UNIVERSIDADE CATÓLICA PORTUGUESA

COMPOSITION WITH COMPLEX DATA: A CONTRIBUTION ON THE MAPPING PROBLEM THROUGH PRACTICE-BASED RESEARCH

Dissertation submitted to the Portuguese Catholic University in partial fulfilment of requirements of the Doctoral Degree in Science and Technologies of the Arts – Computer Music

by

Samuel Van Ransbeeck

ESCOLA DAS ARTES

March 2015

UNIVERSIDADE CATÓLICA PORTUGUESA

COMPOSITION WITH COMPLEX DATA: A CONTRIBUTION ON THE MAPPING PROBLEM THROUGH PRACTICE-BASED RESEARCH

Dissertation submitted to the Portuguese Catholic University in partial fulfilment of requirements of the Doctoral Degree in Science and Technologies of the Arts – Computer Music

by Samuel Van Ransbeeck

Dissertation supervised by António de Sousa Dias

ESCOLA DAS ARTES

March 2015

This work was developed with the financial aid of

the Fundação para a Ciência e Tecnologia (FCT),

under the Programa Operacional Potencial Humano

SFRH/BD/72601/2010

PROGRAMAS DE DOUTORAMENTO FCT

This research has been performed at the Catholic University of Portugal- Research Center for Science and Technology of the Arts (CITAR) in Porto.

Abstract

Composition with complex data is a field of computer music composition/interactive art that uses extra-musical data from various sources like stock exchange data, weather data or seismic data. Despite the fascination that one can have in his exciting field of composition, there is still a lack of data management applications for artistic use. Hence, one is generally forced to create one's own applications, at the expense of the time that should be spent in the artistic side of the work. As the technology can be part of the work but never the sole constituent, we decided to develop a toolbox, DataScapR, which allows artists to work easily with data, helping them in focusing on the artistic side. This toolbox will help the user to quickly advance beyond the technical development and focus on the artistic side of the project.

This project consists of four components: a theoretical framework for sonification art, a state of the art and discussion on mapping techniques, the development of a sonification toolbox for composers who wish to use complex data (more specifically stock market data) as the source material for their music and a series of works as case studies to show the capabilities of the toolbox.

Bringing theory and practice, art and technology together, this project can be seen as a practice-based one embedded within a theoretical framework.

Keywords: sonification art, algorithmic composition, data, information, concept, technology

Resumo

Nesta dissertação, abordo o tema da sonificação em contexto artístico. Sonificação, a tradução de dados en sons, é um largo campo de investigação e existe todo o interesse na exploração deste domínio num contexto artístico. Esse projecto visa contribuir para a criação de um ramo teórico para a arte da sonificação e, em simultâneo, apresentar uma aplicação que facilite o uso de sonificação em contexto artístico. Com efeito, actualmente, existem poucas aplicações que permitam usar sonificação na composição musical de uma forma acessível. Por esta razão, um compositor pode ter que aplicar um tempo considerável no desenvolvimento de uma aplicação própria o que muitas vezes, pode ter um efeito prejudicial na criação da obra em si. Se o compositor tem que aplicar demasiado tempo no desenvolvimento tecnológico pode correr o risco de considerar a aplicação como sendo a obra em si, o que não é o caso: pode ser uma parte mas nunca a obra. Para combater o pequeno leque de aplicações acessíveis criei uma caixa de ferramentas, DataScapR, desenvolvido em Max, e disponibilizada para o público em geral. DataScapR é um projecto aberto: o utilizador pode estendê-lo livremente e adaptá-lo às suas necessidades. Todos os patches são comentados extensivamente para facilitar a sua edição e extensão. O uso prático de DataScapR é exemplificado através de Através de estudos de caso demonstrando que a sonificação pode ser uma prática interessante integrada num contexto artístico.

Nesse projecto de doutoramento foco um tipo de dados especifico: dados da bolsa das acções. Isso vem dum interesse pessoal e no dinamismo inerente à bolsa. Sempre considerei a bolsa fascinante e penso que pode ser interessante para usar os dados para sonificação.

A dissertação consiste em quatro partes. A primeira parte aborda questões teóricos: procuramos uma definição de arte de sonificação e integramos essa prática no contexto da composição. Tratamos da questão da natureza e definição de dados e como eles podem ser aplicados na música. Depois de construída uma base teórica, descrevemos o estado da arte. Nesse segundo capítulo descrevemos obras que usam sonificação como componente importante da própria existência e discutimos os diferentes métodos de mapping.

Seguidamente, discuto o software existente bem como a necessidade duma nova aplicação. No terceiro capítulo apresento *DataScapR*, um dos componentes práticos do doutoramento. *DataScapR* é uma caixa de ferramentas para sonificação de dados da bolsa de acções. Assim, apresento os três módulos que permitem usar dados em tempo real e dados históricos. Os métodos de mapping são explicados e a estrutura interna dos patches é apresentada. Finalmente, no quarto capítulo apresento as obras realizadas usando DataScapR: *For A Fistful Of Data* (flauta de bisel), *4D Brokers* (instalação), *Vapourwaves* (instalação), Mirage (obra sobre suporte). Para cada caso, apresento a obra, discuto a sua estrutura, os mappings utilizados e as questões técnicas e termino com uma avaliação da obra. No final do capítulo concluo com uma avaliação geral das peças. Na discussão final realizo uma avaliação do trabalho feito e aponto direcções para trabalho futuro.

A dissertação é da caixa de ferramentas *DataScapR*, de quatro estudos de caso e dois blogs: datascapr.wordpress.com onde o *DataScapR* está disponível e sonifcationart.wordpress.com onde discuto vários projectos de sonificação.

Esse projecto de doutoramento mostra apenas uma das posições possíveis em arte de sonificação e, por esta razão, deve ser considerado como uma abertura para novos caminhos a explorar.

Palavras chave: sonificação, composição algorítmica, dados, informação

To Cecilia

ACKNOWLEDGMENTS

It would not have been possible to undertake this PhD without the assistance of other people. Therefore, I would like to acknowledge their contributions and thank them for their valuable help. In the first place I would like to thank my supervisor António de Sousa Dias, who always found new angles to approach the project. I would also like to thank Katharina Vogt and Visda Goudarzi, whose expertise helped in progressing during my stay in Graz. Furthermore, I would like to thank Pedro Pestana for assessing the software. Andrea Agostini and Daniele Ghisi from the *Bach* project were very helpful in order to understand *bach* and to solve problems while I was developing *DataScapR*.

My PhD colleagues were a valuable source of inspiration and criticism. Vitor Joaquim's decades-long experience helped me to re-evaluate concepts. Discussions with João Castro Pinto were challenging and inspiring, making me want to indulge more in philosophy. Furthermore I would like to thank André Baltazar, João Cordeiro, José Vasco Carvalho, Mailis Rodrigues, Francisco Bernardo, Pedro Patrício, Diana Cardoso and André Perrotta for their interesting discussions over the years.

Professors are the fundaments of a University; they are beacons on the vast ocean of knowledge. Paulo Ferreira-Lopes', Luis Gustavo Martins', Carlos Sena Caires', Álvaro Barbosa's and Sofia Lourenço's words helped to navigate towards the destination. I also would like to thank the professors at the IEM in Graz, especially Gerhard Nierhaus for his interesting discussions in the field of algorithmic composition.

Many people whose names I do not know helped through forums and mailinglists such as cycling74.com, microsound.org, acma mailinglist and many more. They were very helpful in finding papers, explaining programming concepts and clarifying artistic issues.

I would like to thank all artists whose works I describe in this dissertation and on the blog for their answers explaining their works in a detailed way.

This project would have been impossible without the investment of the Foundation for Science and Technology. The investment in this project proved an open spirit and willingness to research topic without an apparent immediate economic return. We can only hope that these investments will continue to happen in the future. As the Foundation is financed through taxpayers’ money, I would like to thank all these taxpayers for their contribution to this project.

I would like to thank the Catholic University and the Centre for Research in Technology and the Arts-CITAR for accepting me as a student and letting me pursue my PhD here.

My parents who stood behind me were a big support during the years.

Cecilia Peçanha has been the most important person in my life during the past years. She deserves every credit for her kindness and accompaniment throughout all these years.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

Motivation

In this dissertation I discuss sonification art through various viewpoints. This project started out from a personal desire to create a framework for the sonification of stock market data. Using this idea as a starting point, I realized that this would be a big undertaking, obliging me to delve deeper into the field of sonification. Upon embarking on this PhD project, I saw from my experience as a composer that there was a discrepancy between technology and the composer's output. Whereas technology opens a gateway to a compositional ocean, it also opens the risk of drowning in the vast deepness of technological complicatedness. Many fellow students would embark on writing complex patches in *MaxMSP* but forgot to take that technology to the resulting artefact. In other words: when the patch was ready, the resulting composition was treated poorly in comparison. While the composer invested a lot of time in making the patch to work, he would not have sufficient time to create a compelling composition with his patch. In my conviction that sonification of stock market data offers a great potential for musical material, I wanted to create an application for other composers that they would be able to use, without having to loose too much time in programming the framework. This framework is *DataScapR*, presented here. Indeed, programming *DataScapR* was not an easy task, not being a software engineer myself. Nevertheless, having a compositional background, I thought I would be able to keep musical perspectives in a way that is generally not possible to a software engineer. While the application is ready to use, I call it a toolbox as it is an open work: users who wish to expand the application can do so according to their needs. As the programming environment was *MaxMSP*, a widely used environment in the musician's community, it is easy to open up the patches and edit them. The application as an open work means that *DataScapR* offers possibilities that I did not explore myself but nevertheless can be interesting for other people to delve into. Of course

I wanted to create my own works using this application, which I present in this thesis as case studies.

Ultimately, the goal is to create an understanding of sonification art and prepare possible pathways for artists to explore. In no way this Thesis should be seen as a definitive work: it is a particular perspective on the topic and one can study it from many different angles. My contribution is thus one of the many studies possible.

Structure of the thesis

This thesis consists of four chapters. In the first chapter, the field of sonification art is defined and delineated. This involves a comparison between existing definitions and a redefinition for the current concept. Having defined the field of research, I then pursue a localisation within the philosophical-aesthetical field in order to make the research position clear. Having made the predisposition clear, it is possible to embark on the questioning of the problem set I want to treat in this project.

After having posited the questions, I proceed to discuss the state of the art through discussing existing sonification projects and dissecting the mapping methods, which the creators used. This allows for future researchers-artists to see which mapping methods exist and how they can be used in potentially interesting ways. I devote a section as well to sonification software packages and discuss the differences, advantages and disadvantages.

The third chapter presents the *DataScapR* toolbox: the architecture, the different components and how to use it are discussed.

Finally, in chapter four, the practical outcomes of using *DataScapR* are presented: an installation work (as well as an embryonal version of an installation work), an acoustical composition and an electro-acoustic piece.

The reader will find in the appendix and the accompanying DVD the *DataScapR* toolbox, the documentation of the artworks and published papers.

The dissertation is accompanied by two blogs: datascapr.wordpress.com, where the user can download the *DataScapR* toolbox and sonificationart.wordpress.com, where sonification works are discussed.

Situation: orientation of the dissertation concepts

This thesis aims to research sonification in an artistic context. While sonification has been researched extensively for projects that are mainly scientific in nature, sonification art has been not been studied in a systematic way. This thesis aims to fill the existing void though both theoretical research as well as an application of the concepts put forward. There are many perspectives possible on this topic and in no way this thesis will be the Holy Grail in sonification art research. Nevertheless, I hope that this thesis will be a good contribution to the level playing field.

Limits of this thesis

In order to perform this research, some limitations had to be set. These help to keep focus attention on the topic.

As the dissertation is focused on sonification in an artistic context, scientific sonification is not discussed at-length. Sonification is a very large field so we need to limit ourselves to avoid wanting to write everything.

The State of the art discusses a selection of projects where substantial information on mapping techniques has been available. Other projects are presented on the research blog sonificationart.wordpress.com, which accompanies this dissertation.

Although originally intended, a user study on *DataScapR* is not included. I believe it would not add value to the dissertation as the toolbox is meant to be used in a very personal way: every composer will use the toolbox in a different way and tracking those different uses would not be useful unless after a few years when the toolbox is in use. At the moment, I deem it not appropriate to add a user study.

Technical details

The dissertation has been typeset using *Microsoft Word 2011 for Mac*. The font used throughout the text is *Times New Roman*, except for code snippets where *Andale Mono* was used. Citations and bibliography were handled using *Mekentosj Papers 3* and the used style is *APA 6th edition* as provided by *Mekentosj Papers 3*.

1 THEORETICAL CONCEPTS

1.1 Introduction

Sonification art requires the use of auditory displays. Therefore, prior to discuss or present sonification we need to contextualize and define/present auditory displays. Various definitions of auditory displays (AD) are juxtaposed, followed by a discussion on algorithmic composition. In the end I will propose a definition of sonification in an artistic context.

1.2 Auditory Display: Definitions, Types, and Functions

Sonification represents a subset of what we call an auditory display (AD). As with a visual display, an auditory display acts as an interface between the information source and the recipient. (Walker & Nees, 2011) define AD as "any display that uses sound to communicate information…". This goes in the same direction as (de Campo, 2008) who refines this, stating that "Auditory display is the rendering of data and/ or information into sound designed for human listening."

In AD we can distinguish two subsets:

1. The well-understood information subset. In auditory information display well-understood information is communicated through sound. Examples are public service announcements, auditory feedback sounds on computers, alarms and warning systems, process monitoring systems, etc. These messages do not require explanation: their meaning is immediately perceptible.

2. The Sonification subset. Sonification or Data Sonification is the rendering of (typically scientific) data into (typically non-speech) sound designed for human auditory perception. The informational value of the rendering is often unknown

beforehand, particularly in data exploration. Through (repeated) listening of the sonic result we are able to grasp the meaning of the data.

We can further refine these distinctions in various ways according to their intended function, as do Walker and Nees who describe four function categories in auditory display:

1. Alerting functions: any sound that warns the user for an event, e.g. an alarm clock[1]
2. Status and progressing indicating functions: sounds to keep the user informed about the status of a process
3. Data exploration functions: The sound is used to give a person an impression of the data set as opposed to alerting and status indicating functions that only show a momentarily state.
4. Art and entertainment: In addition to yielding warnings and sonifications, events and data sets can be used as the basis for musical compositions. For example: one can generate musical sequences and build a composition with them.

Looking at these types, we can say that type 3 and type 4 share similarities; we are exploring a dataset, however, the goal is different. (Polansky, 2002) makes this distinction between data exploration functions (sonification) and sonification for arts practices by dividing the practice in a sonification subset and a manifestation subset; *sonification* is where the objective is to sonify a process in order to better understand it. Its purpose is pedagogical and illustrative. (Hermann & Hunt, 2004) describe an example of this kind of sonification: Helicopter flight data is sonified and listened to in a compressed time space. The listener can discover abnormalities in the flight data by listening to changes in the sound. On the other hand, in *manifestation* we are instantiating an aesthetical experience, in musical terms: a composition. Although alarm sounds or status report indicator sounds should not be seen as totally unaesthetic, however the functionality is primary in those types.

Using the above-mentioned definitions, we are able to draw an auditory display map pictured below.

---

[1] We can consider this binary information: there are only two possible states.

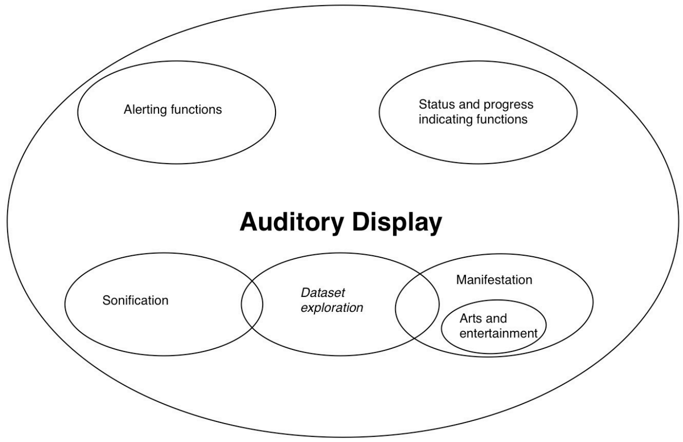

Figure 1 – Auditory display and its subtypes. Sonification and Manifestation are connected through the dataset exploration function.

1.2.1 Sonification Modes and Techniques

In sonification practice for data exploration (scientific or artistic), it is necessary to choose a proper mapping method. Arbitrary mappings that have no relation to the data or cannot be easily related to them will impede the listener to gain a better understanding of the data, offering no advantage over visual or other displays. This problem is commonly known as the mapping problem. In this research, it is necessary to have a closer look to possible mapping methods. They will help us in understanding the mapping problem and devise strategies to create interesting mappings.

(de Campo, 2008) states that most literature describes three sonification modes:

Audification: If we use high sampling rates, the data can be translated into waveforms. For example, seismic data have been translated into waveforms for example (Bioacústiques, n.d.) and (Aiken et al., 2012). This process is called audification (or audication). As the data patterns do not follow physical laws that exist in nature, the resulting waveforms can sound unearthly and difficult to interpret for the user (Worrall, 2009a). On the other hand, these unearthly waveforms can be an interesting source for sound design and composition. Furthermore, although the sound might not translate the data in an easy-to-interpret way, this sonification mode is probably the closest to the data. Indeed, the only way to change the sounding result is to either use a different sampling resolution or to play the waveform at a different speed (which is the same as reading the data at a different speed). There is no layer in-between the data and the resulting sound. In other words: the mapping is straightforward. Of course, one can use an intermediate mapping to manipulate the data's ranges but there we enter the realm of parametric mapping.

Event-based sonification using parameter mapping: Each data point is mapped to a sound event, for example: Each update of a stock price results in a new note where the frequency of the pitch is controlled by the price. Synonims are *ratiometric* mapping (Polansky, 2002), auditory scatterplots (Flowers, 2005), unit-swapping (Saladin, 2007) or n^{th}-order parametric mappings (Scaletti, 1994) as cited in (Worrall, 2009a). The advantage of parametric mapping is its flexibility: changing values can be done easily and give different sonic results. The disadvantage is that the change in one parameter can influence how we perceive the others: loudness can affect pitch perception for example (Worrall, 2009a). It is thus necessary to understand the destination parameter's nature and structure.

Model-based sonification: In this type, the user is able to interact with the data mappings. Instead of listening to a static representation of the data, he can manipulate parameters in real time. (Hermann & Hunt, 2004) defines Model-based sonification as:

> "a sonification model is a dynamic system, formed from the data under scrutiny, *plus* a set of interactions determining how the user may excite the system *plus* a fixed mechanism describing how the resulting dynamic behaviour determines the sound".

Model-based sonification is particularly useful in sonifying multi-dimensional datasets where a time-axis (or another equal-distance element to organize the data) is absent. One can compare it to playing a string instrument: in order to make sound, the string has to be excited by the user. The length and thickness of the string will then control the resulting sound. In model-based sonification, virtually bowing the string will excite the virtual body, which is controlled through the data. The physical modeling of instruments can be considered a prime example of model-based sonification.

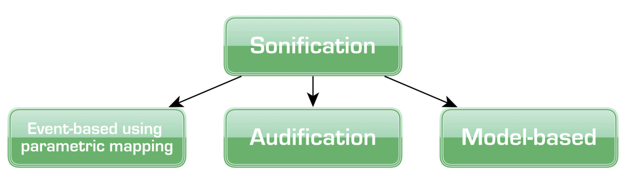

Figure 2 – The three sonification modes according to most literature

DeCampo himself makes a slightly different categorization:

1. *Sonification by Continuous Data Representation*: using an equal distance on at least one dimension and a high enough sample rate so that interpolation between data points becomes useful. Audification and parameter mapping can use continuous data representation.

2. *Sonification by Discrete Point Data Representation*: every data point is tied to an individual event.

3. *Sonification by Model-Based Data Representation*: a model, whose properties are controlled by the data, is used.

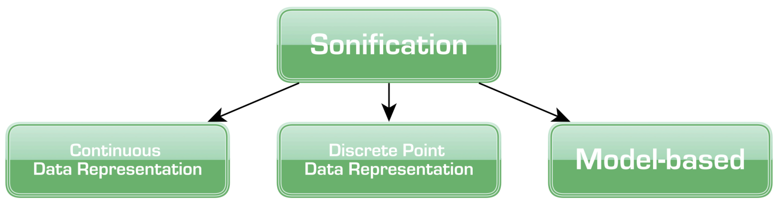

Figure 3 – De Campo makes a slightly differently categorization

(Worrall, 2009a) also distinguishes between parametric mapping and audification but adds *homomorphic modulation sonification*. This kind of mapping similar to parametric mapping but with the difference that while in parametric mapping, the amplitude goes to 0 between every event/data point, in homomorphic modulation mapping, the amplitude points of the events are connected so that a continuous pulse wave arises. The resulting absence of rapid onsets (as opposed to parametric mapping) can have a lower auditory load on the listener. The best way to describe this is a gliding colour scale: all individual colours are present; however, they smoothly morph over into each other so that we cannot distinguish discrete colours.

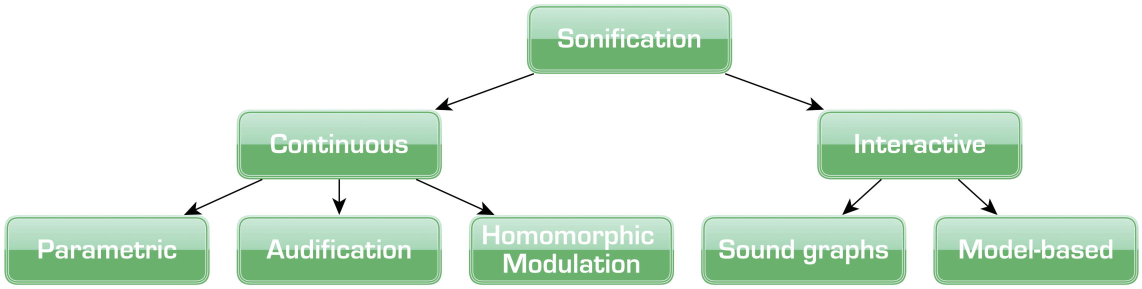

Figure 4 – Sonification modes according to Worrall

In the interactive division, Worrall distinguishes between sound graphs and model-based sonification. Sound graphs are navigable spaces where the user can navigate freely, in a non-linear fashion to hear the data of his choice.

Each of the sonification modes has its peculiarities, advantages and disadvantages. Aesthetic choices will play a significant role in choosing the sonification mode. Once the mode has been chosen, one can choose a *mapping technique*: what conditions will one set to map the data from one domain onto another? Indeed, to map a value we need to set a condition. For example: a value of X will yield a mapped value of Y, a value of 2X will yield a value of 2Y *et cetera*. The process of choosing this mapping technique can be considered to be empirical: we test possible outcomes and through observation we choose the one that best fits our purpose (whether that be scientific or artistic). This aesthetic experience is informed by earlier experiences and learned conventions. This will have an influence on our thinking about mapping, hence we will return to discussing mapping later on in the dissertation (see 1.6).

While the sonification modes can exist as categories, there are no exact borders and we can use multiple modes. For example, (de Campo, 2008), in creating his sonification map, lets the different modes spill over in each other. One example was discussed above: audification can share parts with parametric mapping. Indeed: in audification we still control the amplitude of the wave through mapping a data point.

After having explained the technical side of sonification modes, we can now try to embed sonification in the compositional practice.

1.2.2 Sonification and Composition

As *manifestation* (the term posited by Polansky) can be related to musical composition, we will focus on the exploration of the artistic possibilities of sonification in the context of auditory display. In fact, both manifestation and algorithmic composition use numbers. Indeed, music always involves numbers somehow; notes, rhythms, tempos et cetera can be expressed in numbers. As such, it is a logical step to let those numbers be controlled by external data, be they created through mathematical processes or through observations of a phenomenon. To use these data, we need to drive them through a mapping process. Thus we can see that both algorithmic composition and sonification rely on numbers and on a translation process. Nevertheless, there are differences that we need to discuss in order to understand the premise of sonification.

Indeed, algorithmic music relies on an extra-musical process that is being sonified through mapping the data into musical parameters. This practice is not unique to 20th and 21st century music. Warren claims that in the 14th century Guillaume Dufay used the proportions of the Santa Maria del Fiore cathedral in Florenceto to control the tempi in his motet *Nuper Rosarum Flores*[2] (Warren, 1973). In his *Musical Dice Game* Mozart used two matrixes of 8 by 11 cells. By throwing dices, a series of numbers is created. These numbers correspond to the 196 cells in the matrixes, each containing a musical fragment. Each unique series results thus in a new musical piece. Although algorithmic, Mozart still adhered to the tonal doctrine. In the twentieth century, with fewer restrictions, more and more composers started using algorithms in their music.

Many types of algorithmic music exist. (Roads, 1996) divides the field in stochastic and deterministic music while (Maurer, 1999) distinguishes between stochastic and rule-based music. In stochastic music, the composer uses a series of numbers generated by a computer. The composer still has quite some liberty in mapping the numbers to musical parameters. This is less the case in rule-based/deterministic music where rules are

---

[2] However, Fallow (Fallows, 1987) and Turner (Turner, 1991) contend that Dufay never used the proportions as a control parameter. Wright in continuation of the other critics states that Dufay used the biblical passage Kings 6:1–20, which gives the dimensions of the Temple of Solomon as 60 x 40 x 20 x 30 cubits. Freire (Freire, 2014) totally refutes Warren's claim and proposes a new rational relationship of 3:2:1:1. Putting the controversy aside, the idea of using architectural proportions to control musical parameters is interesting.

established before the composition takes place. The best-known example of rule-based music is probable the treatise on counterpoint *Gradus ad Parnassum*, written by Johann Joseph Fux (Fux, 1965) .

Supper makes a more refined distinction of rule-based algorithmic music types:

1. the modelling of traditional compositional processes, not controlled by algorithms;
2. the modelling of new and original compositional processes, different from existing ones;
3. the use of algorithms through extra-musical processes. (Supper, 2001)

Examples of the first type are SpeciesChecker (McKay, 2002), an application to write *cantus firmi* for species counterpoint and Choral (Ebcioglu, 1990), a program that harmonizes chorales in the style of *Bach*. *Illiac Suite* , created by Lejaren Hiller and Leonard Isaacson in 1957, is an example of a composition that used algorithms to model traditional compositional processes (Hiller & Isaacson, 1957). A composition of the second type is *Çogluotobüsisletmesi*, by Clarence Barlow in 1978.

> "The metric and harmonic intensity, rhythmic and melodic uniformity, chordal density, density of attack, dynamics, and articulation were calculated according to a composed meta-structure: the algorithms. The actual piece is only one of many possible realizations (Supper, 2001)".

The third type uses extra-musical processes to generate material. These processes are for example Lindenmayer systems. Developed by Aristid Lindenmayer in 1968, a Lindenmayer system or L-system is a parallel rewriting system and a type of formal grammar. Lindenmayer used L-systems to describe the behaviour of plant cells and to model the growth processes of plant development.

The system consists of an axiom and rewriting rules, for example:

axiom X

Rule: X--> XYX

If we apply this rule to the axiom, we eventually see the following pattern emerge:

X
XYX
XYXYXYX
XYXYXYXYXYXYXYX
XYXYXYXYXYXYXYXYXYXYXYXYXYXYXYX

An interesting characteristic of L-systems is the auto-similarity that occurs during the recursions of the system. Various composers have used L-systems, such as (Hazard, Kimport, & Johnson, 1999) and (Sodell & Soddell, 2005).

What type of algorithmic music is *manifestation* (using Polansky's term) now? On one hand it takes an ever-changing input, which we cannot replicate when taking a new selection of data. Obviously, there are recurring patterns but there are always subtle differences. It is impossible to encode these patterns, to set rules for their evolution. In that aspect, we should consider manifestation a form of stochastic music. However, this is not entirely true. Although we cannot be sure what is going to be the next data point, there exist patterns, which make it different from random datasets. We are talking about an evolutionary system (type 3 in rule based music) but we cannot encode this evolution, it is an ever-changing body of data. Especially in real-time musical works, it is difficult to create predetermined rules. A solution is to use interactive, dynamic rules, which the user can manipulate in real time. This helps him or her to adapt to new situations. We see the following problems that impede us from placing sonification sound art in either stochastic or deterministic music. These problems are:

1. The use of real data as opposed to mathematical models.
2. Although it is difficult to predict, there is no randomness in the data. Patterns emerge but we can only see them after the data has been 'created'. There is a certain degree of information entropy.

As we cannot place this practice in neither stochastic nor deterministic music, we believe that we thus need to add a third category to algorithmic composition: sonification art.

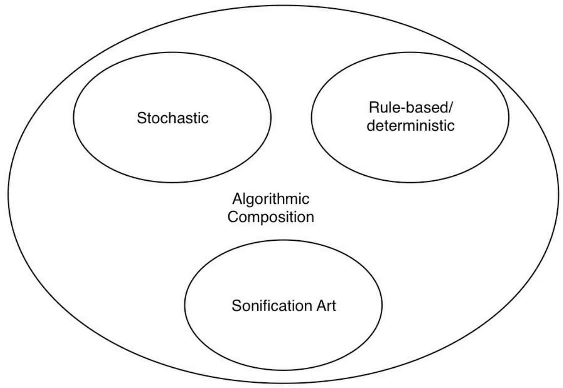

Figure 5 – A third type of algorithmic composition

1.2.3 Locating Sonification Art on the Map

Having compared separately the types of auditory display and algorithmic music, it is now time to bring them together. Where does sonification art find itself on the map? If we look at the auditory display types explained in the first section, we see that the arts and entertainment are applications of sonification. Algorithmic composition in itself is a subset of the arts, but not all algorithmic music is part of the sonification field. If we visualize this, we get the following figure.

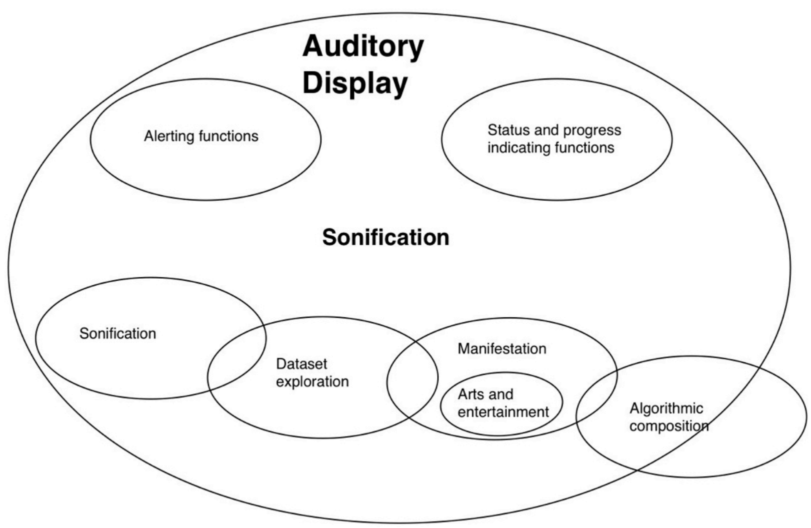

Figure 6 – The place of algorithmic composition on the map. Sonification in an artistic context is located in the common sector of Arts and Entertainment and algorithmic composition.

1.2.4 Towards a Definition

In the preceding sections, we discussed the difference between science and art as the goal / as main driving approaches to data. One element, which is essential to artistic practice remains to be discussed, is the appropriation by the artist of the data implying creative and subjective aspects. As the artist uses data from observations of world phenomena and maps them onto musical parameters, it is possible that he or she will not like the whole result. He can than decide to make alterations in the places that are not to his liking. For example: a line that is going up with one value drop can result in a broken melody, which might not be desirable in the composition. Thus the composer may decide to alter the value to make the melody continuously rising[3]. This results in a re-interpretation of the data: *sonification as a creative trigger.*

Creativity is an essential element in sonification art. In our definition, we will use Polansky's *manifestation* term to refer to the creative element in sonification. We will discuss the creativity aspect more in-depth later on in this dissertation.

At the end of this section, we can finally propose a definition for sonification in an artistic context:

> *Sonification Art is an arts practice that uses data from observations of world phenomena, which are mapped onto musical parameters through fixed or dynamic mapping procedures as an essential input element and uses sound to manifest itself.*

1.3 Data, Their Mapping and the Sounding Result

We can roughly divide a sonification process in four stages:

1. Data fetching;

---

[3] One example of local decision-making can be found in Xenakis' composition *Herma*. In discussing inconsistencies in the score, Xenakis admitted the ear as one of three elements (the other two being slips of the pen and theoretical or computational errors) that could explain these discrepancies (Vriend, 1981) and (Montague, 1995). Xenakis himself writes on this subject: "At the service of music, as well as of every creative human activity, scientific and mathematical thought has to be amalgamated dialectically with intuition" (Xenakis, 1976a).

2. Data reading;
3. Data mapping;
4. Making the data sound.

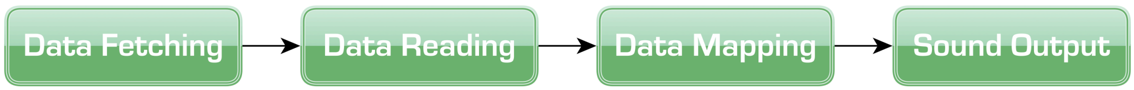

Figure 7 – The four stages of a sonification process.

When we talk about sonification, we first have to take a look to our source material, namely the data. We can basically take any available dataset and sonify it. These data come from various areas such as seismic data, weather data or financial data. Many of these data sets can be found online, either for free (like data.gov: a myriad of datasets from the United States Federal government) or for a charge (Xively.com and Bluekai.com offer access to several data streams))[4]. In a first stage we have to look for a dataset that is interesting enough to sonify. This means that we need a sufficient dynamic system to study. Not all data show interesting dynamics. For example: the birth rate of the Dodo in the last 200 years has been stable during all that period. Thus it comes as no surprise that a sonification would be redundant[5] for most people. On the other hand, in a total random number series, we would be confronted with such a sheer amount of data that it becomes impossible to handle. Hence, it is paramount to choose a good dataset to get started.

The second aspect, equally important as to choosing a useful dataset, is the way we are going to store and read those data. First we have to choose if we are going to use a live stream or a recorded dataset. Both have their advantages and disadvantages. When using a livestream, we are confronted with the unexpected: we can never really know what will happen next. (Devisch, 2008), in his study on urbanism, calls such behaviour *organized complexity*: through their various interactions, the constituent elements create a distinct macro-behaviour, generating recognizable patterns or shapes. Unpredictability does not mean that the data are random: we simply cannot know the future completely. Only after

[4] A quite large list of websites to get data can be found on http://www.kdnuggets.com (KDNuggets, 1997)

[5] I originally used the term *interesting* here. This term would however imply a value judgement. Maybe somebody could find such a sonification interesting. Redundant seems then a better term to sue.

the fact we can assess the data to our assumptions.

Regardless of using historical data or real-time data, we need to create a sufficiently interesting mapping method to create a compelling sonification. We can make this interaction happen on two levels:

1. During the conceptual stage, where the artist tries out different mapping methods.
2. During the showing of the finalised artwork, where the user can interact with the system, within the limitations set out in the conceptual stage.

When using a recorded dataset we can read the same dataset various times and improve our mapping method through multiple iterations. In either case, we can choose to let the user interact with the data and mapping or not. This interaction happens in a different stage than the conceptualization stage.

Whether the work is interactive or not and whether we use real-time data or a recorded dataset, the goal is to make the artwork a compelling[6] experience. If we want the user to listen for extended periods to the sound, we need to pay attention to the aesthetical element. Kramer cit in (Barrass & Vickers, 2011) writes: "Improved aesthetics will likely reduce display fatigue. Similar conclusions can be reached about the benefits of a composer's skills to making displays more integrated, varied, defined, and less prone to rhythmic or melodic irritants".

The fourth stage is the making the data audible: how will the data sound? It is up to the composer to choose his instrument: he can choose to create a score to be performed by acoustic instruments or can map the data to synthesis parameters. The composition itself is not music but "a command structure for the production of sounding music" (Mazzola, Park, & Thalmann, 2011). The interpreting musician[7] can than make personal choices when reading that command structure. The four stages described above do not constitute a linear process: indeed, if we want to, we can go back to the previous stage and make modifications to the process. Hence, process and result become intertwined. Of course, it is up to the public to make the final interpretation: every listener will hear the same

[6] I use the term *compelling* to avoid the charged term of *pleasant.*

[7] Obviously, in a fixed electronic composition a computer will not make interpretational decisions. Nevertheless, the tape or CD remains a command structure to be interpreted in order to get a sounding result.

composition differently. This interpretation however is not part of the compositional process.

1.4 Coming-into-being and Creative Processes in Sonification Art

"Ideas alone can be works of art; they are in a chain of development

that may eventually find some form. All ideas need not be made physical."

(LeWitt, 1971)

The rise of conceptual art in the twentieth century has shifted focus from the artefact to the process of coming-into-being: how the artwork is created is at least as important as the resulting artefact. (Danto, 1998) proposes the term *posthistorical* to describe the period roughly from 1950[8] until now in which “art was no longer possible in terms of a progressive historical narrative” and exemplifies this with Andy Warhol’s *Brillo Box*. Originally designed as a box to hold scouring pads, Warhol transformed the mundane object to an artwork. Indeed, in *Brillo Box*, we came to see that the conceptual part was more important than the result itself. Warhol detached an object from its intended meaning and gave it a new meaning as an artwork[9]. In sonification art (or in algorithmic art in general), we detach the data from their original (informational) purpose and attribute a new meaning. For example: whereas stock market data serve to inform traders on the markets movements, by sonifying them to use in an artwork, we detach the informational aspect from the data and we interpret the data differently to serve our artistic purpose. This does not mean that the artwork does not refer anymore to the informational aspect (for example: one can use climate data in an artwork to make people aware of climate evolution); the data get a new significance assigned. Furthermore, when the data are in mapped to sound, they go through a processing stage. This process is highly conceptual; as the artist needs to think how he is going to make the mapping that best conveys his artistic message. Hence, process

---

[8] Of course Duchamp comes already 40 years earlier with his *Fountain* work, a repurposed urinoir.

[9] Looking at the object was not suficient to grasp its meaning: the audience had to grasp the concept behind it as well. This was clearly not the case when Canadian customs officials considered the work to be merchandise instead of an original sculpture and wanted to levy a customs duty when the artwork was imported in Canada. (Danto, 1998)

becomes paramount and creativity manifests itself on different levels of the artwork.

How can we dissect the creative process? (Mazzola et al., 2011)[10] traces this process back to a cycle of seven steps through which an artists moves, constantly revising his work. These seven steps are:

1. Exhibiting the open question: We start our quest with asking what we want to investigate.
2. Identifying the semiotic context: The question does not come alone. We need to find the surrounding context to be able to solve our question.
3. Finding the question's critical sign or concept in the semiotic context.
4. Identifying the concept's walls; What is limiting
5. Opening the walls
6. Displaying extended wall perspectives
7. Evaluating the extended walls

By going through these steps, the artist constantly rethinks his work. As he gets better acquainted with his work by questioning it constantly, he is able to create the most compelling work. Of course, at one point he will end the process but the goal of these seven steps is to go beyond one's (imagined) limits. This process is not linear: the artist can go back and forth in this series. The question is if we can apply this way of questioning on sonification art. Before we do so, however, we first have to investigate the ontology of data in order to gain a better understanding of our source material.

[10] Many theories on the creative process exist and it would take a separate book to discuss them all. In researching the creative process in music, Mazzolla's theory on musical creativity seemed interesting as it was focused on music; hence I chose to develop my argument through his theory. This does not mean that other theories do not have a value; I merely made a choice in a broad spectrum of theories.

1.5 An Ontology of Data

Raw data is an oxymoron

(Bowker, 2008)

When discussing sonification, we have to be aware that the sounding matter is just the artefact at the end of a chain of operations. In order to understand the concept of sonification better, we thus need to disseminate the different parts that lead to that sounding artefact.

The common assumption would be to consider the data as our starting point. This however is erroneous: Data have to come from somewhere, there has to be a causal event to trigger the data: without cause, there are no data. Data thus do not come up out of nothing[11], the data are related or can be related to a causal event. These causal events can be anything one wishes to investigate. It is there that we make our first choice: in choosing the event we want to investigate, we limit our search space. As such, we are not talking about the raw everything, but about a specific event/phenomenon. In the preceding section we divided the sonification process in four action stages. However, in order to come to the data fetching stage, we need to have a causal event. From that causal event we can distil data. The word is derived from the Latin *datum*, a past form of the verb *dare* (to give). As such, *datum* means 'something given'[12]. (Merriam Webster, 2014) defines data as:

1. Factual information (as measurements or statistics) used as a basis for reasoning, discussion, or calculation <the *data* is plentiful and easily available — H. A. Gleason, Jr.> <comprehensive *data* on economic growth have been published — N. H. Jacoby>

[11] Indeed, it would be erroneous to assume that data come out of nothing. (Bogost, 2015) contends that the rise of *Big Data* has lead to a kind of data-divine: whereas in the past we believed in a God, todays religion seems to be data-based. This is a misassumption: "data is [sic] created, not simply aggregated, and often by means of laborious, manual processes rather than anonymous vacuum-devices". Even with a natural causal event such as earthquakes, the data-collection is a process initiated by a human. The data do not come together themselves.

[12] This does not contradict the earlier statement that data has to be created: the causal event (whether it is natural or man-made) can be measured in an infinite number of ways. The event thus gives us data that we filter to get our desired dataset. (Butler, 2010), writing about friendship visualizations on Facebook, writes: "Visualizing data is like photography. Instead of starting with a blank canvas, you manipulate the lens used to present the data from a certain angle". (We can substitute visualization with sonification).

2. information output by a sensing device or organ that includes both useful and irrelevant or redundant information and must be processed to be meaningful;
3. information in numerical form that can be digitally transmitted or processed.

While the premise of the Merrian-Webster definition is promising, the inclusion of the word information makes it incorrect: Information comes out of data. In order to get information, we need to interpret the obtained values. Vesna (Vesna, 1999) writes “Data is the raw form that is shaped and used to build architectures of knowledge exchange and as an active commentary on the environment it depends on—the vast, intricate network with its many faces.” Another, not so subtle citation comes from Clifford Stoll: “Data isn't information, any more than fifty tons of cement is a skyscraper” (Stoll, 1995). We should distinguish carefully between the building element and the building itself.

Taking in consideration the above remarks we can improve the Merrian-Webster definition as follows:

> *Data are values output by a sensing device or organ that must be processed to be meaningful.*

Although the data may seem raw at the beginning of the study of them (as it is not information yet), they have been filtered already: Through inclusion/exclusion of certain data, the set is interpreted for a first time: the data collector attributes relevance to certain data while deeming other data of none importance. Daniel (Daniel, 2007) writes: “A collection [of data] is produced through processes of selection and differentiation— sorting, classifying, rejecting anomalies—making patterns”. Data thus become non-neutral: even before starting our argument, we have already attributed a certain degree of meaning to the data. The obtained dataset is subject to a second filtering: we read (interpret) the data and irrelevant data remain unused, they do not go to the next stage. After the filtering stage, the remaining (relevant) data are ready to be processed. At this stage, the rhetoric comes into full play: we want to tell a story with the data. Hence we need to find a mapping method, which conveys our message. Here the artist leaves his position as a technologist[13] and has to make artistic, subjective decisions. These decisions can be located in the rhetoric system. Through rhetorics, we attribute meaning to the data.

---

[13] It would be incorrect to call the technologist position a purely objective one: through the choices the technologist makes in including/excluding data, he occupies a non-neutral position.

The next step is to transform data into information. How do we get from data to information? Let us first define what information is. (Rosenberg, 2013) makes a clear distinction with *fact*, derived from the Latin verb *facere*: 'that which was done, occurred or exists'. *Fact* also contrasts with *evidence*, coming from the Latin *videre*, to see. Rosenberg points at the important distinctions between the three terms:

> "Facts are ontological, evidence is epistemological, data is rhetorical. A datum may also be a fact, just as a fact may be evidence. But, from the first vernacular consideration of corresponding, the existence of a datum has been independent of any consideration of corresponding ontological truth. When a fact is proven to be false, it ceases to be a fact. False data is data nonetheless."

We can thus say that data receives its meaning through argument. It has no significance outside a rhetoric system. As rhetoric can change over time due to historical-contextual developments, the data can take on another meaning as well. (Rosenberg, 2013) writes: "… the meaning of data must always shift with argumentative strategy and context — and with the history of both." Thus, data as such mean nothing but a stream of values: we construct a truth or reality with the data through rhetoric. If we apply this to data used in composition we can say that data put in the context of a composition means contextualizing it within the compositional argument set out by the composer: data get significance through the composition. We can visualize this flow as follows:

Figure 8 – The causal event gives birth to data, which have to be interpreted to become information.

In transforming data into information, we need to interpret them. We do this through a mapping process. This mapping process has some philosophical fundaments which we have to explore in order to understand this concept better.

1.6 Beyond Mapping: Artistic Issues

In section 1.2, we discussed sonification modes, which was a technical oriented discussion. In this section, we want to explore the philosophical concepts behind mapping.

As stated above, data has to be interpreted in order to become information. The interpretation happens through a mapping process: we translate the data, turning them explicit. (Daniel, 2007) defines mapping as follows:

> "Mapping is intersubjective communication: the visualization or representation of data and information. The term "map" applies both to a clear representation, one capable of communicating intersubjectively, and the act of analysis required to create such a representation. A map has no single author. To map is to locate, to assign a correspondence. A map fulfills the functions of both record and statement—it is a history of the subject's, or mapmaker's, relation to that which is mapped and an act of communication with others who will interpret and use it. To map is to locate—but position is always "relative to . . . " associative and perspectival. Intersubjective communication occurs when the meaning of data or information is accessible to, or established for, two or more subjects. In intersubjective communication, values and truths are inseparably intertwined. Interpretations and representations are produced dialogically—in cooperation with a "text" or data set."

Mapping implies thus that everything is relative to something: we cannot see a data point in isolation; we have to consider it within the dataverse. This does not mean that the dataverse has to be considered in its totality: we can take a portion out of it and use that as our relational frame.

This definition takes into account that we have two sides to the story: not only the data matter, the interpreter matters as well. As stated before, in sonification there is a substantial difference between sonification and manifestation (the terms posited by Polansky) that we can now correlate to pragmatic and artistic sonification strategies. In fact, referring to data visualization, Manovich states that there are two options to visualize, that we can adapt to sonification as well. He makes a distinction between pragmatic and artistic visualization, stating that on one hand, there is pragmatic visualization, on the other hand artistic visualization. The pragmatic visualization is meant to be useful: making the unrepresentable representable. This is also called the visualization of the anti-sublime. The artistic visualization is made in the first place to think, to reflect about the subject. If the visualization is useful, that is just a mere side effect. The main goal of artistic visualization is "to map such phenomena into a representation whose scale is comparable to the scales of human perception and cognition". The artistic visualization can then be called the visualization of the sublime. The visualization of the sublime has a unique role: to show us other realities embedded in our own, to show us the ambiguity always present in our perception and experience, to show us what we normally don't notice or don't pay attention to (Manovich, 2014). Whereas Manovich speaks of the sublime, Xenakis compares the 'discovery' of the non-scientific a *revelation*: outside the scientific realm, we gaze with awe to what we discover, we do not know what constitutes it, yet we are struck by it.

When mapping data, we are confronted with the abundance (or even overload) of possible mapping choices: Why do we use the mapping we used? Certainly with the computer we can easily map data and try out different combinations. In doing so, and in continuation of the importance of process, it becomes an interesting element to discuss, to motivate why we use a certain mapping above another. It is not bad to have chosen an arbitrary mapping as long as we can motivate why we chose that mapping and not another. This does not mean that there has to be a degree of efficiency: it simply shows us why we deem one mapping to better convey our message than another. One can, simply said, map everything to everything. So why do we choose a certain mapping? These choices are embedded in the whole of the artwork: what do we want to tell and how do we want to experience what has to be told?

1.7 Storytelling with Data

After having defined sonification art, its source material (the data) and mapping, the next question that arises is why an artist would use sonification or: what makes sonification an interesting venture for the composer? We can distinguish three reasons:

1) The composer wants to convey a message related to the data through his music.

2) The artist believes that one can gain a deeper understanding of the data through the artwork.

3) The data could form an interesting source for creating compelling art. For example: a highly dynamic stock market could be translated as a wildly moving melody.

The first reason is not that far removed from programmatic music: the data serve as a source guide to the message. Like Berlioz' *Symphonie Fantastique* that conveys a love story or Debussy's *Sunken Cathedral* conveys an ancient Breton myth, one can choose sonification to convey a certain meaning that goes beyond the music. One cannot deny the influence of the dataset as a political motivator: the artists can try to transmit a message. One can use climate data to bring a message how climate change is affecting the world. Polarseeds, which sonifies climate evolution (cfr infra), is an example of transmitting a message through art. This does not mean that the artist takes a stance on climate change: he transmits the data in a way the audience can experience a phenomenon in a different way. (McKinney & Renaud, 2011) in discussing *Leech*, which sonifies a peer-to-peer network write:

> No judgement is passed on the activities, legal or not, of the pirates, record companies, or musicians involved. Rather, the goal is to illuminate an often overlooked and dismissed part of modern day culture, and to spark a dialog about the nature and value of music and sharing.

On a more political field, Hashimoto in sonifying nuclear tests (the project will be discussed in the following section), wants to let people think about the "extremely grave, but present problem of the world." Whereas McKinney and Renaud take on a distanced position, Hashimoto takes a more direct stance. His objective is to let people not only think about the testing itself but about the implications of nuclear testing. This way, he makes his objections against nuclear weapons explicit.

In discussing the second reason, (Straebel, 2010) suggests that the idea of gaining a deeper understanding "seems to be derived from certain concepts suggested by early romanticism". The author refers to a "*Natursprache*, a poetic language in which one could directly experience nature as the embodiment of a divine being". Indeed, just as programme music wants to convey a narrative, sonification can serve as a messenger of an extra-musical meaning.

With sonification, the representational or the objective observers' point of view becomes a fundamental part of the work or as Straebel writes: "The representation aspect was internalized in the composition itself." Instead of referring to nature by imitating the sounds (for example a flute melody refers to a birdsong), the composer referred to the nature on a different level by using algorithmic procedures. This does not mean that the composer has to reveal his sources: he can choose to obfuscate it and merely give a suggestion of it. In such a way, the music can be compared to pure poetry: while the content is present, pure poetry is concerned with the intrinsic value of pure structure (Keane, 1986). Xenakis, in discussing his work *Pithoprakta,* speaks of *logical poems*

(Xenakis, 2001). The goal is not so much to “translate data into musical parameters, but rather emphasizing the connection between the arts of music and mathematics[14] as established by ancient Greek philosophers and the theorists of the Middle Ages” (Straebel, 2010). Steve Roden (Roden & Polsenberg, 2004), in discussing his *Eart(h)* installation work says the following on this matter:

> "Just as the installation would look and sound different if the data source was the ocean or Shakespeare; ear(th) in no way attempts to illustrate the earth's movement or to recreate an earthquake experience through sound. I am much more interested in simply allowing the earthquake data to generate a sound composition and to allow for my own misreadings of the data to suggest placements, sound ideas, performances, and sculptural forms. For me, this process is a kind of alchemy - to allow the materials to be transformed into something completely connected to, yet seemingly distant from, the source".

Through the narrative system of sonification, the composer thus creates his story and tries to transmit a compelling story.

If we take a close look to the second reason, the belief that one gains a deeper understanding of the underlying data, we can make an extension to Levi Strauss who compared myths to music in his book *The raw and the cooked* (Lévi-Strauss, 1964). Both share the aspect of unfolding over time: to understand them, one has to listen to the progression of the story. In sonification the dataset has to unfold over time. Thus, to grasp the meaning of the data, one has to listen to the artwork unfold over time. The listener needs to experience at least a significant part[15] of the panorama to be able to fully understand the complete story.

The third reason is clear on itself: the numbers that constitute the dataset can contain interesting patterns of organisation and thus it can be interesting to project those intricacies on a musical surface. These data streams are not just random numbers: they embody an intricate structure of patterns that arise during the course of the tracking of the causal phenomenon. Vesna (Vesna, 1999) writes: “Artists working with the net are essentially concerned with the creation of a new type of aesthetic that involves not only a visual representation, but invisible aspects of organization, retrieval, and navigation as well”.

[14] Although Xenakis speaks of mathematics, we can easily open up this statement to sonification.

[15] I use the term *significant part* as it is uncommon to be present at an installation work during the whole period it is being exposed.

Manovich, talking about data-visualization (Manovich, 2014), continues on this line: to show us other realities embedded in our own, to show us the ambiguity always present in our perception and experience, to show us what we normally don't notice or don't pay attention to.

1.8 Sonification as Soundscape

We can find an interesting parallel between sonification and soundscape composition. In both approaches we are taking an external phenomenon and use it as a defining component of the work. Sonification in that point of view only differs from soundscape composition in the fact that there is an additional layer between the source and the final sound: the data have to pass a mapping stage before becoming sound objects[16]. Furthermore, just as with natural sounds, a data point can never be replicated in its context: although we could have the same value returning, the values around it (context) will be different. In the sonic result the artists can strive to totally detach the sound from any association with its source (Shaeffer contended this approach is his sound-objects theory). For example: through layering various recordings one can make the sound unrecognizable. In sonification we obviously do not have a sonic starting point. However, the composer can use sonification and obfuscate the nature of the data in the program notes. Hence, there is a detachment between data and sounding artefact.

1.9 Technology in Sonification Art

Technology plays an essential role in sonification art. It is linked with process, which in turn stresses the idea of concept. As Artur Danto said on the famous *Brillo Box*, it is not only the artefact that counts but also the concept behind it. While the artefact looks like a normal product container, in its recontextualization it becomes a work of art. The concept-behind is an important factor in the existence of the artwork. In contemporary

---

[16] (Gomes et al., 2014) does try to break the border between data and sound: The URB project records sonic data such as the centre frequency and overall loudness. By combining the recording of the sonic landscape with the data extracted out of that recording, one can intertwine reality and interpretation.

music we can see a parallel: while John Cage's *4'33"* would be nothing more than a 'live field recording', contextualized in a concert setting it becomes an artwork. Among other notable examples that explore the conceptual path we can find the Fluxus event scores.

But between concept and artefact stands the process. More and more composers treat the composition process as an inseparable part of the artwork: how the score comes to be is equally important as the sounding artefact itself. Often the composer explains these processes in detail. One of the most prolific composers to do so is probably Iannis Xenakis, explaining in detail[17] his methods to create his music. It is without a doubt that the processes used by Xenakis and others used technology extensively to help them in their quest for the composition.

In the process, we can also see the importance of technology in contemporary composition. Over the last decades, the computer has become ubiquitous as a tool to assist in the compositional process. While Xenakis performed his first processes by hand like in *Achoripsis*, he saw the potential of the computer to perform the calculations and find unsuspected, surprising musical material that he could not have thought of in a traditional way. Herbert Brun's (Brün, 1970) summarizes this spirit succinctly:

> "Whereas the human mind, conscious of its conceived purpose, approaches even an artificial system with a selective attitude and so becomes aware of only the preconceived implications of the system, the computers would show the total of the available content. Revealing far more than only the tendencies of the human mind, this non-selective picture of the mind-created system should prove to be of significant importance".

Similarly, Berg sees three reasons to use the computer: "to hear that which without the computer could not be heard; to think that which without the computer could not be thought; to learn that which without the computer could not be learned…" (Berg, 1979).

We can thus see that the computer is considered of great value with composers: the computer helps to uncover possibilities not thought of by the human mind. Furthermore, it can speed up processes that would take much time or even be impossible when done manually.

---

[17] Although Xenakis wrote down his methods in great detail in *Formalized Music*, Pierre Schaeffer criticized him for his obscure explanations: "As far as Xenakis is concerned, let me emphasize at once that I'd be much more interested in his research if he hadn't set out so obviously to reduce its accessibility and its credibility in a manner which is immediately apparent as soon as you open his book on formal musics." (Schaeffer, 1970)

Using algorithms to create musical material does not mean that the result is some random gibberish: “The output is not simply random, freely interpretable “raw data”—rather, it constitutes a notion of a material that is prefigured in the design of the system through which it is realized” (Hamman, 2004). Effectively, composition becomes then a form of system design:

> In creating the artwork, technology is used throughout the process. We can ask ourselves then: does technology influence the outcome of the artistic process or is the output determined? Feenberg does not agree with the view that technology is neutral or deterministic. A common misconception is to place the finalized artefact to a neutral, deterministic frame, as if the technology itself did not influence the way the artefact became-to-be. If we only consider the artefact, we are missing out an important aspect of the work. Maeda says: "When the output of a computer in the form of traditional visuals is seen, it is trivialized because it is only one facet in that conceptual space" (Hackworth, 2014).

Technology is not deterministic: it plays an important steering role in the work. In creating an application for compositional purposes, it becomes thus a challenge to guide the composer but leave enough space to lie in his own creative mind. A push-the-button-and-you-get-music application is not the way to go as it would restrict the user too much in his interaction. There will be always a limitation in using/interacting with a certain application but these restrictions can be used at the advantage of the artist. Reiterating Mazzola’s idea of breaking the wall, the artist should question the limitations of the technology and encounter a path that suits his ideas. Last, we should not forget that the computer is essentially a number-crunching feature: it cannot interpret freely; it needs instructions from a human being. As such, it cannot compose autonomously: the composer has to create a set of rules to which the computer must adhere. As said earlier true openness does not exist: there will be always borders present. The objective of the composer is to use those walls to his advantage. Reiterating the number-crunching feature of technology, the discussion between Messiaen and Xenakis is especially interesting: In his doctoral defence, Xenakis argued with Messiaen about the use of the computer to get his material. Messiaen states that without knowing all permutations of a melody, he cannot love and choose the one he loves best. Xenakis, in response, states the following:

> "When I look at a starry sky, I love it in a certain way because I know it in a certain way; ... Consequently, I can handle the concepts of things themselves without being in direct possession of them, under the condition that I may conceive of them and feel them from within in some way.... [E]ven if I am incapable of dominating a certain phenomenon, I am capable of obtaining a truth which is inherent to the conceived or

> observed phenomenon, thanks to a kind of immediate revelation. Henceforth, I can accept and use this, in and as itself." (Xenakis, 1976b)

From this quote we learn that the apparent lack of total knowledge does not impede the composer in pursuing his endeavours. Indeed, the computer can help us in revealing something we do not see in clear daylight. One can compare this to looking through a prism: when we let light shine through the prism, it is revealed that the light consists of multiple colours. Thus, although we have no intimate knowledge about the observed phenomenon, the computer can be an asset in composition.

1.10 Different Aesthetics = Different Structures?

Exploratory sonification can serve different purposes: either a scientific or artistic goal can be the point of focus. One question that arises is: does the goal have an influence on the aesthetics of the sonification? To answer this question, we briefly return to the definition in posited in section 1.2. Both in science and art, *the sonification process is the rendering of (typically scientific) data into (typically non-speech) sound designed for human auditory perception*. This definition does not impose an aesthetic ideology: the data should be audible for the listener, how those data sound is part of the follow-up question. Nevertheless, in scientific sonification, we want to make the information understandable for the user. Making it understandable does imply that the listener should be able to grasp the informational meaning that was given to the data earlier on. For example: in sonifying stock market data for use in trading, we want the listener to perceive trends, rise or falls in the stock price. The goal is utilitarian and the sound should be as clear as possible. In an artistic context, this criterion falls away and the composer has more freedom in his aesthetic choices. In creating an artwork, the artist tries to convey a story of some kind, whether that is abstract or concrete, there is a message involved in each artwork. Even in the most minimal[18] artwork one can find a message. For example, although the score of Cage's *4'33"* does not command any sound, the events surrounding the artwork, make the artwork present. Another work where the absence of obvious content makes the actual artwork get its presence is +/- by Rioji Ikeda. A high frequency sine wave is played at a very loud level.

[18] *Minimal* should not be confused with repetitive music.

However, the high frequency makes the sine wave inaudible for the listener. As such, when the piece ends, the sound pressure falls away. It is at that moment that the listener can frame the work. Silence becomes a frame[19].

To create an artwork, one needs to structure his story. In visual arts this can be a diagonal axis upon which elements are connected. In classical rhetorics the order of elements was always fixed and built for dramatic effect. In music we dealt for a long time with arsis-thesis gestures, which drive higher-level forms like the sonata form or ABA structures or even arch forms. In these structures, i.e., homeostatic structures, the goal is to regain stability (eg. Tonic – Dominant (unstable) – Tonic (stable) movement). By mapping one phenomenon onto another, the traditional musical forms cease to exist or at least they appear by accident: Instead of imposed structured forms like ABA-forms, the nature of the data guides the progress of the resulting artwork. It is without doubt that the resulting structure does not adhere to traditional archetypes where the arsis-thesis pair is present such as the sonata but more likely a linear[20] structure. (Pizzi, 1998) calls this heterostatic: hierarchical structures and functions disappear to give way to different situations. This does not mean that behaviours typical of homeostatic systems cannot appear: repetition, tension-resolution can arise without destroying the heterostatic nature of the work. The difference is that these homeostatic elements will appear as a residual consequence, not as a fundamental structural part. Klein describes this as follows:

> Data are also filled with an unmistakable absence. Data cannot "conclude" a story; they cannot deliver a "suspense" ending, like a murder mystery—not in the traditional way (and I am not convinced that interactively choosing your own adventure solves this problem, even with high-resolution effects). Data are part of a process *without* an arc that requires a dramatic ending. Instead, they proceed by insinuation, by involution—toward a *beginning,* toward an *aporia* (the road without a name). That kind of journey can be extremely charming, like "a making-of" that is so *Massive* that it does not even require a movie. (Klein, 2007)

Another interesting point of view comes from Paul Butler. In discussing the visualization of friendships on Facebook he writes: "Visualizing data is like photography.

[19] Of course we can see this as well in more traditional music: the silence before the piece is being played makes it clear that the piece is a separated entity from its surrounding sounds.

[20] Although there are works that counter this linear approach, for example *Leech* (discussed in 3.1.4)

Instead of starting with a blank canvas, you manipulate the lens used to present the data from a certain angle". (Butler, 2010) There are two important elements in this citation:

1. The artist starts with a given material. One can compare this to a sculptor: out of a rock, he removes unnecessary elements to come to his final work.
2. When using a camera, one can only capture a part of the world, not the whole reality. The same goes with sonification: In using a dataset, we only use a small part of reality to interpret. Korzybski's: "the map is not the territory" (Korzybski, 1941) or Bell's (from who Korzybski took his inspiration) "the map is not the thing mapped" statements are useful in this context: When mapping we take only a part of the phenomenon and of that part we only take certain aspects to map. For example: when tracking *Amazon*'s stock between 2010 and 2014, we would not take in account the rest of the time *Amazon* was listed on the market and we would only take some parameters like the stock price and volume, but not intraday data.

As the outside world controls an important aspect of the sound, one can find a great deal of parallels with soundscape composition: we are confronted with an ever-changing sound palette. The sound will always be different. The Heraclites saying "No man ever steps in the same river twice..." perhaps conveys best this line of thinking.

Returning to storytelling, Barthes writes: "The text is a tissue of quotations drawn from the innumerable centres of culture... the writer can only imitate a gesture that is always anterior, never original" (Barthes & Duisit, 1975). In sonification this becomes very clear: one uses a pre-existing event and uses that to build his story. As such, the author becomes a storyteller, a mediator. To tell that story, we use a dataset; a collection of previously gathered data. To tell our story, we navigate through this database (Bachmann, 1973) and try to find the most compelling thoroughfare (he does not look for the most efficient one, although the most efficient could be the most compelling as well) for our voyage. Interestingly, the idea of the navigator is used also (albeit slightly differently) by Xenakis (Xenakis, 2001): "With the aid of electronic computers the composer becomes a sort of pilot sailing in the space of sound, across sonic constellations and galaxies that could formerly be glimpsed only in a distant dream".

1.11 Systematization in Sonification Art

As stated before, algorithmic composition does not produce random gibberish. The systematization of he compositional process allows a structure to arise. Herbert Brun goes as far to say that the creation of a system is the only way to make something significant: "For anything to be of relevance to something, to be of significance to someone, a system has to be created; an artificially limited and conditioned system has to be imagined and then defined." *Only artificial systems will clearly show that they have been elected by choice*... ." (Brun, 1969) Systematization and the subsequent limitation are thus paramount to create a meaningful work. Krenek[21] on this matter says:

> "I was fascinated by the notion that music was not a vague symbolization of 'Gefühl' [emotion] instinctively conjured up into a pleasant sounding matter, but a precisely planned reflection of an autonomous system of streams of energy materialized in carefully controlled tonal patterns."

Even in so-called random composition such as in Cage's *4'33"*, systematization is present: Cage chose to make a three-part composition, performed in the centre of New York. Had he chosen to have the piece performed in a rural town in the midwest, the resulting sounding artefact would be quite different. True randomness or true openness thus does not exist; borders are always present[22]. In Bourdieu's theory of the field of cultural production we can see this as well: in the big search space that constitutes all possible compositions, the composers takes a position from which he works. A dissonant voice comes from Boulez (Boulez, 1986b): "[Invention] should not be satisfied with a raw material come upon by chance, even if it can profit from such accidents and, in exceptional circumstances enlarge upon them" (Manning, 2006). Boulez omits here the limitations inherent to systematization: the material is not raw; it is prefigured through the systematization of the process. We can see similarities with the ontology of data: *there is no such thing as raw data*. However, if we consider Boulez' opinion in a later stadium of the compositional process, his thought becomes legit again: the composer should not be happy

[21] Krenek read Ernst Kurth's *Grundlagen des Linearen Kontrapunkts*, a book on the rules of counterpoint. Here he found out that this seemingly simple music was constructed as a planned complex of notes.

[22] João Castro Pinto (amongst others) addresses this subject in (Pinto, 2011)

with the output of his prefigured system[23]. The composer should assess the results and, if he deems it necessary, make changes to the output. Boulez' statement should thus be seen in a fear that the technical would be confused with the aesthetical. In a study by (Eaglestone & Clowes, 2001) the general consensus among composers is that "composition is a mix of intuition and methodology." The composers refer to the famous neutrality issue (posed by Heidegger on technology in general): software cannot be neutral. In some way it steers the composer to an outcome because of its inherent limitations. The idea of total freedom is fantasy. The composer has to keep these limitations in mind in order to not lose himself in the techno-complex.

Making the process explicit does not necessarily mean that an aesthetic experience arises; we believe that that is exclusive to the artefacts. This does not mean that an aesthetic experience cannot arise during the creation process: fragments of the work in progress are artefacts and can as such arouse aesthetic experiences. The artist acts as an interpreter of his own work. Morris (Morris, 1970) however contends that during the creation process 'forms of behaviour' arise: "These are forms of behaviour aimed at testing the limits and possibilities involved in that particular interaction between one's actions and the materials of the environment". We should consider the making explicit as a revelation of information.

The system is not determinative: it can change during the creation process: "… subject and object become mutually determinative": through the compositional steps, the composer changes his mind. By changing his mind, he changes also the way the composition progresses. Di Scipio states the dilemma or the two questions that composers face when confronted with technology:

> 1) How can I use the available existing task-environment to realise my own ideas of composition?
>
> 2) How can I design the tools that are necessary to realise my own idea of composition? (Di Scipio, 1995) as cited in (Manning, 2006)

---

[23] Barbaud and Babbit however are more strict in this point of view: they contend that for the sake of consistency of their composing methods, the computation had to be accepted as such. (Hoffmann, 2009). Boulez (Boulez, 1986a) in contrast stated that integral serialism raised a formalist utopia without direct effectiveness. More general, Williams believes that since the 1960, there has been a steadily flexibilisation in the structural obsessions and that by the 1980's this obsession had almost disappeared (Williams, 2011)

This dichotomy is central in ‘technological’ composition. Roads and Straw give voice to some Utrecht school composers who plea for a digital aesthetic, which is idiomatic for the computer. In their point of view, technology can be shaped and we should shape the artefacts to be unique to the computer. Koenig seems to go the farthest in this:

> “Primarily, I’m very annoyed with composers using the most modern tools of music making, ... and making twelve-tone series for instance, or trying to imitate existing instruments. That has, of course, its scientific value, but not necessarily a creative value in new music making.... So, ... to open up new fields of sounds you would not be able to produce... in classical terms, I have chosen [a] non-standard approach [to sound synthesis]” (Koenig & Roads, 1978)

Culkin contends that “we shape the tools and in turn they shape us” (Culkin, 1967)[24]. We can thus see a bi-directional process of determination: In order to find new elements, one creates a tool to mine those elements. The way we work with those tools will eventually change our way of working.

Thus, technology shapes the compositional pretext but the composer wants to break new ground, something he would not have thought of had he not used a computer. Hence, he will adapt the tools to his will and the resulting tools will in turn reshape the work he is working on. This is before starting to write the composition, it is at the start of the conceptual groundwork for the composition.

Reiterating Feenberg’s view that technology is not neutral, technology and artwork become mutually determinative and the essence of the work should thus be looked for *“in the activity of making”* (Hamman, 2004).

If we see the artefact as the only constituent of the artwork, we disembody the object from its technique. Splitting the artefact from the process thus takes away a big part of the artwork. Whereas in other productions of artefacts the technique is being hidden and only the end results counts or is showed to the public, in art-making the artefact is that which *arises within* its technique. Hamman writes: “the focus of interaction shifts from a concern for the heuristics of the result, toward investigation of the means of it’s own production. The end result reflects a development both of the material and of the thought by which it is conceived and shaped. Such a shift in emphasis penetrates the technological

[24] This quote has been misattributed to McLuhan hemself. The quote is from Culkin however. McLuhan used the quote in a recording in 1967 (McLuhan, Fiore, & Agel, 1968). Nevertheless, the quote is consistent with McLuhan’s thought.

with the aesthetic—a joining of techne and poiesis.

Hamman, probably using Heidegger's philosophy as a pathway, sees art since the 1950's with the primary objective of engineering breakdowns. Heidegger tells us that through a breakdown[25], we are confronted with the true nature of the artefact; we go back to the roots. Art since the 1950's, in its engineering of breakdowns gives thus primacy to process. Instead of hiding the technique, it is brought to the foreground as an essential part of the final artwork.

Although many of the above-mentioned composers use the unexpected as a source, this does not mean that they abolish their final control: the composer is and stays the responsible for the material. He serves as an evaluator of the output and puts that output in the order he wishes. Let us briefly return to Mazzola's idea of the wall. Hamman considers that composition from the twentieth century (and we can extend this to the twenty-first century as well) is empirical: through observation and experimentation one comes to the destination composition.

> "Technology preserves the problematic of compositional process rather then attenuating it, while the technical thing is transformed from an object for the social mediation of cognitive and epistemological activity to an object through which humans explicitly and experimentally participate directly in the shaping of that activity". (Hamman, 2004)

This empirical idea is parallel to Mazzola's idea discussed earlier: by cycling through the seven steps, the composer comes to new understandings and finally to his desired result. Like Mazolla, Vaggione sees walls in (algorithmic) composition: By setting constraints, the composer constructs reflecting walls. I believe that the concept of reflection is particularly interesting with regard to Mazzola's breaking walls: While Mazolla focuses on breaking the walls, Vaggione sees the reflection of the walls as an element of inspiration: through the collisions with the wall, the ideas get a renewed freshness and can be refocused upon returning to their starting location.

Creativity is being researched extensively and many interesting definitions arise. Discussing and comparing them would take a thesis on its own[26], hence I took the elements

---

[25] Maeda's statement is interesting as well in this optic: "If someone has After Effects on Flash they can produce what I've made here, but it's flawed in a way only I know about. Flawed and thus human. My mind blood is in there" (Hackworth, 2014)

[26] For a thorough dissemination of creativity theories I refer to the The Cambridge Handbook of Creativity.

I deemed most interesting for use in my dissertation. Sternberg and Lubart posed the investment theory of creativity: Comparing the artist to a stockbroker, the artist is willing to buy low (investing in an unpopular idea) and sell high (earn respect for the idea). The authors define creativity as "the ability to produce work that is both novel (i.e. original, unexpected) and appropriate (i.e. useful, adaptive concerning task constraints)" (Sternberg & Lubart, 2010). To do so, one has to use divergent thinking: a problem can lead to many answers, allowing innovation to occur. This way of thinking is opposed to convergent thinking, where only one answer is formulated and as such does not allow for unexpected results to occur. In a study by (Eaglestone & Clowes, 2001) the general consensus among composers is that "composition is a mix of intuition and methodology." The composers refer to the famous neutrality issue (posed by Heidegger on technology in general): software cannot be neutral. In some way it steers the composer to an outcome because of its inherent limitations. The idea of total freedom is fantasy. The composer has to keep these limitations in mind in order to not lose himself in the techno-complex.

1.12 Context and reception

As the artwork as artefact is exposed or mediated to the audience through performance, the context plays an important role in the evaluation of the artwork. We cannot see the artwork as an isolated object that exists without the environment and the audience. Goodman (Goodman, 1999) gives the example of a rock: While a rock in nature is just a natural object, put in a museum space, it takes on a different meaning. The object put in the museum context becomes subject of reflection, of thought and can arouse an aesthetic experience. The initial meaning is disturbed and in this disturbance the artwork gets its significance. Cage's *4'33"* is no different in this perspective: taking the music out of the concert hall and putting it in an urban setting makes the observer reflect on the status of the artwork, not hearing noise as residual waste from an urban environment, it becomes object of contemplation.

This leads us to our next point of interest: the audience as an interpreter. Duchamp said that an artwork was not complete without the observation from the spectator. Without a public, the artwork cannot come to fruition: the object is there but the aesthetical experience is absent. Only when he observes experiences the artefact, the artwork comes to complete fruition. This vision comes as a complement to Hume's idea of beauty: "Beauty is no quality in things themselves: It exists merely in the mind which contemplates them; and each mind perceives a different beauty." (Hume, 1965)

To come to the aesthetic experience, the artwork has to be interpreted. If we continue on Mazolla's conceptualization of creativity, we see in the artefact (be that a score or a recording) a command structure to be interpreted. While in autographic arts (to use Goodman's term[27]), there is no need for a mediator, in allographic arts like music the performer does the first stage of interpretation after which the listener can interpret the sounding artefact. Only at that moment, the artwork reaches its completion. This implies that an artwork is inherently different at every moment in time and for every observer-interpreter. The command structure concept applies not only to symbolic, score music but to electronic music as well: the information contained on the medium (CD or other media) has to be interpreted (through a DA-converter) in order to get to the sounding artefact. From there on, the listener can then make his or her interpretation and subsequent aesthetic contemplation. In the interpretation, art cannot be seen independent from its direct context (the place it is experienced in) but also the socio-historical complex it finds itself in: Petersen, based on a reading of Dewey's *Art as Experience*, argues ". . . that art is not an abstract, autonomously aesthetic notion, but something materially rooted in the real world and significantly structured by its socio economic and political factors". (Petersen, Iversen, Krogh, & Ludvigsen, 2004)

Apart from the actual context during the experience of the work, earlier experiences have an influence on how we perceive the work. Formal (for example art school) and informal (for example social interactions) influence our interpretation of the work (McCormack & Dorin, 2001). A blank, neutral viewpoint is thus impossible. For example, history plays a role in interpreting the artwork. For example: Shostakovitch's 7th Symphony became a symbol of resistance against the Nazi besiege of Leningrad and is today

---

[27] Goodman contends that "a work of art is autographic if and only if the distinction between original and forgery of it is significant; or better, if and only if even the most exact duplication of it does not thereby count as genuine arts". (Giovannelli, 2010) In music or performing arts in general, the interpretation of the work from a score does not constitutes a forgery yet an instance of the same work.

considered a heroic work. Without the war, we would experience the work differently, although the artefact stayed essentially the same. Furthermore, the learned musical conventions, inherent to any culture, undoubtedly have an impact on the perception and appreciation of the artwork. Using extra-musical processes, we will create musical artefacts that are not completely tied to the learned conventions. Chord progressions, which we consider natural in the conventional system, are replaced by new compound sounds, which may not sound 'natural'. This change can be problematic: we need to learn the new conventions. If we are not open to new experiences, we cannot appreciate new music and embed the new experiences in our cultural thought-system.

Barthes' thought, although it focuses on text, is essential to underscore the postulated comments: "a text consists of multiple writings, issuing from several cultures and entering into dialogue with each other, into parody, into contestation; but there is one place where this multiplicity is collected, united, and this place is not the author, as we have hitherto said it was, but the reader…". (Barthes, 1967) We can project this onto the listener: only in his mind the music unfolds in its totality. This does imply that we can hope to convey an aesthetic experience but we cannot impose it: we cannot force somebody to think that something is beautiful. The quote from Tarkovsky perhaps best conveys this thought: "we cannot impose our experience on other people or force them to feel suggested emotions. Only through personal experience we understand life" (Tarkovsky, 1984).

1.13 Summary

In this chapter we described the elements that constitute sonification and more specifically sonification art. In fact, we discussed systematization, the role of technology and the context. The practice was embedded in a theoretical framework encompassing data, the mapping process, the compositional process and the interpreter and a definition of sonification art was proposed.

Even if one could criticize sonification as a process that disembodies the composer from his work, I think that we show in the preceding sections the undeniable interest of this art form: as shown through different composers' and theorists' comments, it can offer new pathways to musical invention.

Hence, having established a ground for sonification art, we can now look at practical implementations: artworks that use sonification as an important constituent of their being.

2 AN OVERVIEW OF EXISTING SONIFICATION ART

In this chapter a selection[28] of sonification artworks is discussed. It serves to show what has been done and what possibilities exist. The discussed projects use different data sources and data mappings. Even if many projects are multimedia artworks, I want to focus on the variety of used sound mappings and hence the non-sonic components are not discussed in-depth. The projects where enough information about mapping strategies was available (either through papers, webpages or personal communication) are included in this document while other interesting projects from which more accurate information on mapping methods was unavailable are discussed solely on the research blog. This allows the reader to focus on mapping strategies while for discovering projects he can visit the blog. The projects ordered according to their data source and their presentation is followed by a general reflection about the used techniques and backgrounds of the artworks.

In looking at sonification works we have to take in account that the subject matter, the data, unravels over time: we need to track a certain phenomenon over a period of time. The time resolution is important here: at what sample rate will one track certain data and is it useful to track at a certain resolution. Taking in mind that an infinite resolution is impossible, the composer has to decide with what sample rate he thinks that the observed phenomenon will be interesting to observe and what is practically feasible[29]. Scale becomes paramount in deciding how to deal with the phenomenon to observe.

First we have to set our range: what timespan will we observe? If we are going to observe climate evolution[30], we need to observe large timespans. If we would like to observe climate evolution since the last ice age, we have to track a period of roughly around 13000 years[3]. In such a large timespan we are confronted with two intertwined problems: the availability of the data and the sample rate to use: we need to determine what

[28] In no way this selection is exhaustive and at the moment of writing, new projects are appearing. Therefore, new projects are gathered on the blog sonificationart.wordpress.com as a complement to this thesis.

[29] This is not to question Di Scipio's dichotomy: the composer can choose to work with existing tools or develop his own to get his desired result.

[30] I use the word evolution to avoid the word politically charged term "change".

data are available: in the case of tracking the climate evolution in the past 13000 years, it is possible to extract data from drilled ice cores such as the ones drilled during the Greenland Ice Core project (European Science Foundation, 2000).

Next, we have to assess the data on their potential: what is the maximum sample rate that we can use or what is the smallest chunk of time for a data point? Once we know the potential, we can set our own sample rate: we are not obliged to sample at the highest rate: we need to determine the rate at which the reading is going to be interesting to use. Returning to the climate evolution example: if we want to see a global evolution over the 13000 years[31], sampling every day would give us a dataset with 4.748.250 entries. Maybe it can be more interesting to sample every year or even every century. This allows us to get a dataset that is easier to handle and avoid unnecessary details. Of course, this is related to what a human can perceive: It is impossible to map 13 million years of data in a 1:1 ratio. We need to find a way to make the data perceivable for the audience or as Manovich says on data-art: “we need to bring the data to a human scale”. (Manovich, 2002)[32]. Of course one mapping is not the only correct one: one can choose to try out different sample rates in order to find the most interesting result.

Lets explain the above with another example: when looking at stock prices, we might want to study the long-term evolution over a period of a year. However, as some stocks are traded heavily, their prices can move a great deal during the day. As such, it would not help us in looking at a long-term evolution of the stock. Therefore, we could chose to track only the daily closing price and still get an adequate representation of the stock’s evolution.

The data sample rate has to be related to the sounding artefact: how are the data going to be used? Whereas in a visualisation the data can come to us in an instant, the sound needs time to be perceived and understood. In this light it is interesting to see how sonification deals with this unfolding: whereas we could display a time series in one image,

---

[31] The last ice age happened approximately 110,000 to 12,000 years ago

[32] This view contradicts with Cage’s conception of time in *Organ2/ASLSP* (As slow as possible), a composition with no specific tempo. One of the interpretations is happening in Halberstadt, Germany where the tempo is set so that the piece lasts 639 years, which is outside the human scale. (Cage, 1987). Another example of this out-of-human-scale pieces is *Soundtrack for an Exhibition* (Stenger, 2006) in which a normal pop song is played at a tempo so that it lasts 96 days (the duration of the exhibition).

no matter how short the dataset, we need time to expose it. If one set one note with a duration of one second for every data point, then listening to a very large dataset would take a very long time which could be undesirable for the artist/ listener (of course Cage (Cage, 1987) challenges this limit in *Organ2/ASLSP*).

The elements discussed above show us that the data have an influence on the coming-into-being of the artwork: the criticism that one could use any dataset interchangeably is thus a fallacy. Just as data influences the artwork, the artwork influences what data we will use: when preparing, 'designing' our artwork, drawing out the plans, we set rules, constraints on how we want our artwork to sound. If we map a value to a frequency we can set he input and output ranges to adhere to certain limits. The criticism that data-driven music sounds random is thus proven an incorrect one. Hence, while data (the content) has an influence on form, structure will determine the interpretation of the data. Content and form become thus mutually determinative.

Summarizing: each dataset has its own peculiarities and needs to be examined prior to transforming it in sound.

2.1 What Data are Used and What Data Not?

In order to perform good research, we need to limit our search space. In selecting projects, the data source was an important element to limit the search space. I used the nature of data as the limitation criterion: all project are *data-based*: the data have been collected and can be presented in a table as variables. Card, in an opposing viewpoint, defines this as *abstract data*: "… information – such as financial data, business information, collections of documents, and abstract conceptions – that, unlike scientific data, does not have any obvious physical space to map to". Aside from the fallacy that certain observations are not scientific while others are[33], Card's definition represents a too narrow viewpoint for artistic applications: the artists can use scientific as well as non-scientific data. Either way, the data are a set of variables that are disembodied from their

[33] Indeed: In distinguishing between scientific and non-scientific data Card makes a value judgement. What is scientific to one, is not scientific to another person. However, just as with the data/fact dichotomy (see 1.5, an observation, being scientific or not, is still an observation.

causal event. Hence, the term *databased art* seems more adequate: the artworks discussed here use databases (either fixed ones, with historical data, or dynamic ones, with real-time data).

2.1.1 Projects Using Environmental Data

The earth offers a great potential for sonification. The constant shifting of tectonic plates, volcanic eruptions and climate evolution all offer an exciting body to work with. As such, it comes as no surprise that many projects use the earth's intrinsic dynamics as a source material. Helped by available datasets mainly used for scientific purposes, there lay big possibilities for using them in the arts.

2.1.1.1 The Climate Symphony (2001)

Marty Quinn's *The Climate Symphony* uses data extracted from a 3053 meter ice core drilled up in Greenland (Quinn & Meeker, 2001). Through the chemical composition of the ice core one is able to analyse the climatological evolution. Quinn uses the top 2960 meters of the ice core, which amounts to 110000 years of climate history. Out of eight chemical time series, three were extracted through principal component analysis[34]. Series 1, the "Polar Circulation Index" describes the atmospheric circulation response to the growth and decay of the continental ice sheets while series 2 and 3 describe the biological response of changing climate. The patterns that are present in the data offer a great opportunity for a composition. Quinn superposes the various patterns and maps them to different instruments (played by a synthesizer). The data are read with a speed of 150 years per second for the first 20000 years and 350 years per second for the latter 90000 years. As such, 110000 years of climate evolution is compressed into a 7-minute composition.

The mapping is done as follows:

1. The 550-year solar intensity cycle is mapped onto a scale of melodic patterns played on a vibraphone. The hotter the sun, the higher the melodies are and vice versa.

---

[34] Principal component analysis is a method to describe a multivariate dataset using relevant parts of the data.

2. The ebb and flow of the ice sheets happens in 6300-year cycles. These movements are mapped onto 3 rhythmic patters of cowbells for expansion and 3 rhythmic patters of tom toms and conga for contraction. The amount of expansion/contraction controls which of the patterns is played.
3. Volcanic activity is sonified through cymbal crashes and timpani whose volume depends on the intensity of the eruption. The pitch of the timpani gets lower with a higher intensity as well.
4. The earth's wobble causes some summers to be hotter and some colder. Quinn maps this influence on the lower two octaves of the organ; the bigger the influence, the higher the pitch, the lower the influence, the lower the pitch.
5. The earth's tilt is sonified through a 3-note arpeggio where the notes are higher if the tilt is bigger. The instrumentation (clarinet, trumpet or muted trumpet) is chosen according to the 1450-year ocean cycle.
6. The slight change in the elliptical orbit around the sun transposes all music up or down up to seven steps depending on the increase/decrease of the elliptical shape of the orbit.

Quinn creates a multidimensional structure where 5 datastreams are sonified and a sixth one manipulates those streams with transpositions.

2.1.1.2 Polarseeds (2012)

Polarseeds, created by the Polar Seeds Group, takes data from a melting Glacier in Greenland and created a series of sonification miniatures presented below (Tedesco, Ham, Perl, & Saltz, 2012). Whereas Quinn's Climate Symphony takes on 110000 years of climate evolution, Polarseeds sonifies data between 1958 and 2012. The sonifications are split up in two parts: one part uses annual smoothed[35] and stepwise averages; the other part uses daily data.

[35] Smoothed means that the values ramp gradually from one data point to the next. In stepwise mode, the value jumps from one to the next without a ramp.

Albedo[36] takes the smoothed summer albedo averages and applies them onto a lowpass filter. A single bass choir drone is played and the albedo value controls the lowpass of the filter. The lower the albedo, the deeper and darker the drone sounds. The subsequent sonifications build upon this piece.

In *Albedochoir versus Meltrate Geiger*, the Albedo sonification is accompanied by the average summer melting rates, which are sonified by a Geigercounter effect: the gigher the meltrate value, the faster the Geiger counter.

In *Albedochoir with temperature filter/reverb/ exiter effect*, the the smoothed summer averages are used to control filter parameters over a ‚swooshy' sound.

Albedo choir vs Melt rate Geiger vs Temperature Filter combines the three sonifications, with the Albedo choir panned center, the Geiger Effect panned left and the swoosh sound panned right.

Summer Melting sonifies annual summer melting values. The value determines the intensity of random note selection of pre-programmed rhythm parts: the higher the melting value, the more intense notes are being selected. Unlike in the Albedo sonifications, the values are not smoothed.

Summer is essentially the same as *Summer Melting*, with the exception that annual air temperatures are sonified. An extended piece involves four groups: Rhythm, melody, arpeggiated and orchestral instruments. Each MIDI-track contains a series of notes and for the first three of them; the probability percentage with which a note will be played depends on the (unsmoothed) melt value. Thus: if the value is high, the more notes will be played. The notes themselves are randomly chosen from the MIDI-track. The fourth group has its volume controlled by the (smoothed) melt rate. All four groups are linked to an exciter, which generates partials depending on the smoothed summer melt value. The same mapping technique is used in another piece with annual air temperature as the data source instead of the melting value.

[36] Albedo is the reflection coefficient of a surface. The higher the value, the more reflective the surface is.

The second group of sonifications uses daily data. In *Daily Melting*, a low-pass filter filters a sustained 9-interval chord. When the melting value rises, we hear more of the sound. As melting occurs only during short periods, we actually hear mainly silence interspersed with short soundbursts. In a variant of this piece, the cumulative daily melt is used. As a result, the melt activity is more perceptible and easier to compare with other years. Whereas in the previous piece, we heard the individual daily values, we now hear an average.

In *Daily Albedo*, albedo values are mapped to 128 possible steel drum pitches and then fitted within a scale. Volume is inversely mapped so that lower albedo sounds louder. Furthermore, the application of a distortion effect and a high pass filter intensifies the dark character of the low steel drum sound: the lower the pitch, the bigger the distortion and the more high frequencies pass.

Daily temperature uses the same mapping process as *Daily Albedo*, although here, pitches are constrained to the nearest notes of a Bb minor 7 chord. An FM synthesizer sound is used instead of steel drums. Temperature changes also influence the attack time of the modulation envelope and the frequency modulation amount.

In a cumulative series, the Daily Albedo sonification is combined with each of the other sonifications individually and then all three sonifications are combined.

To make trends clear, sonifications with a slower playback speed and with years filtered are created.

The Polarseeds sonifications are part of a larger exhibition to make people aware about climate evolution in Greenland.

2.1.1.3 Matanuska Etude (2004)

In *Matanuska Etude*, Mara Helmuth uses data from sediment granulation in a lake formed by the melting of the Matanuska glacier in Alaska during a 24-hour period. Time, grain size and grain frequency were measured and mapped onto parameters for waveform, additive, granular and physical modelling synthesis. (Helmuth & Davis, 2004)

By mirroring the diurnal cycle of d10 grain frequencies (the frequency of the smallest 10% of grains), Helmuth created a waveform period, which produced a complex set of partials for drones.

Sediment grain frequency data was mapped to partial frequency and grain size to duration. In a further phase, Helmuth mapped sediment grain size to partial frequency and grain frequency to amplitude, which resulted in slowly changing events.

For granular synthesis, Helmuth mapped sediment grain frequency to grain rate and grain size to grain frequency. Another approach was mapping sediment grain frequency to grain frequency and grain size to grain duration. Finally she used *Stochgran* (a granular synthesis application) to generate events of grain distribution where the frequency value was derived from sediment grain sizes and rates were derived from sediment grain frequencies and variations on this method.

Helmuth used the *STK Synthesis toolkit* to create stochastic sounds like multiple-attack shaker instruments such as maracas. Grain frequency was mapped to the number of objects, grain size to energy, with all the 22 instruments playing at randomly varied time about 10 times per second. The gradual ascents and descents in energy and density result in a pulsing texture.

On a macrolevel, grain parameters were mapped to event density on the two dimensions. The increasing frequency of grain events is mapped to more additive synthesis layer and higher grain rates near the end of the *Matanuska Etude*. There are also more stochastic sounds present near the end of the piece.

2.1.1.4 Massachusetts Geophonic (2013)

In *Massachusetts Geophonic* Arvid Tomayko uses data from the USGS Massachusetts geological map as a palette on which the user can place markers to 'travel' over the map (Tomayko, 2013). The map covers the following data:

- Bedrock Type
- Bedrock Age
- Faults

- Surface Covering
- Surface Covering Depth
- Gravity anomaly (which is a measure of the density of rock beneath the surface)

By placing one of the four paths on the map, a playhead will transverse the map. At each change of the data, the corresponding musical parameter will change. For example: if we map bedrock type to pitch, at every place where a different type starts, the pitch will change. Alternatively, the user can set up rhythmic patterns. These rhytmic patterns are a division of the spatial path (and thus playtime) in a regular interval. For example: if the looplength is set at 16 seconds and we set the division at 8, the sample will be triggered every 2 seconds. Rhythm thus becomes spatial.

The user can set general parameters, such as the master loop length (how long it takes for the playhead to traverse the path), the melodic scale, the key and the overall transposition. These general controls can then be transposed in each of the four voices. For example: one can transpose voice 1 up two semitones and voice 2 up five semitones. Tomayko sets up fixed mapping destinations.

- Rock Age determines pitch, and also when to play notes in “contacts” mode. Musical scale and tonic are user-defined. Through the curve value, one can set the exponent of the Rock age against the pitch.
- Rock Type (sedimentary, metamorphic or igneous) determines timbre, or instrument type, for each note:
    - Sedimentary rocks are noisy, percussive sounds
    - Metamorphic rocks are a saw wave
    - Igneous rocks are represented by a resonant filtered oscillator.
- Rock chemistry (mafc to felisc) determines note overdrive
- Grain size determines tremolo or modulation for sedimentary rocks. This is represented as different shades of blue on the Rock Type map.
- The metamorphic grade of a rock (represented as different shades of green) determines granular pitch randomness.

- Faults produce record skip-like sounds whenever a playhead crosses one.
- Surficial Geology (the surface “stuff” that covers the bedrock) determines the length and feedback of a delay or echo.
- Lastly, isostatic gravity anomaly determines the attack and decay characteristics of the notes.
- Areas outside the state or in the ocean will not produce sound.

Tomayko’s work should be seen as an artwork and not as a software instrument. While it could be included in the software section, the closedness of the work in terms of sound generation puts it in the realm of an interactive artwork.

2.1.2 Projects Using Biological Data

2.1.2.1 Life Music (1999)

Biological data such as DNA sequences and protein structures have been used in quite some projects to create music. One example is *Life Music* by Dunn and Clarke. Using Dunn’s *Artwonk* software (see section 2.3.5), they created genetic music using amino acids. The proteins show an internal organisation not unlike musical phrases and themes. It came thus as no surprise that there was a musical potential

The DNA-coding consists of four sub-units T, C, A and G (thymine, cytosine, adenine and guanine), which constitute the helix of a DNA-string.

> These four coding elements are combined into groups of three called codons. There are 64 possible codon combinations, of which 61 are used to encode the 20 amino acids [only 20 are unique] and three are “stop” codons that indicate the end of a protein sequence, just as a period indicates the end of a sentence. (Dunn & Clark, 1999).

In earlier experiments, Dunn mapped the protein amino acid data directly to a fixed pitch and in another experiment, using histograms; he assigned more consonant intervals to the more frequently appearing acids and less consonant to the less frequently appearing ones. Clarke introduced a third method to better accentuate the acids properties. First, they were ordered on their water solubility: the least soluble acids were assigned the lowest pitches on a three-octave diatonic scale, a two-octave chromatic scale and four octaves or whole-tone and pentatonic scales. To allow aesthetically pleasing harmonies to occur, acids with similar R-groups (R-groups being the ‘pendants’ of the protein ‘necklace’) were placed at consonant intervals. This harmonic placement proved to be fruitful. The protein chain winds in and out of the interior and two contrapuntal melodies arise. The sonification of the different acids resulted in distinct compositions, which were easy to distinguish one from another.

2.1.3 Projects Using Astronomical Data

2.1.3.1 Solar Wind Sonification (2010)

In 2010, a team from the university of Michigan used solarwind data from 2003, obtained using NASA’s Advanced Composition Explorer Satellite. Composer Robert Alexander used these data and created *Solar Wind Sonification* (Alexander, 2009). To create the sounds, he started by mapping Helium velocity data to a frequency between 20Hz and 6kHz. The resulting frequency was used to control the cut-off frequency of a bandpass filter filtering pink noise. The bandwidth was held at a constant of 1. The result is a sweeping wind sound. The density of the Helium controls the loudness of the sound. At some points in time, a Coronal Mass Ejection[37] occurs. When this happens, the sound is send through an overdrive effect.

Alexander used the carbon state charge distribution to control the amplitude of 6 vocal layers. This creates an ambient atmosphere. The value average of the charge state is sonified by voices in a higher register.

The HE/O (helium/Oxigen) ratio is represented through a chord built out of triangle waveforms in a high register (the corresponding MIDI values of the frequencies are 108, 103, 98, 110, and 105). The ratio is linked to the amplitude of the chord.

[37] A coronal Mass ejection is the burst of solar wind and the magnetic field into space. This phenomenon is responsible for the aurora borealis and aurora australis.

In the global mix, the Coronal Mass Ejection also controls a reverb. When a CME occurs, the reverb will go up fast and recede slowly to the base level, dramatizing the effect of the CME. (Alexander, 2012)

The data are read with a speed of one rotation every 8 measures and a bpm of 150. The result is a three-minute composition.

2.1.3.2 Earth's Magnetic Field (1970)

In 1970, composer Charles Dodge, together with three physicists Bruce R. Boller, Carl Frederick and Stephen G. Ungar, sonified the variations in the earth's magnetic field

(Somerecords, 2007), (Dodge, 1970). Solar Winds make the earth's magnetic field fluctuate and these fluctuations are registered in the KP index[38]. This index consists of 28 possible values and every three hours, a new value is registered by the system, which results in 2920 values for the year 1961, which was used in the composition. These values were tabulated in a so-called Bartel Musical Diagram[39]. One of the physicists had made a five-line staff representation of the data and mapped the values to both a 7-note diatonic scale as well as a 12-note chromatic scale.

As the pitches were given, Dodge focused on working on the rhythm and timbre. The 2920 values were compressed in an 8-minute composition within those 8 minutes Dodge used algorithms to organize the rhythmic values. Shields writes: "In the first half of the piece, there would be accelerando-ritardando patterns; in the second half, a fixed tempo within which two patterns, A and B, would alternate, the A pattern having one note to a beat, and the B pattern 2 to 14 notes to a beat." (Shields, 1997) The B-pattern was derived

---

[38] The KP-index is a global geomagnetic storm index with a scale of 0 to 9. The Kp-index measures the deviation of the most disturbed horizontal component of the magnetic field on fixed stations worldwide with their own local K-index. The global Kp-index is then determined with an algorithm that puts the averages of every station together. The result is the global Kp-index. The Kp-index ranges from 0 to 9 where a value of 0 means that there is very little geomagnetic activity and a value of 9 means extreme geomagnetic storming. (SpaceWeatherLive, n.d.)

[39] The Bartel music diagram takes its name from its appearance, which is not unlike a musical score. Without much hassle, one can easily compile a musical score out of the graph.

from the sudden commencements, rises in the values due to solar winds and flares which had a bigger impact on the earth's magnetic field.

Besides its importance in sonification, *Earth's Magnetic Field* is an important piece in the electro-acoustic music scene as it is the first piece that explicitly uses comb filters to control the timbre. Dodge used the comb filter in the first part while in the second part he used allpass filtering. The use of filters is a pure aesthetical choice while pitch and rhythm are dependent on the data.

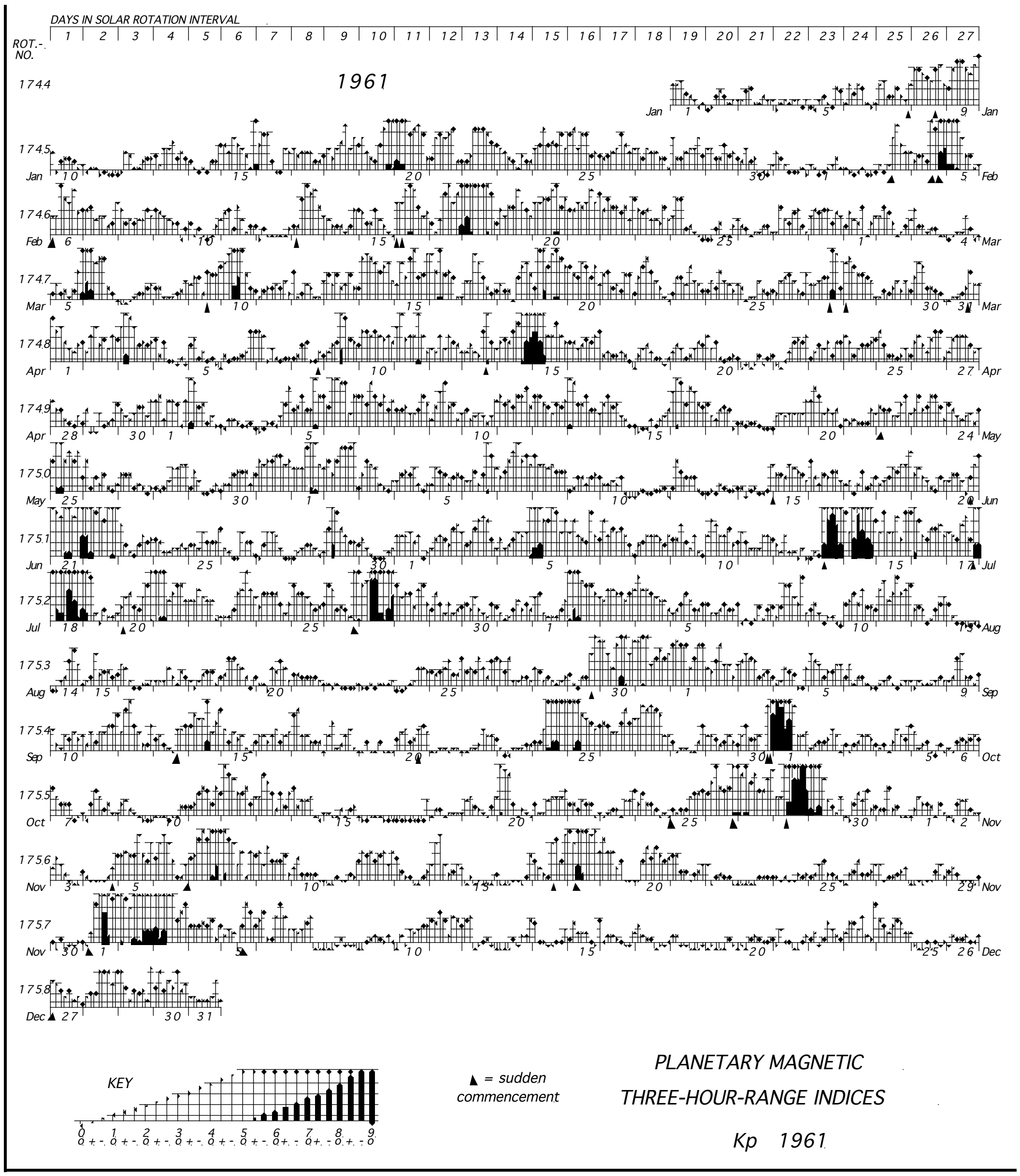

Figure 9 – The KP index visualized as a Bartel diagram

2.1.4 Projects Using Urban[40] Data

In sonification, we do not only have to use natural events, we can also sonify man-made events. These events involve interactions between humans and technology. Without a doubt, the surge of social networks such as Facebook and Twitter gives huge opportunities for sonification. The following projects use this kind of data as a source for their coming-into-being.

2.1.4.1 Twittersynth (2011)

Twittersynth (Harman, 2011), written in *MaxMSP*, uses public tweets as its data source. Every 30 seconds the software pulls out the latest public tweet and converts the characters into ASCII-values. Each ASCII-value is then run through a pseudo randomizer and then to an oscillatorbank containing 144 oscillators. Each character thus drives the frequency of one oscillator and if the tweet contains less than 144 characters, the non-used characters will result in a silent oscillator. The combination of all active oscillators results in a pad-like sound.

A second module allows the user to route the oscillators through a selector object, which chooses a random oscillator to be output at a tempo specified by the user. When setting a high tempo, a twittering effect arises.

Besides these two sonification modules, the user can also use granular synthesis. Here the user is in almost total control over the sound, the length of the tweet controls only the buffer of the soundfile. For example: a tweet with a length of 78 characters will result in a buffer of 78 milliseconds.

[40] I use urban as the projects described here are created with data created in an urban context. Etymologically speaking, urban comes from the Latin *urbanus*, meaning "of or pertaining to a city or city life". Indeed, while Twitter and Facebook are virtual worlds, they are created from within a citylike reality.

2.1.4.2 Tweetscapes (2012)

Tweetscapes (Nehls & Barri, 2012) takes twitter hashtags from public tweets within Germany. Their location controls the stereo position: the more to the west the tweet comes from, the more to the left the associated sound is played. The hashtags values are summed and further mathematical operations are applied, which results in a unique number for each topic. This number (between 1 and 1.000.000) controls the parameters of the sound generation. Several other strategies apply:

- The ratio of vowels in a hashtag determines the softness/harshness of the sound: the more vowels, the more harmonious.
- When the same hashtag is trending, a continuous background noise is created to make this trend audible.
- *@Replies*, answers directed to another specific user, are sonified using whispersounds. The authors say that during the night there are more *@Replies* than during the day.
- Tweets without hashtags or without a direct recipient are sonified with short generic sounds. The number of followers of the tweeter determines the volume: the more followers, the louder the sound.
- When a tweet is retweeted, the sound is echoed. When a message is popular, this results in the sound being denser and the fade out longer.
- In the unusual event that Twitter is down, *Tweetscapes* plays whalesounds. This refers to the image Twitter shows when it is down.

2.1.4.3 The Listening Machine (2012)

The Listening Machine was a work that used tweets from a group of 500 participants in the UK, selected proportionally from eighth different sectors (sports, arts, technology, education, politics, business, health and science). (Jones, Gregson, Britten Sinfonia, 2013)

Another part of the group was selected totally at random. Tweets are analysed on sentiment (positive, negative, neutral) on the basis of keywords like “brilliant”, “rubbish”, and “awesome”. Furthermore, the tweets are classified in eight categories. This classification is used to analyse how much overall twitter activity there is as well as the activity specific to one category.

The actual text of the tweets is analysed on prosody: The consonants are taken out and the remaining vowels are mapped on the fundamental frequency of the vowel as spoken by an average English speaker. The notes are played using pre-recorded orchestral samples and a continuous pattern emerges. The tempo of the music is determined by the overall activity, thus, during the night, when there is less activity, the tempo can very low so that the sonification of a tweet can take several minutes. Besides the individual notes, there are pre-recorded patterns and concrete sounds, which are triggered by topics and keywords. The authors give the example of an airplane sound, which can emerge when the topic is airports.

The project has a dedicated website and during six months in 2012, the sonification was available as a real-time stream on the BBC’s digital art channel *The Space*.

2.1.4.4 Rhythms of the City (2011)

Rhythms of the City (Guljajeva & Sola, 2011) sonifies online activity in specific cities. The overall activity on Youtube, Twitter and Flickr are measured and compared to historical activity. The resulting value is mapped to a tempo value on a metronome. The combination of various metronomes, each one representing a different city, results in a polyrhythmic total. This works bears some resemblance to Ligeti’s *Poéme Symphonique for 100 Metronomes* however, as the metronomes have to keep a tempo, they are fitted with Arduino-controlled motors to allow the never-ending social activity to be sonified.

2.1.4.5 Quotidian Record (2012)

On a more individual note, Brian House sonifies his life during a year in *Quotidian Record* (House, 2012), (House, 2013b). The artist assigned a note to each location he visited and attributed a different key to every city (for example: New York was assigned G major), thus modulating whenever he goes to another city. Within cities, locations are ordered by the frequency with which they were visited. This is mapped loosely onto a composed harmonic series but not strictly ordered by consonance. For example, the most common location is a third, second most common is a sixth, etc., and seconds and sevenths are further down the list. The mapping of these intervals is the same for each city. A pulse which tempo represents two hours, making the whole composition last 11 minutes, accompanies the melody. As the music is pressed on a vinyl record, one day corresponds to one rotation of the disc.

2.1.4.6 You'll have to take my word for it (2013)

Besides *Quotidian Record*, House also created *You'll have to take my word for it*, in which he used the data from a car crash (House, 2013a). In 2011, a US politician had a car accident but survived unharmed. The dubious circumstances in which the accident happened lead to an investigation and the car's black box data were made public. House used a slice of 20 seconds of the data just before the crash and used these data in a five-minute composition for two electric guitars and saxophone. A repeated B on the first electric guitar represents the percentage that the accelerator is floored. The faster the car goes, the faster the guitar plays the note. The second guitar plays continuous arpeggios whose speed is dependent on the RPM of the car. The tenor saxophone follows the overall accelerating motion of the car. This is made explicit by a slowly rising melody in C-Lydian. When the car crashes, the temporal scale changes and one second of data amounts to 30 bars. At this moment, the saxophone plays solo in a blues scale in A the contortions of the car as its flips over before coming to a rest.

2.1.4.7 AERO (2014)

AERO (Reeves, 2014a) is a MacOS app (made with *MaxMSP*) that uses flight departure and arrival data from four airports in the USA. The flight data are downloaded at a random time interval within a set range from the airport's websites and mapped to MIDI parameters like midi parameters for pitch, velocity, rhythm, attack, decay within set ranges. These MIDI values then trigger a custom instrument. The custom instrument is part off a bank of instruments that all receive the data in parallel fashion but play out the data in different ways as a result of using different ranges. As such the data trigger events and parameter changes in an overlapping fashion, which can result in dissonances because of different scales being played at the same time (this is intentional by the creator). However, all MIDI-generators are linked and the different ranges are designed to avoid excessive dissonance. Silences are caused by the random time interval at which data is downloaded: sometimes the separate MIDI generators have not received new data yet. Gregory Reeves quotes Brian Eno's *Music for Airports* as an obvious inspiration. Clearly, they share similarities; nevertheless, while Eno's music is an example of programme music, Reeves' uses the data to drive the narrative. In other words: he set a system in motion, just as Eno wanted to do. (Reeves, 2014b)

2.1.4.8 Stanley Cup Summed Up (2012)

Bard Edlund (Edlund, 2012), (Edlund, 2014) sonified the 2012 NHL Stanley Cup playoffs goals. When a team scores, a specific note, chosen from a G major pentatonic scale (unchangeable during the whole piece), sounds. The teams are divided into a western and eastern division, which is reflected by using a different instrument for each division. The assigned pitch is dependent on the ranking of the team (the higher the ranking, the higher the pitch). The goals (sourced from nhl.com and Wikipedia) are played at a speed of two goals per second and are played in a parallel way: all games of the first round are put next to each other and are the goals are played following the order in the games. For example: Team A scores at the seventh minute in game 1. Meanwhile in game 2, team C scores at the eleventh minute. Then team A scores again in their game at the twentieth minute. The sequence will then be *goal team A- goal team C- goal team A*. To embellish the sonification, the artist included a steel lap guitar loop and a drum beat. Edlund describes this work as *ambient data art*.

2.1.4.9 1945-1998 (2003)

1945-1998 by Isao Hashimoto sonifies and visualizes the 2053 nuclear explosions between the first test in New Mexico until the test by Pakistan in May 1998 (the three tests announced by the DPRK since 2006 are not included in the work). A metronome is set at a BPM of 60 and plays out the time: one month is scaled into a second and plays a sound at every second an explosion took place. This sound is different for each nuclear power and a melody arises as a result. Starting slow with just three explosions in 1945 (the test bomb in New Mexico and the two bombs on Hiroshima and Nagasaki), the sound gets denser as the USSR and the UK are performing nuclear tests. Furthermore, every New Year is denoted by a high-pitched sound and when a new nuclear power arises, a note is repeated four times. (*ctbto.org/specials/1945-1998-by-isao-hashimoto/*, n.d.)

2.1.4.10 Listen to Wikipedia (2013)

Listen to Wikipedia sonifies and visualises activity on Wikipedia through a dedicated website (LaPorte & Hashemi, 2013). Activity such as editing articles, new users arriving, all contribute to the overall soundscape. When content is added, a celesta sound is triggered while a clavichord sound represents a removal of content. It is impossible to tell if there was made both an addition and a subtraction, hence all of the size amounts are the compound net change. The pitches belong to a pentatonic scale; the more bytes are changed, the lower the note. When a new user joins the site, a swelling string sound will be randomly selected from three samples. To keep the soundscape from getting monotonous, the authors implemented a few extra rules:

- It won't play the same sound too many times in a row (it will randomly "fuzz" the sound to a higher or lower pitch) to avoid getting too annoying.
- It will only play a limited number of simultaneous sounds, to avoid getting too chaotic.

The interaction is quite limited on the sound level: The user can choose which language versions of Wikipedia he wants to include in the sonification. Of course, as English is the most used language on Wikipedia; the most dynamic sonification takes place with the English language version. The sonification is accompanied by a bubble visualisation of the same data. The user can click on these bubbles showing a keyword is subsequently taken to the related Wikipedia article’s revision page.

Some of this logic and the celesta instrument are borrowed from Maximillian Laumeister's *Listen to Bitcoin* (Laumeister, 2013) and adjusted for Wikipedia traffic. In comparison to *Listen to Bitcoin*, the authors added a clavichord sound, partially rewrote the code and compressed the sound files to make the website more efficient for high traffic volumes.

2.1.4.11 WikikIRC ou La sonification de Wikipedia (2012)

WikikIRC ou La sonification de Wikipedia (Baudu, Templier, & Blocquaux, 2012), (Baudu, 2013) is an installation that takes Wikipedia as well as its source. Changes in the French language Wikipedia are converted into pulses that control servomotors, which strike diatonic piano keys ranging from C3 to C8. Characters (letters, numbers and punctuation) are mapped onto notes, <a> being the highest note, <b> the note left to it and so on. There are 50 notes (26 for the letters, 10 for the numbers and 14 for punctuation and accented characters). Only the first 80 characters from a modification are used in a sonification phrase. The rhythm depends on the number of characters in a modification. Every sonification phrase lasts four seconds. The notes are spaced evenly across that four-second span. Thus, the bigger the modification, the denser the phrase will sound. For example, if there are only two characters, the notes will be played two seconds apart. A new sonification phrase is triggered every time the IRC-log receives a modification. This allows for overlapping phrases to unfold.

2.1.4.12 *N.A.G* (2004)

N.A.G. (Freeman, 2004) was an application that used data from the *Gnutella* peer-to-peer network. The user would enter search words and the application would start looking for mp3 files matching those keywords. Due to the nature of *Gnutella*, an mp3 file is downloaded in fragments. These fragments form the core of the musical result: Instead of mapping data, the sound files themselves are transformed based on various parameters that can be set interactively by the user. The downloaded fragments are played back at a speed depending on the available bandwidth. The available bandwidth to a file being downloaded also controls the rate at which the most recently downloaded fragment is repeated as well as the volume. When there is no new data available, the user can set the last files (the two last seconds from the file) to be downloaded to create a kind of stuck-record feeling. In case he does not use that option, the system will be silent until new data comes through. This looping feature is automatically interrupted when new data comes through: the system jumps right to the new downloaded fragment (Freeman, 2013). The user can also set the number of sound files being played simultaneously, which can lead to a polyphonic soundscape. Besides the sound-altering parameters, the search for specific files has an influence as well. Exploiting these possibilities, Every Man made a collage using two computers running the program, slowly changing the genres so to allow a varied yet soothing sound to appear. Doing this process on two computer made it possible to do a live improvised collage (Man, 2003). *Illegalart*, a record label centred on audio collages made a mixup of the Beatles and Radiohead (*http://illegalart.net/mp3s/02.04.html*, n.d.). Additionally, the author offers some audio samples on his site. Since writing the application, the *Gnutella* p2p network has almost disappeared and some frameworks are obsolete. Hence, the software does not work anymore.

2.1.4.13 Leech (2011)

Leech (McKinney & Renaud, 2011) is a multimedia composition which sonifies and visualises Bittorrent traffic. The work plays back mp3 files that are being downloaded and uses data related to the download to control sound and visual parameters. The data are mapped according to the following table.

Table 1 – *Leech*: data and mapping destination

Mined Data	Mapping
Torrent Progress(%)	Timbral Complexity
Download/UploadRate(kB/s)	Envelope Attack time
File Names/Sizes(mB)	Visual presentation
Number of Peers(int)	Visual presentation
Leecher vs. Seeder(%)	Synthesis Type
Peer Location(φ/λ)	Pitch/Timbre
Packet Transfer(φ/λ)	Pitch/Timbre
MP3	Processed Playback

2.1.4.14 Maelstrom (2011)

Maelstrom (Jones & Bulley, 2011) is an installation that addresses the data-overload on the internet. Every minute, 48 hours of user-generated audio are uploaded on the net and that amount is increasing exponentially. *Maelstrom* uses these sounds to perform a pre-written score[41] for octet (double bass, French Horn, violin, piccolo, flute and two drones). Audio fragments are sourced from *Youtube*, *Freesound* etc. using keywords and stored in a database for use in the piece. When the score commands the violin to play a middle C, the system will look up a sound that fulfils this condition. Thus, while there is a fixed structure in place, the sonification component ensures an almost infinite timbral variety.

2.1.4.15 Playing the Lottery in Plano (2009)

Playing the lottery in plano (Waren Burt, n.d.) is a composition which takes results from the New South Wales lottery[42]. Each winning number consists of six digits. A modulo division is performed on these numbers to fall within a specified range[43]. The sequential reading and modulo division of the numbers results in a sequence, which can be used to control a sound parameter. Taking inspiration from chaos theorists[44], Burt read through the array in a canonical way. The composer writes:

... several data streams read out simultaneously, and each data stream is the same data, but each one is delayed. Within the limits of text format typing, I'll try to give an example:

Voice 1: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21

---

[41] The score serves here as a command structure. The written notes are not played literally; they are triggers for sound samples.

[42] More specifically the New South Wales *Lucky Lottery* ($2 version) - draw number 9354, drawn on Sept 24, 2009. The *Lucky Lottery* is comparable to Portugal's *Lotaría Clássica*.

[43] For example,with 654332 as a number and a 3-octave range (37 pitches) the 654332 would then be divided modulo 37, resulting in 24.

[44] Delay differential equations are used in various science sectors to create a kind of 2D or 3D reading of the data to make the observed phenomenon clearer to investigate.

Voice 2: 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

Voice 3: 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41

Each of the lists of numbers is the index of the array that is read. All 3 lines happen simultaneously. So at the first moment, voice 1 gives you element 00 of the array, but voice 2 gives you element 10, and voice 3 gives you element 20. (Warren Burt, 2012)

As such, three sequences are generated and can be used to control different sound parameters. There are two versions of the composition. The first one uses a Phrygian mode in equal tempered tuning. According to Burt, he used:

> the original data for pitch, 10 steps later for duration, 20 steps later for velocity, 30 steps later for note duration, for the first voice (piano). The second voice (solo oboe) also has a slow sine wave Func Gen to make variations in CC1 (Mod Wheel) for more expressivity with the Garritan Personal Orchestra instrument.

In the second version, the composer added a third instrument (trumpet), playing together with the second instrument with adjusted parameters specifically for the trumpet. This second version used a microtonal 12-note scale.

2.1.4.16 Submap 2.0 Ebullition (2012)

Submap 2.0 Ebullition, according to their authors "visualises and sonificates data pulled from one of the biggest news sites of Hungary, origo.hu. In the 30 fps animation, each frame represents a single day, each second covers a month, starting from December 1998 until October 2010." The mentioning of names of cities/villages triggers a distortion on the visual map that is accompanied by a continuously evolving drone. Although no information is given on how the mapping is done exactly, it is clear that every location has a certain frequency attached. (Bircsák et al., 2012)

2.1.4.17 FMS Symphony

FMS Symphony, created by *CSV Soundsystem*, tracks the United States government cash spending and borrowing. The account balance is mapped onto a melody in C major, while the federal interest rates determine whether a chord is major (rates going up) or minor (rates going down). The distance between accumulated federal debt and the legal debt ceiling drives a contrapuntal line. With a BPM of 280, each beat counts as one workday. (Abelson et al., 2013), (Levine, 2013)

2.1.4.18 Stock Exchange Piece (2007)

Mathieu Saladin took the price and the 50-day Moving Average[45] of crude oil and gold during 50 days (between 4th of March and 22nd of April, with a resolution of one data point for a day) and mapped them one-to-one (or unit-swapping as Saladin calls it) onto sine waves. The prices are panned to the left while the 50-day MA sine waves are panned to the right. Each day lasts one minute, which allows the listener to experience the acoustic phenomena resulting from the slow-moving sine waves. These movements make the composition into a 50-minute slowly-evolving, minimalist drone piece. (Saladin, 2007)

2.2 Mappings in the Discussed Artworks

Although the works presented in the previous sections are only a very small part of what exists, they can give us a useful insight in the variety of mapping techniques. We can use this overview then as a primer to develop our own mapping methods.

As seen in chapter 1, there are three sonification modes: parametric mapping, model-based sonification and audification. It would not make sense to make a table because the works shown are only a small portion of what exists[46]. Nevertheless, we can make some general remarks.

Most works discussed use a form of parametric mapping: one data point is mapped to an individual event. Reading the dataset results then in a sonic artefact be that a

---

[45] The 50-day Moving Average is the average price of the stock during the last 50 days. The last 50 days prices are added together and divided by 50. The resulting number is the 50-day Moving Average. The Moving Average can be used to predict the trend of a stock.

[46] Indeed: the selection of discussed works is too small to make a statistic comparison. The choice of artworks was chosen on the availability of explanations of their mappings. Unfortunately, many artists do not document their mapping methods in detail which made them unfit to use them in this discussion. Nevertheless, the artworks that do not have sufficient documentation, but which I deem interesting are presented on the research blog.

composition or an installation work. Most works use multi-variate mappings: multiple data points are used to map to multiple destinations.

Furthermore, we can discern two ways of doing the parametric mapping:

1. Mapping to musical parameters such as pitch, velocity and duration. One example is Quinn's *Climate Symphony*: there is no prefigured material (besides the instrument choice itself) and data map directly to pitch. To use Saladin's term again: a unit-swap is performed.

2. Mapping to control parameters that will influence the outcome of a prefigured sound or structure. *Maelstrom* is an example of this kind of mapping: a fixed compositional structure exists; however, the mapping is influencing the timbre. *Leech* is somewhat different: a dynamic structure is created from a file that is being downloaded and that structure is being influenced by data being mapped to filter parameters amongst others. *Quotidian record* also uses a higher-level mapping: the change of city will trigger a key change. This is more than a unit-swap as it goes beyond the individual event: the subsequent notes will be in the different key as well.

Sound graphs can be found in *Listen to Wikipedia*: choosing a language will navigate the listener to another virtual sonic location. In *AERO*, one can change the airport, which obviously will change the sonic artefact. I consider Tomayko's *Geophonics* and *Climate Controlled* (the latter one which is discussed on the blog) as a mixture between sound graphs and model-based sonification: through interaction, the user can traverse the map in multiple dimensions. Especially interesting in Tomayko's work is the use of non-temporal data: while the other works all use time-series data, *Geophonics* uses a map. If we briefly return to section 1.6, Daniel's definition of mapping "to map is to locate" becomes clear in Tomayko's work.

Although *Submap* is parametric in essence, the use of a continuous drone can make us place it in the homomorphic modulation mode. Indeed: the amplitude never goes to 0 and we have a constantly fluctuating soundwave.

The mappings in all their variety yield very different results. It is thus interesting to see how different mappings are used. We cannot tell which mapping is the best, we are talking about an artwork, and hence efficiency is not the defining element to value the sound. Of course the data mapping is not the only defining element of the work, but we can clearly see a symbiosis between the data on one side (the uncontrollable aspect) and the limits set by the artists on the other side. Having discussed mapping techniques, we can now move on to discuss the technology behind sonification: what means do we have to sonify data in an artistic setting?

2.3 Sonification software

To sonify data, one can create one's own applications or use existing software. Over the years, a few software packages have been developed that should facilitate sonification in artistic practices. All applications show potential and their features are shortly presented. Thereafter we discuss the advantages and disadvantages of the applications, which lead me to develop a new toolbox.

2.3.1 sMax

Ciardi's *sMax* (Ciardi, 2004) is a bit of an outsider but at the same time one of the inspirations for *DataScapR* (and its predecessor *StockWatch*). Just as *DataScapR*, it sonifies the stock market using *MaxMSP*. The only information available about *sMax* is the ICAD paper Ciardi presented in 2004 and through personal communication. Ciardi claims a double-mapping process: instead of mapping the data directly to MIDI messages, the data are fed into an algorithm to generate dynamic musical patterns. Unfortunately, *sMax* is not available to the general public and audiovisual material is unavailable as well. The project was "put in sleeping mode" in 2012 (Ciardi, 2015).

2.3.2 *Maestro Frankenstein* 2

Maestro Frankenstein 2[47] (Tomayko, 2012), (Tomayko, 2015) is maybe the closest related to *DataScapR*. The application is focused on using geological data but allows other types of data to be used as well. Originally created in *MaxMSP*, the author has made a standalone version available in 2013. The user starts by importing a dataset. One column is mapped to pitch using a simple scaling (the input data range linearly mapped to the output MIDI range, while allowing logarithmic and exponential mapping as well). The rhythm is controlled by the timepoints the data were obtained but scaled to a smaller timescale. *Maestro Frankenstein* allows up to 256 tracks/datasets to be played simultaneously.

While the software clearly offers potential, there are some drawbacks: The mapping operations are quite limited and the software cannot be modified which restrains the potential a bit.

2.3.3 *Sonart*

Sonart (Yeo, Berger, & Lee, 2004) has been developed at the CCRMA. It was intended to sonify image data (for example RGB values from pixels) and general statistical data. It is only available for PowerPC, and hence is obsolete.

2.3.4 *Sonipy*

Intended as a sonification and auditory display research framework, *Sonipy* (Worrall, 2009b) is a collection of Python modules that allow the user to build his own custom sonification tools. The disadvantage sits in the language used: Python is not an easy language to learn.

[47] There was an earlier version in 2004 however that version was unstable. MF2 seems to be a more mature release.

2.3.5 Artwonk/Musicwonk

Artwonk (and its music-only version *Musicwonk*) is a visual toolbox for algorithmic composition. Connecting modules to create a data flow is similar to Native Instrument's *Reaktor*. However, whereas *Reaktor* offers audio capabilities, *Artwonk* is a MIDI-only application. John Dunn writes on this choice: "*Artwonk* is not music production or scoring or multitrack editing software and it does not attempt to replace them. It is pure creation software. As such, MIDI gives the widest possible choice of sound producing options. With *Artwonk*, you are not stuck with one particular sound set, you have the entire world of MIDI synths and samplers at your command". (Dunn, n.d.)

Artwonk is not specifically aimed at sonification but the modular nature makes it a useful tool for experimentation. Composers such as Warren Burt, Dunn and Clark and Hans Van Raaij (among others) used the application in their compositions (which are discussed in the previous section).

2.4 Conclusions

Finally, regarding sonification software and having looked at different packages we can say that there are various shortcomings in all applications.

1) The application is simply not available: *sMax* could be useful if the creator had made the software generally available. The *Sonipy* modules equally are unavailable. *Sonart*, while available, is obsolete and does not run on newer Macs. For the reason of obsolescence, *N.A.G*, the *Gnutella* sonification work described in the previous section, is not included in this section.

2) The application is limited: While *Maestro Frankenstein* offers an initial good start the further processing is limited.

3) The software is too broad: *Artwonk* is definitely an interesting application but should be seen as a variant of *Reaktor* or as a higher-level version of *MaxMSP*. As such, the user has to start from scratch, which leads us back to the problem of

having to spend too much time in building the system and risking losing out on the artistic aspect.

4) The data themselves come in various formats. Designing a generalized approach can be an option. However, in wanting to be generalist, one can get lost in a maze of options.

The shortcomings discussed above call thus for a new application. It should be widely expandable, easy to use and widely available. It should focus on one on type of dataset to allow maximum usability. Enter *DataScapR*, a toolbox for stock market sonification.

3 DATASCAPR

DataScapR constitutes one of the practical components of this PhD. It is a toolbox of *MaxMSP* patches, which enables the user to map stock market data to musical parameters. I first will describe the background and goals of the application, taking into account the considerations posited in the preceding chapters. I then go on to describe the architecture and usability of the software.

3.1 Background

In embarking on a sonification project, one can ask why I chose to work with stock market data. This question can be answered on a subjective-personal and a more objective level.

On a personal-subjective level; stock markets have always attracted me: the idea of becoming rich by trading the stocks at the right moment was of course a big reason. Furthermore, the abstraction of the real world into a financial maze seemed interesting: indulging in the stock market tables in the newspapers, one could lose himself and become immersed in a virtual world reigned by data (at that moment nobody was speaking of Big Data yet). Films like *Wall Street* (Weiser & Stone, 1987) showed the dynamics of the stock market. Although there were negative sides, the bustling dynamic found in the screaming of traders was compelling and made me want to know more about it. I never pursued the job of a stock trader as I had other interests but I was fascinated by the whole phenomenon of the stock market.

On the objective level, stock markets can show complex behaviour, which can be interesting for musical applications. Much research has been done to determine whether the stock market shows temporal patterns or simply random walks that would make it possible to predict the markets (Goetzmann, 2001), (Agrawal & Tandon, 1994). Furthermore, technical analysis of charts is an important methodology to try to predict future movements. Furthermore, technical analysis is an important way of analysing the stock markets. This type of analysis assumes that all information about the stock price is present in the charts themselves. One type of technical analysis is *Fibonnaci retracement*; two extreme points (highest and lowest) on a chart are taken and the space between is divided using the Fibonacci ratio. For unclear reasons, the price seems to change direction around these lines. Although meant for historical analysis, one can draw arcs through these Fibonacci lines so that they protrude 'into the future' and use these arcs to set limits when to buy or sell a stock.

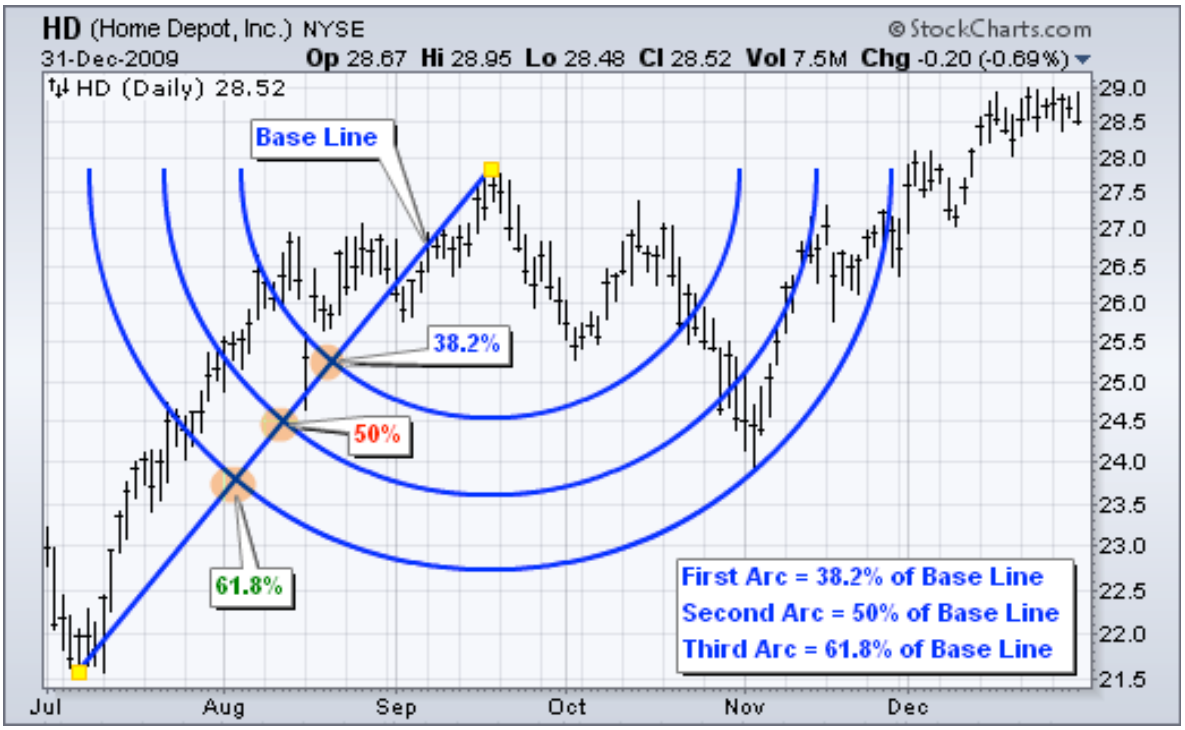

Figure 10 – By drawing arcs, one can predict future movements and levels of change (Investopedia, 2005)

The Fibonacci retracement technique is just one of the many technical analysis techniques. There are many other techniques available such as the *Elliot Wave* theory or *Dow* theory. It has to be said that all these techniques are not 100% correct, sudden movements can distort the movements. Thus, one cannot completely predict the market, just like one cannot completely predict the weather. It would be the Holy Grail if we could predict the markets, as it would allow the trader to buy and sell at the right moment. It would be out of the scope of this dissertation to discuss this topic as the field is very big and it certainly deserves its own dissertation. For more information on technical analysis, I refer to the website *stockcharts.com.*

The financial markets have seen enormous shifts in the last decades. Since the Second World War and certainly with the end of the cold war, globalization has fundamentally changed financial trading. Deregulation of the markets as mandated by Bill Clinton's administration in the 1990's (United States Congress, 1999) made it possible to shift more capital than ever before. The dotcom bubble burst in the early 2000's and the housing crisis in 2008 have proved that this holds risks.

Fuelled by IT-innovation, humans have been steadily been replaced by algorithms to perform trades. As such, high-frequency trading, in the brink of milliseconds, allows the best algorithm to win instead of the best trader. Nevertheless, these systems can break like they did in the 2010 flash crisis. (Kirilenko, Kyle, Samadi, & Tuzun, 2011). High-frequency trading also gives rise to ethical questions: it gives more power to bigger corporations that are able to invest in fast computer systems and the small timeframes can outplay smaller traders (Wagner, 2011).

All elements described above combined, we can see that the stock markets are interesting phenomena and can be an interesting venture to explore in an artistic context.

3.2 Goals

Stock market sonification is not new. Worrall gives an excellent account of stock market sonification going back as far as the beginning of the 20th century where traders interpreted the telegraphed signals from the tickertape to discern stock symbols and prices. The experiments that Worrall presents are all meant to be used to improve trading (although he does refer to *sMax*) and he does not discuss artistic stock market sonifications. The need for a toolbox for stock market sonification in art becomes then very clear.

DataScapR is a toolbox of *MaxMSP* patches that allow the user to control musical parameters using the stock market values as input values. To make this possible, the application downloads and stores values from stocks specified by the user. This happens in realtime[48] for intraday data as well as non-realtime for historical data. The data are sent to a mapper module where the data are scaled and put into specific ranges, which are user-definable. From there on, the mapped data are sent out to the sound and MIDI modules where the mapped data control sound parameters. The composer can then tweak the sounding result or leave the output untouched, as he prefers.

DataScapR takes on the problem of technology in an artistic environment. While technology has helped composers to explore uncharted musical territories, it can become a burden for that exploration if too much emphasis is placed on the technological aspect. In part two of this thesis, the theoretical side of this problem is explored and discussed. The goal of *DataScapR* is to offer composers an easy-to-use application to sonify the stock market and integrate this process in their system. While it is possible to use the toolbox out-of-the-box, the modular nature of the system and the implementation in *MaxMSP* make it possible to extend the toolbox and adapt to one's wishes.

3.3 What *DataScapR* is and What It Is Not

DataScapR is explicitly not a one-button-application. While it is easy-to-use, the user will have to invest time in setting up all parameters. My goal is to keep the user conscious of his operations on the data while making the software accessible. Hence, every operation in the chain has to be initiated explicitly by the user. While it would be technically possible to create a score by clicking just one button, I believe that the constant engagement with the dataflow will benefit the composer in his voyage to the final artwork[49]. As John Maeda says: "Technology needs to be humanised rather than optimised, and yet properly understood on its own terms" ("john maeda: painting by pixel," 2014). In using *DataScapR* we have to be aware that the compositional artefact is just one aspect of

---

[48] Yahoo delays the data for some markets for commercial reasons. A list of current delay times can be found on http://finance.yahoo.com/exchanges. To get real-time data, the user has to subscribe to paid services. Hence, in order to keep *DataScapR* a free toolbox, it was impossible to use non-delayed data. An alternative to Yahoo Finance was Google Finance (started in 2006). However, Yahoo Finance seemed to have more users and getting information was easier, hence I chose to use Yahoo Finance.

[49] One can consider voyage as the position taking posited by Bourdieu or Jacob's search space: everything exists, we just search for what we want. (Bourdieu, 1993) and (Jacob, 1995)

the work. „Composition becomes a form of system design and musical artefacts become traces of that design". (Hamman, 2002).

3.3.1 What *DataScapR* is Not?

In the process of developing *DataScapR*, I made some decisions to omit certain features. As developing a toolbox is not a linear process, I stepped back back and forth many times to add and remove features. While the absence of features might be an uninteresting element at first, I think explaining those decisions will help in understanding *DataScapR's* premise better. This motivation of omittance can be seen as well in the perspective of the process of coming-into-being: the artefact (the toolbox) is not the only important element: the development process is a fundamental part of the work.

No Save function: The most striking omittance is probably that there is no save function present in *DataScapR*. The motivation for this ommitance stems from Giorgio Sancristoforo, the creator of *Gleetchlab*. His statement conveys perfectly why I omitted a save function:

> I intentionally avoided including save and load functions of gleetchlab settings. (That is since the first version of gleetchlab) Why? It is an important part of my musical approach. In my analog synthesizer days there were no save fuctions at all but pencil and paper. If you approach each time a reset machine, you are forced to do something new and with little time and patience, you can master the software much better.

Besides this, the data are always changing. It would thus be illogical to keep a fixed frame for each different dataset we use. There is one instance where I decided to include a save function: in the historical data components, where the user can draw curves, he can subsequently save them and load them later. While the other parameters can be set quite easily, it is impossible to recreate the exact curve used in earlier experiments. I believe that it benefits the user if he can reuse the curves he drew earlier. The addition of the save function is not incompatible with the no save idea: the user will still have to be conscious in setting up the domain and range of the function object. The idea of Heidegger's breakdowns is thus preserved.

DataScapR is **not a one-button-application**: *DataScapR* is explicitly not a one-button-application. While it is easy-to-use, the user will have to invest time in setting up all parameters. My goal is to keep the user conscious of his operations on the data while making the software accessible. This is meant to make the user more aware of his actions and their effects on the resulting artefact. In using *DataScapR* we have to be aware that the compositional artefact is just one aspect of the work. As I discussed in the theoretical part, the process of coming-into-being is equally important as the result of the productive activity. If we would only take the result in account then we would "fetishize the musical work, converting it from a catalyst for experience into a commodity to be traded within an economy, whether financial or ideological" (Hamman, 2004). Making the process of *DataScapR* explicit, every operation in the chain has to be initiated explicitly by the user. While it would be technically possible to create a score by clicking just one button, I believe that the constant engagement with the dataflow will diminish the distance between the composer and the technology and benefit him in his voyage to the final artwork[50]. We can link this idea to Heidegger's idea of breakdowns: only in the breakdown, we become aware of the object's function. The technology becomes human or as John Maeda says: "Technology needs to be humanised rather than optimised, and yet properly understood on its own terms" (Hackworth, 2014).

Not a one-stop-shop: *DataScapR* creates musical material; it does not create a finished artwork. As such, the user has to further sculpt the material in the VST's or in external applications. Related to the previous argument, not delivering a fixed artefact makes the user more aware of his choices and forces him to think about his actions.

No audio-generation within *DataScapR*: While originally I intended to include some small patches to generate audio, I removed them later on. There are many good VST's available and it would be a waste of resources to develop something from which there are way better versions available.

Besides that there are good VST's available, the addition of audio generation modules could impose a certain style on the composer. For example: if I would include a granular synthesizer, there would be a risk that the composer would feel restrained to use only that type of audio generation. This leads us back to Feenberg's argument that no

[50] One can consider voyage as the position taking posited by Bourdieu or Jacob's search space: everything exists, we just search for what we want. (Bourdieu, 1993) and (Jacob, 1995)

technology is neutral. Excluding audio generation is probably tone best way to avoid stylistic constraints..

No visualisation: *DataScapR* is intended for sonification. Adding a visualisation component would certainly drive attention away from the sonification part. There are many visualization applications and languages available such as *R* and *DataViz*. (“www.r-project.org,” n.d.) and (“www.dataviz.org,” n.d.)

3.3.2 Expanding *DataScapR*

DataScapR is created to be used out-of-the-box while at the same time allowing easy modifications. The modular approach of breaking up the patches in parts, each part corresponding to a specific function, makes it possible to insert own patches. Furthermore, to facilitate interaction with other patches, I included send objects at the end of the mapping modules and a patch with the corresponding receive objects. As such the user does not have to go inside the patches to add send objects, which makes it both easy and safe (in the sense that if the user does not have to edit the patches, he will not accidentally delete objects). Finally, every part of *DataScapR* has comment boxes next to it, explaining the function of the parts in the chain. This should allow easy understanding and subsequent modification of the parts in a whole.

DataScapR is written in *MaxMSP*, a visual programming language mainly focused on audio and video processing (“Cycling '74 Max 7,” n.d.). The toolbox allows both real-time and historical datatracking. While the modular architecture is similar, the real-time version differs from the historical module in various aspects. I will discuss the two parts separately and then deliver a side-by-side comparison. The toolbox consists of three main units: a data fetching- and-reading unit, a mapping unit and a sound or score-generating unit. While the most logical thing to do is to use all units, the composer can choose to use only the module he wants. Built in *MaxMSP*, the toolbox offers flexibility in repurposing the software as it suits the user’s needs.

3.3.3 System Requirements

DataScapR has been developed using *MaxMSP* 6.1.8 32-bits under MacOS 10.9 Mavericks. The software should work on computers with at least MacOS 10.6.8 Snow Leopard. The hardware used to develop is a MacBook Pro Retina mid-2014.

3.3.4 Installing the Toolbox

DataScapR relies on the use of some external objects. They need to be installed to make the toolbox work properly. The externals used are:

1) mxj.*DataScapR*: the custom JAVA object that tracks the stock market data in real-time

2) the *bach* automated composer's helper: a set of objects to assist in algorithmic music. I used this toolbox mainly for the ability to show symbolic notation in *MaxMSP*.

3) the *ej* objects by Emmanuel Jourdan. More specifically I use the *ej.function.js* object as it provides more possibilities than the standard function object.

The *mxj.DataScapR* object is installed by copying it in the Java externals folder as well as the Java help folder. The *Bach* toolbox is installed using an included installation script. The ej objects collection has to be copied to the max packages folder.

The historical data patches consist of five modules: data fetching, data reading, mapping, VST or MIDI output and score creation. The patches are laid out next to another in a tabbed interface, allowing the user to focus on a specific part of the process.

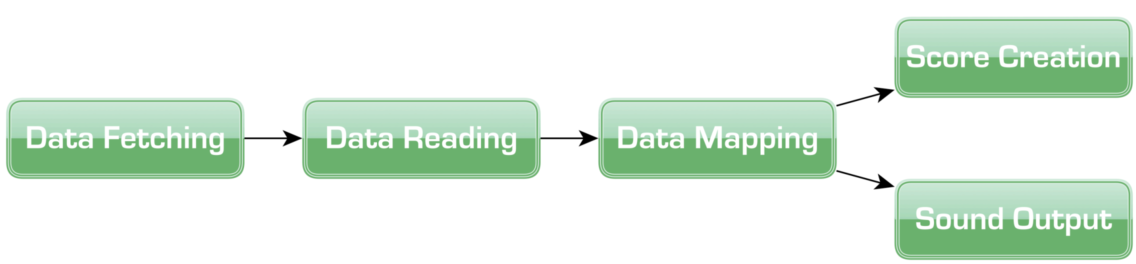

Figure 11 – *DataScapR*: the 5 modules allow the user to follow a logical path from data fetching until the sounding artefact.

The *DataScapR* toolbox allows historical data to be downloaded and used to create compositional material. Yahoo Finance offers historical data on most stocks worldwide and going back as far as 1970.

The datasets can be downloaded directly from the Yahoo Finance site or using the datafetching-patch. To get a dataset, the user can select a stock using the menus. 17 market indices of markets and their respective stocks and symbols are included in the toolbox by default. Should the user want to track other stocks, he can look up the stock and symbol on the Yahoo Finance page. As stocks are listed and delisted, it is impossible to create a definitive its of all stock symbols. Furthermore, Yahoo formats the stock symbol by adding a suffix for non-American markets. This suffix is not used by all other websites hence it can cause confusion. Therefore, it is advisable to double-check the symbol on the Yahoo webpage.

To download the dataset, the user enters the stock symbol, start and end date[51] and the dataset will be downloaded as a csv-file and saved in the chosen folder. The filename is always table.csv and cannot be altered. Downloading a new dataset will overwrite the existing file.

[51] If the data requested is beyond the range of historical prices available through Yahoo Finance, all available data within the range is displayed. Historical prices typically do not go back further than 1970. (Bourdieu, 1993)

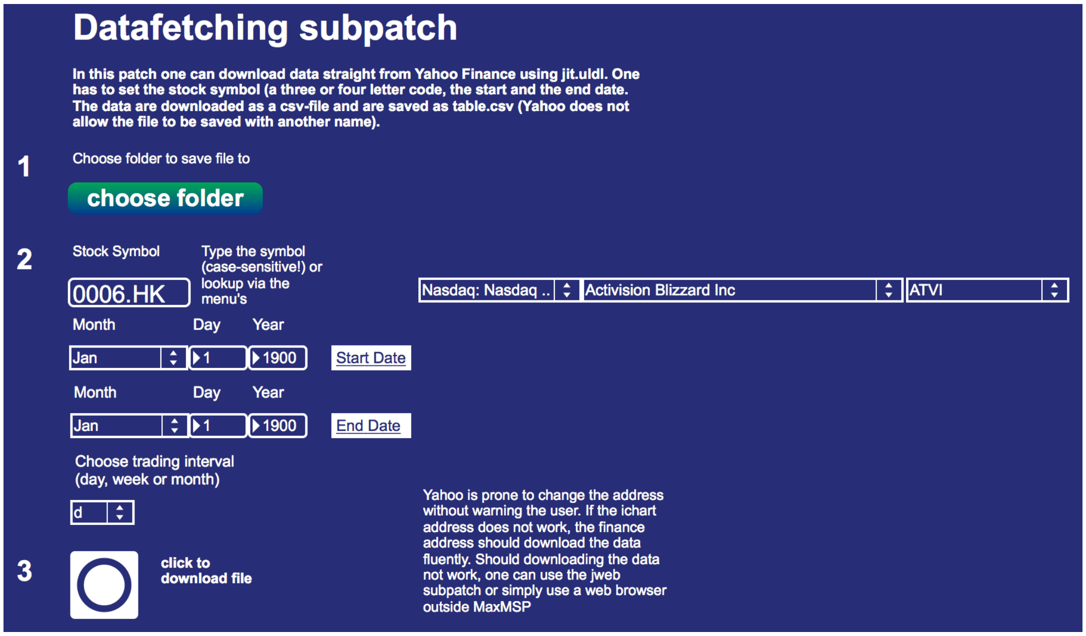

Figure 12 – *DataScapR*: datafetching subpatch window showing three easy steps allow downloading a dataset. The stock symbol can be set manually or by using the umenu's.

The variables included in the datasets depend on the time resolution (day, week, month) chosen.

Daily prices include: the open, high, low, close, and volume for each trading day shown.

Weekly prices include:

- Opening trade of the first trading day for the week.
- High and low price quotes of the week.
- Closing price on the last trading day of the week.
- Adjusted close, based on closing price.
- Weekly volume is the average daily volume for all trading days in the reported week.

Monthly prices include:

- Opening trade from the first trading day of the month,
- High and low price quotes for the month,

- Closing price on the last trading day of the month.
- Adjusted close, based on closing price.
- Monthly volume is the average daily volume for all trading days in the reported month. (“About historical prices,” n.d.)

After the dataset is downloaded, the next step is to read the dataset. By dropping the csv-file on the jit.cellblock object, the dataset is imported in a coll object and can be inspected visually. The csv-file is formatted in reverse-chronological order and comes with a header line. The csv2coll.js javascript object strips the csv file of its header and sorts the data chronologically. The seven parameters are converted from symbols to floats and in addition to those seven parameters, the spreads[52] open-close and high-low are calculated. Finally, the length of the file (the number of data points) is output as well. The date is converted from a symbol in a floating number; however, it is not used further down the patch.

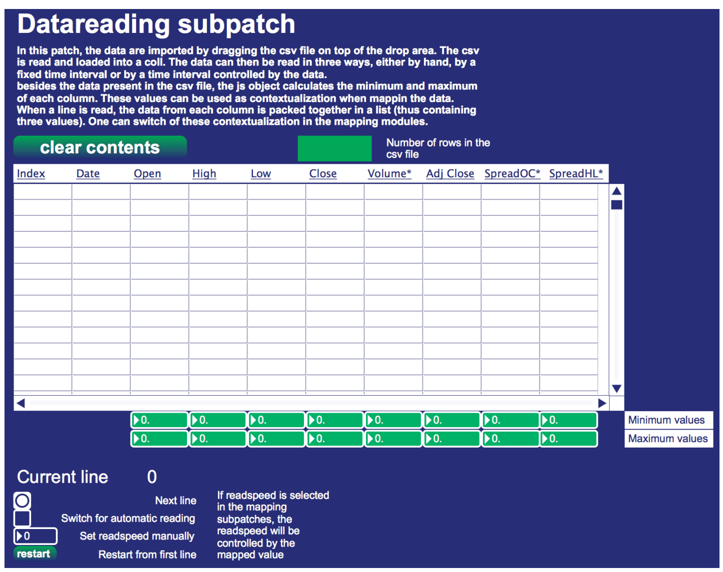

Figure 13 – *DataScapR*: the datareading module allows a visual inspection of the dataset. Minima and maxima are displayed below each column.

---

52 The spread is calculated by subtracting the open from the close and the high from the low. The resulting values are made absolute, meaning no negative values will exist.

After the dataset has been imported and sent to the coll-object, we are ready to read it. This can be done one-row-at-a-time or in a dump operation where the whole dataset is sent out in its totality. Here the mapping modules diverge, as they have to handle the incoming values differently.

3.3.5 Mapping the Data in the Row-by-row Component

To actually use the data in compositions, we need to map them using the mapping module. The mapping module routes the data variables to their output destination. Using the umenu's on the left side, the user selects the data he wants to use. These data are then sent to the mapping method that is chosen through a menu as well.

There are four mapping methods: a scaling of the incoming values using zmap, a scaling of incoming values using function to scale the values, a modulo operation and a modulo operation followed by a function object.

The mapping methods or techniques are parametric: the parametric sonification mode allows the most flexibility and control hence the choice for this mode. The use of parametric mapping makes it easier to integrate *DataScapR* in other patches and use it with other applications. However, when using a VST and a sound input, we can consider the mapping to be a homomorphic modulation (the term posited earlier by Worrall). Indeed, the amplitude is dependent on the incoming sound and interpolation is applied between two consecutive data points.

The data to be mapped are chosen at the left side with a umenu (default is the opening price). After choosing the input data, the mapping method is selected. The first object to the right of the umenu is a text object that displays the value to be mapped. To the right side of that object are the values and contextualization switch. Finally, at the right side, there is another text object, which displays the first element(s) of the mapped list. In the VST and MIDI patches, one can then choose the output destination through a umenu.

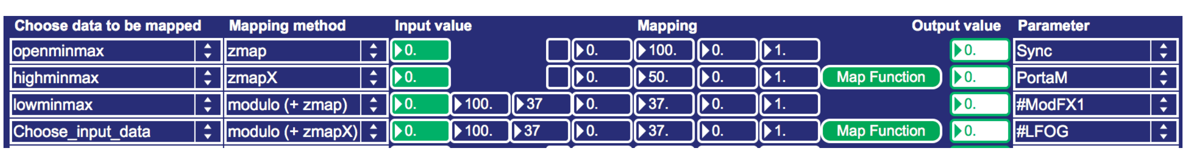

Figure 14 – *DataScapR*: the four mapping methods.

The first method uses incoming values that can be contextualized, meaning that the minimum and maximum value in the column is send to the input range of the zmap object. The user sets the output range. As such, what could be a small movement in absolute values could be a far bigger movement in relative values. For example: a stock price oscillates between € 77,01 and € 77,16. In absolute values this would give a maximum movement of € 0,15. If we would map the value directly to frequency, the sound result would be relatively static. However, if we extrapolate this € 0,15 range to a MIDI range between 0 and 100, the sounding result would be far more dynamic (see the table and chart below). The user can use all variables except for the date variable. Contextualization is active by default and can be disabled by clicking on the toggle box to deselect it. When contextualization is disabled, the user can set the input range manually.

Table 2 – Absolute and contextualized values example (see text for full description)

Uncontextualized values	Contextualized values
77.14	86.663956
77.1	59.996948
77.06	33.33
77.01	0
77.07	39.997967
77.09	53.328926
77.06	33.33
77.13	79.995934
77.13	79.995934
77.16	100

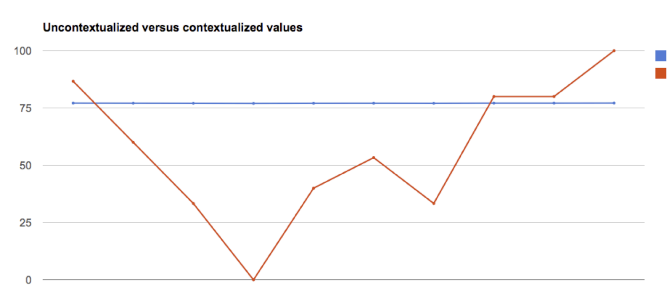

Figure 15 – Absolute and contextualized values example: the contextualization makes the profile of the line sharper.

Instead of setting the range with a minimum and maximum value, the user can opt to use the function object, which allows creating a diversified mapping depending on the incoming value. For example, incoming values between 0 and 50 can be mapped to a range of 0 to 10, while values between 51 and 100 can be mapped to a range between 20 and 80. The function object will open in a separate window to avoid cluttering on the main patch. Pressing the option key wile dragging the segments allows the user to create curves.

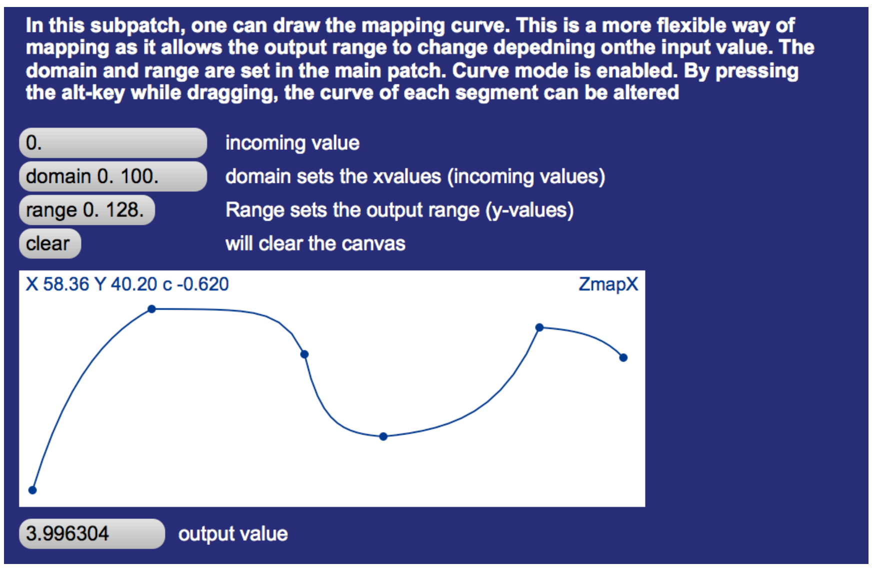

Figure 16 – *DataScapR*: The *ej.function* object allows non-linear scalings

A third mapping uses the *modulo* object, followed by the *zmap* object. As Yahoo delivers its data with two decimals, the incoming value is first multiplied by 100 after which the modulo operation is performed. The modulo value[53] can be set by the user. After the modulo operation, the value is sent to a zmap object as the value needs to be prepared for its output destination. For example, one can perform a modulo 12 operation but then send it to the zmap object to have the value between 0 and 12 be mapped to MIDI-values between 60 and 84. In the case of using modulo, the subsequent zmap does not use the contextualization from the data but the minimum (0) and maximum value of the modulo operation. Of course, the user can set a custom output range.

[53] The default value is 37, as a reference to Warren Burt's *Playing the lottery in plano* (see section 2.1.4.15)

The fourth method is essentially the same as the third one, with the difference that instead of a simple *zmap*, the *function* object is used, allowing a more diverse mapping.

All mapping results can be routed to a VSTi (using the vst~object) or a generic MIDI-output. A mapped value can also be sent to the metro object that controls the reading speed of the coll.

To allow contextualized mapping with the function object, I used the *ej.function.js* external as it allows the domain range to start at a non-zero value, which is impossible with the standard function object.

3.3.6 Using a VST

To use a VST, the user has to load a VST by clicking on the *plug* button at the left side of the window. Each VST has a set of proper parameters. Hence, after the VST has been loaded, the parameter names have to be initialized by clicking the *params* button (however, the MIDI events parameters stay always the same). Loading a new VST will automatically clear the parameter names. When selecting a parameter in one of the menu's, that parameter will be automatically disabled in the other menu's, preventing sending out conflicting values. By clicking *open VST window*, the user can open the VST window to inspect the VST visually. In the audio abstraction box, one can choose the audio input to use, as the user wishes. Clicking the loudspeaker icon at the left side enables the audio; moving the sliders sets the volume.

Finally, one can control the reading of the dataset in three ways:

1. Clicking the bang button will send out the data on the next line.
2. The dataset can be read at a fixed speed by toggling the switch for automatic reading. Additionally one can set the rate at which a new line is read by using the numberbox.
3. The readspeed can be controlled by a mapped value.

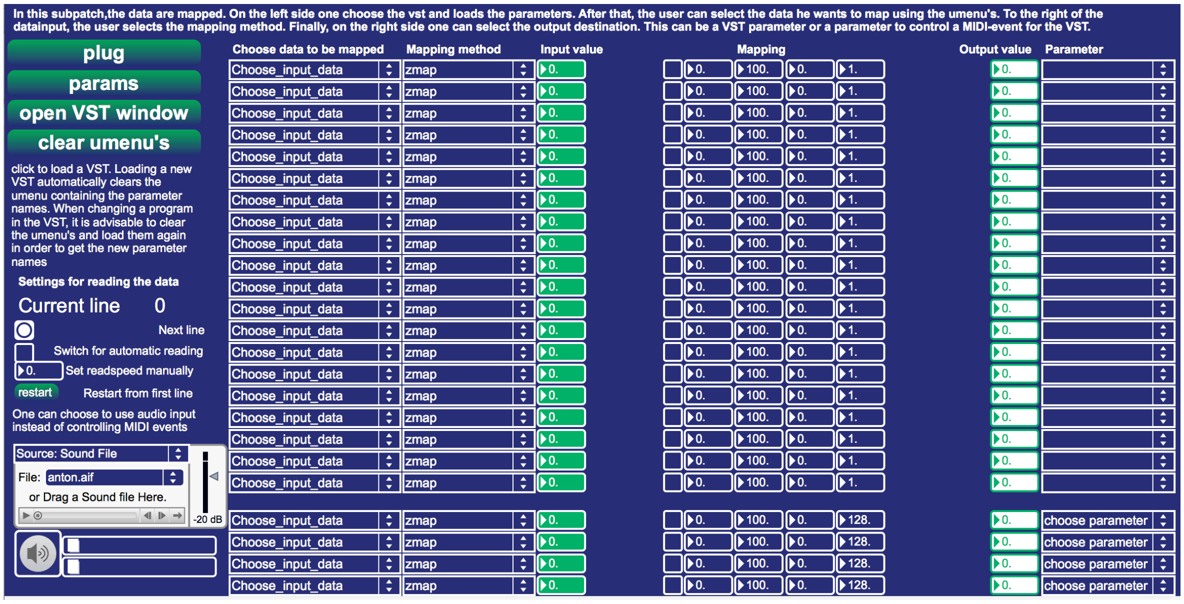

Figure 17 – *DataScapR*: the VST mapping module window

The MIDI patch is essentially the same except for the following differences:

1) The default output range for the VST patch is between 0 and 1[54] for the parameters (for the MIDI events, the range is between 0 and 128, for the read speed the range is between 0 and 1000). For the MIDI patch, the default output range is between 0 and 128.

2) In the VST patch, one has to load the VST he wishes to use and then the parameter names (as they are different for each VST). In the MIDI patch, the menu is loaded with all possible MIDI parameter names at startup.

3) The VST patch allows using live input instead of generating MIDI events. This makes it possible to use effects VSTs for use in live electronics.

---

[54] The VST standard, as set out by Steinberg, means that "All parameters - the user parameters, acting directly or indirectly on that data, as automated by the host, are 32 bit floating-point numbers. They must always range from 0.0 to 1.0 inclusive, regardless of their internal or external representation." Hence, all mapping modules (except those for the MIDI-events) map between o and 1. (Steinberg Media Technologies GmbH, 2003)

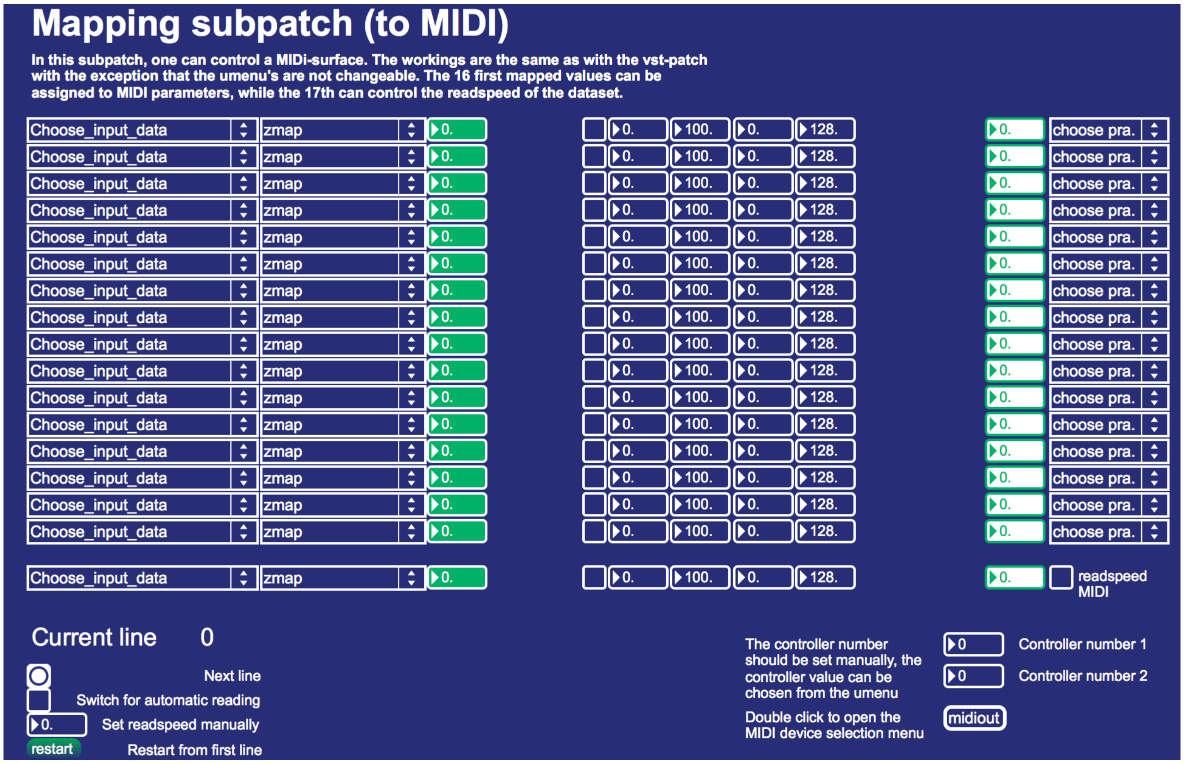

Figure 18 – The MIDI patch is slightly different from the VST patch

3.3.7 Score-creation

Besides reading a dataset row-by-row, one can create musical scores. The score creation module uses the *bach: automated composer's helper* framework developed by Andrea Agostini and Daniele Ghisi. *Bach* is a set of patches and externals to help in computer-assisted composition. The framework is comparable to Ircam's OpenMusic and the open-source variant PWGL but being written in *MaxMSP* it allows real-time operations and it has a somewhat easier learning curve for people who use *MaxMSP*. The ability to use symbolic notation was the main stimulus to use *bach*. This allows the user to quickly create a provisory score so that he can experiment easily with different mappings without having to export the data to another application. Indeed, I initially intended to create *DataScapR* with only a textual output that could be used in AC Toolbox or other applications. Using *Bach* however, allowed an easier, yet still hands-on approach.

For the score creation, one can control up to four monophonic voices. Every voice has the following parameters that can be controlled by data: pitch, onsets, duration, velocity, articulation and instrumentation. Not all parameters have to be used: one can disable the mapping to go through by toggling the toggle boxes at the right side of the patch. Instead of sending out the mapped data, the mapping module will send out an empty list, which will result in showing the default value. The default values are shown in the table below.

Table 3 – *DataScapR* Score Creation: parameters and default values

Parameter	Default Value
Pitch	Central C (Midicents value 6000)
Onsets	1000 (All notes start at a multiple of 1000 ms)
Duration	1000 ms
Velocity	100
Articulation and Instrumentation	Nil (no articulation/instrumentation will be shown)

3.3.7.1 Components of the Score-creating Module

The score creation is divided in four patches: data-fetching, data-reading, data-mapping and score creation. The data-fetching and the data-reading modules are the same as in the Row-by-row module, with the only difference that a whole data set is read at high-speed (using an uzi-object) instead of row-by-row. The mapping methods are the same, however, the mapping destinations are fixed. Of course, the score creation patch and the graphical conversion are not present in the *Row-By-Row* component.

3.3.8 Working with the Score Creation Component

After importing a dataset, the user can set all the desired mapping parameters in the mapping subpatch. Clicking map it will trigger an *uzi* object, which will send a series of bangs (the number of bangs being equal to the length of the csv file) to the reading module. As such, the data are mapped sequentially and collected at the end of the mapping in a list. This list is wrapped in parentheses so it can be used by the *Bach* framework (see further). As there are four voices, the lists have to be combined using the join object. Certainly when using large lists, the list might not come in order which could confuse the join object and make max crash. To avoid this from happening, mapping only sets the lists in message boxes through the right inlet. After mapping, the user has to click combine lists to trigger the join object.

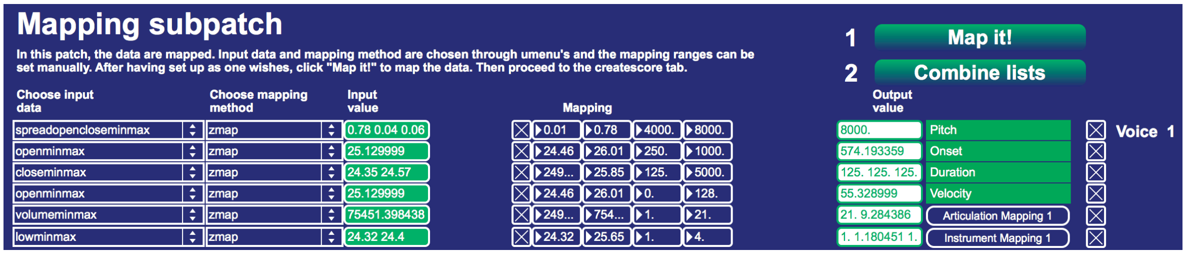

Figure 19 – *DataScapR* mapping subpatch window detail showing the first voice of four. The mapping destinations are fixed but can be disabled.

After having mapped the values and combined the lists, the user can move on to the score creation patch. In that patch the user first sets the number of voices (by default that is four). Clicking the *Make Roll* button will feed the mapped data into a *bach.roll*, an object that displays the notation in a proportional fashion. If desired, the user can make adjustments by selecting a portion of the roll and move the notes around manually. Alternatively, one can use the menu's to transpose the parameter values by adding, subtracting or multiplying them. Furthermore, one can set the tone division to allow quartertones or even smaller divisions[55].

[55] While allowing smaller divisions (up to 1/100 of a tone), *Bach* does not support those smaller divisions graphically. Playback however will adhere to the chosen tonedivision.

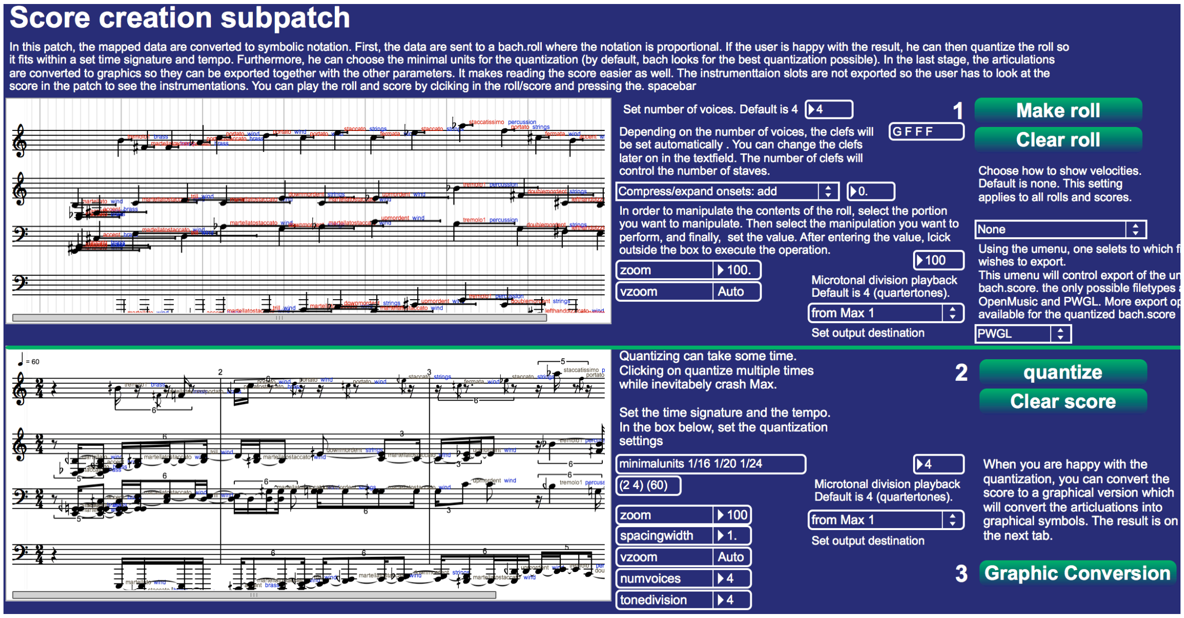

Figure 20 – *DataScapR* scorecreation subpatch window showing the roll and quantized score

In order to obtain a classical score the proportional roll has to be quantized. This is done by clicking the quantize button. By default *bach.quantize* is provided with quantize definitions that the developers think are best for general operation, however, the user can change the quantization parameters if desired. For detailed information how the quantization works, I refer to the *bach* help files. One can set the time signature and the tempo (default is 2/4 and q = 60) Due to *bach's* handling of articulations; the score has to be sent out to another score object to convert the articulations into graphical symbols. In the quantized score, one can select notes and drag them to transpose or displace them.

Due to *bach's* handling of articulations and instrumentation (see down below for an in-depth description), the quantized score has to be converted to show the articulations as graphical symbols. After clicking the *graphic conversion* button, the result can be seen in the graphical conversion tab. The graphical conversion is placed on a separate tab to make the score object bigger and allow easier inspection of the score.

3.3.8.1 Exporting the Score

It is possible to export the unquantized *bach.roll* as a *PWGL* file or *OpenMusic* file. The *bach.score* can be exported to more file formats: XML, MIDI, *PWGL*, *OM* and

Lilypond. The articulations will be exported however the instrumentation will only be visible in the Max patch.

3.3.8.2 LLLLs: Lisp-like Linked Lists

bach uses Lisp-like linked lists. This implies the use of parentheses to make the hierarchy of the lists clear. In order to send a list to the *bach.roll* notation object, a list has to be properly formatted to be used. A list for a voice is contained within one level of parentheses. Thus if the list output by the mapping module is 1 2 3 4 5 then it will be wrapped in one layer of parentheses `( 1 2 3 4 5 )`. In order to control four voices, the four wrapped lists are joined using *mxj.join*. In case there is only one voice used, the others will be expressed with `null`.

3.3.8.3 The Different Parameters Explained

Pitch is expressed in midi cents. This makes it possible to use microtones. By default, the minimum division are semitones but using the tone division box, the user can select a smaller subdivision. If no data are received, all pitches will be middle C (midi cent value 6000). The default mapping range is between 4000 and 8000 midi cent.

Onsets are described in milliseconds and set the point at which a note starts, counted from the beginning. The onsets are calculated by adding the *n*th value to *n -i*.[9] This list of increasing values is sent to the onsets parameter. If no values are received, the onsets will be at a distance of 500 milliseconds.

Durations are expressed in milliseconds. Negative values constitute rests. If no data are received, all notes will have a duration of 500 milliseconds.

Velocity is expressed in MIDI values. If no data are received, all velocities will have a value of 100.

While pitch, onsets, durations and velocities are sent to normal inlets, articulations and instrumentation are sent to *slots*. Slots are different from normal parameters as they are tied to a specific note. Hence, if a quantization takes place, there will be no shift of data if notes are removed through the quantization. There are different slots possible, however, in this implementation I only use two text slots. The formatation is different, as each articulation/instrumentation has to be tied to a specific note. Hence a list 1 2 3 4 5 will be formatted as

```
( slots ( ( ( ( 10 1 ) ) ) ( ( ( 10 2 ) ) ) ( ( ( 10 3 ) ) ) ( ( ( 10 4 ) ) ) ( ( ( 10 5 ) ) ) ) ) )
```

Slots have to be declared at the beginning of the list and the type of slot has to be declared as well though a separate message. For articulations the slot number is 1, for instrumentation the slot number is two. The user can invoke up to 30 slots should he want to expand the patch).

Articulations use an intermediary *coll* to convert the mapped values to text articulations. There is a list of 21 available articulations in *bach*, but the use can always limit the available text articulations or modify the *coll* file.

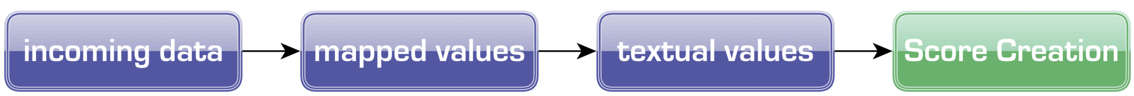

Figure 21 – *DataScapR* flowchart: the mapped values are mapped again to text expressions

Instrumentations use the same principle as the articulations: The data are converted into text expressions as well.

3.4 Real-time

The real-time patch is divided in three modules: data fetching/interpretation reading, the mapping-to-VST and MIDI module and a mixer module.

3.4.1 Data-fetching Module

The data-fetching module is built around a modified java-object (mxj StockWatch) from Max's included externals to fetch the data. The *mxj DataScapR* object receives stock market data in almost real-time[56] from the Yahoo Finance API[57] through a dynamic URL. This URL is consists of four parts:

1. the basic URL: http://download.finance.yahoo.com/d/quotes.csv?s=
2. the stock symbols are added at the end of the basic URL: http://download.finance.yahoo.com/d/quotes.csv?s=AAPL,GOOG Multiple stocks are separated by commas.
3. Each stock property is represented by a letter[3]. Adding these letters will get the desired information. For example, In this URL, the name n, stock symbol s, Last Trade Price l1, the open o and closing price p are are asked using f=nsl1op

   http://download.finance.yahoo.com/d/quotes.csv?s=GOOG,AAPL&f=nsl1op
4. Finally to download the data, one adds &e=.csv

   http://download.finance.yahoo.com/d/quotes.csv?s=GOOG,AAPL&f=nsl1op&e=.csv ("csvQuotesDownload," n.d.)

The *mxj* object sends an update request data every 30 seconds. Although a higher refreshing rate is possible, this rate seemed the best choice, as it does not overload the server, which would result in service denials. This rate allows as well having sufficient differences in the data. It would not make sense to have to use the same trading price every when updating every second. Nevertheless, if the user wants to ask data at a higher refreshing rate, he can set the desired time interval using the number box located above the ticker.

---

[56] Yahoo delays the data for some markets for commercial reasons. A list of delay times can be found on http://finance.yahoo.com/exchanges. To get realtime data, the user has to subscribe to paid services. Hence, in order to keep *DataScapR* a free toolbox, it was impossible to use non-delayed data. An alternative to Yahoo Finance was Google Finance (started in 2006). However, Yahoo Finance seemed to have more users and getting information was easier, hence I chose to use Yahoo Finance.

[57] The API is not officially supported by Yahoo.

3.4.2 Stock Properties

Yahoo tracks up to 51 properties of a stock, shown below. Sometimes less data are available for smaller stocks or more exotic markets. In general we can say that for the main markets, all properties are available. In testing the software, I excluded properties that yielded no data or only showed N/A.

- Symbol
- Last Trade Price Only
- Last Trade Date
- Last Trade Time
- Change
- Open
- Days High
- Days Low
- Volume
- After Hours Change (Realtime)
- Ask
- Average Daily Volume
- Bid
- Book Value per Share
- Change from 50-day Moving Average
- Change from 200-day Moving Average
- Change from Year High
- Change from Year Low
- Change in Percent
- Currency
- Days Value Change
- Days Value Change (Realtime)
- Dividend Pay Date Trailing Annual Dividend Yield
- Trailing Annual Dividend Yield in Percent
- Diluted EPS EBITDA
- EPS Estimate Current Year
- EPS Estimate Next Quarter
- EPS Estimate Next Year
- ExDividend Date
- 50-day Moving Average
- Market Capitalization
- One Year Target Price
- Orderbook Realtime
- Peg Ratio PE Ratio
- Percent Change from 50-day Moving Average
- Percent Change from 200-day Moving Average
- Percent Change from Year High
- Percent Change from Year Low
- Previous Close
- Pricebook
- Price Estimate Current Year
- Price Estimate Next Year
- Revenue
- 200-day Moving Average
- Year high
- Year Low
- Stock Exchange
- Short Ratio
- Name

3.4.3 Interaction with the Datafetching

Upon opening the patch the tracking starts automatically with predefined stocks. The user can track up to eight stocks at the same time. To track a specific stock, he clicks on the stock symbol at the upper side of the patch window. A pop-up window will appear, prompting the user to enter the stock symbol. Alternatively, he can choose a stock from 17 market indices worldwide[58]. The data are displayed in a moving ticker as well as message boxes on the lower side of the patch. The output of the JAVA object is sent through a route object, sending the stock data to the correct message box. From these message boxes, the data are then send to a subpatch where the variables are unpacked, transformed from symbols to floats and sent to the mapping module.

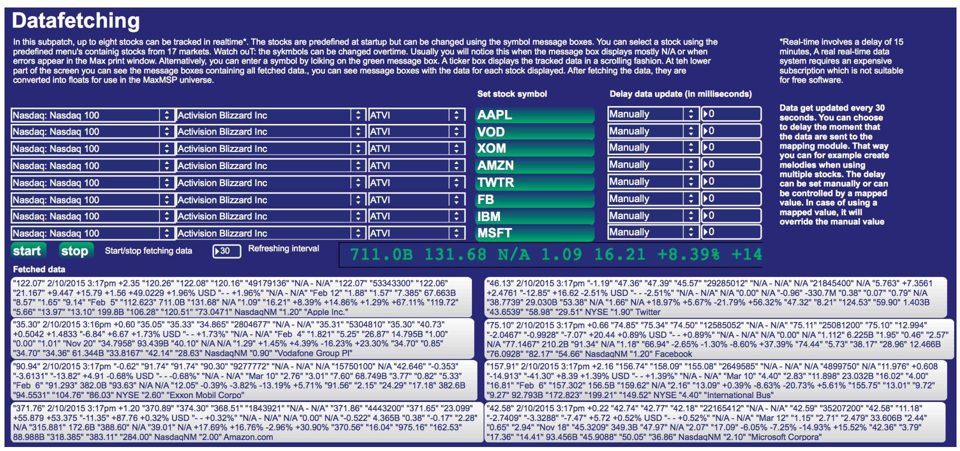

Figure 22 – *DataScapR* datafetching window allowing easy stock selection and a quick visual inspection of the received data

Initially, I included coll objects to store the real-time data. However, one would end up with partial datasets or very small ones unless one lets the system run continuously for a long time. As such, I disconnected the dolls from he incoming data. However, if the user wants to record the data, he would only have to connect the message boxes containing the data to the *coll* objects.

When the data are fetched they are sent to message boxes from where they are sent to the reading and mapping components. The data for each stock can be delayed so that

[58] The full list of indices is included in the help files of the toolbox.

updates do not come all at the same time. This can for example be useful to create compound melodies.

The data received through the *DataScapR* object are output as symbols in one string. Hence, they need to be unpacked and converted to floats. This happens in the datastorage and unpacking subpatch (not shown in presentation mode). For most properties a simple *fromnumber* is sufficient. However, there are three instances where more operations are necessary:

1) Data containing the percentages symbol % need to get stripped of that symbol. Herefore the symbol is first converted to ASCII values and the value for % is stripped using the *zl.filter* object. The values are then transformed back in a string after which the *fromnumber* converts the symbol to a value.

2) Data containing two points: some data properties hold two values such as the Days value change give the lowest and the highest change during the day. The symbols are sent through the *ATOI* and *ITOA* objects just as with the percentages. At the end, an unpack object splits the list in three: the lowest value, the - sign and the highest value. Only the two floats will be sent out.

3) Data with values higher than a billion: Some data have a very high market capitalization. Yahoo displays this data with a capital B (for Bilion) and T (for trillion) after the value instead of using many zeroes. As such, the B is filtered out and the resulting value is multiplied by 1000000.

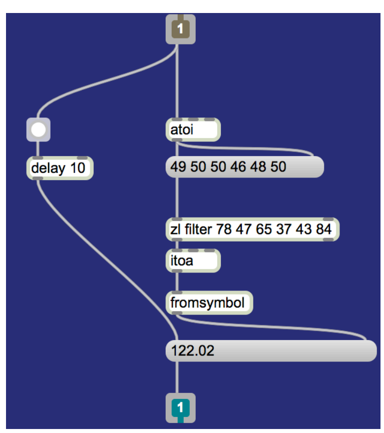

Figure 23 – *DataScapR* conversion subpatch window: the conversion from symbol to float. In case a data point is not available (N/A), the last received value will be sent out.

After the symbol is converted to a value, it is sent out via a send object, for example: the first stock's Last trade is send `1LastTradepriceOnly`.

Data mapping

To actually use the data in compositions, we need to map them using the mapping module. This mapping happens in the *DatatoVST* and *DatatoMIDI* patches. Each stock is linked to a VST and a MIDI patch. During development, loading the 16 abstractions at start-up resulted in loading times of over a minute. As most users will likely use one voice to start with, the user can load the abstractions when he needs them. It takes approximately four seconds to load one abstraction, which I deem acceptable. The abstraction will be added to the tabbed view at the top of the patcher window.

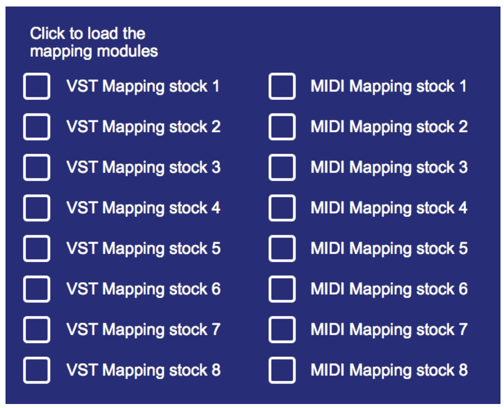

Figure 24 – *DataScapR* real-time main window detail. The user can load 8 *datatoVST* abstractions and 8 *datatoMIDI* abstractions.

3.4.4 Mapping Methods

The mapping module routes the data to the specific mapping method and subsequently their output destination. The patch is laid out so that the data come in at the left side, get mapped and go out at the right side. Using the menu, the user can select the data he wants to map. To the right of the data menu he can then select the mapping method he wishes to use. According to the selected mapping method, the layout of the following number boxes changes.

There mapping methods are almost the same as in the historical data patch: a scaling of the incoming values using the *zmap* object, a scaling of incoming values using function to scale the values, a modulo operation and a modulo operation followed by a function object. The difference lies in the incoming data: The user can choose 53 + 2 data points to map. While in the historic data components, one can contextualize all data; the nature of the incoming data in the real-time does not allow this. However, Yahoo offers the minimum and maximum day- and year-price. Hence, it is possible to use contextualization for the last trade. The contextualization does not have to be activated as in the historical data patches; just selecting the contextualized data in the menu will send a list containing the value and its context values.

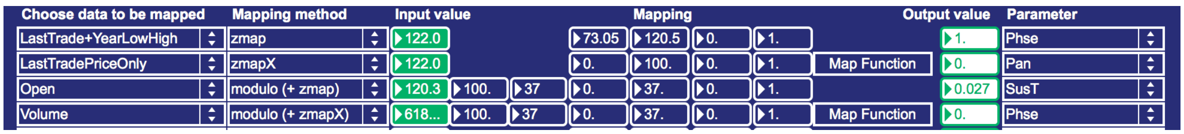

Figure 25 – *DataScapR* mapping window detail showing the four different mapping methods. On the left side on selects the incoming data, followed by the mapping method. The input value is displayed, followed by the mapping values (input range, output range, modulo) and finally the resulting mapped value. The rightmost menu is used to select the VST or MIDI parameter to send the value to.

3.4.5 Output

The mapped values can be sent out to a VST or a generic MIDI instrument. This is equal to the row-by-row component.

3.5 Evaluation and future work

As shown above, *DataScapR* offers many possibilities to use stock market data in the creation of artworks. Let us revisit the requirements set out at the end of chapter 2:

1. The application has to be available: *DataScapR* is available for download on a dedicated blog datascapr.wordpress.com
2. The application has to be flexible: While offering an easy start to use the toolbox, it is flexible to be adapted to one's specific needs
3. The software should not be too broad: *DataScapR* is admittedly limited in its operations. However, in theses limitations, we can understand its workings better

and improve our interaction with it. If the toolbox would be too broad, we could lose the focus over our intended outcomes.

4. There is only one type of dataset used; *DataScapR* uses a specific type of datasets (Yahoo Finance datasets), which allow the toolbox to be consistent in its operations. If we had to cater for every possible dataset, it would inevitably make the toolbox bloated.

From the above, we can say that *DataScapR* has achieved its objectives but there is always room for improvement and extension. Beyond this research, *DataScapR* will continue to be developed to ensure that the project does not become obsolete immediately. Let us list future improvements:

1. At the moment, *MaxMSP* 7 is still not stable enough. With its release in January 2015, I deemed it irresponsible to transition immediately to the new version due to some stability issues. Furthermore, *bach* had compatibility problems, which caused *MaxMSP* to crash intermittently. In a few months, when Max will be stabilized again, *DataScapR* will make the transition to version 7.
2. Currently, there are basically two mapping methods: the linear method (using *zmap*) and the non-linear (using the *ej.function* object). Using mathematical expressions can be an interesting third way of setting up mapping rules.
3. Inherently present in the current version of *DataScapR*, one can use the toolbox for sound design. By using the row-by-row module and controlling a VST, one can effectively discover new sounds. The case studies in the next chapter were focused more on the structural compositional level, but one can use *DataScapR* for sound design. If one sets up the VST parameters and reads a dataset, one can discover new and interesting sounds that he would not have encountered without using this process.

4 CASE STUDIES

DataScapR is intended to create music using stock market data. After having explained the structure and workings of the toolbox in the preceding section, it is time to show concrete examples of how one can use the toolbox to create music, be it in the form of fixed pieces or installation art. Four works created using *DataScapR*, each one focusing on a different component of the toolbox, are presented and analysed:

- *For a Fistful of Data*, for solo recorder, using the *score creation* module;
- *Vapourwaves*, an installation work, using the *real-time* component;
- *4D Brokers*, a multichannel work/sound sculpture, using the *Row-by-row* component;
- *Mirage*, a fixed-media piece, using the Row-by-row component.

More specifically, the concept and the used mappings are discussed and the case studies are related to the concepts posited earlier in the dissertation. These case studies offer a glimpse of what is possible using *DataScapR*: in no way these case studies show all possibilities. Nevertheless, they can give an insight of how the toolbox can be used in a practical way.

4.1 For a Fistful of Data

For a Fistful of Data is a piece for solo recorder. The monthly data, from *Amazon.com* starting at 16 May 1997 (the day after *Amazon's* IPO, Initial Public Offering at *Nasdaq*) until 1 May 2014[59] are the source material and amount to 210 entries. The dataset as a csv-file and the score are included in the appendix.

[59] I actually requested the start date to be 15 May and the end date as well. However, Yahoo divides the month in two and hence gave me 16 May as the start date. From there on, all data are from the first day of the month; hence the end date is the 1 May.

4.1.1 Instrument Choice

Choosing the recorder was a purely subjective choice. Cecilia Peçanha is my partner, which facilitated trying out the piece. Although there were some technical limitations inherent to the instrument, I used those to my advantage.

4.1.2 Choice of the Dataset

The choice for *Amazon* was based in the first place on a personal preference for the company. *Amazon* is without a doubt one of the most disruptive companies in the last decades: starting as a small bookseller, they managed to become one of the world's largest stores through mass scale operations, thereby having a big impact on traditional high-street commerce.

In the second place, besides its impact on traditional commerce, the company seems to defy the laws of the financial markets. Despite posting meagre profits in comparison with the total revenue, the stock has been steadily rising. *Amazon's* CEO, Jeff Bezos, sees the company as a long term project and hence sees no negative side to the small profits.

The disruptive nature of *Amazon* combined with its odd stock-pricing make it an interesting company to have a closer look at.

4.1.3 Mappings Used

The candlestick chart below shows us the monthly evolution of the stock price while the bars at the bottom of the chart show the trading volume. Furthermore, the sloped line underneath the candlesticks show the exponential moving average[60], the average stock price in the 50 preceding data points.

[60] More specifically, the exponential moving average places more weight to the most recent prices as opposed to a simple moving average where all prices are treated equally.

Figure 26 – *Amazon*'s stock price evolution (Yahoo Finance, 2015a)

4.1.3.1 Pitch

It became clear that the rising stock price would be a focal element of the composition. However, mapping the price to the pitch (using the whole register of the recorder between C4 and D6) in a simple linear fashion (using the *zmap* mapping) would prove uninteresting. First, the price stayed in a low register for a long time (for example, the EMA only passed the $ 100 mark in 2011). Furthermore, there are no big shocks in the stock price. Hence, I decided to map the contextualized opening price using the *modulo* operation: multiplying the value by 100 (to 'get rid' of the floating values) then performing a modulo 54 operation (there are 27 notes available in the recorder's register but the double value seemed to yield a more interesting result). Finally the resulting value was mapped to the register of the recorder. The modulo mapping yielded a very dynamic pitch line, inevitably making the piece more interesting than it would have been with a simple linear mapping.

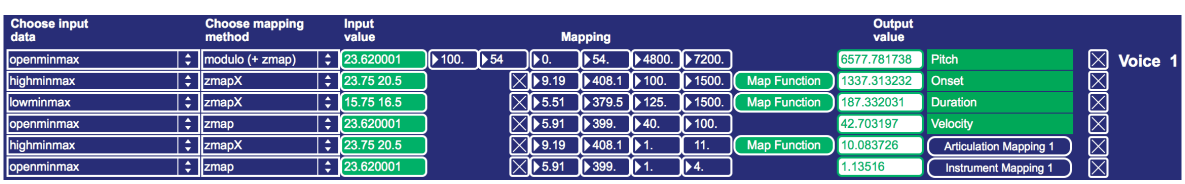

Figure 27 – *For a Fistful of Data:* an overview of the mappings used. The top row shows the pitch mapping.

4.1.3.2 Onset and Duration

Although the rising stock price was not interesting to use for the pitch, I envisaged to make this rise perceptible is some way. Hence, the onsets of the notes would be inversely correlated with the price: the higher the price, the closer to each there the onsets were. On the contrary, the notes' durations were correlated with the price so that the durations would be longer when the price went higher. Through experimentation, I chose to map the contextualized highest price (for the onsets) and the contextualized lowest price (for the durations) to use the *ZmapX* mapping, which allowed me to put more weight on certain values. While a purely linear mapping showed a potential interesting result, the non-linear mappings proved to give more interesting results.

The mappings resulted in a dual piece: the first part shows spare notes while in the end, the notes are spaced very closely to each other and the notes continue when a new note has been started already.

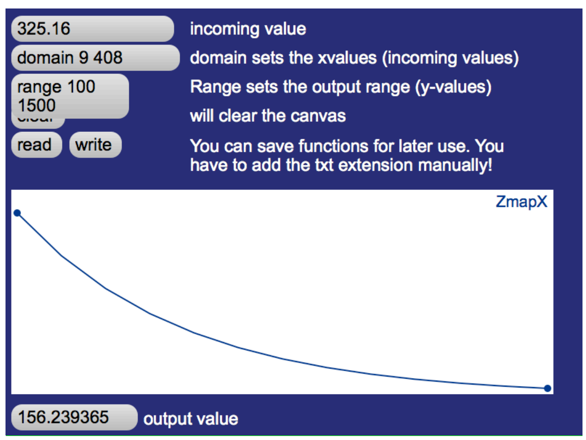

Figure 28 – *For a Fistful of Data*: the sloped curve causes the low values to have onsets that are placed far from each other, while the higher values will result in smaller onset values, hence placing the notes closer to each other.

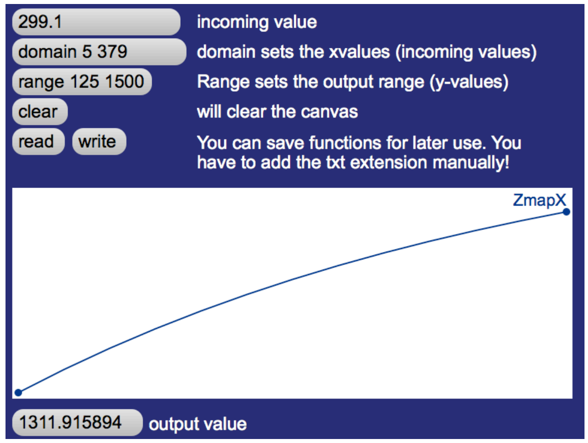

Figure 29 – *For a Fistful of Data*: the durations use the inverse of the onset mapping: low stock prices result in short notes, while higher prices yield longer notes.

4.1.3.3 Velocity

Velocity uses the contextualized opening price and a linear mapping method (see the overview image). The output range is from 36 to 100. I read those values later on manually and wrote the related dynamic symbols on the score. There are two places where the velocity is high: in measure 9 (just before a stock split) and in the last part of the piece where the stock price is rising a lot.

4.1.3.4 Articulations

As the notes in the beginning of the piece would be spaced quite distant one from another, I wanted the articulation to be a part of the piece. The default articulation list of the *bach* toolbox is not tied to a specific instrument and some articulations are not available. Hence, the default *articulationcoll* file was replaced with a different one where the non-applicable ones (for example upbowing and downbowing) were removed. The final list is as follows:

1) bogus;
2) staccato;
3) accentstaccato;
6) upmordent;
7) trill;
8) downmordent;

4) tremolo; 9) accentstaccato;

5) staccatissimo; 10) tremolo;

Due to limitations of *bach*, some articulations were not included so placeholder articulations were used. Tremolo is replaced by flatterzunge, accentsstaccato is replaced by slap tongue. Finally the upmordent and downmordent can be replaced by an upwards/ downwards glissando of approximately a half tone (notated with a hairline). These changes are done through editing the score in Finale. The performer has some liberty in interpreting the articulations: for example, the glissando's duration is not defined exactly, so the performer can let the glissando protrude in the rest after the note .

The contextualized highest price was mapped using the *zmapX* mapping method. This choice was based upon the desire to establish a progression from a more gesturalized characteristic (using flatterzunge, mordents) to a more percussive sound. Although articulations, such as the mordents and trills, expand the duration of a note, I consider these expansions to be like a ripple effect on water: The stock has a continuing impact that fades away.

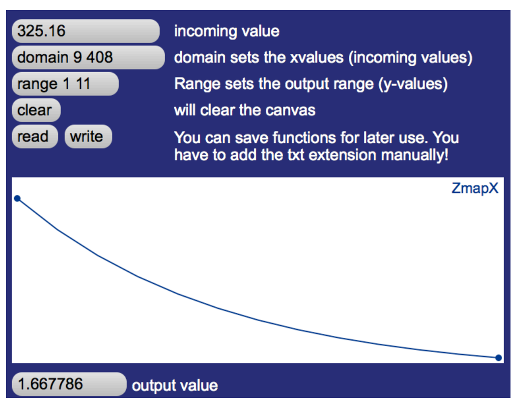

Figure 30 – *For a Fistful of Data*: The sloped inverse mapping gives more emphasis to the expanding articulations. The staccato articulation then only appears when the stock is really high.

One final note regarding the mappings: as the piece is for solo tenor recorder, the instrumentation mapping was not used.

4.1.3.5 Quantization

The mapped values were sent to a *bach.roll* object where they were quantized using the `minimalunits 1/16 1/20 1/24` command line. The time signature and tempo were set to 2/4 and `60`. As such, the lowest division of a beat was a sextuplet, which seemed readable enough. I did not want to use smaller divisions to avoid making the piece needlessly complicated.

4.1.4 Changes After the Mapping (or Bricolage)

While *DataScapR*'s output was interesting, it was not a final score. Some changes were necessary, some depending on technical characteristics of the recorder, others on aesthetic grounds.

After having mapped the data, the resulting material was exported as an xml-file and imported in Finale. In a first reading, the score was cleaned up from anomalies (*bach* had switched upmordents and downmordents with each other) and the dynamics were added.

In the last part of the piece, due to the mapping of onsets and durations, the notes ran over in each other. Of course, playing complex chords on the recorder is impossible hence these chords were transformed in fast arpeggios. Playing these arpeggios at a fast speed would give the illusion of a continuous stream hence this change seemed justified. However, the tempo as it should be was 360 for a quarter note, which was completely unpleasing. Slowing the tempo down allowed the notes to come through while still retaining the sense of a rising stock price. At some places the articulations were the same for various notes. Hence, to introduce variations, the upmordent/downmordent could be replaced with small glissandos of a half tone. Furthermore, on some places the articulations were changed on subjective grounds.

The arpeggios in the last part of the piece were very fast; hence, to facilitate playing, I reduced the tempo while retaining the ratios in that part. Of course, there is a change of tempo then but this change and the other ones described above can all be reduced to *bricolage* decisions: the overall structure of the piece is preserved.

4.1.5 Evaluation

In this piece, it became clear that working with *DataScapR* is an empirical process: Through observations of tried out mappings, I came to the mapping that yielded the best personal subjective aesthetic response. Various mappings, while showing interesting results in one parameter showed flaws in other parameters. Through experimenting, going back and forth between various combinations, I finally settled at the mappings as described above. The going-back-and-forth shows the adherence to Mazzola's idea of creativity: cycling through the seven steps I questioned the walls and searched for ways to break those walls. Of course, the limitations reflected on my thought (hereby I refer to Vaggione's idea of using the walls at one's advantage). For example: the limitations of the recorder made me rethink the way the chords should be played. Technology and the result became mutually determinative: as some results were more pleasing I pursued them while other results were discarded. While I had a general structure in mind (mainly based on the general rise of the stock price), the different results incentivised me to change the input data or the mappings. For example, some outcomes generated very large onset spans. To have the notes' onsets closer together, I experimented with different mapping values. Finding the best outcome was a going back and forth between the technical decisions and the aesthetic experiences that arose each time I obtained a temporary artefact.

The composition is not completely objective[61]: subjective decisions are present in the piece. I do not think that this impoverishes the piece, quite to the contrary: combining objectivity with aesthetic decisions can lead to a more compelling piece.

From the arguments above, it has become clear that this piece (as well as the others) is meant as a *manifestation*: The goal was to create a piece where the data served as an inspirational starting point. One will not be able to understand the stock's movement and use that knowledge to trade but there is a kind of *revelation* happening: we experience the stock market phenomenon in a different way and can appreciate it in that way.

[61] No composition can be completely objective: there will always be some kind of personal decision present.

4.2 4D Brokers

4D Brokers is an experiment using the 4D sound system. The 4D sound system is an array of 45 loudspeakers arranged in a grid of 5x3x3. Through a custom application, the system interacts with *Ableton Live*. During a workshop at the *Amsterdam Dance Event*, I got the chance to use this system. While the work is not presented in its definitive form (nevertheless it can be seen as a work-in-progress), I nevertheless wanted to include it in the dissertation as it shows the potential of spatialized sonification.

4.2.1 Data Used

4D Brokers uses historical data from the *IBM* stock: the data go from January 1995 to October 2014, with one data point for each month. This resulted in a dataset of 237 data points.

Figure 31 – IBM's stock chart (Yahoo Finance, 2014)

Using the *4D* system, the space was divided in 20 equal slices, and each slice was assigned a sine wave. Each slice represented a data point and through reading the data, 20 slices were constantly changing in content. As such, one would effectively walk through time-space. *Shape of things past*, a work by Joachim Sauter et Dirk Lüsebrink (Sauter & Lusebrink, 1995), can be considered an inspiration for this work. *Shape of things past* is “a parametric translation of films into space”. Individual frames are placed next to each other, according to the movement of the camera with which the sequence was shot. The result is the appearance of "single pixels (picture elements) into space, objects of voxels (volume elements)" (Sauter & Lusebrink, 2006). *4D Brokers* is similar in the translation of a one-dimensional phenomenon in three dimensions. The difference of course being that I used audio instead of visuals. Besides the frequency of the sine wave, each slice's size (height and width) was controlled by data as well.

4.2.2 Mappings

The mapping used contextualized values and was done as shown in the table:

Table 4 – *4D Brokers*: input data, destination and destination range

Data	Parameter	Parameter values
Last Trade	Sine wave frequency	20-440 Hz
Volume	Rectangle height	0-4 (meter)
Spread Open-Close	Rectangle width	0-16 (meter)

The presentation consisted of a 3-minute run of the system. The limited time necessitated the use of a more radical, dynamic mapping method to allow maximum diversity in the short timespan. To make the mapping result as diverse as possible, I used the *modulo* mapping. This would create big differences in the mapped values, even when the difference in the input values would be minimal.

As an additional layer, I correlated a linear increasing amount of distortion to the slices as they went back in time. This would be comparable to the obfuscation that takes place over time in remembering events.

As the 4D system worked with Ableton Live, I created a custom version of the *DataScapR* patches to use with Max4Live.

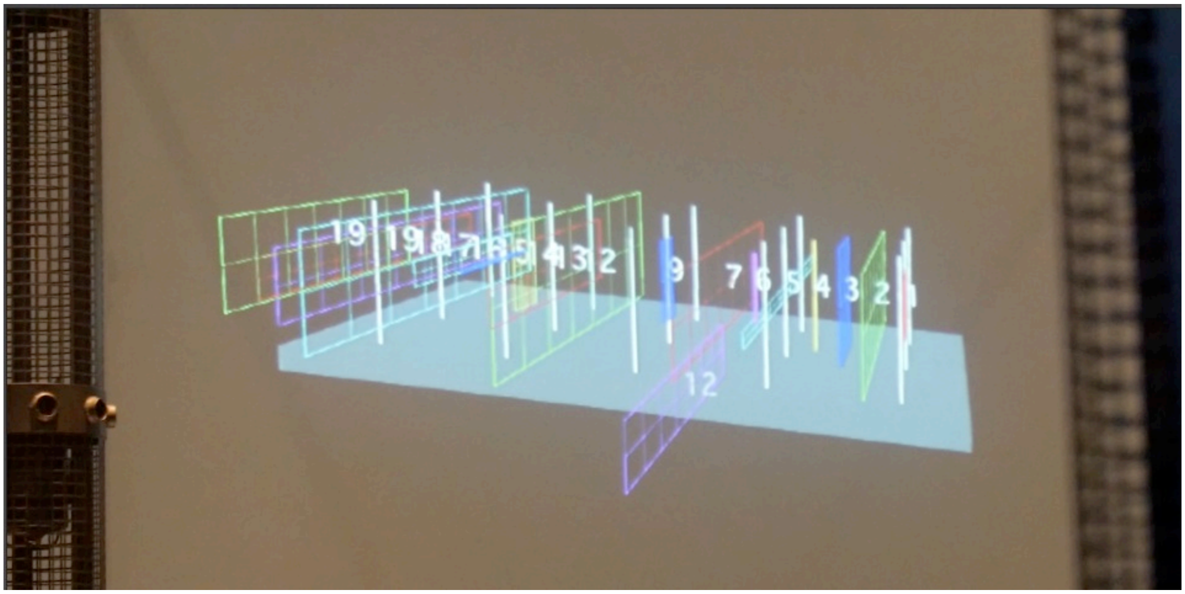

Figure 32 – *4D Brokers*: The 20 slices spaced at equal distances. The oldest slices' sounds were more distorted to symbolize the disappearance of the data in the past.

4.2.3 Evaluation

4D Brokers was the first work in which *DataScapR* was used 'in the wild'. As such, it was an excellent opportunity to assess the toolbox capabilities. The toolbox proved to be solid and provided everything necessary to complete the project set out for the workshop. The flexibility was proven by adapting the patches for use inside *Max4Live* and the use of sine waves instead of VST's. Originally, I wanted to use real-time data. However, as the event was taking place in the weekend, there were no markets open and I resorted in using historical data instead (I would use real-time data later in *Vapourwaves*, see below). This change proved to be beneficial as it allowed more time to experiment with a restricted dataset, which allowed repeated testing and comparison. The short time to demonstrate the work forced me to use a more radical mapping than intended. However, the groundwork laid in *4D Brokers* would be used in the later works.

4.3 *Vapourwaves*

Vapourwaves is an installation work that tracks stocks on eight markets located all over the world. The aesthetic I am trying to convey in this work is one of dehumanization and ephemerality: the markets are a phenomenon where the human element is increasingly absent and where the last received data will soon remain just a vague memory.

The name *Vapourwaves* is a name in reference to the Vaporwave genre, “a micro-genre of electronic music that draws on the corporate sonic ephemera of the 80s and 90s – such as lift muzak, ad soundtracks, ‘hold’ music and cocktail jazz – to satirise the emptiness of a hyper-capitalist society” (Ward, 2014). This emptiness is certainly present in *Vapourwaves*: the stock’s last prices are just ephemerous phenomena and they soon dissolve in the vast ocean of data, only a vague memory remains.

4.3.1 General setup

The installation tracks stocks from eight markets located in different places of the world. The markets were chosen for their availability of data (for some markets, Yahoo does not provide data) and for their relative place in the world (I wanted the markets to be somehow distributed in different parts of the world)[62]. For each market, I took the stocks in the main index and so I could choose from them to change a stock if I wanted to. The markets chosen are shown in the table below. If the market would be closed, we would get stuck with a non-changing value. To make variation possible I change stocks every hour, choosing a stock randomly from the market’s index.

Table 5 – *Vapourwaves*:The eight cities, their markets, indexes and longitude

City	Stock Market	Index	Longitude
New York	New York Stock Exchange	Dow Jones Industrial Average (DJIA)	-74.011276
New York	Nasdaq	Nasdaq 100	-73.985489
Mexico City	Bolsa Mexicana de Valores, BMV	Indice de Precios y Cotizaciones (IPC)	-99.164322
São Paulo	Bolsa de Valores, Mercadorias & Futuros de São Paulo (BM&F BOVESPA)	Ibovespa	-46.722172
Hong Kong	Hong Kong Stock Exchange	Hang Seng 50	114.15768

[62] The only exception on this rule is the proximity of the New York Stock Exchange and the Nasdaq. I believe that these two exchanges are the centre of stock trading hence the exception on the rule.

City	Stock Market	Index	Longitude
Moscow	Moscow Exchange	MICEX 10	37.606232
Sydney	Australian Securities Exchange	S&P/ASX 20	151.208527
London	London Stock Exchange	FTSE 100	-0.099034

To create *Vapourwaves*, I used an adapted version of *DataScapR*. In the datafetching module I added a *metro* object that would send a bang every hour to change the stocks being tracked, the mapping module was repurposed so to contain all mappings for all parameters I wanted to control. Furthermore, I created a separate patch for the oscillatorbanks, the granularisation and the MIDI handling for the Disklavier. All sounds are mixed in a separate mixer patch.

4.3.2 The Different Instruments Their Mappings

Vapourwaves uses three distinct voices: eight oscillatorbanks (one for each stock), the *Yamaha Disklavier* and a granularisation of a soundfile[63] of traders screaming on the pit floor.

To control the oscillatorbanks, I used the Last Trade Price, contextualized with the Year's Low and Year's High as the input values. These values were mapped onto a frequency range between 27,5 Hz (which corresponds to A0, the lowest note on the piano) and a varying upper limit between 500 and 2000 Hz[64]. To ensure variety in the sound, the upper limit would change every 15 minutes. As such, if the markets would be closed, there would be still variation possible. The mapped value was sent to an oscillatorbank, which would hold the 28[65] last values, with the older ones having smaller amplitudes. The overall amplitude for each oscillatorbank was dependent on the opening or closing of the market: if the market were open, the volume would be higher.

[63] The file was downloaded from the freesound.org website

[64] The other upper limits being 750, 1000, 1250, 1500 and 1750

[65] The number 28 refers to my birthday. The value 28 (and multiplex 56 and 84) are also used in other works.

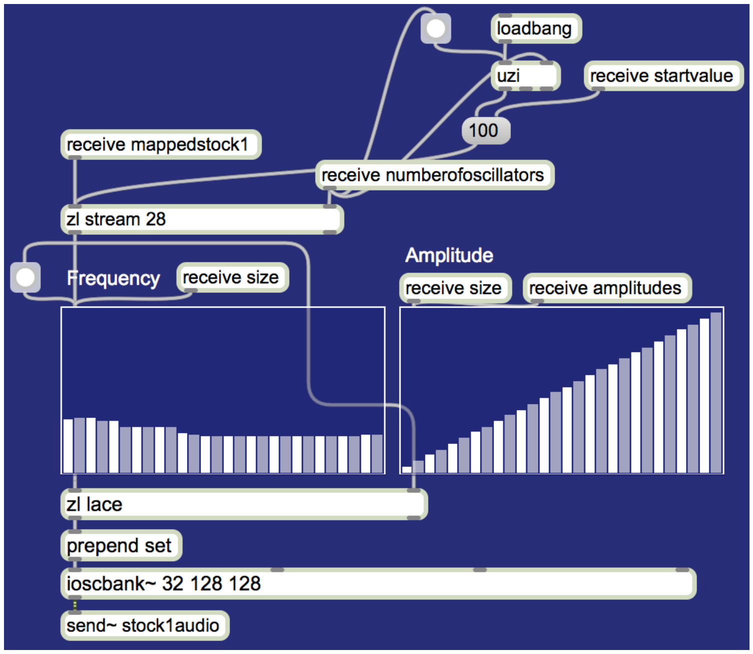

Figure 33 – *Vapourwaves*: the oscillatorbank for stock 1. The last value is sent to the zl stream object which outputs the last 28 values it received..

The frequency sent to the oscillatorbanks was translated to MIDInotes using the *ftom* object so the *Disklavier* could play the notes and enhance the oscillatorbanks' sound. As *DataScapR* gets new data every 30 seconds, I delayed the notes to allow a melody to appear. To set the amount of delay, I used the Last Trade Price and sent those data through the *modulo* mapping: First the incoming value was multiplied by 100 to get rid of the decimals and then send to a *modulo* object. The resulting value was then mapped between 0 and a varying upper limit between 15000 and 29999[66] milliseconds. The resulting value would then delay the note to be played so a melody would arise. I used the same modulo operation to control the velocity as well, however, the output values were between 45 and 85 when the market was open and between 25 and 65 when the market was closed. The note duration is fixed to 1000 milliseconds. However to allow sympathetic vibrations to arise, I put a weight on the sustain pedal to keep it down permanently. As a result the notes last much longer.

---

[66] The other upper limits being 17500, 20000, 22500, 25000 and 27500. As the value sets a delay, the note will only be played after a certain amount of time. If I use the value 30000, there is the possibility that the delay can be 30000. Consequently, this would cause a new value to be loaded before the previous delay value (of 30000) had been used and potentially 'silence' the note. Hence, the highest upper limit must remain 29999.

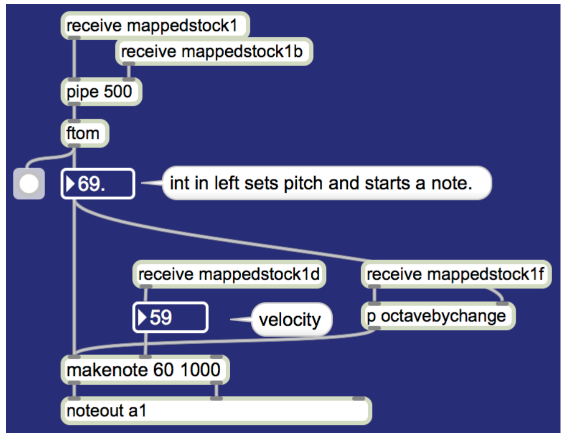

Figure 34 – *Vapourwaves*: the MIDI patch for stock 1. The mappedstock1 value will set the MIDI note, the mappedstock1b value the delay with which the value is sent out. Mappedstock1d sets the velocity and mappedstock1f receives the change in percent.

To enhance the *Disklavier*'s voice I used an octave switch: if the change-in-percent would be higher then 1, the MIDI note would be played together with a transposition of one octave higher. If the change would be greater then -1, the MIDI note would be played together with a transposition of one octave lower.

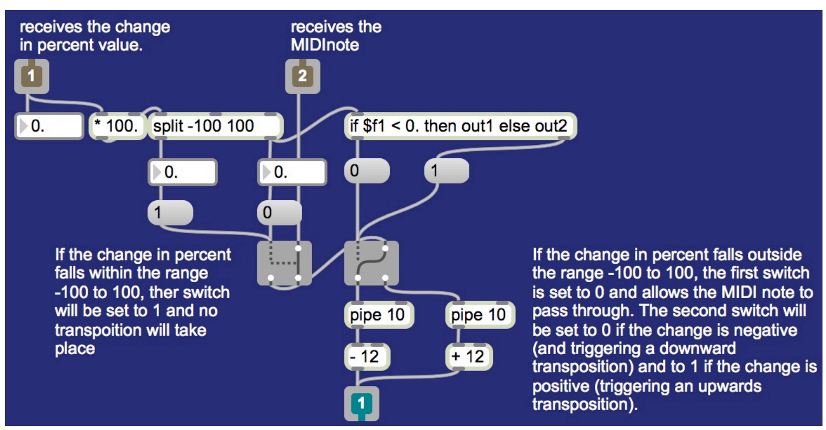

Figure 35 – *Vapourwaves*: the octavechange subpatch will add a transposition of the MIDI note if the change in percent is large enough.

The sound file with the screaming traders was controlled only by the stock of the New York Stock Exchange, as it is the only exchange used in this installation where the open outcry system is still used (all other markets are electronic). To muddle the voices and to avoid placing focus on the meaning of the words, the mapping was chosen in such a way that the soundfile would be transposed down for most of the values. I used the change in percent variable as the data input and set the input limits to -10 and 10. The output limits were 0 and 64. Instead of using the simple *zmap*, I used *zMapX* which allowed me to make the mapping non-linear: the bigger the change in percent (either negative or positive), the less transposition there would be so that with bigger changes you would hear the people scream clearly.

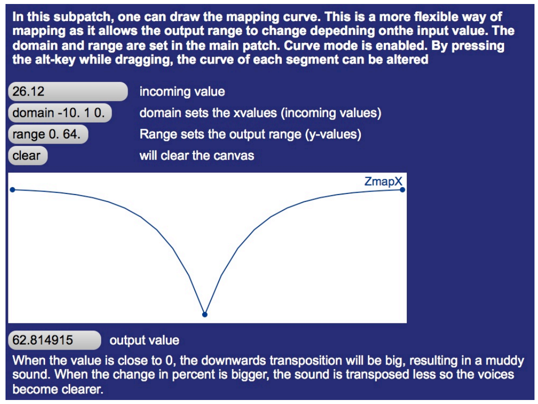

Figure 36 – *Vapourwaves*: the ZmapX ensures that transpoition is the biggest with a change in percent value that is close to 0.

There were some glitches at irregular intervals. I was not able to trace those glitches back to the system but they are a nice addition as they give an unexpected accent to the sound.

4.3.3 Spatialisation

I originally conceived *Vapourwaves* for the surround sound system in the MoCap hall at the Catholic University of Porto. The installation follows eight stocks at eight different stock markets around the world. Their position in the hall would be correlated to their position on a world map. These eight sound sources would then move in a 24-hour time period over the hall so that you would get a dynamic sonic constellation. Unfortunately, unforeseen circumstances made it impossible to use the 24 speakers hence I re-imagined the installation and used a stereo system. While it obviously it is a smaller setup, the reworking as described below is technically less resource demanding and thus has bigger chances to be presented in other places that do not have a surround system at their disposal.

I mapped the eight stock markets' longitude coordinates to the left-right stereo range. This resulted in the following disposition.

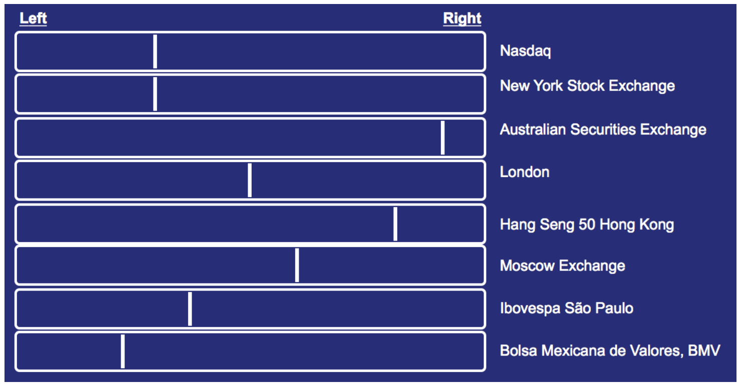

Figure 37 – *Vapourwaves*: the initial locations of each market in the stereofield

The oscillatorbanks and the granularisation are tied to the markets locations. It might be difficult to distinguish the different sound sources in a small space. However, the sound sources move to the left in a time period of 24 hours. When they reach the left side of the stereo field, they will flip to the right and start moving again to the left. This creates a change in the sonic constellation: what you were hearing for a while on your left side suddenly appears at your right side. The *Disklavier* is placed at the opposite side of the room. As such it creates a sonic reflection to the oscillatorbanks.

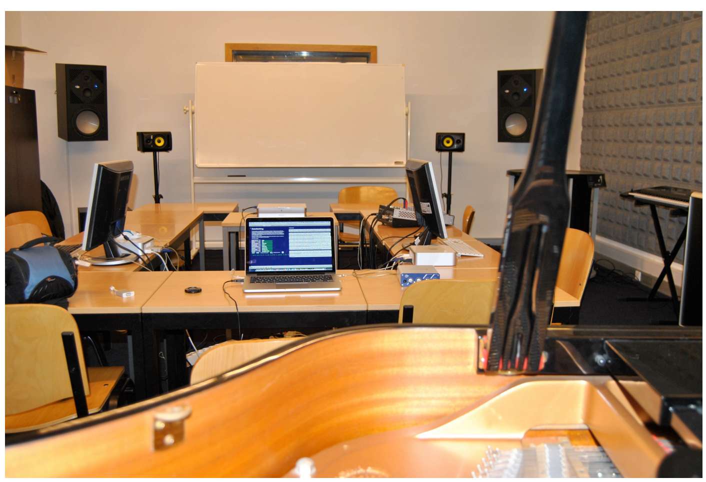

Figure 38 – *Vapourwaves*: the setup at the post-production room

4.3.4 Aesthetic Considerations

By using sine waves I wanted to create a dehumanization of the sound: computers do most trading nowadays, hence the sterile character of sine waves seemed a good way to bring forth this characteristic. Furthermore the eight drones create beating frequencies which are not present in the system but only appear as a result of it. Hence I named this work *Vapourwaves* to symbolize these ephemerous frequencies.

The *zl.stream* object records the last 28 frequencies of the stock. This results in a local drone where beating frequencies appear as well. Besides the vapour waves that arise, the 84 frequencies bring forth the history of the stock. The diminishing amplitude of the older frequencies makes them disappear slowly, receding in the vastness of the ocean of data.

As I argued in an earlier section, technology is not neutral: it shapes our practice. In *Vapourwaves* this idea is brought to the foreground: the failure of the surround sound technology forced me to rethink the spatialisation aspect. While the spatialisation aspect was apparently impoverished, the presence of the *Disklavier* compensated for this. Without the *Disklavier* I would have used a VST, but in the final result, I believe that the presence of the *Disklavier* enriches the installation through its physical presence and natural sound.

To contrast with the sterile sine waves, I added a granularization of sound files of traders on the stock market. This contrast is meant to create an opposition between the technological (sine waves) and the human emotions (the voices). The *Disklavier* is in the grey zone between the two. To avoid that the public would try to discern the meaning of the words, I mapped the data in such a way that most of the time the sound would be transposed down so that it would sound muffled. Only when the percentual changes were big enough, the transposition would be less so that the voices would be more prominent in the whole. In other words: when the stock is more dynamic, the traders are brought in the foreground.

The combination of the three voices is meant to follow the rule of three as three elements inherently prove to be a more interesting feature than just two opposing voices[67]. Indeed, the first idea was to use only the oscillator banks with the *Disklavier* as its opposite. Adding the third voice helped in balancing those two voices: One can link all elements without giving precedence to one other. On one hand, the sine waves and the *Disklavier* form a unity as they both use the same value to control the frequency or pitch. On the other hand, the granularisation and the *Disklavier* form a unity, as they are more physical: the voices are human; the *Disklavier* has a physical presence. However, we can link the sine waves to the granularisation as well: both are outputted through the stereo speakers. From the above we can thus see that all elements are entangled together in a way that each one enriches the other. Take one element away and the sound would be less interesting.

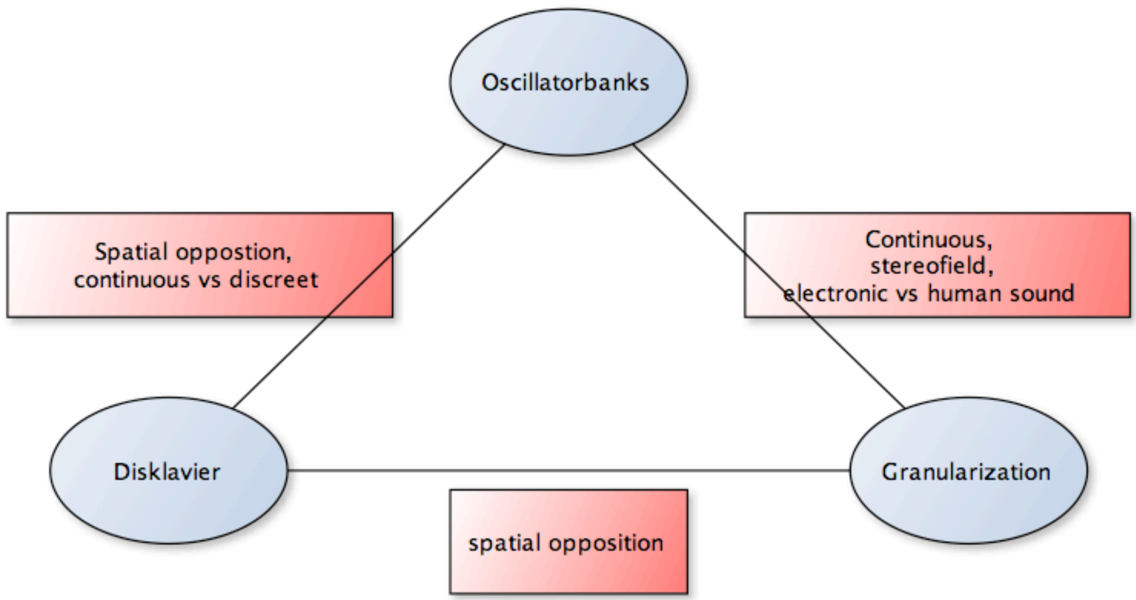

Figure 39 – *Vapourwaves*: the triangular entanglement. Each element can be linked in a relationship.

While recording the installation work, there were some glitches at irregular intervals. I was not able to trace those glitches back to the system but they are a nice addition as they give an unexpected accent to the sound.

[67] I hereby refer to the rule of three which asserts that things that come in three are more satisfying.

Although the room was not apparently important for the setting up of the work and I easily could ignore it, the post-production room where I installed *Vapourwaves*, seemed the ideal place to show this work: During the week I prepared the installation there, I was alone, totally isolated from other rooms. Furthermore, the technology present in the room could be seen as a dehumanized aspect: filled with technology but devoid of humans (besides the public), the room became an ideal place to show the dehumanization of the stock markets.

Figure 40 – *Vapourwaves*: a room where technology rules.

I made a recording of the installation on Tuesday 27 January between 6 pm and 7 pm. As the recording was less than one hour, the stocks tracked (shown in the table below) remained the same. For the recording I used a Zoom H6 handheld recorder, positioned at the middle left side of the room. This recording setup does not completely convey the panning in the speakers; however, I deemed it the best option to hear the Disklavier. Of course, the best is to experience the installation in person.

4.4 Mirage

Mirage is a fixed-media piece using the *Row-by-row* component. The piece tracks the data of *Citigroup*, one of the biggest US banking corporations over a period of 20 years ranging from 1 July 1994 to 1 July 2014, which amounts to a dataset of 241 entries. A recording is included on the DVD.

4.4.1 Choice of Dataset and the Compositional Concept

Citigroup's stock evolution shows an interesting rise in the early 2000's followed by a big fall in 2008 when the housing crisis started in the USA. This fall brought Citigroup back at the level it had in the 1990's. The spectacular rise to 10 times the price in the 1990's can be considered an illusion of what the company was worth. Indeed, the deregulation that started with Clinton's administration allowed banks to go on financial adventures, realising constantly rising stock prices. The crash in late 2007 can then be seen as a wakeup. The mirage or illusion ceases to exist and business returns to realistic levels.

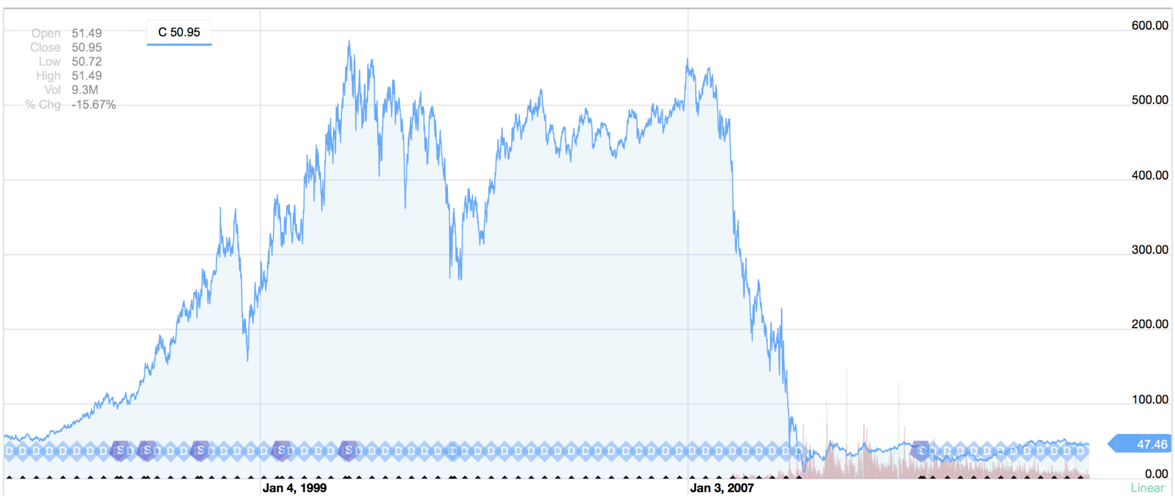

Figure 41 – The evolution of *Citigroup's* stock price over 20 years (the small icons at the bottom of the chart denote stock splits and dividends) (Yahoo Finance, 2015b)

Although the idea of a mirage is only part of what made Citigroup stock price rise and fall spectacularly, I think that the idea that people become delusional when stock prices go up is compelling. I this piece I wanted to convey the transformation of the realistic into an illusory one, hence the name *Mirage*.

In keeping in line with the other works presented above (or: having learned from earlier experiences), I wanted to create a slowly-evolving piece, which unfolds over time. A drone-like sound is combined with sparse sound events, both procedural and individually triggered ones. The piece consists of three voices, contrasting and complementing each other. The use of resonators and reverb makes the continuous sounds roll over in each other which references again to the history aspect used in the other works: the stock price has an influence on later events or: *past becomes future*.

4.4.2 Mappings

There are five elements used: two continuous sounds and three voices consisting of individually triggered events. All data use only the adjusted closing price, as it was the most dynamic line in the dataset. This also made the piece consistent in the sense that one column can be interpreted via many channels, allowing a deeper understanding, albeit conceptually, of the data.

4.4.2.1 Pit Trading Sound

Just as in *Vapourwaves* I used the pit trading sound file[68] as source material and fed it through a VST whose parameters were controlled by the data. The goal was to let the spoken words be recognizable for some moments while letting the sound be transformed during the course of the piece. To manipulate the sound, I used the *R3son8* preset from Native Instruments' *Absynth* VST. Out of the possible 21 parameters I chose 7 parameters, to be controlled by the data. While there were more parameters available, through experimentation and observation I settled for the following 7:

1. Harmony Resonator A
2. Harmony Resonator B
3. Release
4. Resoverb Feedback
5. Liveness
6. Tone Sharpness
7. Volume

Using more or different parameters transformed the sound too much and resulted in sounds that were not suitable for use the composition. The mapping ranges go from 0,5 until 0,95. A value higher than 0,95 yielded unwanted sonic artefacts while a value lower than 0,5 did not transform the sound sufficiently to my liking. As the volume was too low, I

[68] I had to resort using the same sound file for the simple reason that it was the only sound that was clean enough to be used. Other sounds (or the sound from vídeos) was too distorted to be qualitatively useful.

set the volume range from 0,8 to 0,95. The data was read at a speed of one data point every 2000 milliseconds, which resulted in a composition of approximately 8 minutes.

4.4.2.2 Individual Events

It was clear from the start that sonifying each data point individually would not yield an interesting sonic artefact, certainly not if all notes were placed equidistant. Furthermore, the constant stream of note events would take away attention to the other sounds. Instead of using all data points, I searched the adjusted closing price column for peaks, troughs and recoveries (the point at which previous losses or gains are 'neutralized)[69]. This resulted in three series, with events taking place at different distances from each other. These irregular distances blocked creating rhythmic patterns, something I wanted to avoid. However, the irregularity and relative sparseness would allow the listener to detect these sounds while in a regular rhythm, the individual notes would disappear in an endless stream.

Each of the three series is played by a different instrument in Native Instruments' *Absynth* and *Reaktor Spacedrone.*

I wanted the sounds to be equal sounding, keeping in mind with their signalling function: the events show important changes in the stock's movement. As the sounds were to my liking, I only mapped the data to MIDI parameters (pitch, velocity and duration), each one played by a different instrument.

The peaks were played using the *Falling Waves* instrument. By using this sound, a peak is sonified through a series of downward sounds. While it would at first seem illogical to use downward sounds, it does make sense: at the moment we hear the peak, the price is already going down. Pitch was set in the upper-midlevel, as it was the most 'neutral': the extremer ranges would make the sound too prominent. The original sound has a long release. Hence, to prevent that the notes would sound for a too long period, I mapped the

[69] I used Microsoft's Excel for this filtering, as it was the most adequate for this operation. Of course I could have designed a filtering mechanism inside DataScapR. However, I believe that adding such features would unnecessarily bloat the toolbox hence I did not include it. Zawinski's law of software envelopment conveys best the unnecessary bloating of software: "Every program attempts to expand until it can read mail. Those programs which cannot so expand are replaced by ones which can" (Zawinsky, n.d.).

data to very short durations. Velocity was set at relatively high values to ensure that the sound would come through in the ensemble.

Table 6 – *Mirage:* parameters output range for the peaks in the stock price

Parameter	Output range
Pitch	70-84
Duration	56-84
Velocity	84-112

The troughs were played through the *Absynth's inharmonic droplets* instrument. The sound of this instrument was especially apt for mapping the troughs as the sound goes up. Hence, the upward movement of the stock price is reflected in the sound. This is the same logic as used in the sonification of the peaks. The pitch is kept in the middle register as the extreme values resulted in uninteresting sounds. Although the duration values may seem short, the long release times make them longer than expected.

Table 7 – *Mirage:* parameters output range for the troughs in the stock price

Parameter	Output range
Pitch	28-56
Duration	28-56
Velocity	56-84

4.4.2.3 Embelishing Voice

In keeping the trinity as set out in *Vapourwaves*, I included a third voice to complement the composition. The *birds* preset in *Reaktor*'s *Spacedrone* ensemble was used and data were linearly mapped to resonance, density and decay using the *zmap* method and ranges between 0,5 and 0,95. The mapping ensured that the sound would be birdlike in lower prices while being more resonant and thus transformed a lot when the prices were higher. One can see here the symbolism that high stock prices delude us.

4.4.3 Evaluation

Mirage can be seen as a logical follow-up to the other works: earlier structural elements were re-used and integrated in the piece. These elements include the history aspect: a sound continues to exist after the initial triggering, hence living forward in the future. The long releases make sounds blend in together, symbolizing the continuing impact of the stock's price. Secondly, sounds are born as a result of processes: the resonators give birth to secondary artefacts. For example, one can hear the resonating sound as somebody bowing a suspended cymbal although that event is not created directly. We can see the idea of secondary sounds in *Vapourwaves* as well where the compound sine waves created secondary frequencies. Using only certain specific data points (peak, through and recovery points) to trigger individual events and dividing those in three distinct voices created different rhythms and the melodic contours were more sharper as adjacent notes were virtually absent as a result of the data filtering. Where all data points were used in one voice, they were not triggering individual events but set processes in motion. Thirdly, trinity is present in *Mirage*: the modulation of the human sounds, the procedural bird-like sounds and the three individual-event-voices. I consider the three individual-event-voices as one because of their similarity in structure. Just as in *Vapourwaves*, the trinity turns the piece more balanced and interesting.

Of course, as said earlier, everything can be mapped to something. However we cannot try out every possible mapping. I experimented with various mappings and from those experiments I chose to use the ones that I deemed the best in the context of this composition. In no way I contend that the chosen mappings are the best or correct ones: I merely contend that the chosen mappings are the best ones within the limits of the tried-out mappings.

As stated previously: in sonification we track a portion of a reality: the dataset is just a snapshot of the total (hi)story of Citigroup. As such, we are interpreting reality and reconstructing it through a narrative system.

All elements considered, I believe that *Mirage* lives up to its promise: reveal the stock market (and more specifically *Citigroup*) to the listener and offer him a symbolic connection to the phenomenon. Nevertheless, in no way *Mirage* is intended as an analysis of the price evolution, it should be seen as an artwork, speaking for itself.

4.5 General Evaluation

The four works presented above show the potential of *DataScapR* and the potential of using stock market data in general in an artistic context. As stated earlier in the dissertation, these works are not meant to improve trading performance, but to make the general public aware of the phenomenon of the stock market in a symbolic way. Although we do not understand fully the price evolution (that is, our interpretation will not allow us to grasp every small change in the data values), we can get a general image and thus an aesthetic experience. This experience can then be seen as a kind of revelation: something not scientific in nature, we experience the markets in a different way. In art, there is no scale to which we can measure the aesthetic experience. As each person interprets the work differently, he will get a different aesthetic experience.

Although not intended from the beginning, we can see similarities in the works. All works somehow let history flow over into future, either by letting older events co-exist with the newer ones like the frequencies in *4D Brokers* and *Vapourwaves*, or by long release times like in *For a Fistful of Data* and *Mirage*. All works use time as an aspect of the revelation: we need to listen to the work for an extended period to grasp its evolution.

After having presented these works, we can now evaluate the requirements set out to create *DataScapR* again and see how they were fulfilled in a practical sense. We set out that the software had to be freely available, flexible, not too broad and using only one type of data. Of course *DataScapR* is freely available (via the research blog) and is very flexible. Both *4D brokers* and *Vapourwaves* used portions of the default toolbox and had other parts seamlessly attached to it. *DataScapR* is focused on mapping stock market data coming from Yahoo Finance. No other datasets are used to avoid compatibility problems. The toolbox is style-independent: the unity in the case studies is a result of the process of writing, not a primary intentional goal.

Finally, the toolbox is easy-to-use but does not hide the mapping process: at each moment in the chain, one can retrace his steps and continue experimenting with different values. In the end we can thus say that *DataScapR* has fulfilled its promises and it is a solid toolbox suitable to be used by the general public.

LOOKING BACK, LOOKING FORWARD

Reaching the end of this dissertation, we can look back at what was accomplished with this project. In financial terms: what has been the return on investment? We have set up a theoretical framework for sonification art, contextualized this framework with practical examples, created a toolbox for stock market sonification and used the toolbox in a practical way. This dissertation is just one of the possible viewpoints amongst many others on this topic. There are many other opinions and philosophical constructs that one can use to travel through the concepts discussed here.

The theoretical framework offers the reader a pathway on how to consider sonification art within the bigger context of composition and art in general. The overview of existing art projects allowed the reader to get acquainted with current practices and how these works used sonificaton as a constituent of their being. This can help him to develop his own artworks, learning from the experiences here. The *DataScapR* toolbox offers unprecedented possibilities for an easy-to-use sonification of the stock markets while retaining the openness necessary to extend the software for personal use. Finally, the case studies showed how *DataScapR* can be used in a practical way and how the theoretical concepts can be applied to the works.

Although this project ends formally here, it will continue to be developed. In the first place *DataScapR* will be extended (see section 3.5) and I will continue to gather sonification art on the research blog. Furthermore, although *DataScapR* was meant as an artistic project, it would be interesting to develop it to use in the real markets. I thus hope to be able to turn *DataScapR* into a truly interdisciplinary project, joining artists, programmers and economists. Not only will this potentially enhance trading performance, it will bring, though the artistic component, the general public more in contact with the stock market.

The voyage has been interesting so far; we are merely making an intermediate stop, and we will continue our discovery in the endless sea of data.

BIBLIOGRAPHY

Abelson, B., Bialer, J., DeWilde, B., Keller, M., Levine, T., & Podku, C. (2013). *FMS Symphony*. (CSV Soundsystem)*Bicoastal Datafest* (Vol. 6). Columbia, USA.

About historical prices. (n.d.). About historical prices. *Help.Yahoo.com/Kb/Finance/Historical-Prices-Sln2311.Html*. Retrieved January 15, 2015, from https://help.yahoo.com/kb/finance/historical-prices-sln2311.html

Agrawal, A., & Tandon, K. (1994). Anomalies or illusions? Evidence from stock markets in eighteen countries. *Journal of International Money and Finance*, *13*(1), 83–106.

Aiken, C., Peng, Z., Simpson, D., Michael, A., Kilb, D., Enescu, B., & Shelly, D. (2012). Shaking up Earth Science: Visual and Auditory Representations of Earthquake Interactions

(pp. 250–251). Presented at the the International Conference on Auditory Display, Atlanta, Georgia, USA.

Alexander, R. (2009). *Solar Wind Sonification - Scientifically Accurate Music!* (R. Alexander)*Youtube* (Vol. 8). Michigan, USA.

Alexander, R. (2012, January 22). Question about *Solar Wind Sonification*. Michigan, USA.

Bachmann, C. W. (1973). *The programmer as navigator*. CACM. doi:10.1145/1283920.1283928

Barrass, S., & Vickers, P. (2011). Chapter 7 Sonification Design and Aesthetics

. In T. Hermann, A. Hunt, & J. G. Neuhoff, *The Sonification Handbook*

(pp. 145–171). Berlin, Germany: Logos Verlag.

Barthes, R. (1967). THE DEATH OF THE AUTHOR

. (B. O'Doherty)*Aspen Magazine*, (5-6), 1–6. Retrieved from http://www.ubu.com/aspen/aspen5and6/threeEssays.html#barthes

Barthes, R., & Duisit, L. (1975). An Introduction to the Structural Analysis of Narrative. *New Literary History*, *6*(2), 237–272.

Baudu, O. (2013, November 12). WikikIRC... Orléans, France.

Baudu, O., Templier, A., & Blocquaux, S. (2012). wikikirc ou la sonification de Wikipedia. *Labomedia.org*. Orléans, France. Retrieved 2012, from http://labomedia.org/oeuvres-interactives/wikikirc-ou-la-sonification-de-wikipedia/

Berg, P. (1979). PILE: A Language for Sound Synthesis. *Computer Music Journal*, *3*(1), 30. doi:10.2307/3679754

Bioacústiques, L. D. (n.d.). LIDO: Listening to the Deep Ocean Environment. *Listentothedeep.com*. Retrieved November 20, 2010, from http://listentothedeep.com/

Bircsák, E., Braspenning, J., Bujdosó, A., van de Craats, R., Csík-Kovács, Z., Diaz, V., et al. (2012). *Beyond data*. (A. Plohman & M. Sipos). Eindhoven, Netherlands: Lecturis.

Bogost, I. (2015, January 15). **The Cathedral of Computation**. *Theatlantic.com*. Retrieved February 8, 2015, from http://www.theatlantic.com/technology/archive/2015/01/the-cathedral-of-computation/384300/

Boulez, P. (1986a). **Le système et l'idée**

. *InHarmoniques*, (1), 316–390.

Boulez, P. (1986b). Technology and the Composer. In *The Language of Electroacoustic*

Music (pp. 5–14). Routledge.
Bourdieu, P. (1993). *The Field of Cultural Production*. Columbia University Press.
Bowker, G. C. (2008). *Memory Practices in the Sciences*. MIT Press (MA).
Brun, H. (1969). Infraudibles. In H. Von Foerster & J. W. Beauchamp, *Music by Computers* (pp. 117–121).
Brün, H. (1970). From musical ideas to computers and back. *The Computer and Music*.
Burt, Waren. (n.d.). Notes on Plano. *Algoart.com/Help/Artwonk4/ArtWonk/*. Retrieved November 13, 2013, from http://www.algoart.com/examples/PlanoNotes.htm
Burt, Warren. (2012, February 2). Re_ Form Submission - Contact Me. Melbourne.
Butler, P. (2010, October 13). **Visualizing Friendships**
. *Facebook.com*. Retrieved February 23, 2015, from https://www.facebook.com/notes/facebook-engineering/visualizing-friendships/469716398919
Cage, J. (1987). *Organ2/ASLSP*. *Aslsp.org*. Halberstadt, Germany. Retrieved January 10, 2015, from http://www.aslsp.org/de/home.html
Ciardi, F. C. (2004). sMAX A MULTIMODAL TOOLKIT FOR STOCK MARKET DATA SONIFICATION
. In S. Barrass & P. Vickers, (pp. 1–3). Presented at the Proceedings of the 2004 International Conference on Auditory Display, Sydney, Australia.
Ciardi, F. C. (2015, January 23). Question on sMax- stock market music. Rome, Italy.
csvQuotesDownload. (n.d.). csvQuotesDownload. *Code.Google.com*. Retrieved February 10, 2015, from https://code.google.com/p/yahoo-finance-managed/wiki/csvQuotesDownload
ctbto.org/specials/1945-1998-by-isao-hashimoto/. (n.d.). *ctbto.org/specials/1945-1998-by-isao-hashimoto/*. CTBTO.
Culkin, J. M. (1967). *A schoolman's guide to Marshall McLuhan*. Saturday Review.
Cycling '74 Max 7. (n.d.). Cycling '74 Max 7.
Daniel, S. (2007). ***The Database: An Aesthetics of Dignity***
. In V. Vesna, *Database aesthetics* (Vol. 20, pp. 142–182). Minneapolis: U of Minnesota Press.
Danto, A. C. (1998). *Beyond the Brillo Box: The Visual Arts in Post-Historical Perspective* (3rd ed.). University of California Press.
de Campo, A. (2008). Toward a data sonification design space map. In G. P. Scavone, (pp. 343–347). Presented at the Proceedings of the 13th International Conference on Auditory Display, Montréal.
Devisch, O. (2008). Should Planners Start Playing Computer Games? Arguments from SimCity and Second Life. *Planning Theory & Practice*, *9*(2), 209–226. doi:10.1080/14649350802042231
Di Scipio, A. (1995). Centrality of techné for an aesthetic approach on electroacoustic music. *Journal of New Music Research*, *24*(4), 369–383. doi:10.1080/09298219508570691#
Dodge, C. (1970). *Earth's Magnetic Field*. (Columbia University Computer Center)*record*. New York, USA.
Dunn, J. (n.d.). **ArtWonk & MusicWonk Manual**. *Algoart.com/Help/Artwonk4/ArtWonk/*. Retrieved January 15, 2015, from http://algoart.com/help/artwonk4/ArtWonk/
Dunn, J., & Clark, M. A. (1999). Life Music: The Sonification of Proteins. *Leonardo*, *32*(1), 25–32. doi:10.1162/002409499552966

Eaglestone, B., & Clowes, N. F. M. (2001). Do Composition Systems Support Creativity? - An evaluation (pp. 22–25). Presented at the 2001 International Computer Music Conference, Havana, Cuba: Ann Arbor, MI: Scholarly Publishing Office, University of Michigan Library.
Ebcioglu, K. (1990). An expert system for harmonizing chorales in the style of JS Bach. *The Journal of Logic Programming*, *8*(1), 145–185.
Edlund, B. (2012). *Stanley Cup Summed Up*. *edlundart.com*.
Edlund, B. (2014, August 25). Re_ question about Stanley cup sonification. New York, USA.
European Science Foundation. (2000, February 14). The GRIP Ice Coring Effort. *Ncdc.Noaa.Gov*. Retrieved December 1, 2014, from https://www.ncdc.noaa.gov/paleo/icecore/greenland/summit/document/gripinfo.htm
Fallows, D. (1987). *Dufay*. Vintage.
Flowers, J. H. (2005). Thirteen Years of Reflection on Auditory Graphing: Promises, Pitfalls, and Potential New Directions (pp. 406–409). Presented at the Eleventh Meeting of the International Conference on Auditory Display, Limerick, Ireland.
Freeman, J. (2004). N.A.G. Presented at the Proceedings of the 12th annual ACM international conference on Multimedia - MULTIMEDIA '04. doi:10.1145/1027527.1027567
Freeman, J. (2013, December 12). N.A.G question. Atlanta, USA.
Freire, T. S. (2014). A new reading of Nuper rosarum flores and its controversial numeric relation with Santa Maria del Fiore. In M. Rossi, (pp. 101–106). Presented at the Nexus, Milan.
Fux, J. J. (1965). *The Study of Counterpoint from Johann Joseph Fux's Gradus Ad Parnassum*. (A. Mann & J. Edmunds). New York: W. W. Norton & Company.
Giovannelli, A. (2010). Goodman's Aesthetics. *The Stanford Encyclopedia of Philosophy*. Retrieved from http://plato.stanford.edu/archives/sum2010/entries/goodman-aesthetics/
Goetzmann, W. N. (2001). Patterns in Three Centuries of Stock Market Prices. *The Journal of Business*, *66*(2), 249. doi:10.1086/296603
Gomes, J. A., de Pinho, N. P., Lopes, F., Costa, G., Dias, R., & Barbosa, Á. (2014). Composing with Soundscapes: an Approach Based on Raw Data Reinterpretation, 1–14.
Goodman, N. (1999). Quand y a-t-il art? In M.-D. Popelard, *L"oeuvre d"art* (pp. 153–157). Flammarion.
Guljajeva, V., & Sola, M. C. (2011). The Rhythm of City. Geo-located Social Data as an Artistic Medium (pp. 1–5). Presented at the International Symposium for the Electronic Arts, Istanbul.
Hackworth, N. (2014, June 29). John Maeda: painting by pixel. *Dazeddigital.comartsandculturearticlejohn-Maeda-Painting-by-Pixel*. Retrieved November 6, 2014, from http://www.dazeddigital.com/artsandculture/article/20241/1/john-maeda-painting-by-pixel
Hamman, M. (2002). From technical to technological: The imperative of technology in experimental music composition. *Perspectives of New Music*, *40*(1), 92–120.
Hamman, M. (2004). The Technical as Aesthetic: Technology, Art-making, Interpretation. *Music, Arts and Technologies: Toward a Critical Approach*, 375.
Harman, S. (2011, February 6). Twitter Powered Synthesis. Retrieved October 2, 2013,

from http://samharman.com/wp-content/uploads/2011/03/Twitter-Powered-Synthesis.pdf
Hazard, C., Kimport, C., & Johnson, D. (1999, October 1). Fractal Music. *Tursiops.Cc*. Retrieved October 1, 2013, from http://www.tursiops.cc/fm/
Helmuth, M., & Davis, T. (2004). Rock Music: Granular and Stochastic Synthesis based on the Matanuska Glacier (pp. 619–622). Presented at the International Computer Music Conference 2004, Miami.
Hermann, T., & Hunt, A. (2004). The importance of interaction in sonification, 1–8. Retrieved from http://hdl.handle.net/1853/50923
Hiller, L., & Isaacson, L. (1957). Illiac Suite for String Quartet. New York: Theodore Presser Co.
Hoffmann, P. (2009, April 29). *Music Out of Nothing?* Berlin, Germany.
House, B. (2012). *Quotidian Record*. (Eyebeam Art Technology Center)*brianhouse.net*.
House, B. (2013a, May 25). Brian House | You'll Just Have to Take My Word for It. *Brianhouse.Net*. Retrieved November 5, 2013, from http://brianhouse.net/works/youll_just_have_to_take_my_word_for_it/
House, B. (2013b, November 10). Questions about quotidian record. New York, USA.
http://illegalart.net/mp3s/02.04.html. (n.d.). *http://illegalart.net/mp3s/02.04.html*.
Hume, D. (1965). *Of the standard of taste, and other essays*. MacMillan Publishing Company.
Investopedia. (2005, August 22). What is Fibonacci retracement, and where do the ratios that are used come from? *Investopedia.com*. Edmonton, Canada. Retrieved February 28, 2015, from http://www.investopedia.com/ask/answers/05/fibonacciretracement.asp
Jacob, B. L. (1995). Composing With Genetic Algorithms. Presented at the International Computer Music Conference, Banff, Alberta: unknown.
john maeda: painting by pixel. (2014). john maeda: painting by pixel, 1–6.
Jones, D., & Bulley, J. (2011). *Maelstrom*. *jjbdjj.com*.
Jones, D., Gregson, P., Britten Sinfonia. (2013). *The Listening Machine*. *The Space*. London, UK.
KDNuggets. (1997, February 1). kdnuggets.com. *Kdnuggets*. Retrieved January 10, 2015, from http://www.kdnuggets.com
Keane, D. (1986). At the treshold of an aesthetic. In S. Emmerson, *The Language of Electroacoustic Music*.
Kirilenko, A. A., Kyle, A. S., Samadi, M., & Tuzun, T. (2011). The Flash Crash: The Impact of High Frequency Trading on an Electronic Market. *SSRN Electronic Journal*. doi:10.2139/ssrn.1686004
Klein, N. M. (2007). Waiting for the World to Explode: How Data Converts Into a Novel
. In V. Vesna, *Database aesthetics* (pp. 86–94). U of Minnesota Press.
Koenig, G. M., & Roads, C. (1978). An Interview with Gottfried Michael Koenig. *Computer Music Journal*, *2*(3), 11. doi:10.2307/3679451
Korzybski, A. (1941). ***Science and Sanity: An Introduction to Non-Aristotelian Systems and General Semantics*** (5 ed.). New York, USA: Institute of General Semantics.
LaPorte, S., & Hashemi, M. (2013). *Hatnote Listen to Wikipedia*. *listen.hatnote.com*.
Laumeister, M. (2013). *BitListen.com - Listen To Bitcoin*. *bitlisten.com*.
Levine, T. (2013, September 8). Question about *FMS Symphony*.
LeWitt, S. (1971). **Sentences on Conceptual Art**. *Art Now*, *3*(2).
Lévi-Strauss, C. (1964). *1: Le cru et le cuit*. Paris: Plon.

Man, E. (2003, July 28). *Episode 318: Can't Stop the NAGGING!* Presented at the pressthebutton.com, Cleveland, Ohio, USA.
Manning, P. (2006). The significance of techné in understanding the art and practice of electroacoustic composition. *Organised Sound*, *11*(01), 81–11. doi:10.1017/S1355771806000112
Manovich, L. (2002). The anti-sublime ideal in data art. *Manovich Net*. Retrieved February 14, 2015, from http://meetopia.net/virus/pdf-ps_db/LManovich_data_art.pdf
Manovich, L. (2014). **The Anti-Sublime Ideal in Data Art**
. *Animation*, *9*(3), 281–298. doi:10.1177/1746847714545938
Maurer, J. A. (1999, March 1). The History of Algorithmic Composition. *Ccrma.Stanford.Edu*. Retrieved October 1, 2013, from https://ccrma.stanford.edu/~blackrse/algorithm.html
Mazzola, G., Park, J., & Thalmann, F. (2011). *Musical Creativity*. Springer.
McCormack, J., & Dorin, A. (2001). Art, Emergence, and the Computational Sublime (pp. 67–81). Presented at the Second Iteration A Conference on Generative Systems in the Electronic Arts, Melbourne, Australia: unknown.
McKay, C. (2002). *Final Project Report SpeciesChecker 1.0*. Montréal.
McKinney, C., & Renaud, A. (2011). Leech: BitTorrent and Music Piracy Sonification. Presented at the Sound and Music Computing, Barcelona: Bournemouth University, Fern Barrow, Poole, Dorset, BH12 5BB, UK.
McLuhan, M., Fiore, Q., & Agel, J. (1968). ***The Medium is the Massage***
. (J. Simon)*ubu.com*.
Merriam Webster. (2014, September 15). Data. Springfield, USA. Retrieved September 15, 2014, from http://www.merriam-webster.com/dictionary/data
Montague, E. (1995). The Limits of Logic: Structure and Aesthetics in Xenakis's Herma. *Ex Tempore*, *7*(2).
Morris, R. (1970). Some Notes on the Phenomenology of Making: The search for the motivated. *Artforum*, *9*, 62–66.
Nehls, A. V., & Barri, T. (2012). #tweetscapes – listen to Twitter. *Heavylistening.com*. Berlin, Germany. Retrieved October 1, 2014, from http://heavylistening.com/tweetscapes/
Petersen, M. G., Iversen, O. S., Krogh, P. G., & Ludvigsen, M. (2004). Aesthetic interaction: a pragmatist's aesthetics of interactive systems (pp. 269–276). Presented at the DIS '04: Proceedings of the 5th conference on Designing interactive systems: processes, practices, methods, and techniques, New York, New York, USA: ACM Request Permissions. doi:10.1145/1013115.1013153
Pinto, J. C. (2011). The Status of Interactivity in Computer Art: Formal Aporys. *CITAR Journal*, *3*(1), 10–19.
Pizzi, F. D. (1998). Towards Heterostatic Systems. In X. Hascher & C. Miereanu, (pp. 709–712). Presented at the quatrième Congrès international sur la signification musicale, Paris.
Polansky, L. (2002, April 15). Manifestation and Sonification. *Eamusic.Dartmouth.Edu*. Retrieved September 1, 2013, from http://eamusic.dartmouth.edu/~larry/sonification.html
Quinn, M., & Meeker, L. D. (2001). Research set to music: The climate symphony and other sonifications of ice core, radar, DNA, seismic and solar wind data (pp. 56–61). Presented at the *Proceedings of the 2001 International Conference on Auditory*

Display, Espoo.
Reeves, G. (2014a). *AERO* (1st ed.). Los Angeles, USA: Bandcamp.
Reeves, G. (2014b, May 13). Questions on *AERO*. Los Angeles, USA.
Roads, C. (1996). *The Computer Music Tutorial*. MIT Press.
Roden, S., & Polsenberg, A. (2004). *Ear(th)*. Retrieved from http://ddata.over-blog.com/xxxyyy/1/96/04/42/Textes-1/steve-roden-earth-installation-at-art-center.pdf
Rosenberg, D. (2013). Data before the Fact. In L. Gitelman, *Raw Data is an Oxymoron* (p. 182). MIT Press.
Saladin, M. (2007). *Stock Exchange Piece*
. (Mattin)*mattin.org* (32nd ed., Vol. 50). w.m.o/r.
Sauter, J., & Lusebrink, D. (1995). *Invisible Shape of Things Past*. Berlin, Germany: ART+COM Studios.
Sauter, J., & Lusebrink, D. (2006). The Invisible Shape of Things Past | ART+COM Studios. *Artcom.De*. Berlin, Germany. Retrieved January 12, 2015, from https://artcom.de/en/?post_type=project&p=970
Scaletti, C. (1994). Sound synthesis algorithms for auditory data representation. In G. Kramer, (Vol. XVIII, pp. 223–252). Presented at the Santa Fe Institute, Studies in the Sciences of Complexity Proceedings, Reading, USA.
Schaeffer, P. (1970). Music and computers. In W. Skyvington, (pp. 57–92). Presented at the Stockholm Meeting Music and Technology, Stockholm.
Shields, A. (1997). ***The Columbia-Princeton Electronic Music Center***
(pp. 1–20). New York, USA: New World Records.
Sodell, J., & Soddell, F. (2005, December 28). microbes, L-systems and music. *Cajid.com*. Retrieved October 1, 2013, from http://cajid.com/jacques/lsys/index.htm
Somerecords. (2007, March 17). Charles Dodge - Earth's Magnetic Field [1970]. *Somerecords.Wordpress.com*. Retrieved October 12, 2014, from https://somerecords.wordpress.com/2007/03/17/charles-dodge-earths-magnetic-field-1970/
SpaceWeatherLive. (n.d.). The Kp-index. *Spaceweatherlive.com*. Retrieved February 13, 2015, from http://www.spaceweatherlive.com/en/help/the-kp-index
Steinberg Media Technologies GmbH. (2003). What is a VST Plug-In?
. *Gersic.com*. Retrieved October 1, 2014, from http://www.gersic.com/vstsdk/
Stenger, S. (2006, March 8). SOUNDTRACK FOR AN EXHIBITION. *Mac-Lyon.com*. Lyon, France. Retrieved January 10, 2015, from http://www.mac-lyon.com/mac/sections/fr/expositions/2006/soundtrack_for_an_ex/
Sternberg, R. J., & Lubart, T. (2010). The concept of creativity: prospects and paradigms. In R. J. Sternberg & J. C. Kaufman, *The Cambridge Handbook of Creativity* (pp. 3–15). Cambridge, UK: Cambridge University Press.
Stoll, C. (1995). *Silicon snake oil: Second thoughts on the information highway: Anchor*. New York.
Straebel, V. (2010). The Sonification Metaphor in Instrumental Music and Sonification's Romantic Implications
(pp. 287–294). Presented at the Proceedings of the 10th International Conference on Auditory Display, Washington.
Supper, M. (2001). A Few Remarks on Algorithmic Composition, *25*(1), 48–53. doi:10.1162/014892601300126106
Tarkovsky, A. (1984). Andrei Tarkovsky: A Poet in the Cinema. (D. Baglivo)*Vimeo.com*.

Italy.
Tedesco, M., Ham, E., Perl, J., & Saltz, I. (2012, September 8). POLAR SEEDS - Home. *Polarseeds.org*. New York. Retrieved September 9, 2013, from http://www.polarseeds.org/
Tomayko, A. (2012). Maestro Frankenstein. *Arvidtp.Net/Sw/Maestro_Frankenstein.Php*. Retrieved February 12, 2015, from http://arvidtp.net/sw/maestro_frankenstein.php
Tomayko, A. (2013, June 4). Geophonics. *Arvidtp.Netgeophonics*. Boston, USA. Retrieved October 1, 2014, from http://arvidtp.net/geophonics/
Tomayko, A. (2015, January 9). Questions about *Maestro Frankenstein*. Providence, USA.
Turner, C. (1991). Proportion and Form in the Continental Isorhythmic Motet c. 1385-1450. *Music Analysis*, *10*(1/2), 89. doi:10.2307/854000
United States Congress
. Gramm-Leach-Bliley Act
, U.S. Government Printing Office 1–154 (1999). 106.
Vesna, V. (1999). **Database Aesthetics: Of Containers, Chronofiles, Time Capsules, Xanadu, Alexandria and the World Brain.** *Database Aesthetics: Issues of Organization and Category in Online Art*.
Vriend, J. (1981). *Nomos alpha* for violoncello solo (Xenakis 1966) analysis and comments. *Interface*, *10*(1), 15–82. doi:10.1080/09298218108570328
Wagner, R. (2011, February 26). High Frequency Trading - Financial Ethics - Seven Pillars Institute. (K. Tan Bhala)*Sevenpillarsinstitute.org*. Lawrence, KS. Retrieved January 4, 2015, from http://sevenpillarsinstitute.org/case-studies/high-frequency-trading
Walker, B. N., & Nees, M. A. (2011). Chapter 2 Theory of Sonification. In T. Hermann, A. Hunt, & J. G. Neuhoff, *The Sonification Handbook*
(pp. 9–39). Berlin, Germany: Logos Verlag.
Ward, C. (2014, January 29). Vaporwave: Soundtrack to Austerity. *Stylus.com*. London, UK. Retrieved November 1, 2014, from http://www.stylus.com/hzwtls
Warren, C. W. (1973). BRUNELLESCHI"S DOME AND DUFAY"S MOTET. *The Musical Quarterly*, *LIX*(1), 92–105. doi:10.1093/mq/lix.1.92
Weiser, S., & Stone, O. (1987). *Wall Street*. (O. Stone) (Vol. 126). 20th Century Fox.
Williams, A. (2011). Ageing of the new: the museum of musical modernism. In N. Cook & A. Pople, *The Cambridge History of Twentieth-Century Music* (pp. 506–538). Cambridge: Cambridge University Press. doi:10.1017/CHOL9780521662567.021
Worrall, D. (2009a). *Sonification and Information*. Sonification and Information: Concepts, Instruments and Techniques, Canberra, Australia.
Worrall, D. (2009b). *Sonification and Information: Concepts, instruments and techniques*. (M. Whitelaw & R. Dean). University of Canberra, Canberra.
www.dataviz.org. (n.d.). www.dataviz.org. *Dataviz.org*. Retrieved December 26, 2014, from http://www.dataviz.org
www.r-project.org. (n.d.). www.r-project.org. *R-Project.org*. Retrieved December 26, 2014, from http://www.r-project.org
Xenakis, I. (1976a). *Musique, architecture* (2nd ed.). Doornik, Belgium: Casterman.
Xenakis, I. (2001). *Formalized Music: Thought and Mathematics in Composition* (2nd ed.). New York, USA: Pendragon Press.
Xenakis, I. (1976b, May 18). *Arts/sciences, Alloys*. (S. Kanach). Pendragon Press, New York, USA.
Yahoo Finance. (2014). *IBM Interactive Stock Chart* (pp. 1–1). Yahoo Finance. Retrieved

from
http://finance.yahoo.com/echarts?s=IBM+Interactive#%7B%22range%22%3A%7B%22start%22%3A%221995-01-27T12%3A00%3A00.000Z%22%2C%22end%22%3A%222014-10-01T11%3A00%3A00.000Z%22%7D%2C%22lineType%22%3A%22combo%22%2C%22scale%22%3A%22linear%22%7D

Yahoo Finance. (2015a). *AMZN Interactive Stock Chart*. Yahoo Finance. Retrieved from https://finance.yahoo.com/echarts?s=AMZN+Interactive#%7B%22range%22%3A%22max%22%2C%22lineType%22%3A%22candlestick%22%2C%22indicators%22%3A%7B%22ema%22%3A%5B%7B%22id%22%3A%22ema50%22%2C%22name%22%3A%22ema%22%2C%22params%22%3A%5B50%5D%2C%22lineType%22%3A%22line%22%2C%22color%22%3A%22%234a86e8%22%2C%22weight%22%3A1%7D%5D%7D%2C%22scale%22%3A%22linear%22%7D

Yahoo Finance. (2015b). *C Interactive Stock chart*. Yahoo Finance. Retrieved from http://finance.yahoo.com/echarts?s=C+Interactive#%7B%22range%22%3A%7B%22start%22%3A%221994-07-01T11%3A00%3A00.000Z%22%2C%22end%22%3A%222014-07-01T11%3A00%3A00.000Z%22%7D%2C%22lineType%22%3A%22combo%22%2C%22scale%22%3A%22linear%22%7D

Yeo, W. S., Berger, J., & Lee, Z. (2004). SonART: A framework for data sonification, visualization and networked multimedia applications. Presented at the International Computer Music Conference, Ann Arbor, MI: Scholarly Publishing Office, University of Michigan Library.

Zawinsky, J. (n.d.). Zawinski's Law of Software Law of Software Envelopment. *Catb.org*. Retrieved February 23, 2015, from http://www.catb.org/~esr/jargon/html/Z/Zawinskis-Law.html

APPENDICES

1. DVD contents

The DVD that accompanies this dissertation contains all relevant materials referred to in the text. The contents are as follows:

1. Digital version of the dissertation in pdf and doc format
2. *DataScapR* toolbox folder: this folder contains all files used in the toolbox and is the folder that is made available to the general public as a download
3. *4D Brokers* video (this video is the complete documentation video about the workshop)
4. *Vapourwaves* recording and video: a recording made on 27 January 2015 showing the sonic output of the installation as well as a short video showing the space where the installation was placed.
5. *For a Fistful of Data*: *DataScapR* output file and score for the recorder piece in pdf
6. *Mirage*: fixed-media piece in WAV format
7. The csv files used for *For a Fistful of Data, 4D Brokers, and Mirage*
8. Parts of this dissertation have been published in altered forms earlier. The contents of these papers and posters are used in this dissertation. The papers and posters are:
    a. *Sonification Art*: short paper, presented at *The Global Composition* soundscape conference in Dieburg, Germany in July 2012
    b. *Sonification Art*: book chapter for inclusion in *Playful subversions of Technoculture: new directions in creative, interactive and entertainment technologies Singapore*, Springer (in press)
    c. *DataScapR: A Toolbox for Stock Market Sonification* poster presented at DMRN+9 in at Queen Mary University London on 16 December 2014

2. Scores

1. unformatted score *bach's* xml output imported in Finale in pdf format
2. final score in pdf format

Score

For a Fistful of Data/ XML Output

Samuel Van Ransbeeck


42
48
54
60
66
71
76
80


82
6
6
6
84
6
6
85
6
6
86
6
6
87
6
6
88
6

Score

For a Fistful of Data

Samuel Van Ransbeeck


p
mp
p
pp
p
♩ = 180


staçatto (until ord)
ord