used the VST object in Max/MSP to include a software instrument directly in the patch. In turn I anticipate that this object be replaced with a Max based synthesis sub patch to keep it all in the family as it were.

The generative and improvisational material created through the sequencer is then sent, using a pre fade send, to the Drone Maker plugin by Michael

<sup>39</sup> An Mac OS internal system for the communication of MIDI information between applications.

Norris<sup>40</sup>. I have found this spectral manipulation plugin really useful for creating thematically related beds of sound from melodic material.

The final element of the system is a simple three channel mixer allowing the sine drone, the melodic material from the sequencer and the spectral drone material to be balanced against each other.

### **Set in Stone**

This piece is really a test bed and final prototype of what I hope to be is something like the final system combining a number of the modules discussed in relation to the other experimental pieces. It consolidates a number of ideas explored throughout my research and offers up a system that goes a long way towards reaching my goals of a real time composition and performance system. In this instance the system uses self built piezo contact microphones and stones as the only input source, the title and concept being based around the constant evolution of the system so far, is anything ever really set in stone?

---

<sup>40</sup> <http://www.michaelnorris.info/software/soundmagic-spectral.html>

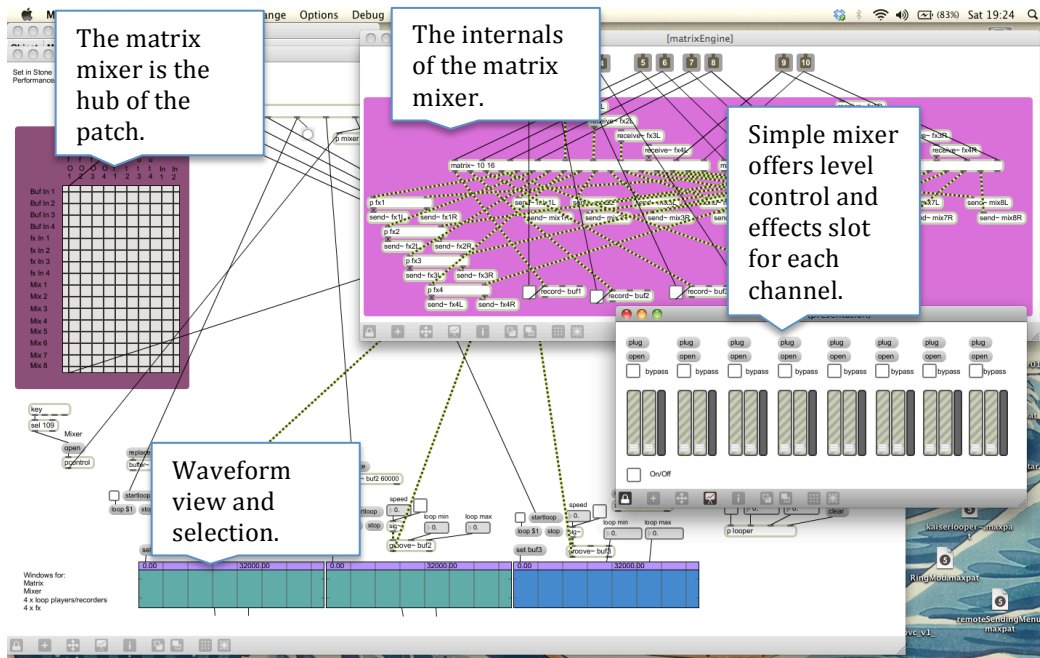


Figure 11 Set in Stone patch

Any input source material can be patched to various processes through the matrix mixer in a pretty much anything to anything else manner. (see Fig 11) This means that textures can be simple or quite complex and processed added and recorded at will throughout the performance. The options and tools are provided to create both ambient beds as well as much more glitch and angular material and move between quite sparse and very dense textures.

The dynamics of the piece are a combination of the original material used and the elements used to manipulate this within the system with again the scope for the creation of pieces varying from gentle drones to walls of noise.



Elements are available for the automation or semi automation of parameters within the system so it could effectively be used as a pure performance tool or utilized in installation work.

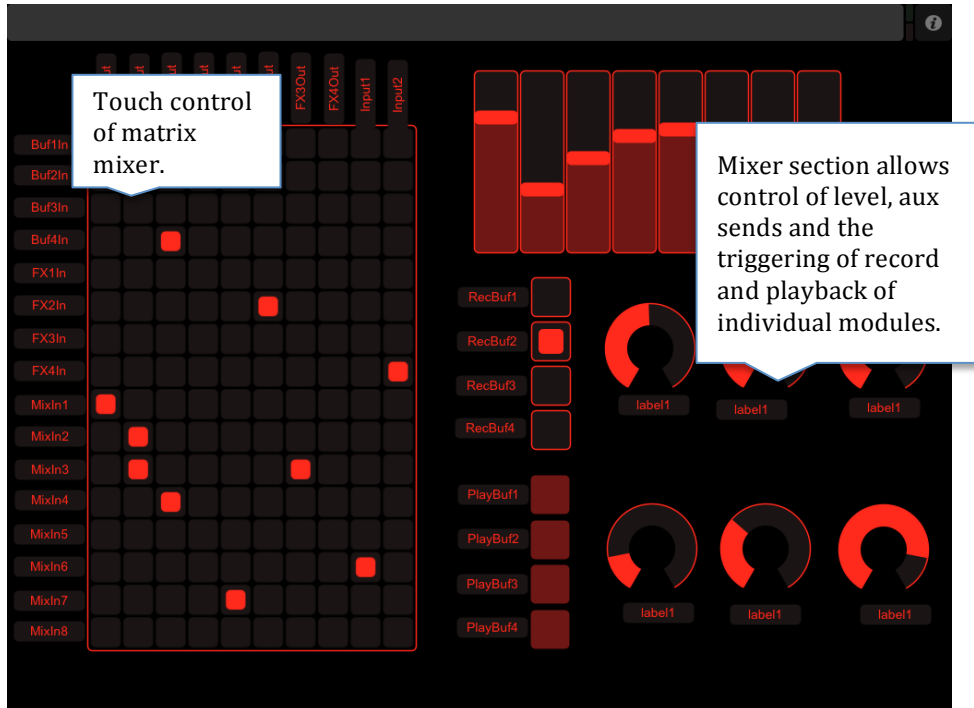


Figure 12 Set in Stone iPad UI

The iPad interface utilized is the most complex yet, but still relatively straightforward in use. (see Fig 12) I found with this setup though that I really missed the tactile elements available in other controllers. It would be much more effective to feel some feedback when making selections in the matrix, to know that you have pressed a button without having to check visually.

## Conclusion

I feel that this research has taken me quite some way in developing a laptop based system that affords me a level of improvisation in both composition and performance. Having said that it feels as though this is just the

beginning and there is a lot of scope for the further development of this system in a number of possible directions.

Whilst the choice of source sounds and the combinations of DSP elements allow me to impart a certain level of identity and possibly character on the audio products of the system it will take a lot of practice to fully develop the kind of vocabulary that will really take this towards being an instrument.

Even though I know the system literally inside out the physical interface still needs some work on the details to allow fluid movement around the system.

TouchOSC has been extremely useful in building flexible interfaces that can be customized in terms of the placement and size of the components. The iPad that hosts it is also great in terms of bang per buck. There are however some elements that really need a more physical and tactile sensation and I will be exploring this further through the LaunchPad and the Arduino.

The modular nature of the system has proved very useful and whilst the flexibility of swapping VST effects in and out has been valuable during the development, as I am settling on specific effects and processes I can begin to develop my own variations and replace them module by module.

One particular area I am now interested in exploring is the use of code in the audio generation elements of the system. SuperCollider especially, with its OSC interfacing, looks useful however I am also contemplating investing the

time and effort required to learn a 'real' programming language and moving in that direction.

In some ways I think that it is very likely that the software will never really be finished and ultimately I am conscious that there is a very real danger of spending too much time on the system and not enough on the art.

## References

2004. *Scanner Live Performance*, perf. Scanner. Leeds College of Music.
2007. *Pauline Oliveros Solo Performance*, perf. P Oliveros. De Montfort Univeristy.
2009. *Leafcutter John live performance*, perf. J Burton. Flapper and Firkin.
- Burton, J. *About Leafcutter* | [leafcutterjohn.com](http://leafcutterjohn.com). [online]. [Accessed 25 August 2011]. Available from World Wide Web: < HYPERLINK "http://leafcutterjohn.com/?page\_id=21" [http://leafcutterjohn.com/?page\\_id=21](http://leafcutterjohn.com/?page_id=21) >
- Casserley, L. 2001. Journeys, Instruments and Networks - 1966- 2000. *Leonardo Music Journal*. **11**.
- Casserley, L. 2011. *Re: Potential Max Patches* [email] (Personal Communication, 3 May 2011).
- Casserley, L. A Digital Signal Processing Instrument for Improvised Music. *Journal of Electroacoustic Music*. **11**.
- Charles, JF. 2008. A Tutorial on Spectral Sound Processing Using Max/MSP and Jitter. *Computer Music Journal*. **32**(3), pp.87-102.
- Collins, N. 2003. Generative Music and laptop Performance. *Contemporary Music Review*. **22**(4), pp.67-79.
- Collins, N. 2003. Live coding in laptop performance. *Organised Sound*. **8**(3), pp.321-329.
- Dubois, L. 2007. Extension 5: Sound. *In: Processing: a programming handbook for visual designers and artists*, London: MIT Press, pp.579-601.
- Garton, B. 2007. *Multi-language Max/MSP*. [online]. [Accessed 15 Feb 2011]. Available from World Wide Web: < HYPERLINK "http://cycling74.com/2007/03/26/multi-language-maxmsp/" <http://cycling74.com/2007/03/26/multi-language-maxmsp/> >
- Gough, H. 2012. *Re: Composition and Performance*. Berlin. [email] (Personal communication, 9 January 2012).
- Gough, H. *helenagough | composer*. [online]. [Accessed 24 August 2011]. Available from World Wide Web: < HYPERLINK "http://www.helenagough.net/" <http://www.helenagough.net/> >
- Grossman, R. 2008. The tip of the iceberg: laptop music and the information-technological transformation of music. *Organised Sound*. **13**(1), pp.5-11.

- Historical Performances - Toplap*. [online]. [Accessed 13 February 2012]. Available from World Wide Web: < HYPERLINK "http://toplap.org/index.php?title=HistoricalPerformances" <http://toplap.org/index.php?title=HistoricalPerformances> >
- Hunt, A, M Wanderley, and M Paradis. 2002. The importance of parameter mapping in electronic instrument design. *In: Conference on New Instruments for Musical Expression*. Dublin: NIME.
- Inglis, Sam. 2003. Four Tet Kieran Hebden: Recording Rounds. *Sound On Sound*, July.
- Kaiser, J. 2007. *How I Lost 150 Lbs. Thanks to Max/Msp! (Back to My Routes: Freely Improvising with Max/Msp)*.
- Kimura, M. 1995. Performance Practice in Computer Music. *Computer Music Journal*. **19**(1), pp.64-75.
- Lansky, P. 2004. *The Importance of Being Digital*. Amsterdam.
- Lexer, S. 2010. *Max as a live performance and composition tool*. London.
- Lexer, S. 2010. Piano+: An Approach towards a Performance System Used within Free Improvisation. *Leonardo Music Journal*. **20**, pp.41-46.
- Macdonald, A. 1995. Performance Practice in Electroacoustic Music. *Computer Music Journal*. **19**(4), pp.88-92.
- Magnusson, T. 2006. *Affordances and Constraints in Screen-Based Musical Instruments*.
- Magnusson, T. 2008. Expression and Time: the question of strata and time management in creative practices in technology. *In: A Mansoux and M DE Valk, (eds). FLOSS+Art*, Poitiers: GOTO10, pp.232-247.
- Magnusson, T. 2010a. Designing Constraints: Composing and Performing with Digital Musical Systems. *Computer Music Journal*. **34**(4), pp.62-73.
- Magnusson, T. 2010. *The ixi lang: a supercollider parasite for live coding*. Brighton.
- Magnusson, T. 2011. *Confessions of a Live Coder*. Brighton.
- Mccartney, J. 2002. Rethinking the Computer Music Language: Super Collider. *Computer Music Journal*. **26**(4), pp.61-68.
- McLean, A. 2004. *Hacking Perl in Nightclubs - Perl.com*. [online]. [Accessed 15 January 2012]. Available from World Wide Web: < HYPERLINK "http://www.perl.com/pub/2004/08/31/livecode.html" <http://www.perl.com/pub/2004/08/31/livecode.html> >

- McLean, A. 2008. Live Coding for Free. *In: A Mansoux and M De Valk, (eds). FLOSS+Art, Poitiers: GOTO10, pp.224-231.*
- McLean, A., Griffiths, D., Collins, N., and Wiggins, G. 2010. Visualisation of Live Code. *In: Electronic Visualisation and the Arts London 2010.* London.
- McLean, A and G Wiggins. 2010. *Bricolage Programming in the Creative Arts.* London.
- McLean, A and G Wiggins. 2010a. Tidal - Pattern Language for Live Coding of Music. *In: Proceedings of the 7th Sound and Music Computing conference.* Barcelona.
- Murphy, B. 2011. *ProFile: All the Pretty Algorithms | Gregory Taylor.* [online]. [Accessed 12 January 2012]. Available from World Wide Web: < HYPERLINK "<http://www.emusician.com/features-interviews/0777/profile-all-the-pretty-algorithms--gregory-taylor/137101>" <http://www.emusician.com/features-interviews/0777/profile-all-the-pretty-algorithms--gregory-taylor/137101> >
- Oliveros, P. 2005. *Deep Listening: A Composer's Sound Practice.* iUniverse.
- Oliveros, P. *expandedmusicalinstruments | DEEP LISTENING INSTITUTE.* [online]. [Accessed 5 October 2011]. Available from World Wide Web: < HYPERLINK "<http://deeplisting.org/site/content/expandedmusicalinstruments>" <http://deeplisting.org/site/content/expandedmusicalinstruments> >
- Oliveros, P. *outsidethewindow | DEEP LISTENING INSTITUTE.* [online]. [Accessed 6 October 2011]. Available from World Wide Web: < HYPERLINK "<http://deeplisting.org/site/content/outsidethewindow>" <http://deeplisting.org/site/content/outsidethewindow> >
- Prendergast, M. 1995. *AMBIENT TECHNO: Peter Namlook, Mixmaster Morris, Scanner & David Toop.* [online]. [Accessed 14 November 2011]. Available from World Wide Web: < HYPERLINK "[http://www.soundonsound.com/sos/1995\\_articles/mar95/ambienttechno.html](http://www.soundonsound.com/sos/1995_articles/mar95/ambienttechno.html)" [http://www.soundonsound.com/sos/1995\\_articles/mar95/ambienttechno.html](http://www.soundonsound.com/sos/1995_articles/mar95/ambienttechno.html) >
- Rimbaud, R. *scanner.* [online]. [Accessed 15 September 2011]. Available from World Wide Web: < HYPERLINK "<http://www.scannerdot.com/scanner.shtml>" <http://www.scannerdot.com/scanner.shtml> >
- Roads, C. 1996. Early Electronic Music Instruments Time Line 1899-1950. *Computer Music Journal.* **20**(3), pp.20-23.

Sellars, P. 2002. *Designer Label*. [online]. [Accessed 15 December 2011]. Available from World Wide Web: < HYPERLINK "http://www.soundonsound.com/sos/may02/articles/paradinas.asp" <http://www.soundonsound.com/sos/may02/articles/paradinas.asp> >

Stowell, D. 2011. Writing Unit Generator Plug-ins. *In: S., Cottle, D. and Collins, N Wilson, (ed). The SuperCollider Book*, Cambridge, MA: MIT Press, p.680.

Stowell, D and A Mclean. 2011. Live music-making: a rich open task requires a rich open interface. *In: Proceedings of the BCS HCI 2011 Workshop - When Words Fail: What can Music Interaction tell us about HCI?*.

Trueman, D. 2007. Why a laptop orchestra? *Organised Sound*. **12**(2), pp.171-179.

Walby, Sam. 2010. Leafcutter John, Computers never felt so unpredictable. *Now Then Magazine*, November, pp.52-53.

Zicarelli, D. 2002. How I Learned to Love a Program That Does Nothing. *Computer Music Journal*. **26**(4), pp.44-51.

## **Appendix 1 – Portfolio submission and programme notes**

### **Taut - Stereo 4' 18" AIFF**

*01taut.aif*

Taut is a fractured exploration of the range of timbres available from the steel string acoustic guitar. Wire stretched across wood is used to create source tones that are in turn further stretched through arrangement and processing. Emotionally taut, tense and disjointed.

### **Drive By - Stereo 12'08" AIFF**

*02DriveBy.aif*

This piece explores our human perception of time. Whilst time is a physical fact, our perception of it can be influenced by what we are doing. We talk of time flying or dragging, when really it ticks on second by second and in reality it is just our experience of it that alters. The motorway journey was used as the basis here. Depending on where we are travelling to and the purpose of that journey time can appear to expand or contract.

### **The Long Kiss Goodnight - Stereo 9' 42" AIFF**

*03TheLongKissGoodnight.aif*

The Long Kiss Goodnight uses a single organic sound object as the starting point for the exploration of generative rhythms. Step sequencers can often be used in a very prescriptive mechanical way, but here elements of randomness and chaos are introduced into the mix. The piece has a very rhythmic feel and sometimes sounds quite rigid, however there is the potential for the generative elements of the engine to produce a similar but different product with each performance.

### **Bone Machine - Stereo 9' 08" AIFF**

*04BoneMachine.aif*

The coaxing of an electric guitar into sustaining feedback is used as the source sound in this piece. The source files are stretched, frozen,



manipulated and layered to create a dense texture that moves between moments of noisy dissonance and fields of harmonic stasis and suspension.

**What's Going On In There** - Stereo 9' 58"

*05WhatsGoingOnInThere.aif*

An experiment in data moshing. Computer files other than audio files are sonified through use as raw audio data files. The zeros and ones that originally represented images and text files are repurposed and used as input to a signal processing system. Whilst often very noisy the resulting audio can sometimes sound quite organic and often reveals some interesting rhythmic interplay.

## **Appendix 2 – DVD contents**

### **Dissertation Audio**

Between The Lines - Stereo 13' 10" AIFF  
Cooking With Kids - Stereo 8' 29" AIFF  
Hold That Thought - Stereo 05' 06" AIFF  
Inner Space - Stereo 09' 08" AIFF  
Little Machines - Stereo 14' 39" AIFF  
mt02 - Stereo 11' 20" AIFF  
NTWICM v2 - Stereo 9' 52" AIFF  
Plasticity v1 - Stereo 7' 22" AIFF  
Set in Stone - Stereo 11' 39" AIFF

### **Dissertation Code**

Between The Lines – Max/MSP Patch  
Cooking with Kids – Max/MSP Patch  
Hold That Thought – Max/MSP Patch  
Inner Space – Max/MSP Patch  
Little Machines – Max/MSP Patch  
mt02 – Max/MSP Patch  
NTWICM – Max/MSP Patch  
Plasticity – Max/MSP Patch and Ableton Live Project  
Set in Stone – Max/MSP Patch

Where possible all external objects and VST effects are included.

### **Digital copy of Dissertation**

Laptop Performance The Current State of Play - PDF