Processor Art

Currents in the Process Oriented Works of Generative and Software Art

Thor Magnusson

August, 2002.

Coordinator: Mette Sandbye

Contents

Abstract – [English]

This thesis is a research into the sub-genre of computer art concerning works that necessarily must run on the computer in order to exist. To limit the research – and define it from the uncountable and varied art practises which utilise the computer today – the concept of *processor art* is created. Processor art is a generic term that includes software art, generative art, interactive installations and composed hyperinstruments, but in this thesis the emphasis will be laid on works which can be distributed on CD-Roms or through the Internet, thus less concerned with installations and hyperinstruments. The thesis traces currents in the art history of the 20th century, which are important as conceptual predecessors for processor art and serve as ideological foundations for many of the works analysed. In an important chapter on the philosophy of technology, the new digital technology is compared to the modernist technology and the consequences which new technology has for our culture, arts and the human subject are drawn up. This groundwork is necessary to be able to discuss processor art, its aesthetics and ideologies, and analyse different works of art under the categories of software and generative art. The contemporary state of computer technology and its use in art is discussed – showing its qualities and limits – and how it might evolve as a better tool for human expression. By the use of the technology we are able to express and extrapolate the future of our society, our dreams and nightmares about it, and create representations of the utopian or dystopian worlds of the future that we might find us in.

Abstrakt – [danish]

Dette speciale er en undersøgelse af den undergenre af computerkunsten, hvor værkernes eksistens afhænger af computerens processor. For at begrænse undersøgelsen – og definere den ud fra de mange forskellige kunst praksisser som i dag anvender computeren – har jeg skabt begrebet *processorkunst* . Processorkunst er et generisk begreb, som inkluderer 'software kunst', 'generativ kunst', interaktive installationer og komponerede hyperinstrumenter, men i dette speciale er hovedvægten lagt på værker, der kan distribueres på CD-Rom eller via Internettet, og jeg vil i mindre grad komme ind på installationer og hyperinstrumenter. Specialet vil gennemgå begivenheder i det 20nde årshundredes kunsthistorie, som er vigtige forgængere til mange af de værker, der bliver analyseret. I et vigtigt kapitel om teknologifilosofi, bliver den digitale teknologi sammenlignet med den modernistiske teknologi og jeg gennemgår de konsekvenser, som den nye teknologi har for vores kultur, kunst og det menneskelige subjekt. Dette fundament er nødvendigt for at kunne diskutere processor kunst, dens æstetik og ideologi, og for at kunne analysere forskellige kunstværker under kategoriene software kunst og generativ kunst. Computerteknologiens nuværende status og brug i kunstfeltet diskuteres – hvor dens kvaliteter og begrænsninger bliver fremdraget – samt hvordan muligheden for, at den kan udvikle sig som et bedre værktøj for menneskelige udtryk. Gennem brugen af teknologi kan vi udtrykke og fremsige vores fremtid, vores drømme og mareridt om den, og skabe repræsentationer af de utopiske eller distopiske verdener, som vi muligvis kan finde os i.

1. Introduction: An Exploration of a Field

> A well written computer program is the perfect resting place for its author. Programs originate in the human imagination, and so programmers know their creations extremely well. The first run of a program is therefore a precious time. The programmer's imagination is built into something intangible, but with presence gained from timelessness. Thoughts solidify into code, and become fluid once more in execution.
>
> Alex McLean [1]

The objective of the current thesis is to analyse changes that have occurred in the field of technology and art in the latter half of the 20th century. The analysis is concerned with the relationship of science/technology and art and how these fields have related to each other historically. What I will be concerned with is a subset of computer generated art which I will define with the generic term *processor art*, and as such narrowing down the research to certain directions within the field of computer art: generative art, software art and to some degree computer installations and composed hyperinstruments.[1] *Generative art* is not necessarily computer based, but the computer is an excellent tool for making such works of art. It is basically a term given to any practise where the artist creates a process in the form of procedural rules, which are then set into motion with some degree of autonomy, which results in the outcome of the work. *Software art* is probably as old as the computer itself, and it includes any form of programming activity involved with creating artistic systems that generate works of art and/or are works of art themselves. It focuses on the aesthetic aspect of software creation and is often seen as personal expression of the programmer. These two fields are virtual categories, they intersect and overlap and may not even be meaningful in the future, but at present they may help us to understand a certain fascination programmers and artists have with the dynamic qualities of the computer as a host for their artwork.

When artists started working with the computer in the middle of the 20th century, a cornerstone was set in the building of a bridge between the fields of science and art, but these fields had been separated on a large extent since the advent of the Renaissance. By this bridging between technology and art we are witnessing categorical shifts in the roles of the artist, the designer, the engineer

[1] The reason for coining the term processor art and the limitations made with it will be discussed further below. The research will not concern so much the categories of interactive installations and composed hyperinstruments. These categories and the reasons for excluding them in my analysis of processor art will be explained later.

and the technician. The boundaries are blurring and we notice this tendency in the academies and universities around the world, with the advent of new, integrated academic departments and media labs that seek to provide spaces for experimentation in the fields of new media art, aesthetics and information design. Art can question the potential and implications of the current technology, just like the sciences, but it can explore it from different perspectives, conceptual frameworks and cultural associations which the sciences could never dream of.

One of the central ideas of this thesis is to show how contemporary artists are relating to technology in an essentially different way than the artists of modernism. There are many historical, sociological and technological reasons for this change and I will try to outline at least two strong currents in contemporary culture of technology and arts that have resulted in this paradigm shift. Firstly, the artistic field has changed enormously in the last century. The avant-garde has criticised all institutions and traditions to their very core, and art practises have become deconstructed and indeterminable. The relationship that modernist art had with technology was one of mastery of the human reason: that of using technology as instrument for our human endeavours. The modernist view of the autonomous self was amplified with the Freudian philosophy of the human subject and, technology became to be seen as a tool in the process of rationalisation. We find this idea in the art of the Futurists, Dada, Constructivists, Bauhaus, Surrealists and Pop Art, who all related to this modernist notion of the human subject either positively or negatively, emphasising respectively the rationality of mankind or its irrationality. These art movements all used technology in their working process and/or created works of art that were technological in themselves. However, with poststructuralist and postmodernist views of the human subject and the deconstruction of its rationality, art was not seen any longer to be the placeholder of subjectivity and creativeness. Art became fluid, contextual, experimental and did not concentrate on *explaining* the world anymore (*why*), but rather experimenting with the world and *describing* it (*how*).

The second current is due to the drastic changes that our technology has undergone the last fifty years or so. Technology has become extremely complex, so complex in fact that we are not in control of it anymore. We have difficulties in understanding or keeping an overview of our technology, which at the same time is inserting itself as our second Nature. The only thing we can do is to relate to technology in a critical way and try to influence its evolution in ways that we find meaningful for the future of our species. This is a serious problem in our culture and thus we find artists working in the fields of artificial life and artificial intelligence, telecommunication arts, genetic arts, geological arts and chaos arts to name but a few. For many of these artists, their work is an attempt to understand the implications of our new technology and our relation to its terrifying powers.

This modal change of technology, from the modernist, industrial technology of the machine, which Martin Heidegger analysed so elegantly in the 1950s, to the post-modern information technology of our current time, brilliantly defined by Manuel Castells, can be explained simply by referring to epoch-changing works of art of the time.[2] The relationship of the human subject with the modernist technology is the theme of Charlie Chaplin's movie *Modern Times*, where mankind becomes automatised by technology and slave of the machine. The technology that was made to ease the work of humans has turned them into mechanical automatons, an insignificant part of the big machinery. Another movie, *Metropolis*, by Fritz Lang showed how humans have come to see themselves as masters of nature, but then loose their humanity in the strive for progress and social control. The post-modern view of technology is very different. In William

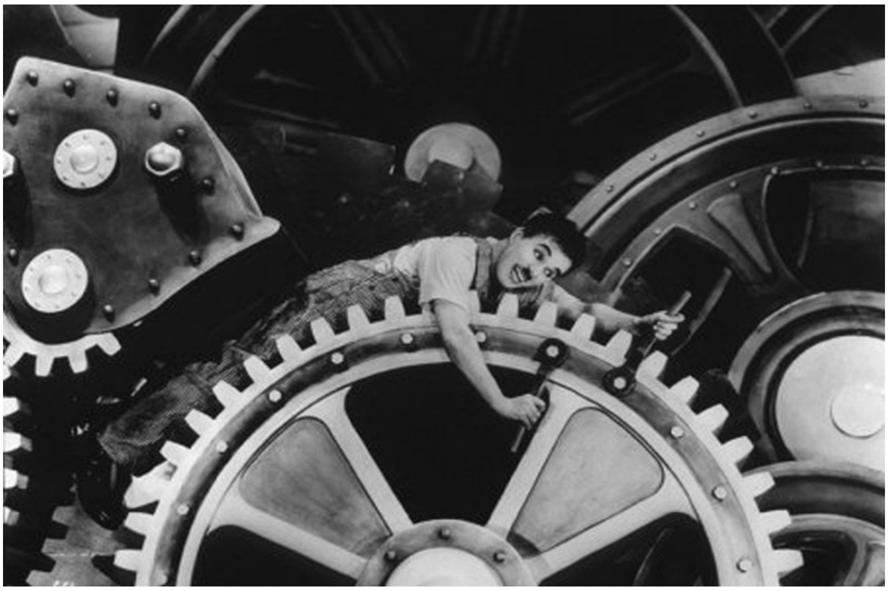


Figure 1. Charlie Chaplin in Modern Times fighting with technology. Many workers and critics found working in factories dehumanising and meaningless.

[2] See Heidegger's essay "The Question Concerning Technology" and Castell's three volume work: *The Information Age: Economy, Society and Culture*.

Gibson's cyberpunk novel *Neuromancer*, the protagonists "jack" into a complex system of computer networks (called "cyberspace," – "a consensual hallucination"), which include artificial intelligence beings, virtual spaces and endless information. Reality has become virtual and the cyberspace is "realer" than reality. Here the human subject is not seen anymore as a master of technology or Nature, as a coherent autonomous self like in the modernist Freudian view, but as a dispersed self, fractured into various realities made up of connections and nodes in the network of computer terminals. An even more drastic view is to be found in the movie *Matrix,* where the humans are the fuel of the intelligent machine, a totally irrelevant creature in the world of the matrix, if it was not for the energy their bodies produce. To quote the media historian Lev Manovich on Matrix: "We loved it because it was making clear what is already there – a rain (reign?) of data, generating the logic for our reality."[3]

Figure 2. A representation of cyberspace from the movie Matrix. The world is now made of numbers and so are we, although oblivious of it.

These works describe well the difference of the modernist and post-modernist technologies and how our ideas towards technology has changed. However, I will not go into deeper analysis of such works, but will rather concentrate on the implications of the information technology, how the computer as a machine is essentially different from the tools we have hitherto made ourselves and finally

[3] Manovich (2001) [2]. The years the works were released: Modern Times : 1936; Metropolis : 1926; Neuromancer : 1984; Matrix : 1999.

how people working in the field of art are using the new technology, experimenting with its qualities and potential.

1.1. The Symbol Processing Machine

This two-line BASIC program kills itself. It’s really sad. It makes my cry when I see it.

```
10 PRINT “Goodbye, cruel world”
20 NEW
RUN
```

Adrian Ward in Levin (2002). p. 80

At the highest level of abstraction, the computer is characterised by the ability to collect, structure, process, store, replicate, transport and display digital information that can be rendered as text, sound, images, video, 3D worlds and other forms that represent a wide world of symbolic constructs. It is basically a calculating machine that deals with formal instructions and processes them. The invention of the microprocessor (CPU – Central Processing Unit) in the 1950s as a silicon chip, drastically changed the size and function of computers. The computer became an information machine and a semiotic machine. It is controlled by information with the use of a program, and it processes information in the form of data. Thus its internal machinery (the software) is made of the same stuff that it is processing (the data).

The computer is obviously a tool. It is a problem-solving machine, as it used to be called. It allows people to work with whatever information that can be digitalised and manipulated in its mechanism. Initially it was seen as a calculating machine that could easily perform tasks that would make humans err frequently, such as file registering, mathematical calculation and manipulation of databases. But its tasks have changed characteristically in the last decades. Text, images, sound and video are examples of media that can be digitalised and manipulated within the computer. As digital information, it can be changed easily by algorithmic procedures, which are the foundation of computer software. A reverb to a sound or a blurring effect to an image are nothing but an algorithm imposed upon the raw data of digital information.

Finally, the computer has become a strong distribution and displaying medium. With the invention of the Internet,[4] we suddenly had a medium that allowed us to receive, display, manipulate and

[4] The Internet’s predecessor, the ARPAnet was invented in 1973 but it was only in the mid 1990’s that the World Wide Web became publicly available and popular.

send information across an interconnected network of computer terminals. It is a hypermedia that allows for various grades of interactivity between people, databases, neural networks and other types of formal structures. The computer has the ability to access data non-linearly, using its Random Access Memory (RAM). This opens up for different kind of interactivity and works that are stored in the memory as formal instructions can be processed differently according to various algorithms and variables when executed. This results not only in a highly interesting and powerful hypermedia, but also in an intelligent medium, which can understand the needs or the state of its user.

When analysing the computer and its role in society, we can learn from the media studies[5] and their research of, e.g., the television medium. They do not limit themselves to the study of the individual program, but rather to the general placement of the program within the *flow* of programs on each particular evening. The social context is based on how and where people see the program, and which programs precede and follow it, not only what the particular program contains. In this manner we can benefit from analysing the social meaning of a computer program, who uses it, where, in which context and by what means. The meaning of computer software does not only lie in what it can do and how it does it (using which metaphors, methods, functions and protocols) but also who uses it, how and for what purpose.

In his article "Computermedier: Computeren som medie, kommunikation og mediekultur," Jens F. Jensen analyses the computer as a discourse. "Discourses are systems of meaning that are situated in a higher organisational platform than the symbol, the message and the text, but on a lower organisational platform than the 'culture'".[6] Jensen defines discourses by three characteristics:

1) They are defined as a special subject.
2) They have a special social function and locality.
3) They produce and distribute a coherent set of opinions and meanings about the particular subject.

[5] For example: Fiske (1987 and 1990).

[6] Jensen (1990) p. 94

Each computer product – whether it is a program, database, code or symbolic representation – is therefore possible to define as a discourse: as a socially and culturally structured set of interactive patterns, representations and social practises. Jensen points to the fact that the computer's interactive qualities allow for different relationships between the text and the reader (music and the listener) to that in the traditional media. The interactivity of the computer allows the user to manipulate the work's structure itself, the *signifiers* of the work as opposed to being only able to actively relate to the *signified*, the content of it. Thus the reader becomes active in the literal sense of the word. This is not only a possibility, but a necessity if the work is to be explored and read.

This unique quality of the computer as a tool and a medium to be able to process formal instructions in the form of code "on the fly" as it were, is what I will explore in this thesis. The way the processor can read generative code and present it as textual, visual or sonic data in ever different ways is the present object of interest, and we will explore the potential of the medium as a host for an art form (which I have given the generic term "processor art", including generative art and software art) that has had a history down through the ages, but has now found the proper medium in which to be created, stored and displayed.

1.2. Arte ex machina

> The way we are doing music, the machines are as important as we are. It goes both ways. When we are building the tracks we have some ideas in our minds; they're still playing us as well. Sometimes it's very difficult to force the track into that direction. It is often a lot easier and more enjoyable when you just let it grow by itself, in a way. Just go, and follow it.
>
> Mika Vainio in "The Wire", March 1997

Generative works or automation in the fields of art, are not new phenomena. There have been various experiments in the history of art that have used automatic techniques to produce or play artistic works. One has just to mention Aeolian harps or wind chimes, water organs, music boxes, nickelodeons, automatic pianos, jukeboxes, etc. After the Industrial Revolution the interest in such automation increased and we find examples of them in the works of the Futurists, in Duchamp's gap music, in the Dadaist experiments with sound art, with the Russian Constructivists and Surrealists; a mentality of technological experimentation that became fortified with the work of John Cage in the mid 20th century.

Ada Lovelace, the daughter of the famous poet Lord Byron, was arguably the first to see the potential of the computer as a tool and a medium for the fine arts. In the 1830s, Lovelace was a close assistant to one of the early inventors of the computer, Charles Babbage, and worked with him on building the famous Analytical Engine that would automate the process of calculating complex mathematical tasks. She understood that a mechanism like the computer would not necessarily have to deal singularly with raw numbers, but its scope could broaden considerably:

> *The operating mechanism [of the Analytical Engine]… might act upon other things besides number, were objects found whose mutual fundamental relations could be expressed by those of the abstract science of operations, and which should be also susceptible of adaptations to the action of the operation notation and mechanism of the Engine. Supposing, for instance, that the fundamental relations of pitched sounds in the science of harmony and of musical composition were susceptible of such expression and adaptations, the Engine might compose and elaborate scientific pieces of music of any degree of complexity or extent.*[7]

It is this mentality of invention, of utilising machines for human processes of work and play, that has resulted in the creation of the computer as we know it today. The modernist idea of progress and automation, of the ever-new in all fields, also had a strong influence in the arts. Certain currents in culture and technological inventions like the photography (an imaging automation machine), freed artists from realist representation and the artistic project changed to a more subjective and personalised activity, which we see in impressionist, expressionist and abstract art. The craving for the new in modernist ideology became the foundation for the avant-garde in the arts, where artists now started to use found objects (e.g. Picasso's collages and Duchamp's *Urinal*), and moved into non-art social settings where they invaded the space of everyday life and context (the Russian AgitProp or Schwitters's Merzbau apartment are good examples). Live works of art, that arguably could be defined as generative works, became popular with the Dadaists and the Futurists. Randomisation and the accidental became accepted as an important factor in the process of creating and performing art works. Industrialisation influenced artists and they began using industrial materials and processes in their art. Photography and film are good examples and

[7] Roads (1994) p. 882.

so is the serialist mentality of the Bauhaus, of kinetic art and the mid-century experiments with electronic music and sound recordings as we find in the works of Pierre Schaeffer or Karlheinz Stockhausen. Duchamp's experiments resulted in the dematerialisation of art and in the latter half of the century conceptual art became widely accepted with artists, the institutions and the public.

1.3. Processor Art

> Ordinary music is like engineering, where everything's built according to a plan, and it's the same every time you play it. Generative music is more like gardening; you plant a seed, and it grows different every time you plant.
> Brian Eno in The Wire.

With the concept "processor art," we are trying to limit ourselves to works of art that use the microprocessor as a necessary element in their development and execution: works that could not be made without its powers of calculation. It is a vague and blurry categorisation, which is also one of its strengths: by conceiving it as a "family resemblance" concept, in the tradition of Wittgenstein, we are able to talk about works that by many different means participate in the dynamic set of the concept. Processor art is a perspective rather than an art form, a view of looking at what is happening when art is created and executed, with a special attention paid to the medium or technology in which it is being created.

The field of computer arts has had an exceptional growth in the last two decades. We are faced with a variety of forms of computer art such as hypertext, net art and web art, interactive movies and video (or hypervideo), interactive installations, computer music, information art and others yet to be defined. Artists working with these art forms are consequently working with various information and materials in different fields where one might mention robotics, physics, ecology, genetics, biology, kinetic sculpture, electronics, telecommunications, virtual reality, artificial intelligence and artificial life, politics and social processes. When constructing the term processor art, and by limiting it to works that have to be processed in real-time on the computer to exist, I can distinguish myself from works of art that could have been made with other media such as television, video, telephone, newspapers, billboards, fax, mail, exhibitions, installations and social meetings. Thus we end up with categories such as generative art and music, software art, interactive installations (such as sound or video), and composed hyperinstruments.[8] These categories are recent constructions and they might take different forms, names and divisions in a few years time. However, at present they serve their purpose and by analysing these art forms under the concept of processor art, we are able to overlook much of net art that is based on social processes; visual design, where the internet is used as a distribution medium; electronic music and video that could have been made with analogue techniques; and information arts that could have used the older media.

The concept of processor art is not limited to works that are screen based or intended for loudspeakers. We are concerned with works that use the processing power of the computer to calculate input in a way which involves a real-time processing of data, whether from external sensors in the physical environment or from structures within the code itself. The output can be of various types; robots, video projectors, lights and lasers, sound systems, screens or whatever the artist chooses to use. However, in my analysis of processor art works in this thesis, I will concentrate on the "smaller" works (to be enjoyed on personal computers) rather than big and expensive installations designed for galleries and museums or performances using hyperinstruments. The reason is that I want to address a field that might in the future become a new art form with commercial potential and public usage. Will we experience a future where people come home from work and start a generative piece of music that could be generated from various algorithms or external factors such as the temperature, light, season, date and time, and even bodily information from the listener himself? Where people have various screens on their walls instead of a painting, and there is a generative piece of imagery evolving according to the same or other variables?

When analysing computer arts after the advent of the Internet, one becomes acutely aware of the enthusiasm artists had when they started to experiment with this new medium. There appeared a strong aesthetic style in the late 1990s related to the people doing net.art[9] which made it almost a prerequisite of a net.art project to use the

[8] By "hyperinstruments" I mean computer based synthesis of sounds or control of sound samples by means of sensors or controllers, that are connected to a human performer, and feed digital data into the computer which acts according to how the composer "composed" the instrument. See Fernström (2002).

[9] Individuals connected to the nettime mailinglist that started working with the net as an expressive medium, for example: Mark Tribe, Matthew Fuller, Geert Lovink, Alexei Shulgin, Heath Bunting, Vuk Cosic and others.

internet's connectivity in their works. A work would typically be rated higher if it received data from other computers connected to the Net to control some internal structures in the work, rather than just getting the data from the source work itself or the user's machine. There was understandably a certain fascination with being able to access dynamic data stored on terminals all over the world, and use that to control or be part of something in your own work.

However, when writing about processor art, this fact – the potential of the Net and its connectivity between terminals – is not of a vital importance. There isn't necessarily a radical difference in our experience of the work whether the data comes from the Net, is generated within the code, or taken from some information stored on the user's machine. Conceptually there are interesting things to be experimented with concerning art production on the Net, but the computer arts are a much older tradition, which did not start in the mid 90's and I think it is important to look at the whole tradition of algorithmic works, and not just how they became popularised in the end of the 90's. We might propose that there are two modes of conceptual understanding going on here: a) The conceptual view: the understanding where the idea of data coming from networked computers and their context is interesting in senses that might be political, social and epistemological. A piece of music that is controlled by the changing data of the stock market is a typical example. b) The aesthetic view: The view where the music itself and its structures are more important than the fact where the data controlling it is coming from. Thus one is not concerned with the kind of algorithms that lie behind the music, but rather, how they work in a musical context. Both approaches are interesting and one could write in length about the differences of these views, but that is outside the scope of this research. What is important, as said before, is *how* the data (wherever it comes from) is used to generate structures that result in an interesting piece of visual or sonic piece of art.

In my analysis of generative and software art, I will be concerned equally, if not more, with music rather than visual media. There are many reasons for that. Historically, music has always been strongly related to technological inventions and science. From Pythagoras' mathematics of music, through the invention of the note script in the Middle Ages to the invention of the analogue recording and now when the powers of the computer are used to benefit musicians, we can see a very strong symbiotic relationship between music and science. With the use of computers and other digital devices, the processes of music composition, storage and distribution have become strongly interrelated with the scientific and technical resources of society to a greater extent than ever before. Not only have science and technology enriched contemporary music, but the converse is also true: problems of particular musical importance in some cases suggest or pose directly scientific and technological problems which have been taken up by computer scientists and engineers. This strong historical relationship between music and technology/science makes it a very interesting field for research when the subject is generative and software art.

In order to create the conceptual and aesthetic tools to analyse processor art, I will in the next chapter go through some important events in the history of western art that serve as the foundation of what is currently happening in the processor art. In the 3rd chapter, I will analyse some changes in the philosophy of technology that have occurred with the information technology, and shed light on changes in the conception of the human subject which then manifests itself in the works of generative or software artists.

2. Historical view: The Origins of Processor Art

2.1. Technology and the 20th Century Avant-Garde

> It’s always been the artist who perceives that alterations in man are caused by a new medium, who recognises that the future is the present, and uses his work to prepare ground for it.
>
> McLuhan (1969)

The avant-garde of the early 20th century developed amongst other things as a reaction to the Machine Age, which had began with the industrialisation and the migration of people into the cities. Some artistic movements opened their arms happily to technology and used it for creating their works, or even, as was the case of the Futurists, made technology an essential part of their aesthetic. Other movements reacted negatively to technology but it was always there as a point of reference with which they would define themselves. Many of the avant-garde movements were experimenting with the accidental, with generative processes and automation of the artistic creation. After the war, Pop artists like Andy Warhol started to use mechanical techniques in their art production and mass-produced works with various techniques that could be described today as generative techniques. The fusion of technical process with aesthetical concern became a fundamental and characteristic aspect of Pop art and in the earlier abstract minimalism of Jackson Pollock the generative process of painting was extremely important and performed with detailed attention. Technological inventions were starting to influence artists working in all fields of art. The invention of the tape machine was a revolution for musicians and composers. Pierre Schaeffer and Edgar Varèse developed new compositional structures when working with it in the 1950s. Schaeffer called his work “Musique Concrète” referring to the raw sound materials he would record on his tapes. This, of course, is under the influence of Duchamp’s ready-mades and the Surrealist “found objects”, where all kinds of materials could now be presented as works of art. Varèse used the term “Organised Sound” for his music, avoiding Schaeffer’s term, but for both of them, and countless followers, the new recording techniques allowed for a fundamentally different way of storing music and working with sonic materials. Whereas before composers would have to write down formal instructions (the score) for musicians of how to perform the work, now they could start to concentrate on the material aspect of the music: the sounds themselves. In the 1950s electronic sound studios were set up in various parts of Europe. The most important and influential of the period was the Westdeutscher Rundfunk (WDR) studios in Cologne, Germany, founded in 1951 by Herbert Eimert, and later led by Karlheinz Stockhausen. At the WDR a new style in music composition was developed, *elektronische Musik*, which concentrated on creating and manipulating raw electronic sounds, a different use of the technology than in the works of Schaeffer and Varèse who both worked with recorded sounds, the former even declared that he was not interested in sound generation and synthesis.

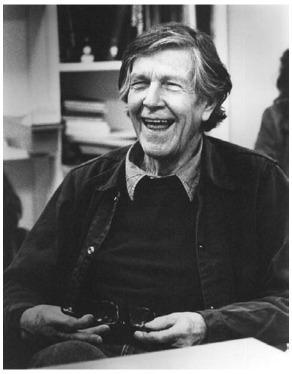


Figure 3. John Cage has been one of most important artists in the 20th century. He saw himself as an inventor rather than artist.

The influence of the artist and composer John Cage cannot be overestimated when studying music and arts of the mid 20th century. In 1939 Cage began experimenting with indeterminacy in his works. In the composition, *Imaginary Landscape No. 1*, multiple performers are asked to perform on multiple record players, changing the variable speed settings. This is one of the initial works that deliberately work with generative and formal instructions for live (or rather, in computer terms: real-time) performances.[10] John Cage's use of indeterminacy culminated with *Music of Changes*, in 1951, a work based on the charts from the *I Ching*, the Chinese book of oracles. In 1952, Cage composed the (in)famous work *4'33*, where he was trying to liberate the performer and the composer from having to make any conscious decisions. The work is in three parts, but in all of them, the musician does not play a single note. The only sounds in the piece are those produced by the audience and the environment.

This process of freeing the artist from the historical constraints and codes of art production and values happened in all levels of the arts. In literature, the Beatniks of the 1950s were experimenting with writing techniques such as trance writing, the cut-up techniques of Brian Gysin and William S. Burroughs, and other generative rules where the writing subject is abandoned for the qualities of random or algorithmic work processes. Burroughs and Gysin also used the tape machine for creating their literary works, recording their readings and cutting the tape back and forth through various procedures. In

[10] Generative methods in the arts are much older and arguably one can define W.A. Mozart's 1787 Musikalisches Wurfelspiel (Musical Dice Game) as generative art in the wide sense. This composition was a series of precomposed measures arranged in random eight-bar phrases to build the composition. Each throw of a pair of dice represented an individual measure, so after eight throws the first phrase was determined.

1960 the French Oulipo group (Ouvroir de littérature potientielle – *Workshop for Potential Literature*) was founded around the poet Raymond Queneau, the mathematician Francois le Lionnais and a group of writers, logicians and mathematicians. Their primary objective was the systematic and formal innovation of constraints in the production and adaptation of literature (they also defined themselves as rats who themselves build the labyrinth from which they will try to escape). The Oulipo believed that all literature is governed by constraints, whether it is a sonnet, a novel, or anything else. By creating new formal constraints, the Oulipo was thus trying to create new forms of literature. In their 1962 manifesto, the Oulipos proposed to use computers for poetic games, where they would, for example, process text with Markov chains[11] and write poetry in the Algol programming language.

2.2. The Minimalists and Fluxus: Experimental Music and Art

"Draw a straight line and follow it" – La Monte Young, Composition 1960 No. 10.

The writer and composer Michael Nyman established the word "minimalism" as a style within experimental music of the 1950s, in his book *Experimental Music: Cage and Beyond* (1974). Like Impressionism, the term was borrowed from the visual arts. Nyman was comparing the style of a certain group of American and European composers to the style of painters such as Frank Stella and Robert Rauschenberg, who reduced their canvas to spaces of black or white, and sculpturists like Richard Serra and Donald Judd, who created huge, uninflected cubes or arcs. Minimalist music is based on the notion of reduction, to strip music to its fundamental elements and materials that the composer would use in his work: harmony, rhythm, dynamics, and timbre. It can be seen as a reaction to the serious and complex serial music that was reigning in Europe and many conservatories in America. Instead of being slaves of complex rules of musical structure, experimental musicians such as La Monte Young, Terry Riley, Steve Reich and Philip Glass – all highly influenced by the work of John Cage – would start to work with simple musical structures, and write compositions that would include a high degree of performer improvisation. These composers

[11] Markov chain is a probability system in which the likelihood of a future event is determined by the state of one or more events in the immediate past.

were influenced by the rapidly changing culture and art scene, but also by jazz, pop music, pop art and not to forget Eastern philosophy and culture.

The minimalists rejected both serialism and the total indeterminacy of Cage, but they owed a lot to both traditions. The serialist music of Anton Webern might be said to be minimal in length and inaudible in dynamics and Cage's composition 4'33" can be seen as the epitome of the minimalist work. It was their interest in non-Western music that influenced the minimalists to work with music that was open in structure, highly flexible and included elements for improvisation. Indian music, with its formal rules for improvisation, was of a great influence, as was the simplistic and minimal philosophy of Zen Buddhism and other Asian thought systems. The influence of pop music can also be seen in the fact that all of these composers were playing in their own ensembles, touring all over the world and often playing in venues that were more associated with rock than classical music. The fact that the music was often rule based – i.e. the score would not be fully precomposed but open in structure – and intended for improvisation, would also make the composers themselves want to play their own music.

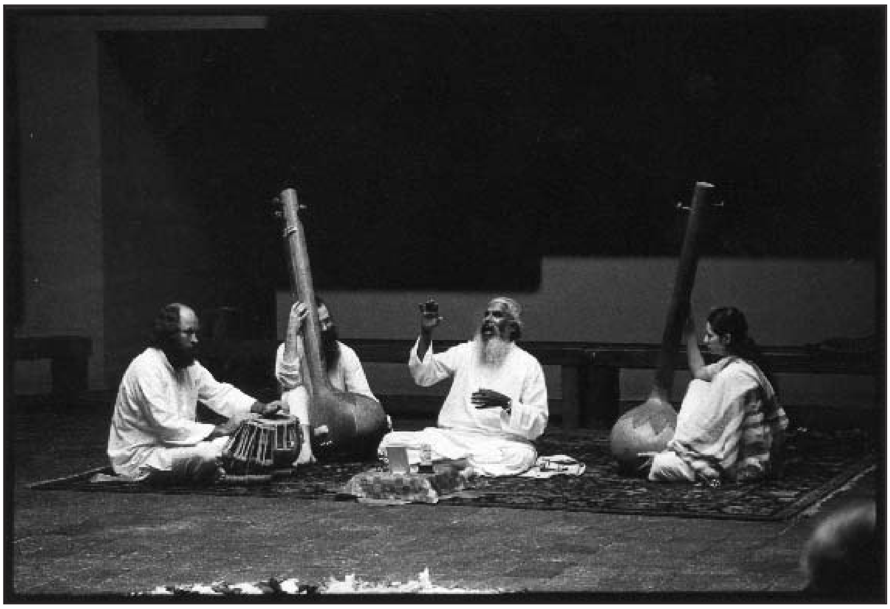


Figure 4. Terry Riley, La Monte Young, Pandit Pran Nath and Marian Zazeela at the Rothko Chapel in Houston 1981.

From our perspective – looking at processor art – one aspect of minimalist music is especially interesting: the attention the minimalist composers paid to process in their work. The simplicity of the form makes the listener concentrate on the musical evolution in the works. For Steve Reich, it was important that the musical process should be audible to the listener. In his essay "Music as a Gradual Process" he writes "I am interested in perceptible processes. I want to be able to hear the processes happening throughout the sounding music. To facilitate closely detailed listening, a musical process should happen extremely gradually."[12] The compositional process and the sounding music become one and the same thing. Again, for Reich, it was a matter of setting up the initial conditions, and then enjoy the result: "Though I may have the pleasure of discovering musical processes and composing the musical material to run through them, once the process is set up and loaded it runs by itself."[13] The minimalists were not so concerned with prescribing a defined time-object whose materials, structuring and relationships are calculated and arranged in advance, but they were rather concentrating on outlining the situation in which the music will occur, the field that was delineated by the compositional rules. Those processes could range from a minimum of organisation to a minimum of arbitrariness, highlighting the fact that music does not have to be defined before it is performed, – rather: the performance is a space where chance and choice are the interesting elements.

Figure 4. Steve Reich performing his piece *Drumming*. Written in the early 1970s, the piece is under strong influence from West African music.

The artistic movement Fluxus was important in many respects. They employed everyday objects and natural materials in their work and emphasized the artistic process over the object. Fluxus wanted a

[12] Reich (1968).

[13] Ibid.

new and closer relationship between art and life, and they explored new media (which they came to call “intermedia”) for the sake of establishing the role of the artist as the destroyer of old forms in order to shape a new consciousness and society. The relationship between visual artists and musicians flourished in New York in the beginning of the 1960s. Yoko Ono was running a small gallery where she invited La Monte Young and others to perform music or happenings, and very often musicians would do performances that had “nothing to do with music”, such as Young’s drawing a line and following it. These Fluxus happenings were popular amongst the artists in New York where Marcel Duchamp, John Cage, Jasper Johns and Robert Rauschenberg would frequent. Disrespect for tradition and a break up of categorisations between the different media characterised the works, as well as humorous approaches to compositions:

> *Bring a bale of hay and a bucket of water onto the stage for the piano to eat and drink. The performer may then feed the piano or leave it to eat by itself. If the former, the piece is over after the piano has been fed. If the latter, it is over after the piano eats or decides not to.*
> *– La Monte Young: Piano Piece for David Tudor #1.*

The Fluxus composers made use of untrained musical performers, and widened the musical performance to non-musical events such as breaking violins, putting flower pots on pianos, rule followed walking or using the piano as a dart board. They also made frequent use of interactivity in their work, an element that has come to be of a central concern in the computer arts. After John Cage and Fluxus, everything was possible in music. Contemporary to the breakdown of musical values and traditions, the academic laboratories were experimenting with music on the computer, with algorithmic music and computer generated sound synthesis. Many of the people that have been mentioned above were invited to participate in the experimentations with the computer as a musical tool, John Cage being one of the earliest.

2.3. The Computer's Induction into the Arts

The new media are not ways of relating us to the old "real" world; they are the real world and they reshape what remains of the old world at will.

(Mc Luhan quoted in Rush (1999) p. 80)

The digital arts trace their roots in a high degree to the development of military defence systems after the second World War, and not to the art academies. The world's first digital computer, the ENIAC (Electronic Numerical Integrator and Computer), which would fill a large garage, had been introduced at the University of Pennsylvania in 1946 and in 1951, the first commercial computer (Univac) was patented. Very soon, people in the research labs and educational institutions were experimenting with the use of the computer for musical composition and sound synthesis. Typically, computers were used to organise the macrostructure material of the music, i.e. the composition and arrangement of notes or tonal pitch. At the time, there were experiments with going into the microstructure of the music, the analysis and synthesis of complex sound structures, but the processor speed and memory of the early computers were not fast enough to perform these tasks efficiently. The first synthetic sounds generated by a digital computer were made by Max Matthews[14] and his group at the Bell Laboratories in the late 1950s and for the first time composers realised that here they were faced with a tool that had potential to change music production forever. It was much later, in 1965, that the first *visual* works produced by a digital computer appeared. These were made by Frieder Nake and Georg Nees in Germany and A. Michael Noll from the Bell Laboratories, K.C. Knowlton, B. Joulesz and others in the U.S.A.

In the 1960s the relationship between art and technology became stronger: people were collaborating between the fields, and artists would often specialise in the field of technology, whereas engineers or programmers would in turn produce work that were to be defined as art. A new cultural awareness was taking place where art institutions, academies and art festivals would realise the powerful connection of art and technology. One of the earliest festivals of this kind was the 1966 "Festival of Art and Technology" in Stockholm.

[14] Matthews, who the electronic music software Max is named after, is perhaps more recognised for making the track "Daisy" which the computer Hal sings when it is about to die in the science fiction movie 2001: A Space Odissey by Stanley Kubrick.

However, this symbiotic relationship between art and technology happened largely within academic institutions and laboratories and some time was to pass before it resulted in attention from the mass media and the public.

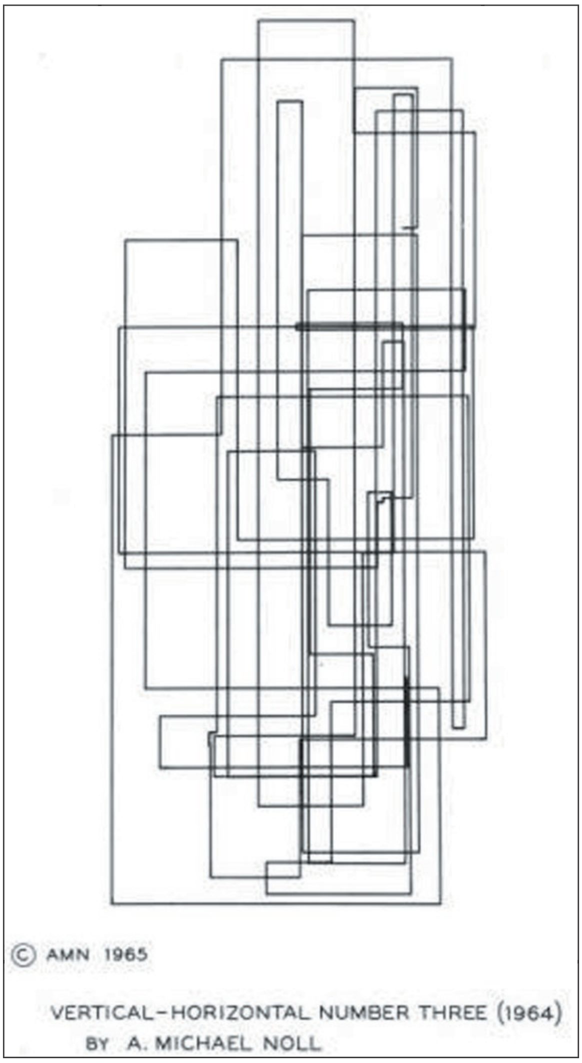


Figure 5. Computer generated art by A. Michael Noll. Noll was one of the first artists to use the computer for visual works.

In the early sixties Robert Rauschenberg, met Billy Klüver, an electronic engineer and sound wizard who had collaborated with several artists, most notably Jean Tinguely on his self-destroying machine. Klüver worked with John Cage and Merce Cunningham on their multimedia stage events, *Variations V*, where Klüver designed an interactive sound system that responded to movements. Rauschenberg and Klüver founded the EAT (Experiments in Art and Technology) in 1967, an enduring and influential collaboration between artists and engineers. Computer art was now getting attention from the general art world and a few important exhibitions were held that helped establish the computer and technological art into the main art scene. The new technological age might have been introduced with the Museum of Modern Art's 1968 exhibition, "The Machine As Seen at the End of the Mechanical Age." It was a response to the demise of the modern machine age and the advent of the post-industrial, post-modern information society culture based on instant communication services. In 1970 there were few important exhibitions that made further influence: "Cybernetic Serendipidy" at the ICA in London, "Information" at the MOMA in New York, where art was seen as pure information, drawing from the conceptual art that was becoming strong at the time, and finally "Software" at the Jewish Museum in New York. "Software's" main concept was that computer software and information technology would serve as main metaphors for art. Jack Burnham, the curator of the exhibition, conceived of software as parallel to the aesthetic principles, concepts, or programs that underlie the formal embodiment of the actual art objects, which in turn parallel the hardware of the computer. Although these exhibitions focused on electronic media, the works exhibited were still largely based on modernist premises in their aesthetic attitudes.

Apart from a few individuals, hardly any influential visual artists were working with the computer before the late 1980s and public awareness (as opposed to the world of art) of artistic experiments with the use of computers was very limited. There are various reasons for this: *firstly*, there was a strong anti-technological sentiment in the late 1960s and the 1970s. Ecological and anti-nuclear groups were strong and they would overshadow the experiments done with the computer as a tool or medium; *secondly*, computers were expensive and big (in physical terms) and only the privileged people at the technology research centres or the academies would be able to concentrate on working on computers; and *thirdly*, computer interfaces in the early days were very primitive and made all interaction with it laborious and complicated. The computer also had to be programmed in complex, logical and dry programming languages that would not attract the more "artistically" minded. It was only in the mid 1980s when the computer became affordable in price and size that people could really start to experiment with it as a tool for creating artistic works. A new generation of digital artists had come on to the scene and they were approaching the medium from a different perspective. Instead of imitating physical art, they concentrated on the qualities of the new medium itself and started experimenting with interactivity, artificial intelligence, sensors, biofeedback and other things that analogue media are not as good at.

It was really in the 1990s that the golden age of the computer art started when computers became powerful and cheap enough to deal with and when, in the middle of the decade, the Internet became available as a public medium. Many of the artists working with computers in the 1990s had grown up with them and learned programming languages like Basic, Pascal or C at an early age, starting by using cheap personal computers like Sinclair or Commodore, but now the situation had changed such that they were able to buy powerful computers which they could use in their work with images or sounds. Software like Photoshop, Premiere, Illustrator or Cubase was widely available at reasonable cost. Designers, musicians, film makers, video artists and writers all started using the computer for their work and this created a growth in aesthetic styles and new musical forms.[15] One could argue, for example, that musical styles such as house, techno or drum'n'bass would not have appeared without the use of computers, due to the software abilities of arranging samples, looping and creating complex rhythms that few drummers would be able to perform. The software determined the artistic process and creativity to a certain sense, but at the same time the artistic experiments with the software expanded it and resulted in its further development.[16] However, the mentality of innovation and experimentation today is such that many people prefer writing their own software for visual or musical creations, which makes the composer Kim Cascone comment, echoing Marshall McLuhan, that "the medium is no longer the message; rather, specific tools themselves have become the message."[17]

The desire for customised, flexible software has resulted in a new art form, software art, where artists-programmers are writing their own software, which is seen as an artistic expression in itself. This can range from individual programs like Adrian Ward's Auto-Illustrator or Netochka Nezvanova's Nato.0+55 to individual patches (software extensions) for musical programs like Max/MSP or plug-ins for commercial programs like Adobe's Illustrator.[18] In fact some companies have starting producing software that allow the user to create their own instruments from scratch by visual programming of which Native Instrument's Reaktor is a good example. In Reaktor, the user is able to program their own instruments, sequencers, samplers and other applications that can communicate with external devices such as keyboards, drumpads or whatever sensor equipment that can be translated into digital information. In year 2001, the media art festival Transmediale in Berlin introduced the category *software art* and this year's (2002) Read_me festival at the Macros Centre in Moscow was probably the first festival that concentrated solely on

[15] The media critic Lev Manovich, claims that the cut'n'paste functions of programs such as Photoshop has created a special visual style which he relates to another current in our culture, that of sampling and rearranging. See Manovich (2001).

[16] This symbiotic relationship can be very powerful as is the case with Macromedia's Flash software. What was intended to be a simple vector animation program has now become a very powerful programming suite for interactive multimedia content on the Net. The people using the software, broadening its scope and applicability, experimenting with its qualities and potential, gave a strong feedback to the developers of Flash at Macromedia who responded to the new needs.

[17] Cascone (2000) p. 12.

[18] These programs will be discussed further later in this essay.

artistic software.[19]

Software for generative arts has also appeared in the last few years. In 1996, Sseyo in England released the Koan software that allowed musicians to create generative music and they even wrote a browser plug-in such that the music could easily stream through the Net. Brian Eno was one of the first to fall for the technology and he released an "album" on a floppy disk soon after he got the program. At the MIT MediaLab John Maeda and others developed the program dbn (Designing by Numbers) which allows the user to program generative pieces of visual works using a simple programming language, but other artists have used software such as Director or Flash for the same purposes. This year the Morpheus CD was released in England with generative music programmed in Supercollider[20] Maybe for the first time in history have we got a mass-cultural and distributed product that contains generative works of music, intended for the computer and not for cd-players. This is what I mean by processor art: an art form that has to be processed on the computer every time it is listened to or watched, and in my view we are going to face complex questions of aesthetic evaluation of such works, the questioning of the author, of the ontological status of such works, the collaborative role of the user, etc. I will address this more fully in the 4th chapter, but before I start to outline the aesthetic characteristics of processor art, the nature of the new technology has to be explored. Additionally, we need to investigate how currents in our culture are responding to new technology resulting in a different perspective of the human subjectivity.

[19] See respectively: http://www.transmediale.de/en/02/ and http://www.macros-center.ru/read_me/index-en.html

[20] Supercollider is a software environment and programming language. Further info at: http://www.audiosynth.com

3. Transitions: Currents in the Philosophy of Technology

3.1. Technological Transitions

> Code shapes technology into whatever form it desires. Before code, any system was fixed by its design, no matter how flexible. With code, despite its structure being fixed and defined by the system on which it is executed, a new area of creativity is opened: a definition process rather than product.
>
> Adrian Ward in Levin (2001) p. 66

To understand the relationship between technology and art we might benefit from studying Martin Heidegger's 1955 essay "The Question Concerning Technology" in which he outlines the essence of modern technology and tries to establish a proper relationship towards it. Contrary to how many people read Heidegger, he was not technophobic at all. His intention was to establish a free relationship between the human being and the essence of technology. An essence which is nothing technological in itself, but rather a result of how western culture has metaphysically defined nature and the human subject all the way back to the early Greek thinkers. For Heidegger "modern technology" has three interrelated meanings: first, the industrialism's new production processes, techniques, devices and systems; second, modernity's association with secular world view and rationalistic, scientific, commercialist, utilitarian, and anthropocentric way of being-in-the-world; third, the modern, technological man's mode of understanding or revealing things which symbolises the way industrialism and the rational mind relate to the world.

What Heidegger is trying to show is the ontological set of conditions necessary for this particular technological worldview. He believed that in the historical epochs of the human mind, we have undergone some conceptual and ontological movements in which we have been "thrown". These are not worldviews in themselves but rather the conditions necessary for the emergence of a particular worldview. The technological man sees the entire world as something existing for him to consume. The whole world thus becomes for him a *standing reserve* [g. *Bestand*]. In this system, for something "to be" means for it to be raw material for the self-enchanting technological system. This essence of modern technology shows itself in what Heidegger calls *enframing* [g. *Gestell*]. „[Enframing] is nothing technological, nothing of the order of the machine. It is the way in which the real reveals itself as standing reserve."[21] Enframing is the manner in which we see the world, and this essence of technology is the way the world now discloses itself for us. It monopolises the way we experience the world and excludes other ways, such as the religious, the aesthetic, or the ethical view. The technological enframing leads to crisis in values, and instead of an authentic world of beings, we get a desacrilised world of standing reserve, which exist only for and because of us; the humans who always think they are at the centre of the universe.

Heidegger traces this productionistic metaphysics not only back to the industrial revolution of the 18th century, but to the Greeks. The history of the West is the story of how the Greek metaphysics degenerated into modern technology. What we need, according to Heidegger, is a proper relationship with Being. Only then can humanity enter into a meaningful relation with the essential nature of technology. For Heidegger, it is here that art comes into the picture. Art is our hope for being able to disclose entities for their own sakes, to let them speak through their creation, without us being the technological agent using our enframing power to force material together into a specific form. Through etymological analysis, he shows that the Greek word *techné* meant not only technology, but also art. Both art and craft production are modes of disclosing, what the Greeks called *poiésis.* „There was a time when it was not technology alone that bore the name *techné*. Once that revealing that brings forth truth into the splendour of radiant appearing also was called *techné*. Once there was a time when the bringing-forth of the true into the beautiful was called *techné*. And the *poiésis* of the fine arts were also called *techné*.“[22] Thus art has the potential to disclose the being of entities, to reveal things *as things*, in their individuality, their depth, their being – not as standing reserve, stock of energy or commodities. Art can free us from the total enslavement to technology by opening up another sensibility, and by teaching us to use technical objects appropriately.

One can smell the modernist mentality when reading Heidegger's essay. He was of course criticising modern technology and how modern man relates to nature as such. But now at the beginning of the 21st century, we are faced with both technology and art that is essentially different from what Heidegger was referring

[21] Heidegger (1977) p. 23.

[22] Ibid. p. 34.

to. First, the *technology* Heidegger had in mind was the technology of nuclear energy, of large-scale industry, of the conveyor belt and the mass production. It is the technology of the Machine Age capitalism, of utilitarian, profit-oriented mentality and international monetary functions. In this model the Rhine becomes a standing reserve for power stations or – which is still possible – we enjoy the beauty of the river, but then as customers (standing reserve in the form of man) of the tourist industry. The Rhine written about in Hölderlin's poems is disappearing because the technological way of revealing things has the power to drive out other ways of revealing. Second, the *art* Heidegger wrote about was not that of the Dadaists, Futurists, Constructivists or Bauhaus (movements which all were influenced, if not fascinated by technology and whose artists often used technological means in creating their works). These avant-garde movements were outlawed from Nazi Germany and it is highly unlikely that Heidegger approved of their nihilistic, post-romantic, expressivistic, and hence anthropocentric and subjectivistic conception of art. No, Heidegger's art was rather that of Hölderlin, Rilke, Cézanne and Van Gogh; artists who believed that their role was to let things *be* through their works.

As I will argue later on, we are relating to something of a different nature when we talk about technology and art today, and the context in which we are situated is essentially different. The romantic Heidegger referred to pre-capitalistic time where the human was defined in relation to Nature and the animal world, a world which evoked fascination and attraction, repulsion and resentment. His own was the world of modern capitalism, of the Machine Age where the human was defined in relation to the industrial machine and what kind of work the subject had in the industrial society. In our world, the world of post-industrial late capitalism, the word "technological" has an altered meaning. Today the human subject is defined in relation to technology, but it is the high technology of cybernetic systems, computer networks, ecosystems, biogenetically engineered organisms, expert systems, robots, smart machines, and cyborgs. It is the age of virtual selves[23] where self-organizing systems, artificial

[23] See Turkle (1996). In her book, *Life on the Screen*, Turkle tells the story of how the computer has has had an impact on our psychological lives and our evolving ideas about minds, bodies and machines. What is emerging with the age of networked computers, Turkle says, is a new kind of identiy: a defragmented, decentred and multiple identity.

intelligence, neural networks and artificial life, genetic algorithms, knowbots, and mathematical wonders like chaos theory or fractal mathematics have come forth, all because of the invention of the computer.

Manuel De Landa, in his book *Thousand Years of Nonlinear History,* defines the virtual environments that the computer provides as "epistemological reservoirs" which can lead to the extinction of many old philosophical doctrines. Essentialism, reductionism, and formalism are the first ones to go and our intellectual habit of thinking linearly will change. The science of Artificial Life brings forth an abundance of concepts and ways of thinking that were seen as trivial before, but now actualise themselves as important tools to understand life in general.

> *Artificial Life is the study of man-made systems that exhibit behaviors characteristic of natural living systems. It complements the traditional biological sciences concerned with the* analysis *of living organisms by attempting to* synthesize *lifelike behaviors within computers and other artificial media. By extending the empirical foundation upon which biology is based beyond the carbon-life chain life that has evolved on Earth, Artificial Life can contribute to the theoretical biology by locating* life-as-we-know-it *within the larger picture of* life-as-it-could-be.[24]

The older Artificial Intelligence top-down approach, where structures were defined above and the system would react to input with predefined behaviours, is substituted in Artificial Life with the bottom-up synthetic approach, where smaller objects are programmed and given their own life-like behaviour, but where unknown input might cause mutations in the objects, resulting in emergent behaviours that might have been unforeseen by its human creators. Artificial Life attempts to understand high-level behaviour from low-level rules, and works in a new paradigm where new scientific discoveries have done away with rigid, mechanistic and hierarchical views of organisms. In this paradigm life is seen rather in terms of dynamics than mechanics:

> *Life depends critically on principles of dynamical self-organization that have remained largely untouched by traditional analytic methods. There is a simple explanation for this – these self-organized dynamics are fundamentally non-linear phenomena, and non-linear phenomena in general depend critically on the interactions between*

[24] Langton (1989) p. 1

> *parts: they necessarily disappear when parts are treated in isolation from one another, which is the basis for any analytic method. Rather, non-linear phenomena are most appropriately treated by a synthetic approach, where synthesis means "the combining of separate elements or substances to form a coherent whole".*[25]

I find this new scientific worldview of non-linearity, complexity, networks, emergent behaviours and other factors to be highly relevant when analysing processor art. Not only do the programmer-artists partake in this mentality, but it also has important consequences when we are to relate to their work. In processor art we are to find innumerable new aesthetic concepts that up to now have not been an issue. The distinction Artificial Life makes between *genotype* and *phenotype* is a good example.[26] As a concept taken from genetics, the genotype might be defined as the specification of a machinery, whereas the phenotype is the actual behaviour of that machinery. Thus the genotype is the complete set of genetic instructions that makes an organism's DNA, and the phenotype is the physical organism itself – the structures that emerge in space and time as the result of the interpretation of the genotype on a particular environment. In genetics the term "epigenesis" is used to describe the process whereby the phenotype grows from a genotype. In much processor art, as we will see, we find the concept of epigenesis useful to describe the concentration on process in the works, as very often the object of interest in the art is not the code itself (the genotype), but rather the process (epigenesis) and the manifested outcome (the phenotype). As I will argue later, it is the *functions* of the code (i.e. the process it creates) that has aesthetic value, rather than the code itself.

The computer has given us a virtual laboratory where systems of unimaginable complexity can be made to deal with natural phenomena, of which the chaos theory, fractals or genetics are good examples[27]. As chaos theory has shown, it is precisely when a space or system reaches a certain degree of complexity and its processes become unstable, unpredictable and chaotic that mutation or emergent properties occur[28]. Consequently, it becomes less likely

[25] Ramos (2002) p. 27.

[26] Ramos (2002); Langton (1989); Wilson (2002).

[27] Peitgen (1992); Abraham (2001).

[28] The chaos theory, cybernetics and autopoietic theory has had influence in all fields, from architectural studies (Koolhas: 2001) to Artificial Life (Maturana &

that the complex series of interactions, alterations, divisions and combinations possible within that space can be foreseen by humans, much less controlled. This is also the case with the complex system of communication networking. The new communication system radically transforms space and time, the fundamental dimensions of human life. Localities become disembodied from their cultural, historical, geographic meaning, and reintegrated into functional networks, or into image collages, inducing a space of flows that substitutes the space of places. Time is erased in cyberspace when past, present and the future can be programmed to interact with each other in the same message, be it in the form of automatic email systems, networked databases, or other complex automatic information systems.

For Manuel Castells, we are experiencing a new relationship with nature once again. First we had an age where nature dominated culture. The codes of social organisation almost directly expressed the struggle for survival under the uncontrolled harshness of nature. In the modern age, we saw ourselves as the masters of nature and we believed in the triumph of human reason and its use of technology. Humankind found its liberation from natural forces and its submission to its own abysses of oppression and exploitation. At present we are entering a new stage in which culture refers to culture, and not nature anymore, having reached the point where nature is artificially revived or preserved as a cultural form. The environmental and ecological movements and political parties are good examples of that: the aim is to reconstruct Nature as an ideal cultural postcard for us to enjoy and withdraw in.[29] This complex nature of our technology and its transitions, which we have lost control over – and has changed our imaginary and scientific codes – has resulted in the transformation of the human into what has come to be called the “posthuman”.

Varela: 1980) and Sociology (Luhman: 1986). Autopoiesis is a concept designating a state in a system where the members of the system interact with each other in such a way that they continually produce other members and the relationships between them. In the IT sector Winograd and Flores (1986) have been using autopoetic theory for analysing enterprise structures.

[29] Castells (1996) p. 477

3.2. Human Transitions

In the [Artificial Life] paradigm, the machine becomes the model for understanding the human. Thus the human is transfigured into the posthuman.
N. Katharine Hayles (1999) p. 239

In his essay "The Cultural Logic of Late Capitalism," the philosopher and cultural critic Frederic Jameson writes about how postmodernism defines our new technology as machines of reproduction rather than of production and how the evolution of digital technology makes different demands on our capacity for aesthetic representation to that of the machinery of modernism. This has resulted in a complex problematic in the way humans view the world and its coherence:

> *My implication is that we ourselves, the human subjects who happen into this new space, have not kept pace with that evolution; there has been a mutation in the object unaccompanied as yet by any equivalent mutation in the subject. We do not yet possess the perceptual equipment to match this new hyperspace, as I will call it, in part because our perceptual habits were formed in that older kind of space I have called the space of high modernism.*[30]

The essay was first published in 1984 and in it Jameson calls for an aesthetic of cognitive mapping which would help us to understand our new cultural form. We have to note that this was before the popularisation of the computer and before artists like Knowbotic Research or artificial life researcher Tom Ray put their experiments onto the Net where people could connect to virtual spaces representing dynamic flows in cyberspace or allowing Artificial Life to breed and evolve on their network connected computers.[31] It was the same year, however, that William Gibson published his cyberpunk novel *Neuromancer* where he created the word "cyberspace" and established the imaginary world it denoted. What Jameson manages to outline with his analysis of global capitalism, multinational corporations and complex networking that characterises postmodernism, is that a new kind of subject is appearing, a subject very different from the ideal humanist subject of modernism.

R.L. Rutsky analyses in his book *High technè* how the culture of postmodernity has ended up becoming one of techno-culture. As our habitual world becomes ever more liable to technological

[30] Jameson (1991) p. 38

[31] Knowbotic Research is: http://www.krcf.org/krcfhome/ and Tom Ray's Tierra: http://www.isd.atr.co.jp/~ray/tierra/

digital reproduction, any distinction between technology and culture begins to vanish. When all our communication and information is transmitted and stored by means of digital media, the technology itself comes increasingly to be seen in terms of cultural data that make up what might be called the techno-cultural memory. A phenomenon that has become too dense and complex to be thought or represented as a whole. Technology has begun to seem beyond human instrumentality and control. Not subjected to the modernist instrumental rationality anymore, technology takes on a much more unpredictable "techno-logic" of its own. Not only has our conception of technology undergone a mutation, but technology itself has come to be seen as a mutational process or logic which is beyond our conceptual understanding.

The modernist idea of the human subject is clearly manifested in the writings of Sigmund Freud. Rutsky theorises that in fact we might see Freud's division of the psyche as a replication of the "psychic" structure of technological modernity. Thus, modern technological rationality becomes the very model for the ego, where the irrational, primitive, animistic or magical is "repressed" back into the unconscious. For Rutsky, the technological unconscious, which is always a techno-cultural unconscious, is a space where technological otherness and cultural otherness are linked by virtue of the fact that both are excluded from and by the Western technological modernity. However, today the representations of this very technological unconscious, especially when relating to conscious artificial life or agency, have begun to change. "Thus, for example, the monstrous computers, robots, and other technological mutants of traditional science fiction have given way to notions of artificial life, artificial intelligence, and intelligent agents, to cyborgs and other biotechnological life forms that, despite their 'inhuman' status, are no longer represented simply as a threat to humanity."[32] In the techno-paganism of John Perry Barlow (the former Greatful Dead lyricist), Timothy Leary (the LSD prophet of the 60s), Howard Rheingold and other subscribers to the "Californian ideology,[33]" we find a spiritual discourse where technology is presented as something that might liberate the human from the burden of everyday rationality and boredom. The human beings are not defined anymore solely in terms

[32] Rutsky (1999) p. 136

[33] Barbrook (1995)

of their control of an objectified, instrumentalised world, but they are rather seen as participants in a world filled with autonomous forces and agencies, with which they must deal, interact and cooperate. The boundaries between the human subject and the world are not seen to be rigid or impassable anymore; instead we get a sense of openness, of connection and interconnection, of mixture between the two. Thus we get the conception of posthumanism, a state where the human has become a kind of a hybrid, a permeable, mixed and complex entity, a cyborg.

As Donna Haraway has shown in her "Cyborg Manifesto", the cyborg is simultaneously an entity and metaphor, a living being and a narrative construction[34]. The world of science fiction and techno-paganism has taken the concept with open arms, but the fact is that cyborgs actually exist. Over 10 percent of the Western population are estimated to be cyborgs in the technical sense, including people with electronic pacemakers, artificial joints, drug-implant systems, implanted corneal lenses, and artificial skin. A much higher percentage are metaphoric cyborgs, i.e. people that are occupied with computers, video games, mobile phones and other devices that connect them to a complex network of interactive loops. This new subjectivity of the cyborg results for Scott Bukatman in a "*[t]erminal identiy:* an unmistakably doubled articulation in which we find both the end of the subject and a new subjectivity constructed at the computer station or television screen."[35]

In her book *How we Became Posthuman*, one of N. Katharine Hayles' projects is to investigate this new subjectivity by analysing the disappearance of the body in postmodern theory, where the body is seen primarily as a linguistic and discursive construction. This happened parallel in two fields: with the cybernetics of the 1950s and 60s that stripped information of its body with their experiments and theories of artificial intelligence and information systems, and with poststructuralist theories such as Foucault's archaeology of knowledge that saw the human body as a play of discourse systems. For Hayles not only has the body disappeared but a new subjectivity has emerged which is not to be seen in the same terms as the modernist subject of control and mastery of nature: "This subjectivity is constituted by the crossing of the materiality of informatics with

[34] Haraway (1991).

[35] Bukatman (1993) p. 9.

the immateriality of information."[36] The disappearance of the body is problematic for Hayles and she provides a new flexible framework in which to think about embodiment in the age of virtuality:

> *This framework comprises two dynamically interacting polarities. The first polarity unfolds as an interplay between the body as a cultural construct and the experiences of embodiment that individual people within a culture feel and articulate. The second polarity can be understood as a dance between inscribing and incorporating practises. Since the body and embodiment, inscription and incorporation, are in constant interaction, the distinctions forming these polarities are heuristic rather than absolute. They nevertheless play an important role in understanding the connections between an ideology of immateriality and the material conditions that produce the ideology.*[37]

When analysing computer art, this difference of the body as a theoretical construct and embodiment becomes of a vital importance for the digital aesthetic, as we will see later. What would be important to mention now is the similar distinction between *inscribing practises*, which are normalised and abstract, usually considered as a system of signs operating independently of any particular manifestation and *incorporating practises*, which are actions that are encoded into the bodily memory by repeated performances until it becomes habitual. Paul Connerton, in his book *How Societies Remember* elaborates further on this distinction of the concept of the body and embodiment. The body is like a Platonic normative construction, an abstract and idealised form to be seen relative to some set of criteria, whereas embodiment is always contextual and immersed in the specifics of place, time, physiology and culture with its infinite variations, particularities and abnormalities. There is an immense difference between these two modes of operating within the idea of human corporality and how it relates to information and here, I believe, we may find one of the biggest divides between traditional and digital arts.

[36] Hayles (1999) p. 193

[37] Hayles (1999) p.193

3.3. Media Transitions

> The history of every art form shows critical epochs in which a certain art form aspires to effects which could be fully obtained with a changed technical standard, that is to say, in a new art form.
>
> Walter Benjamin (1969) p.243

An excess of books and articles have been written about Walter Benjamin's notion in his essay "The Work of Art in the Age of Mechanical Reproduction," that when works of art become reproducible by mechanical means, they loose their sense of "aura", i.e. their uniqueness and material and historical situatedness. Photography and film were the media that Benjamin analysed and he showed how their invention caused a changed role and status of the artwork and our perception of reality itself. The video artist Douglas Davis elaborates on this idea in his essay "The Work of Art in the Age of Digital Reproduction" where he states that there is no sense in talking about the "original" and the "copy" anymore as digital works of art can be copied and distributed without any loss of quality at all. There is no difference between the two. In his essay Davis overlooks the fact that even in the production phase of digital works it is hard to talk about an original – as one is constantly copying, manipulating and mixing different materials together – when he states that it is the "repetitive 'copy' that is dead, not the original."[38] I find this a highly problematic issue and probably one that is haunting us from the pre-digital times when artworks could be falsified or reproduced as copies of the original.[39] In generative art, this issue – which I will argue, is a categorical mistake when relating to digital arts – of the original and the copy becomes even more complex. We are now faced with the disappearance of the referent itself. Generative works may not have a physical object nor a unique idea, which they represent. They are rather a manifestation of a certain family resemblance conception that the artist-programmer had in mind, a phenotype resulting from the coded genotype. Another way to explain it would be to state that the work's signifiers are never the same, and the signified is not an

[38] Davis (1995)

[39] I asked Davis in a lecture where he talked about the disappearance of the copy, what would happen if I would take his "The World's First Collaborative Sentence" (a work that is growing on the Net, and which he sold for good amount of money to a gallery) and copy it onto my server and let it grow there as well. People would then be adding to two different sentences. Now, which of them is the original and which is the copy? We would have two different works, one that was sold to the gallery and another one "stolen" by me. Davis' answer was: "No comment."

object or a single idea, but rather a set of networked possibilities.[40]

To elaborate further on this with an example, we could imagine a photographer using generative methods of arranging the objects he is photographing. The photographer fills many films of images of the objects and develops them perhaps using generative methods (by controlling lights or other elements that affect the development process). Each photo is a "variation" of the theme he is working on, but each photo shows the *material reality*, which is being photographed in different ways. In generative works on the computer, there isn't necessarily any original object, any material referent or situation which the work is representing. The only material we've got is the artist's idea represented in the code itself, and the "realisation" of the code (the visual or sonic outcome) is essentially different from its own substance. There is no analogy. The work is an expression of an idea that is stored in the form of binary digits, ontologically dissimilar to the "materialised" outcome, which again can have innumerable variations.

Works realised this way through code are breaking up the modernistic tradition of the usage of recording technologies. Code that generates images, videos, sounds/music, 3D worlds, architectural forms, etc. does not have to work with any recorded source material, it is able to generate the material from instructions written by the artist. A 3D program like AutoCAD, for example, allows architects and environmental planners to plant trees, flowers or other physical objects that "grow" into a different versions of the genotype or the rhizomatic idea. Each tree will be unique representation of an ideal tree, but note that the concept of an ideal tree is not a Platonic one, but precisely a rhizomatic structure in the sense Guattari and Deleuze give to the term.[41] Many artists working in generative arts have been experimenting with this idea and one could mention here Celestino Soddu's generative architecture and furniture design, the generative music of Brian Eno and the Morpheus team, the visual art of meta,

[40] The signified is not, and cannot be the code itself as that is not of an aesthetic importance in generative art, but rather the set of ideas, the rhizomatic potential of the concept which the artist-programmer had in mind.

[41] "Let us summarize the principal characteristics of a rhizome: unlike trees or their roots, the rhizome connects any point to any other point, and its traits are not necessarily linked to traits of the same nature; it brings into play very different regimes of signs, and even nonsign states. The rhizome is reducible neither to the One nor the multiple... It is composed not of units but of dimensions, or rather directions in motion." Deleuze (1987) p. 21

lia, Golan Levin or Antoine Schmitt, or the 3D objects made by Karl Sims and William Latham which grow in various forms and appearances when they are given life by triggering the code behind them.[42]

The French philosopher and economist Jacques Attali has analysed in his book *Noise: The Political Economy of Music* how the recording techniques that came with the advent of modernity changed musical practises and cultures. Before the recording, people would listen to music in the town halls, upper class homes or on the town squares and markets. Music was dynamic, fluid and often interacted with the environment. With the invention of the phonography we get a new organisational network for the economy of music. The consumption of music became individualised and the main goal of the producers would be to create increasing demand for music so they could produce and stockpile musical products in an increasingly capitalistic market system. Music became *enframed*, to use the generic term of Heidegger. Instead of being primarily a live performance phenomenon, music became something that was recorded, mass-produced, distributed and sold in innumerable copies. This resulted in a metaphysical view of music as something that would have to be certain way, and if musicians diverge from that stored version of their music, they would be performing a "wrong" version of the work. This is less a problem in classical music and jazz than in pop or rock, as classical music is traditionally written down and each performance is an evocation of the composer's work. However, it is a fact that some recordings of classical works have become a standard and conductors have to be careful not to diverge too much from the accepted standard.

Let us imagine a possible world in the tradition of modal logic.[43] Imagine computer technology having been invented before Edison's invention of the phonograph. There is a rule of thumb that whenever a new communication medium is invented, some artists will start to use it in a creative way. We just have to look at the photograph, phonograph, film, video or the Net to see how that dynamic works. In our possible world, we are living in the late 19th century, and we have several different musical practises in Europe,

[42] Most of these works will be discussed later.

[43] Thought experiments which are done to test out possible ideas in the rational world of logic. Developed by Gottfried Leibniz, David Lewis, Saul Kripke, etc.

but two general ones: the classical music of the concert halls and the popular folk music of the town squares and the taverns. These musicians would start to use the computer for their creative output, just as today, and say that they would start to write generative music which would for example be conditioned by algorithms, changes in environment, temperature, and other factors. My bet is that in this possible world, people would not have the same difficulties with understanding the ontology of generative works of art, for the simple reason that they are not accustomed to the recorded work to the same degree as people at the beginning of the 21st century. In our world and at our time, we find it hard to get beyond the idea of the recorded version being the original and alternative versions or cover versions being somehow copies of the "real thing".

The digital media have a certain characteristic that the older media did not have, namely that their functionality – by means of computer programs – makes it a very mouldable medium. A text editor is written in a programming language not different from the natural language we write with it, but both are translated to binary code within the machine. The computer is a meta-machine in the sense that we can build whatever we want on it as long as it is possible. One can write a program for a special purpose, a filter, a plug-in, an extension, or a virus that affects and changes the functionality of other programs. Software art is one of the oldest art forms of the computer, but it has only recently come to be acknowledged as such. It is one of the most natural art forms of the computer, because for a creative person that wants to see or listen to something and use the computer to facilitate the process of creating it, the most obvious thing to do is to sit down and write such a program. Software artists are often people who are not content with standardised commercial programs that lead the user into a certain way of thinking and working.[44] By writing a program, one is involved in a highly creative and aesthetic process, where interface design, interaction design, functionality, speed, strength and other factors are the criteria for evaluation.

The commercial software companies are becoming aware

[44] Whose results we see clearly in the design industry and on the Net. Whenever a new version of some program comes out, the designers working with it come up with the same style of design. A good example would be the embossed Photoshop button found in many html files some years ago or the visual style of Flash that has become an art movement on its own.

of the fact that many people do not want standardised products, but something they can add to, change and extend by writing additions. Thus we find Adobe allowing for plug-in creation for their software, Macromedia has a form of allowing people to write Xtras to their programming environment Director, AutoCAD has an inbuilt programming editor and now developers such as Native Instruments are making programs that allow people to visually construct their own instruments without having to know how to program. That is the commercial and conservative side of things. Other networks, companies or institutions are writing software as open source, where people can get at the source code of the programs, and write their own versions of them. Netscape has released the source code of their Navigator browser and the operating system Linux is the product of a network of programmers working for free, all helping to extend and refine the initial code made by Linus Torvalds. Other programs such as Max/Msp made at Ircam in Paris, Supercollider by James Mc Cartney, Pure Data by Miller Puckette or Imag/ine and Keystroke made at Steim are all programs that have flexibility as their goal, allowing the user to build up their own work processes, instruments, sequencers, data protocols and other things that one might want to build in one's own style. Artists are now experimenting with collaborating in creations of programs that communicate via the Net with the aim to create music, video art or collective social spaces.

One of the strongest aspects of what the new media have brought into our culture is the manifestation of a certain poststructuralist ideas of the network, the hypertext and the quotational nature of our discourses.[45] The science of complexity and computer science have put meat on the bones of poststructuralist thinking and today we find in our society an increased awareness of networks and structures, of non-linearity and connectivity. The "web" has become the leading metaphor of the zeitgeist in the early 21st century (in whatever context the concept is used), just as the "tree", or hierarchy was the main metaphor in the 20th. In the computer arts, whether it is net art, generative art or software art,

[45] I am not going into reciting the various ideas and theories of post-struturalism here, but I am referring to the works of Michel Foucault, Gilles Deleuze, Roland Barthes, and Jacques Derrida amongst others. George P. Landow has written about this in his work on hypertext (Landow, 1992), where he is engaged in rather superficial comparisons between computer technology and networks and post-structuralist theory.

we find this idea strongly represented. We are dealing with webs and nets that are not all-encompassing structures but consisting of associations and strategic coalitions of nodes, modules and circuits. The web is never permanent or static but a dynamic, ever-shifting interaction of local nets. The architecture of networks and webs is bottom-up rather than top-down, parallel rather than serial. We have a rhizomatic, lateral organisation rather than branches and roots in a hierarchical organisation. As they are dynamic and non-linear, these networks are always incomplete, in a state of permanent emergence. The web is a super-complex structure with open gaps that result in factors which are unforeseen and incalculable. This metaphor of the web is not a construction of a discourse, but something we find in our culture wherever we look, whether it is telecommunication technology, bioscience, physics or the arts.

In the next chapter I will analyse some works that I define as processor art and use some of the ideas in my analysis that we have encountered in this chapter. The journey in this chapter through the philosophy of technology was important because I find it impossible to relate to computer arts from a solely art historic or aesthetic perspectives. Artists working in the field of computer arts are equally inspired by science and technology as from the world of art itself.

4. Processor Art

> When I started working with computers I had to learn to visualize what was going on in the machine. Once I had learned how to 'see' the environment of the operating system I had no problem navigating that space. Curators and critics that look at net art have to go through this process as well, but they may not realise it.
>
> Mark Napier (2000) interview.

The computer is a meta-machine, a machine in which one can build other machines that are able to change their own structure during execution. It is a direct result of a centuries old discipline, logic, which we can also refer to as meta-thought, i.e. the thought about thinking. The two forms of processor art I will analyse in this thesis – generative art and software art – are in their own way meta-disciplines as well: generative art is the artwork that contains all possible outcomes of that artwork within its code, and software art might be referred to as a meta-artwork, an artwork for creating artworks. That might be seen as a rather narrow definition of software art, as software art also includes the deconstruction of commercial software, writing plug-ins or additions to it, or writing artistic (some call it "irrational")[46] software that is not aimed for the commercial market at all. It is without any restraints that one defines these practises as art: the history of Western art since Duchamp's conceptual works in the 1920s, has been very concerned with the dissolution of the concept of art. Especially in the 1960s, with the works of Joseph Beuys, Robert Rauschenberg, Oulipo or the Situationist International, the concept has come to include abstract ideas, formal rules or social processes. In this chapter we will have a look at works that could also be defined as scientific research, technological experiments, software development, hacking, virus creation, or instrument production. However, all the creators of the works I will analyse define their activities as art in various senses, whether it is personal expression through code or non-subjective experimentations with technology and/or culture. In this chapter I will illustrate shortly the aesthetic principles I think we can detect in processor art works, show how they are partly derived from issues discussed in the second and third chapter, and then introduce the works of individual artists working with generative and software art.

First I have to make a note about my use of the concepts

[46] read_me festival: http://www.macros-center.ru/read_me/about-en.html.

generative art and *software art*, because both activities are arguably much older than the computer itself.[47] As I mentioned before, these words are crude ways of categorizing a wide field of activities that overlap and go much further beyond the labels they are given. Much generative art is software and many software artists are using generative methods in their code. These concepts have popped up at the surface of the art world due to the increasing amount of artists that have begun working in the field. Thus it has become necessary to categorize the activities, mainly for practical reasons such as when journalists are writing about the works or when art institutions are funding big and expensive projects. And finally, the reason for coining the term processor art is to limit the research of the current thesis to art that is necessarily processed by the computer in real-time to “become alive”. So we have various conceptual sets that interact and below I will write about some chosen processor art works under the – sometimes misleading – categories generative art and software art.

4.1. Aesthetic Principles and Notions

There is no such thing as beautiful code, only fans of clear thinking. Golan Levin, interview.

There are certain metaphors, ideas, knowledge and technological understanding that make up for much of the aesthetic characterising processor art works. I am not saying that we’ve got an artistic movement with coherent aesthetic or agenda such as we might find in movements like Fluxus or Situationist International, but rather cognitive sets which these artist-programmers participate in when creating their works. Processor art is a manifold flora of aesthetic ideas and artistic attitudes, but the reason one can find a certain unity or approach in their work is due to the actual experience

[47] Florian Cramer discusses in his article “Concepts, Notations, Software Art” (Ward, 2002, p. 106) how we might define software as sets of formal instructions or algorithms and not necessarily something unique in the world of computers. Cramer shows how the Dadaists used “software” to create some of their poems and he wrote a Perl program that does the same. He then states: “If software is generally defined as executable formal instructions, logical scores, then the concept of software is by no means limited to formal instructions for computers. The first, English-language notation of the Dadaist poem qualifies as software just as much as the three notations in the Perl programming language. The instructions only have to meet the requirement of being executable by a human being as well as by a machine. A piano score, even a 19th century one, is software when its instruction code can be executed by a human pianist as well as on a player piano.”

and study most of the people have had to go through to be able to produce their works. To learn programming properly one has to work with computers (and their faulty mechanisms) for long hours, study programming languages that are object orientated which opens up for the view of creating individual objects (or "beings") that have their own properties and "lifelike" behaviours. It involves a high degree of awareness of the qualities of the medium, and unlike painting or classical music – as the field is young and every experiment is new without a predecessor or history – the experimentation involves creating new methods of working with the medium. The experiment might be a failure, but out of the failure (or the accident) might spring something unexpected – which leads to new ideas and/or new approaches.

In this section I will try to extract a few tendencies I see in processor art that characterises the aesthetic the artists are working with. I mention them here briefly for a clear overview, but then I will discuss them in greater detail below: *First*, there is the biological metaphor derived from biology and the science of artificial intelligence, artificial life and the science of complexity. Under this category I will also put cybernetics and the network theories of both social and computer science. *Second*, we have the non-referential nature of many processor art works, where artists are not working with representations of actual physical objects, but rather creating the objects from pure code, i.e. simulacra in the highest sense. *Third*, I find that the computer as a medium greatly influences the aesthetic outcome of processor art. I will call this the machine aesthetic, where the good and bad qualities of the machine itself influence the visual or sonic material. This is not out of necessity but rather due to a fascination with the history and low-tech aspects of computing – as when artists work deliberately with very low pixel resolution in their images (called "pixelism"). *Fourth*, there is the aesthetic of the accident. When artist-programmers are testing out a code or the functionality of some software, the result is very often unexpected, but enjoyable nonetheless. The accidental has become both a valid work process and aesthetic (in the sense of strange and unforeseeable results). *Fifth*, the aesthetic of cause. It involves the way artist-programmers set up the interactive modes in which they want to engage the user. They typically try to awake his or her curiosity by setting up strange cause and effect relationships, which makes the system look autonomous and with free will. *Finally*, and here we are talking about a temporary fact of computing, we have the cerebral nature of computer art which influences strongly the character of computer created works. The works are produced in a "mind-machine" relationship where the artist's body is not involved in the same degree as in the traditional arts of, say, painting, music or dance. This is a very important factor and I will argue that this disembodied method of creating art results in a more "emotionally detached" works than works produced where the body is involved. I do not find that a negative fact, but a characteristic one of the computer arts for the time being, and something that will definitively change in the future when technology evolves.

The first aesthetic category is the one using metaphors from biological science, artificial life and network theories. A good example is the artist Philip Galanter who has made both software and generative art where he works with sound and three-dimensional animations. He teaches a course at the New York University on the "Foundations of Generative Art Systems" and his artistic and theoretical work has had a strong influence within the genre. Galanter is involved in studying various algorithms and methods such as genetic algorithms, Markov chains, L-systems, tiling and symmetry, cellular automata, fractals, artificial life and chaos science in his work.[48] As one can see from the examples on his website, Galanter's artistic imagination is not that of representing nature or deconstructing older artistic directions, but rather generating works in the computer that get inspiration from the natural world, and then creating autonomous life within the virtual space of the computer. Karl Sims is another artist who has created systems where the user controls the evolution of three-dimensional beings that are evolving on the screen. His project *Galapagos* has been exhibited widely as an interactive media installation of evolving

[48] An *algorithm* is basically a rule-based method to solve a problem in finite amount of steps. *Markov chain* is a probability system in which the likelihood of a future event is determined by the state of one or more events in the immediate past. An *L-system* is a grammar based system for describing and generating branching structures that (often) exhibit self similarity. *Cellular automata* are systems of individual objects that display a certain behaviour. Each object (it could be tone, colour pixel, or vector in 3D space) has an intelligence that affects the other objects, but all the objects have the same properties. Conway's Game of Life is a good example of cellular automata. *Fractals* are structures that exhibit self similarity at all scales. One can zoom infinitely into degrees of the fractal, but the same structure is emerging in nano as in macro.

virtual “organisms”.[49] Most of the concepts that were introduced in the 3rd chapter like genotype/phenotype, emergent behaviour, mutation, evolution, simulation and autopoiesis are relevant when analysing their works and those of the other artists in this chapter. This aesthetic of complexity and unpredictability of the generative processes in the computer makes us frequently see the computer as the source of autonomous life, where life emerges from some untouchable code and evolves in directions not foreseen even by its creator. In the last decades we have witnessed the science fiction hype of cyberspace and ideas about intelligent life that mutates from our uncontrollable inventions, where the computer becomes non-human and the *other* of our being. In relation to processor art, this is of course a false conception: everything programmed in the computer and the computer itself are made by humans and it is as representative of human invention and expression as a sculpture or a painting.

Secondly, in addition to these concepts taken from the science of artificial life, complexity and chaos theory, we find other aesthetic attitudes that are more related to the nature of the new media and the way information is dealt with when it becomes digital and distributed through computer networks. When one experiences works of art created in the digital media one becomes aware of its non-referential nature: the objects we see or the sounds we hear are often a simulation of reality and not a representation of it as in the older art of painting, photography or film. When the computer is applied in generative art, we find simulations being generated that take their form and appearance from algorithms that might make them appear differently every time. There is not even a unified description behind the art work, it gets its life from various instructions that change it every time it is run from genotype code to phenotype manifestation. Again, this is related to the aesthetic concentration on process as a value in itself and the awareness that the computer is a virtual space where objects, relations and beings can emerge and take on themselves an autonomous life.

The third aesthetic notion is that of the machine aesthetic. It has been part of the hacker culture since the early 1980’s where hackers were creating works on their personal computers[50]. The

[49] http://www.genarts.com/galapagos/index.html
[50] Levi (2001).

machine is not only seen as intelligent tool but a co-player in the creation of the works. But this is a machine that is fragile, constantly crashing and sensitive to viruses and system errors. It is probably the faultiest medium ever created.[51] The computer crashes, misinterprets file formats, disconnects from networks, does not read old files, cannot find external devices or the drivers are old and insufficient to latest software updates, etc. It makes us feel that we are not in total control over the tool we are working with, and we can never take network connections for granted. Artists have been working with this insecure relationship to the machine in many ways, for example Alex McLean's Perl software that won the Transmediale_02 in the software art category.[52] The program – called *forkbomb* – is seven lines of code that reproduces other programs that also reproduce until the memory of the computer is full and it crashes. A simple, innocent and beautiful virus. Other artists, such as the net.artist team Jodi, work with the noise/signal distinction and try to make the viewer realise the fragile state of the technology by making browsers crash, texts move or become unreadable, changing images into such low resolution that they become noise and so forth. The artist-programmer Netochka Nezvanova takes this further and plays with the paranoia and the anonymity of the postmodern information society. The fact that we are living in an age where the amount of

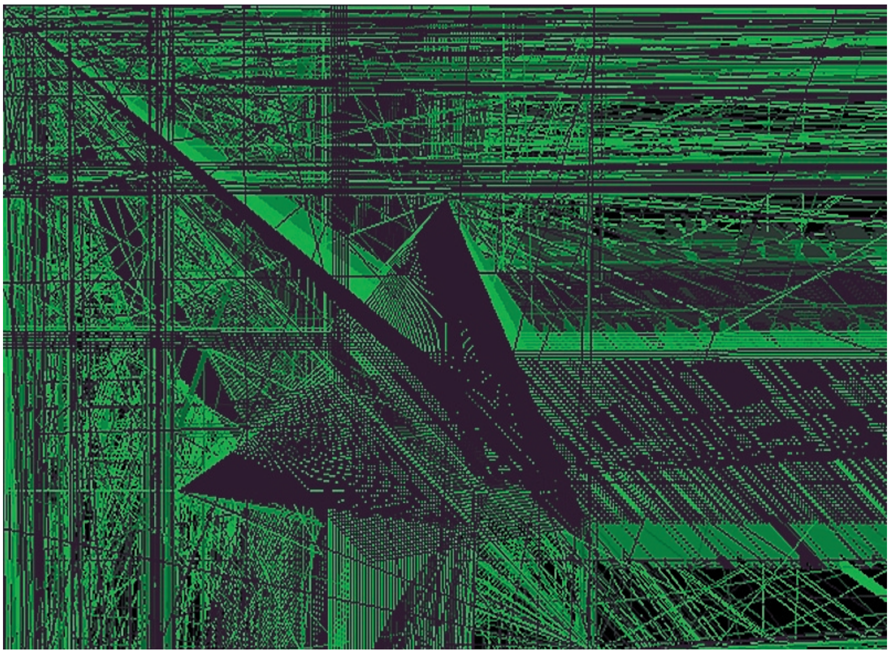


Figure 6. An image created by Netochka Nezvanova using her Nebula_m81+0.2 browser.

[51] Could you imagine buying a television that gives you as many problems as your computer makes you suffer?

[52] Alex McLean: http://www.slub.org and Transmediale: http://www.transmediale.de/en/02/

information we are surrounded with has become noise on its own, and we have to make an effort to tune in to the signals we want to hear, is the basis of the aesthetic agenda in her work. The world of marketing, media, advertising, branding, entertainment industry and relentless consumerism has resulted in a noise society that has become chaotic and irrational if one does not set up one's filters for survival. Nezvanova is the creator of various software and webpages that play with the aesthetic of noise, paranoia, signal distortion and discourse that is at the same time distorted speech and distorted message. Her messages are coded in machine-speak, her webpages crash or take control over browsers and she manipulates the system of the user computer. It is the apparent irrationality and noise element of the machine and communication that attracts her. This aesthetic can be seen as that of noise and accident where data becomes subjected to systems that distort, fragment and compile it into new and unpredictable results.

The fourth category, that of the aesthetic of the accident, is a very general one and not to be found only in the destructive works of Nezvanova and likes. It is a general knowledge of everybody that use computers in their work that accidents often create results that are more interesting than what was intended. The accidents have become accepted as creative input from the unstable media. Paul Virilio has commented on this tendency in our technology: "The accident is the original sin of the technical object. Every technical object contains its own negativity. It is impossible to invent a pure, innocent object, just as there is no innocent human being."[53] And N. Katharine Hayles has written about how much of scientific knowledge has been discovered by accidents in the research labs and further: "The 'accidental' is not so much a fixed category as the boundary between the known and the unknown, the expected and the unexpected; the "accidental" happens where waves break on the beach of knowledge."[54] John Cage is one of the first artists working consciously with the accidental and experimenting with it in all levels of his art. His way of using random factors to determine works, or allowing for chaotic elements from "outside" the work to be included in its final outcome, has influenced younger generations of artists and we sense his influence strongly in processor art. This is clearly visible in the works of artists using

[53] Broeckmann (1998) p. 34
[54] ibid. p. 210

evolutionary concepts, where mutations (a genetic jump in the form of a change in the DNA of the being) take place because of accidents that affect the being. The accidental becomes interesting in itself as it triggers the evolutionary changes in the objects.

The fifth important factor in the processor art aesthetic is the aesthetic of cause. The computer allows for an interactivity hitherto unprecedented where the artist can engage the user in various ways to experience and/or evolve the artwork. Although interactivity is the characteristic of the computer arts which is most commonly examined, it is not the focus of this essay as that has already been done in countless books and articles.[55] There is nonetheless a feature of interactivity that becomes important in processor art works and that is the aesthetic of cause. It activates the curiosity of the user, his or her engagement and interest in helping the work to develop through process orientated activity. What I have in mind – apart from the obvious fact that if the work is interesting, the user will always check what lies behind a link or a node – are possibilities of programming hidden, subtle or strange cause-effect relationships in the work. An easy example would be the possibility to detect the mouse movements of the user and initiate some processes from the user behaviour. A visual object on the screen might move to the right if the user moves the mouse to the left. A mouse-click might result in direct feedback of sound, for example, or latent feedback in form of some other process. One can play with the interactivity and create tension, expectation and curiosity in the user. This intensifies the idea that the computer is a non-deterministic and non-linear intelligent system with an autonomous will. It could be that the programmer is "bluffing" by using those methods, but it could also be that the system is truly intelligent and responds through artificial life-programmed instructions. In any case it activates the user, and illustrates well what I have in mind with the aesthetic of cause.

Finally, the aesthetic of the cause in addition to the aesthetic of accident, which the computer media has injected so much life in, is a direct result of another and very important factor which is *how* people work with the computer. When artists and programmers work

[55] See for example: Bell (2000); Bolter (1999); Castells (1996); Davis (1995); Dinkla (2000); Drotner (1999); Druckery (1999); Goldberg (2000); Huhtamo (1996); Jensen (1998); Lunenfeld (1999); Manovich (2001); Mayer (1999); Moser (1996); Murray (1997); Popper (1993); Rush (1999); Van Dijk (1999); Weibel (2001); and Wilson (2002).

on the computer it is a highly "cerebral" activity that does not require much bodily control. In the 3rd chapter I introduced N. Katharine Hayles' and Paul Connerton's distinction between the *inscribing* and *incorporating* practises, which runs parallel to the distinction between the theoretical and social construction of the abstract body on the one hand and actual embodiment on the other. Programming or making art with computers is a very cerebral activity. The knowledge (apart from training in typing and using the mouse) is that of abstract acts: symbolised acts where the user is using metaphorical graphical user interfaces or working with programming functions and keywords that are also symbols of the much lower-level language of the machine code. Nothing is direct when working on the computer. Artists working in both analogue and digital fields feel this difference, although they may not be aware of it all the time. Let us take a drummer for example. The drummer has been practising his skill for years and he knows his set of drums like his own mind. He has encoded the skill into his bodily memory by repeated performances until it becomes habitual. Every movement can be done automatically as it is the body of the drummer that does the drumming. The drummer can then take conscious decision to break up the beat, either with planned action or some unconscious action that the body performs out of its bodily memory. In a sense, every rhythmic possibility is incorporated into the body of the drummer, and artistic skill is defined as how much has been incorporated and how originally that incorporation is made use of in performance. This is an example of incorporated practise.

Now, this same drummer might have a computer music studio-setup, which he works on. But here different rules apply. If the drummer is not using a midi-drumset[56] (which very few people do due to the limitations the rubber pads have for sensing the very physicality that I am talking about), he is typically controlling drum samples from drum machines or midi tracks on software like Cubase or Logic. He paints in or plays the beats on the midi-track, programs the drum machine or uses arpeggiators or other tools which provide him with algorithms that create interesting beats. He plays the track, evaluates it and changes the things he wants. His head is doing the drumming, not his body. The musician is working in the field of

[56] A setup that looks like a normal drumset, but in stead of drums with skins there are sensitive rubber-plates that send out midi signals to the software or hardware used.

inscribed practise (as opposed to incorporated practise) where he is manipulating a system of signs that are independent of any particular manifestation. Making music on a computer is a symbolic rather than direct activity. There is little possibility for embodied, incorporated skills to act or react intuitively to what is happening in the music. What we get instead is the approach of planning, programming (the word also used when working with graphical music applications), and running the program to test how it will sound. This is partly what I mean with the aesthetic of the accident. In this type of aesthetics, we allow for much more complex and formally difficult experimentation than that which is possible when working with a band or an orchestra. And this is also what I mean by the intactile nature of the medium: the limited interfaces we have when using it and the cerebral nature of the resulting work. The advent of genres like drum'n'bass, jungle, microsound, glitch and other types of electronica would hardly be possible without the computer technology and this disembodied way of working. Included in the aesthetic of the accident, or within its workprocess, is the method of generating and then evaluating the outcome (some call it GAE).

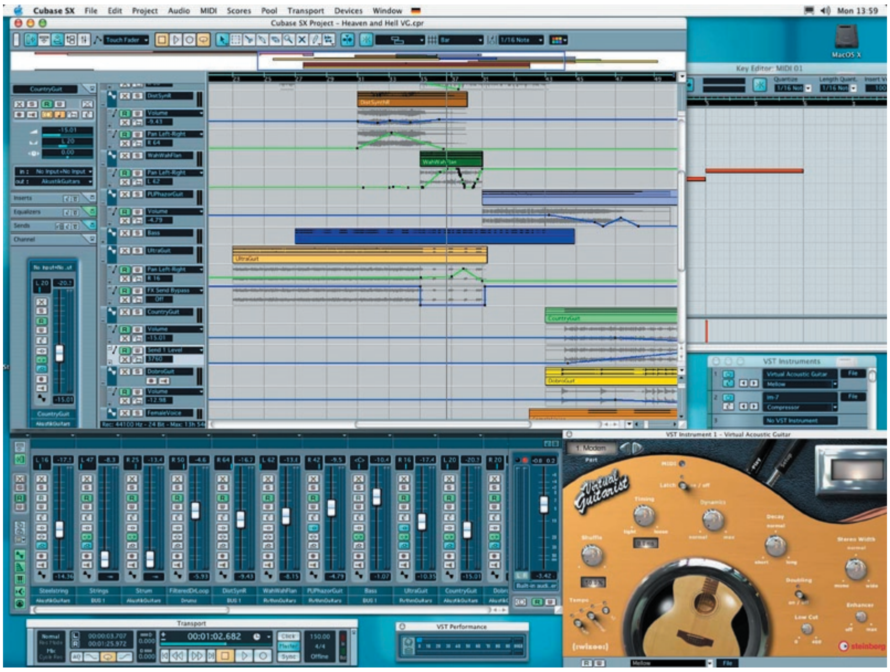

Cubase SX File Edit Project Audio MIDI Scores Pool Transport Devices Window
Mon 13:59
Cubase SX Project - Heaven and Hell VG.cpr
Virtual Guitarist
steinberg

Figure 7. A screenshot of Cubase where audio and midi tracks are arranged linearly, just as in a musical score. At bottom right there is a VST instrument controlled by midi.

The computer is a detached instrument for creating art, but that does not mean a bad instrument. It is just that our communication with the system is through very limited and crude devices like keyboard, mouse, midi instruments, insensitive sensors (the amount of information to detract from sensors like video or haptic sensor is nothing compared to the human eye or skin) are not *yet* capable of allowing for the same degree of control that physical instruments provide. Imagine a guitar player trained in classical music. He might have a very precious instrument from the best instrument maker in the field. His relationship with the instrument is unique and when he plays the guitar, he knows incredibly well the location of the bands, the thickness of the neck, the way the strings vibrate and the smell of its wood. This guitar player will feel alienated and play differently when using another guitar. He will even play differently if he is not using his favourite set of strings. All this subtle information which is involved in tactile instruments has no equivalent in the computer. One might argue that working with different operating systems, programming languages or software, matters in a similar way when we are working on the computer, but it is not the same. We are talking about bodily communication with a physical instrument, and not a cerebral communication with abstract system.[57]

The argument above is the case for the year 2002. However with the evolution in the computer industry where experiments with sensors and intelligent systems are performed in research laboratories all over the world, the future will provide us with controllers and sensors for the computer – and more intelligent and sophisticated software – where we will communicate with the computer in an even stronger and more interactive way than with our physical instruments. What I have in mind here is to some degree being studied and experimented with in the works of e.g., the Sensorband, Pamela Z or SensorChip[58] where its members have invented sophisticated controls that send information to software like Max/MSP and Supercollider in which they have written programs

[57] The analysis above runs parallel to the distinction Jean Francois Lyotard makes between feelings and "intensities" and which Frederic Jameson has elaborated on in his essay on postmodernism. For Jameson, we have lost the capability of expressing true feelings in art (as irony kills such things) and today we have rather got intensities that "are now free-floating and impersonal and tend to be dominated by peculiar kind of euphoria…" (Jameson: 1991, p. 16). For me, it seems that it is easier for a musician to express him/herself emotionally on a physical instrument like trumpet (say Miles Davis in the early 1970's) than using computers for creating music (say Autechre running their generative code). Again, there is no value judgement included in this argument, just an observation on the different modes on how the instruments and tools function at the present time.

[58] http://www.sensorband.com, http://www.pamelaz.com, http://www.pamelaz.com/sensorchip.html.

that interpret the input from the controls into the desired outcome. There are academic conferences being held about the subject of new interfaces for computer music[59] and I suspect that in a decade or so, we will have instruments that are extremely sensitive for the bodily control of the performer, instruments that could even have intelligence on their own, so they can interpret the user movements and respond differently if the context is different. Such is not the case (and never will be) with physical instruments.

The ability to write formal instructions for the computer to perform; to create life that appears conscious and that evolves; to simulate without reference sounds, images and three-dimensional objects, has given us a medium that provides extraordinary possibilities for artists and programmers to work with. This quality of the computer plus the fragile nature of its system and the way it communicates through limited protocols and file formats, the "mystical" aspect of its workings (its machinery is absolutely hidden from us and understanding it is only for the few) and the estranged and disembodied way in which we communicate with the system has resulted in a set of aesthetic concepts and working methods that are characteristic of today's computer art. Below I will discuss some examples of processor art that I have chosen, with the aim of taking as a broad scope as possible within the limits of generative and software art, which I find characteristic of the aesthetic notions I have already discussed.

[59] See NIME 2002. (New Interfaces for Musical Expression) http://seamonkey.mle.ie/nime

4.2. Generative art

4.2.1. Morpheus

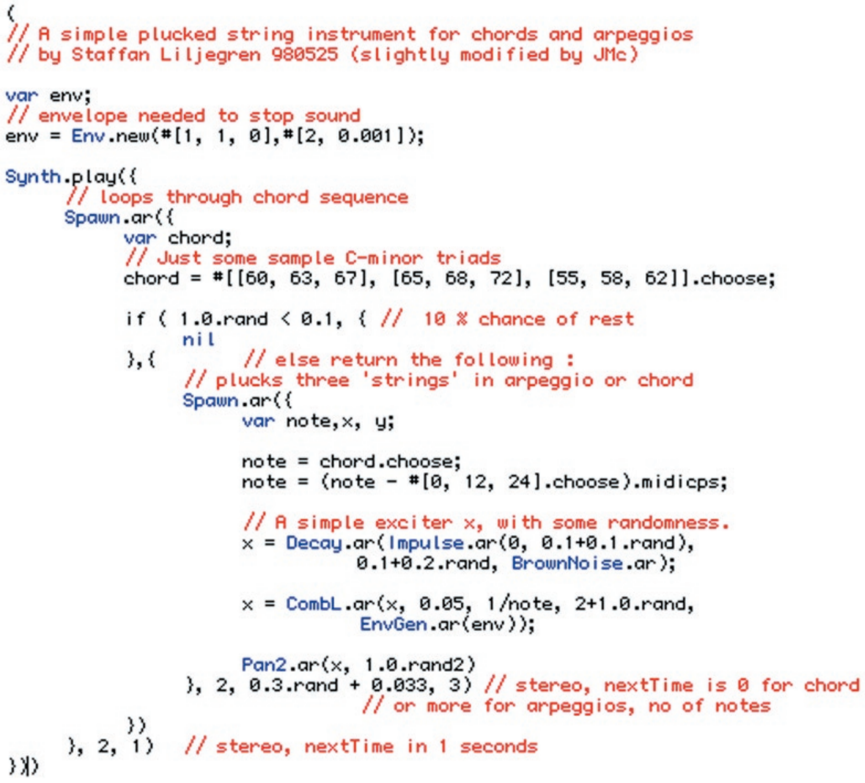
> Generative art refers to any art practice where the artist creates a process, such as a set of natural language rules, a computer program, a machine, or other procedural invention, which is then set into motion with some degree of autonomy contributing to or resulting in a completed work of art.
>
> Philip Galanter (webpage)

The idea of Morpheus was initiated by John Eacott, a musician and lecturer at Westminster University in London. His aim was to create a CD of generative or algorithmic dance music, which would be in constant flux, and never the same when listened to. All the works should be composed in Supercollider, be beat based and have a time range from 3 to 8 minutes. The decision was taken not to allow for graphical user interfaces (GUI) or user interactivity in the works. One might ask oneself: why not? The computer is after all an interactive medium and by allowing the user to control some of the musical variables, the musician might be working in greater accordance with the medium's "nature". But after listening to the CD a few times this question withers away. The fact is that interactivity is a special phenomenon in arts which has its own rules and logic. With interactivity one is breaking up the immediacy of experience and making the user suddenly aware of the medium itself.[60] Thus, in Morpheus, one is listening to the music in passive mode (rather than the active mode of interactive works) and enjoying the variability and surprise that the music constantly provides. As music is a time-based art form it is important for its function not to break up or pause unless the conscious decision of the composer is to do so. Of course, one could also imagine a work along the lines of Morpheus, where the user would be able to interact with painterly visual interfaces or GUI style knobs, buttons and sliders, but that is a very different listener experience. In fact, here we have addressed one of the biggest problems of interactive music or interactive narrative today: how to create interactive structures that engage the user and provide coherent plot that functions in all its possible versions.[61] In

[60] Jay David Bolter and Richard Grusin have, in their book *Remediation: Understanding New Media*, written about the differences of the immediacy of painting, television, film, and virtual reality, in contrast to the hypermediacy of multimedia where the user is constantly aware of hyperstructures of links and nodes. See Bolter (1999) p. 33

[61] See the book *Hamlet on the Holodeck* by Janet H. Murray. She analyses the

music there has to be a certain flow, and in narrative there must be a certain plot, which both have to work with building tension, release, solutions and closure. Formal structures are difficult to work with in interactive narrative and music, and we are yet to see a science of interactive narrative studies.



```
(
// A simple plucked string instrument for chords and arpeggios
// by Staffan Liljegren 980525 (slightly modified by JMc)

var env;
// envelope needed to stop sound
env = Env.new(#[1, 1, 0],#[2, 0.001]);

Synth.play({
        // loops through chord sequence
        Spawn.ar({
                var chord;
                // Just some sample C-minor triads
                chord = #[[60, 63, 67], [65, 68, 72], [55, 58, 62]].choose;

                if ( 1.0.rand < 0.1, { //  10 % chance of rest
                        nil
                },{             // else return the following :
                        // plucks three 'strings' in arpeggio or chord
                        Spawn.ar({
                                var note,x, y;

                                note = chord.choose;
                                note = (note - #[0, 12, 24].choose).midicps;

                                // A simple exciter x, with some randomness.
                                x = Decay.ar(Impulse.ar(0, 0.1+0.1.rand),
                                        0.1+0.2.rand, BrownNoise.ar);

                                x = CombL.ar(x, 0.05, 1/note, 2+1.0.rand,
                                        EnvGen.ar(env));

                                Pan2.ar(x, 1.0.rand2)
                        }, 2, 0.3.rand + 0.033, 3) // stereo, nextTime is 0 for chord
                                        // or more for arpeggios, no of notes
                })
        }, 2, 1)   // stereo, nextTime in 1 seconds
}))
```

Figure 8. Supercollider code. This piece is generative and never plays the same track twice. The "choose" and "rand" functions make the variability in this code.

On Morpheus we find the works of six composers that work very differently with their material in terms of programming, use of algorithms, sound materials and sound synthesis. Morpheus was supposed to be rhythm based, and all of the tracks are working with beats and rhythm structures that range from funk to drum'n'bass and breakbeat. This does not make it easier for the composers as writing generative music with complex rhythmical structure is arguably much more complex than writing generative ambient music in the style of Brian Eno and others. However, the composers on Morpheus manage to deal with the task and on the CD one finds many different approaches on how to compose beat based generative music. With one exception, Nick Collins, we can recognise each of the tracks again without any difficulties, enjoying how varied and different they

problems of the interactive narrative and how we in the future might be able to create narrative forms that use non-linear ways of telling a story.

all sound each time they played.[62]

The composers work dynamically with variables like tempo, pitch, timbre, tones, rhythm and modes, as Supercollider allows for each of these variables to be subjected to algorithmic variations and calculations. For example, in “Memory” Fabrice Mogini uses prerecorded soundfiles of drum beats but adds on top of them a bass line where the pitch and rhythmic structure are varied every time. He uses algorithms for creating “slides” on the bass which results in a very lively performance that passes the test of whether the computer will ever be able to “funk” (which has been a debatable issue over the years). Frederic Olafsson has decided not to use any samples in his tracks, but synthesises his sounds “on the fly” and uses them in his varied and dynamic structures. The fluidity of his music is very rich and he experiments with algorithmic control over timbre, pitch, rhythm and the macro-structure of the piece itself which results in a very strong experience of listening to a “live” piece every time. The music is powerful, and it emphasises the feeling of a wild jazz performance where the musicians are in an oblivious state as they blow away into their instruments.

All of the composers are skilled programmers and have programmed their pieces from the microstructure (synthesis for their instruments, timbre changes, overtones, overlaps etc.) to the macrostructure (the timeline, the rhythm, and the harmonic and melodic content itself). Using the genetic analogy (which I introduced in the 3rd chapter), the composers have composed the genotype of their works with a large amount of planning and testing, and we, the listeners, are experiencing the epigenesis of the works when we run the code in the form of music, the phenotype itself. All in all, listening to Morpheus is a very strange experience because of the fact that, being generative, one is listening to a unique piece of music which might never be heard again in the same way, as well as for the fact that the music is often highly engaging and attractive, although, as it says on the CD cover, “contents may vary.”

[62] Nick Collins’ work is very interesting in the way he investigates breakbeat structures and sound synthesis, and Eacott admits that he was included on the disk due to the sheer ingenuity of his work, although he did not fulfil the requirement that each song should be recognisable. At present Collins is working on genetic algorithms for creating rhythms and melodies. He uses interactive methods for guiding the evolution of the material, as at the end of the day, it is the human ear that is the best judge of what sounds good and what not. (See Eacott, 2002)

4.2.2. meta

The artist calling himself meta has hit the nail on its head with his pseudonym. As I explained above, generative arts are a type of meta-artform, where one creates a composition in code that results in visual or sonic compositions. One composes a machine for composing art, but that machine is an artwork in itself. meta's website is the ground for the artist's activities. There he archives his works on a monthly basis, on a diachronic interface where his pieces are divided into six sections: *flux*, *graphic* and *video* that deal with dynamic visual arts; and *octave*, *rhythm* and *tone* that are sonic experiments. All of meta's works are generative in their nature and he concentrates on the creation of aesthetically appealing works that evolve naturally through time. He frequently plays around with the aesthetic of cause where abstract forms follow (or escape) the mouse, sometimes directly but also in a non-linear way whose logic is harder to detect. The user is thus collaborating with the generative process in "painting" the image. meta is mainly using two programming platforms in his works, the graphical programming language of Max/MSP/Nato and the web based programming suite Director which is programmable with Lingo. Each of these platforms have their strengths and weaknesses which are detectable in the works.

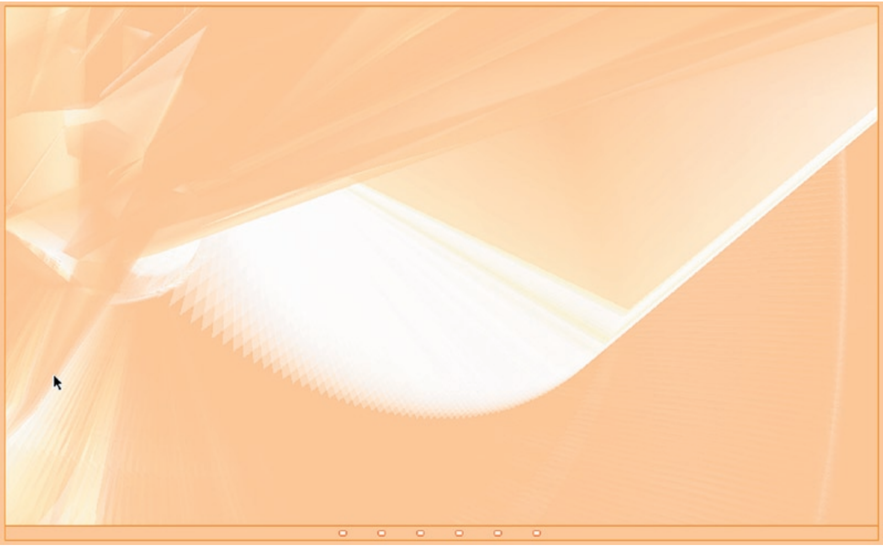


Figure 9. An image from meta's piece *sol*. It is both generative and interactive. The user can affect the process happening on the screen.

We see the smooth vector rendering of Director and the crunchy aesthetic of Nato, and it is obvious how the tools are determining the artist, although there is a personal touch running through all of meta's work. Artists using these tools have to work within their limits, but that is one of the interesting aspects of programming: exploring the limits of what can be done (or said) with a particular language, by which means and how.

In the *flux* category we find works with bitmap and vector visuals that render smoothly on the screen by following generative rules and responding to user activity in the form of mouse movements. For example, in the piece *sol*,[63] we get flickers of white forms that leave colourful trails on the screen. We are able to affect the behaviour of the flickers, but we are not in total control over them. The question arises: what is causing the changes? The user or the code? One is able to affect the work's process and play around with the visuals, thereby participating in making one's "own" version of the piece but in a strange way the work has its own life. When we are happy with the image, we could take a screenshot, manipulate it in Photoshop and print it out. Such a printout might be very beautiful and formally interesting, but it only tells the half story, because it is just a snapshot of a work that was evolving in the computer as a creative process. It is really the evolution of the work, the mutations and transformations it takes on its way to nowhere (as there is no "final outcome"), that is the way we should relate to generative works and not expect a final state or a "conclusion". The process is the conclusion.

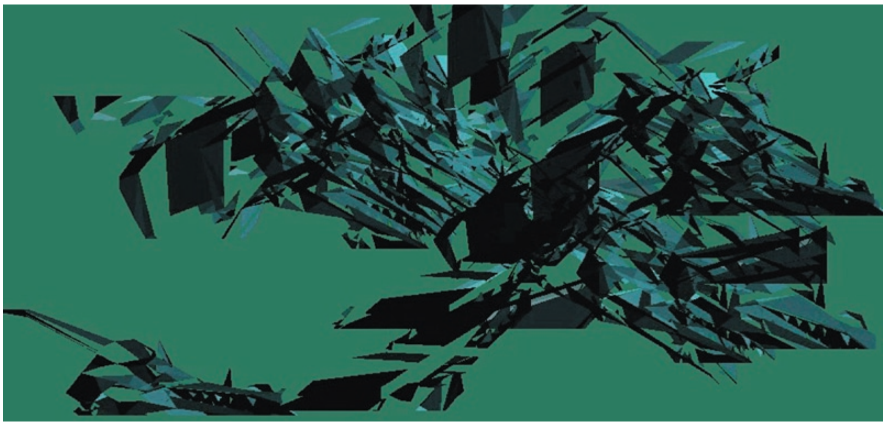


Figure 10. An image from the graphic series. As we see, this style is much rougher than the pieces made in Director.

The categories *graphic* and *video* use the Max/Msp software and the Nato+0.55 software to create generative visuals that have a unique visual style that is partly that of Nato's creator Netochka Nezvanova,[64] and partly that of the artist himself. These works are good examples of how software art can be a collaboration between

[63] http://meta.am/flux/sol/

[64] Netochka Nezvanova's work is discussed below in the software art section.

the software artist and the (here, generative) artist in creating works that are partly the software artist's and partly the artist's. In the *graphic* category, one can browse though a series of images that have been created by algorithmic methods and result in powerful visuals with a strong deconstructive style. We are in a three dimensional space where forms break up and disperse through a black empty space. Each image is a version of the genotype or source instructions.

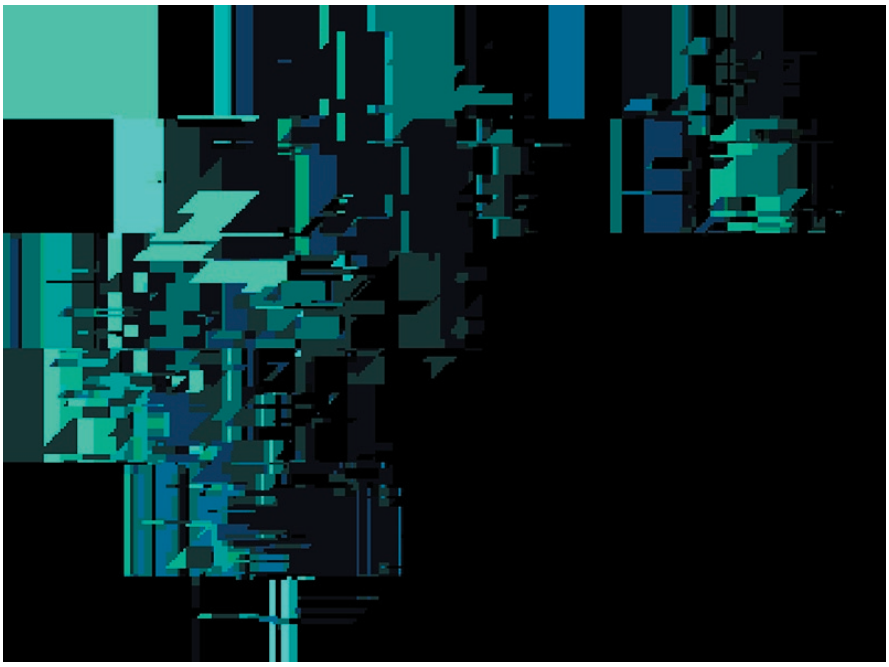


Figure 11. A still from a video piece called *roac*.Parts of it can be seen as video on the Processor Art CD-Rom.

The same can be said of the videos in which we can experience the process of how the graphical works were created. We see how the forms are moving and evolving, taking upon themselves different colours and hue. This part of meta's work, is not interactive on the website, probably because there does not exist a Nato plug-in for browsers, so he has to record the process as video before putting it on the Net. Instead of interactivity, we are passive viewers of processes that have already been created and recorded. I find the Nato work on the website not as interesting as the other pieces due to this lack of interactivity and the obvious Nato style. However meta has made his own software in Nato for image and video manipulation that allows for user interactivity. *arc* is a software that can be downloaded from meta's website and it allows the user to import his or her own Quick Time movie files into the program which then cuts them up, defragments and manipulates them in a style that is part Nato's, part meta's and part the users.

Meta's music is made with partly algorithmic and synthetic methods, but, unlike Morpheus, it is provided in the compressed mp3 format which cannot support real-time generative processes. The music is the same every time listened to. However, meta provides various versions (or phenotypes) of the results of his generative compositions. We find up to 8 different versions of the same source track as in the noisy piece *d.frag.01-08*. The first is an electronica piece with crispy and ironlike sounds, fast rhythm and interventions of softer and smoother background textures. The next ones take different directions in aggressiveness, smoothness, or degree of deconstruction of the source. In other works such as *chaia* we find one source track *chaia* and it's two genetically related siblings *chaia.glt* and *chaia.grl.* I don't know what the subfixes mean in the latter two, but I suspect it has to do with the method they were generated. Again, the aesthetic is that of the relationship between noise and signal. Between meaninglessness and meaning. This style of signal awareness or machine aesthetic has become very strong in the last decade or so, but it has its roots in the cybernetic theories of the 1950s and 1960s where researchers in cybernetic theories were concentrating very much on the meaning of noise and signal. In the mid 1990s people like Jodi, e13, Vuk Cosic and Netochka Nezvanova have all experimented with this thrash aesthetic of the accident, of noise, and distorted signal. The political message is that of the manipulation of the media, the noise and junk that is added to raw material and how the world becomes falsified through the use of media. We are encountered with a strong awareness of the instability of our media and the way we receive information in the post-industrial information society[65].

[65] The nettime mailinglist which I have mentioned before in this thesis is the best reference for this critical awareness of the media, in particular the new media. Since 1995, the list has been a forum for discussions of media critics, artists, sociologists and philosophers amongst others about the influence of the media on our culture and individual psychology. (http://www.nettime.org)

4.2.3. Jem Finer

The former guitarist and co-founder of *The Pogues,* Jem Finer, has been actively creating algorithmic computer music for years. One of his latest – and probably biggest – projects is the *Longplayer: A Thousand-Year Musical Composition*, which started on January 1st 2000 and will play until its completion in December 31st, year 2999. The composition has listening posts in various places over the world, but the permanent one is at the Trinity Buoy Wharf Lighthouse in London, plus being broadcasted through the Net every day, every hour. Longplayer is an algorithmic piece of music written in Supercollider but uses sound samples of Tibetan Singing Bowls which create a pleasurable and meditative soundscape. Finer uses synthesis methods to manipulate the samples, by altering their pitch and tempo or changing the actual wave information of the samples. The music never repeats itself for the 1000 years it will be playing and the composer's idea is that this piece will play continuously unless something drastic happens in the world. Finer is playing with the idea of time, new technology and the transformation of culture. How are people going to relate to an old Apple computer with an archaic sound synthesis programming environment called Supercollider kept in an old lighthouse in the Docklands after 100 years? Not to mention

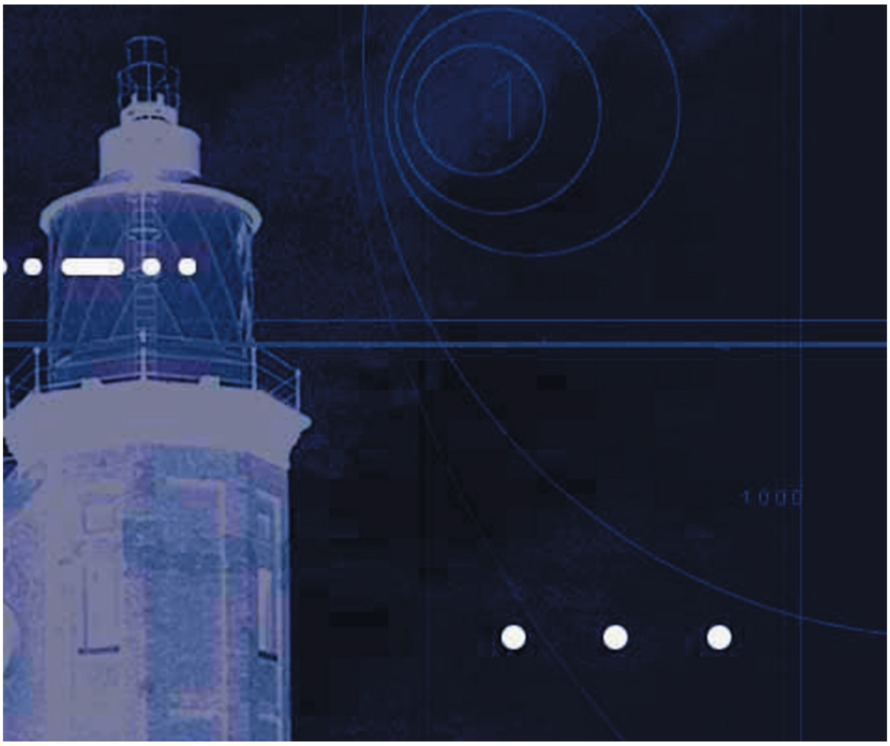


Figure 12. The Trinity Buoy Wharf Lighthouse in London. This is the physical location of the computer that will play Finer's long piece for the next ten centuries.

900 years. It is an interesting question and Finer has established a fund and a committee to take care of the project in the future.

On his forthcoming album “gtr” Finer works with Supercollider code to write generative music programs that manipulate his guitar playing. The work is released on the “dead” medium of a normal compact disk, so we don’t experience the excitement or the “liveliness” of the generative code, but that is not the intention of the composer here. We hear a combination of code composition and guitar improvisation, where the code is the score that interprets the sounds and works with them in various ways from macrostructuring their combinations to microstructuring their sound properties. As Supercollider is a real-time sound synthesis language, it is able to do this within the computer, such that one experiences almost the feeling of a live performance. The music is a strange combination of a guitarist in a meditative state playing his guitar and a programmed system that works on his playing and responds in unforeseen ways. One senses the cybernetics between the performer and the composed system, and it becomes hard to figure out what is being played live and what is being processed within the code. The question could arise: who is the composer here? Which has a simple answer: Finer composed both the code and the music, and the result is a general composition in different levels. One is reminded of Gordon Mumma’s and David Tudor’s statements that making an instrument is a composition in itself.

Finer has played live using his Supercollider patches and connecting his guitars into the computer. The result is a different experience than that of Morpheus because of the physicality of the guitar. He uses the guitar as sound source in his work and that creates a different connotations and experience from the non-referential, synthetic sounds of many of the Morpheus tracks. In a conversation Finer told me that he had become tired of playing his “one finger music” – i.e. when laptop musicians are using the trackpad of the laptop as the only interface to their musical performance – and he desired an interface that was more tactile and instrumental. Thus Supercollider became a way for him to write interesting musical structures but then use the guitar (which arguably has much more evolved interface due to its being developed for hundreds of years) as an interface to his music. The inscribing and incorporating practises (which I talked about in the 3rd chapter and further at the start of this one) are used equally here. The code is written beforehand, tested and refined, but the playing is done spontaneously with the bodily knowledge acquired through years of instrumental practise. Finer says about his technique: “I use both methods of practises. The code is in a sense a “score”. Composing becomes an interaction between what the programs do to what I play, on the interaction between the computer and my playing. They evolve side by side and are continually being adjusted, re-written, added to. I like the tactile nature of a guitar say. I can play it, my fingers have a mind of their own . . . the computer is not tactile at all but I can still “play” it too . . . interface becomes important . . . I use the guitar as the interface a lot, as well as a tone generator etc.”[66]

In Finer’s work we see how the computer is taken as a co-player or a collaborator; an intelligent, autonomous system that has been taught how to respond to certain inputs by calculating, synthesising and finally playing sounds that become audible music related to the musician-programmers input.

[66] From a private conversation/interview.

4.2.4. Celestino Soddu

The architect Celestino Soddu, professor at Milan University, has been involved with creating many interesting generative systems for various purposes. He has written programs like *Basilica*, which creates endless sequences of architecture, and *Argenia*, which is a design machine that makes generative design of industrial objects. Since 1987, Soddu has been experimenting with the creation of systems that can be used to generate unique objects, whether art, city planning, architecture, industrial design objects or graphic design. He works in an opposition to the ideas of modernist industrialisation and mass-production, which we find for example in the writings of the Bauhaus school. For Soddu, designed objects can be unique in form and appearance and it is not necessarily impractical to think in these terms:

> *After two hundred years of the old industrial era of necessarily cloned objects, the one-of-a-kind object becomes an essential answer to the long-neglected human need to live [in a world where] each artificial object mirrors the uniqueness and unrepeatability of every person. In an epoch marked by repeated attempts at the cloning of natural beings, design returns in advanced technological fields such as non-linear dynamic systems to the notions of artificial life and artificial intelligence, the aesthetic and ethical pleasure of rediscovering the processes and character[istics] of nature.*[67]

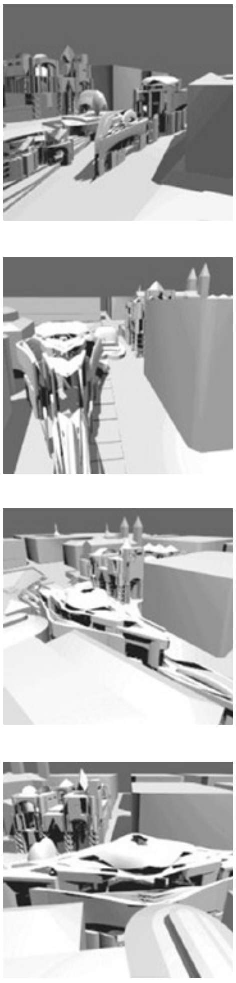
This view is one of the central ideas behind the yearly Generative Art conference in Milan[68] (which Soddu is one of the organisers of) where many interesting projects and papers have been exhibited and published over the last years. One of the concerns behind the conference is to explore how the computer opens up the rediscovery of possible fields of human creativity that would be unthinkable before. If it seemed, at the beginning of the computer era that the new tools were diminishing human creativity, today's technology does the opposite: it opens up new fields and enhances our understanding of creativity as an indissoluble synthesis between art and science. Many of the participants of the conference have been mentioned in this thesis and one can say that we are witnessing an ever increasing group

[67] Generative Art" webarticle:
http://www.celestinosoddu.com/design/GA_soddu_e.htm

[68] http://www.generativeart.com

of academics and artists who believe in the powers of generative design and its modes of human-machine collaboration. The machine is helpful here as it can do things that we otherwise would find hard to perform, but it also points to aspects of the design process that we are not accustomed to contemplate.

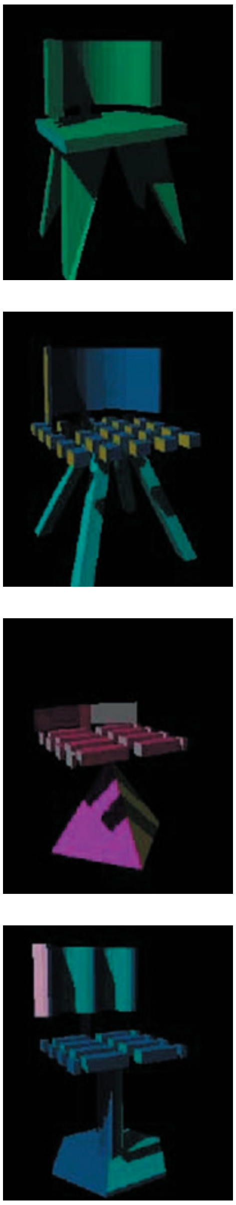
Basilica, which Soddu made in 1990, is a generative program that allows for the generation of endless sequences of architecture, which all have different phenotypes but belong to the same genotype. Soddu sees this as “design of species” or “design of morphogenesis” and bases that on a presumed homology between the natural and the artificial, and their belonging to the same world of chaotic systems. The designer can control the evolutionary procedures of the software and define the complexity of possible manifestations. Thus one is able to tell the system how different each phenotype can vary from the other phenotypes, thereby establishing the subjective identity of every one of them. Soddu has used Basilica to design city planning and architecture, for example the enlargement of the Prado museum in Madrid. He used Basilica to generate various different 3D scenarios as a projection of a single composition idea. For Soddu, the multiple shapes are not degrading creativity, on the contrary, they show the possible representations of the architect’s idea, exploring both its potential and applicability. In Soddu’s work we experience strongly the idea of the rhizomatic structure of creative thought processes. He is involved in creating elegant genotypes (in the form of software code) that produce endless variations of his artistic or architectural ideas. Soddu is aware of the fact that for architects and designers, decision making is often a limiting practise; they burn bridges behind them by choosing one path to work with rather than another, but in Basilica, the fast applicability of computer code means the designer can explore the potential of the ideas in a much wider sense.



Figures 13-16.
A city planning project made with Soddu’s software

Argenia is another of Soddu’s projects. It is a program that sets up the structure needed for the generative design of any number of industrial objects. The designer’s idea is coded into the program, which in turn produces a series of different and unique objects, which still are recognised as belonging to the same idea. Just as family resemblance works within families where each individual bears the traces of being genetically derived from the DNAs of his or her parents. Soddu is working on the (seemingly paradoxical) concept of unique mass produced objects. He points out that before the industrial era, each object was unique, unrepeatable and strongly connected to the identity of the craftsman that created the object. The industrialisation changed all that, and the object became a multiple unindentified; one object of many, who all are equal due to the process of optimisation and repetition.

The modernist belief in the pragmatics of the mass produced object – that serially produced objects are cheaper in production, that the optimisation of function brings forth a necessary identification of a “unique” design, and that the great designer only comes up with one “unique” design, and not many versions – has lost its force in the post-industrial society where new technologies have been invented which challenge the industrial mode of operating. Not only has our technology changed, but many of the things we create with it are of different nature than the material objects created with modernist technology.[69] The expensive thing today is the actual designing of the systems that are used to produce our objects, but once the systems are set up – computers, specialised software, tools to produce material objects – it does not matter at all whether each copy is identical to the other or not.[70]



Figures 17-20. Here Argenia is used to generate various forms of industrial design based on one idea. A chair in this case.

[69] The logo on the Rhizome art-database (www.rhizome.org) is a good example. It is a generative logo that appears differently every time the user logs on. And further, as Soddu points out, it costs the same today to print ten different pages and one page ten times in the advanced print technology today. Such was not the case in the modernist printing technology.

[70] Soddu designed the cover of the book following the first Generative Arts conference in hundreds of different versions. No copy was the same, but they were all similar as the program created the covers from a singular design idea or genotype.

4.3. Software Art

4.3.1. Signwave

> Software not as a functional tool on which the "real" artwork is based, but software code as the material of artistic creation. Software Art can be the result of an autonomous creative practice, but can also refer critically to the general technological and social meaning of software.
>
> - Jury statement, Transmediale 01.

The software artist Adrian Ward is the man behind the software company Signwave. He writes various types of software, but is most famous for two in particular where he parodies the popular Adobe software packages Photoshop and Illustrator in applications that he calls respectively Autoshop and Auto-Illustrator. Both programs are functional and cleverly designed software that is to be used for bitmap and vector graphic creations. The interface and its structural and visual metaphors (icons) are imitating that of Adobe. However, the artist sees them as "personal expression" of his own character and he has written functions in the program where the program takes control of the user and does things that one either cannot or can hardly control. "I definitively treat auto-illustrator as though it were me. Designers who are using my code are collaborating with me in the construction of vector designs."[71] Thus, Ward is calling into question the authorship of the creations made with his software and in fact most software in general. All graphic software has abilities and limits that often lead to a certain "style" of visual design but in the case of Auto-Illustrator and Autoshop, the style is made as a conscious decision of the programmer, i.e. Ward himself. The Signwave software is not trying to be as general as possible (such as Photoshop aims to be), but rather limited, characteristic and personal expressions of the programmer-artist's worldview. This is one of the characteristics of artistic software. Artists are not making the software to please the general user or the typical buyer of software, but are rather making applications with functionality that *they* want to see in the software they use. Paradoxically though, Signwave presents itself as a multinational commercial company with innumerable employees (when users of the software write to a support mailing list run by Signwave, they get answers from people whose names have been generated by a Perl script written by Adrian Ward himself), and the whole structure and product design parodies

[71] Levin et al (2001) p. 257

that of Adobe and other software developers. Auto-Illustrator can be ordered on Signwave's website and comes shipped in a box with the CD of the software and manuals and books about the project. But Signwave experiments with a variety of things in its productions such as these clauses in the user agreement of Auto-Illustrator:

> *6. PERMISSIVE CLAUSES*
>
> *By using this software you agree to permit us to get your computer to do whatever we wish it to do. At our discretion, our software will commit your computer in whatever way we desire without your knowledge nor permission. You agree to be held responsible for all actions your computer takes, even when those actions are 1) those of, or 2) the result of, our software.*
>
> *7. ARTISTIC CLAUSES*
>
> *You realise that your use of this software constitutes a component of a distributed, realtime and non-realtime interactive and autonomous artwork between you, your computer, our software, and us (the artists).*[72]

One can hardly imagine anything like this being written by commercial companies. Ward is deconstructing everything that one could relate to traditional manners of selling and marketing software. Auto-Illustrator is supposed to inspire or frustrate the user, make him aware of the machine, the software and the functions of software that trained designers have become oblivious to, due to familiarity and with natural the use of hardware and software. In Heideggerian terms, the technology as such has disappeared, and it is only when it misbehaves or breaks down that the technology turns "present-at-hand" and the user becomes aware of its nature and reflects on it. The Signwave software is trying to create this awareness of these tools, by providing the user with instruments which open up a questioning into the nature of the tool. It encourages the user to find new ways to do old tasks, and to become aware of the medium's structure and encourage him to write additional code to extend the application.

It is interesting to compare the efforts of the software artists of today to the video artists in the late 1960s. For the video artists, video was the "other" of television and their aim was to liberate the public from the bondage of the mindless mass-medium they found

[72] Auto-Illustrator 1.0 on-line documentation.

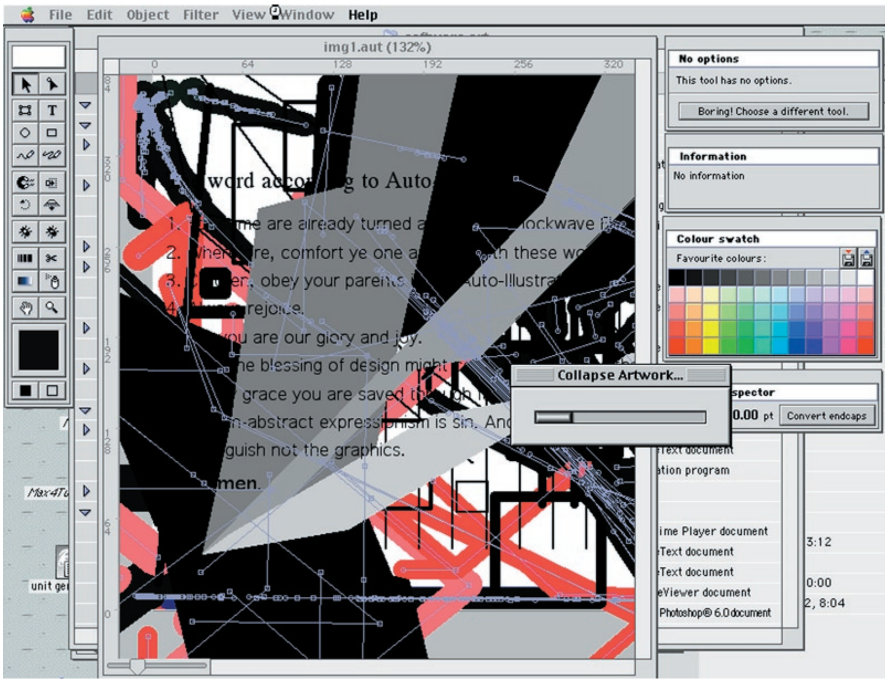


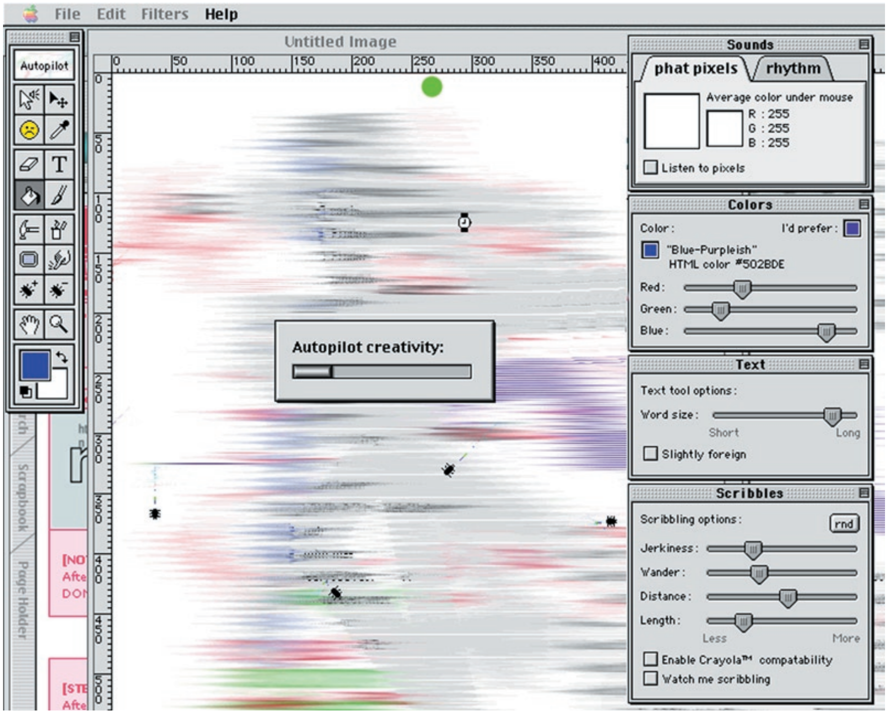
Figure 21. A screenshot of Ward's software Auto-Illustrator. It is a vector graphic software just like Adobe's Illustrator.

in television. It was supposed to be the long dreamed-of, truly democratic medium that would wake up the viewer and allow the artist to express him/herself in a way not possible in commercial television. Today, television has incorporated video art and video artists are working in collaboration with television – the recent popularity of documentaries is a good example of that. By analogy, we find a situation where commercial software companies are producing software which controls the work process of the user, unchangeable and closed, where the source code is not accessible for the general public. That is the side of the determining, established and commercial world. On the other side we find artists such as Ward who produce their "own version" of "commercial" software and others who do not even pretend to participate in the rules and methodology of the commercial world. I suspect that this is just a temporary situation as happened with video art. Two things will change in the near future: a) the open source movement[73] will change the way commercial companies work and relate to competition, and b) the companies are realising that the users want to be able to change the software, write their additions and plug-ins, and get more personalised control over the software. We will then witness a situation where the distinction between a commercial software

[73] The Open Source Movement is a loose term for the people that write open source software. The operating system Linux is a good example of such collaborative programming. Further info: http://www.linux.org/ and http://www.gnu.org/.

company and the artistic software movement is not so clear. The artists might have a day-job in the commercial companies, which in turn show considerable interest in the works of software artists. Who knows, but Adrian Ward might just as well be found at a high post in the development department at Adobe in the near future.

Ward uses various generative methods to control visual elements and the user can for example insert ants into the image he or she is working on, where the ants start to crawl over the picture leaving tracks behind them that often result in chaotic and funny visual structures. In Autoshop there is an *Autopilot creativity* that deconstructs, filters and reconstructs the image in random ways, which the user has no control over. There is a "bot" tool in the tool palette, which is a smiling face if the bot "likes" the image or angry if it "dislikes" it. When clicked on, the autopilot "fixes" the image and the bot becomes happy again. In Auto-Illustrator Ward has added functionality that allows the user to write his own plug-ins for the program. Thus the program has endless possibilities for being extended and it is only limited by the imagination of the user.



Autopilot creativity:

Average color under mouse
R : 255
G : 255
B : 255
□ Listen to pixels

Color:
"Blue-Purpleish"
HTML color #502BDE
I'd prefer:
Red:
Green:
Blue:

Text tool options:
Word size: Short Long
□ Slightly foreign

Scribbling options:
Jerkiness:
Wander:
Distance:
Length: Less More
□ Enable Crayola™ compatability
□ Watch me scribbling

Figure 22. A screenshot of Autoshop. Here we see the Autopilot "fixing" the image after his own taste.

> *… treating code as an expressive language (it is merely more syntactically strict than spoken language) one can see that human creativity can be codified in similarly dynamic results. An impulse, a desire, an emotion can be expressed using code. The code becomes an extension of the programmer, so it makes sense to treat code as*

> *an externalisation of not only your own working process but of your creativity and thus your self too.*[74]

For Ward, code is as much an artistic material as paint and canvas – he has written extensively on the creative aspect of programming and how the software artist is interacting with the user in creating the final visual work. That aspect of Ward's reasoning – that he "is" the software and there is a collaboration between him and the user in creating the work – is perhaps a little bit extreme in some ways. Where are the limits between such personal creation and a more impersonal one? Are the programmers at Adobe just writing some non-subjective code that has nothing to do with their personal worldview or experience? And if we take that argument further, would an instrument maker, say of guitars, not be collaborating with the musician in the final outcome of the music, and he being part of the creative process? The instrument maker is creating both the potential and limits of the instrument, and the characteristics that make a musical piece unique in many senses. I believe Ward is right in stressing the personal aspect of his work, but he seems to be stretching the limits of what can be counted a personal expression, when he states that *he* is always involved in the creations of people using his software.

[74] Levin et al. (2001) p. 67

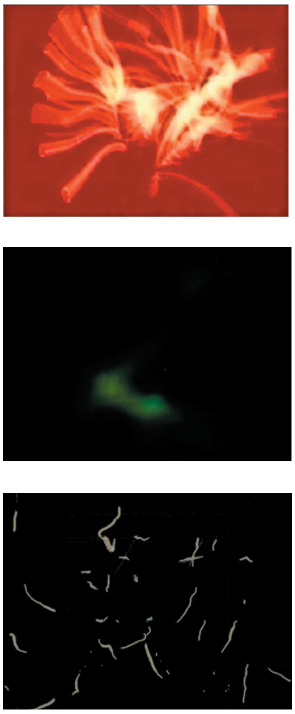
4.3.2. Aesthetics + Computation Group

The Aesthetics + Computation Group of MIT's MediaLab is directed by the programmer and designer John Maeda. The aim of the faculty is to bridge the gap between artists, designers and engineers, by providing an environment that researches the skills of each discipline simultaneously. Maeda's work is concerned with rethinking the way we use our media and the development of new systems for visualising and interacting with information in the hypermediate structure of the digital age. At the MediaLab, the computer is not seen as a box connected to a screen, mouse and keyboard, but rather as a mouldable device which can function everywhere in our daily lives, even in our clothes. The courses Maeda has designed are concerned with the aesthetics of the computation media – how one can design organic lifelike forms in the computer, plus trying to make it an ubiquitous tool in our environment. One has just to take a quick look at the books Maeda has published[75] to see how both his conceptual ideas and visual approach have had a strong influence on the people working at the MediaLab, and this is not meant in a negative way. Quite the contrary, Maeda has experimented ingeniously with the use of animation, time, colours, structures, memory, and interactivity to such a degree that his influence reaches far beyond the experimental, academic world of the MediaLab. Maeda's enthusiasm for introducing programming to designers and artists made him write a program called "Design by Numbers" which he documented and elaborated on with the publication of a book with the same name.[76] The program is basically an empty canvas and a script window where the user programs with a customised Java-style programming language and experiments with forms, colours, animation and interactivity. Another such program, *proce55ing*, created to teach the skill of programming is currently being developed at the MediaLab by one of Maeda's students, Benjamin Fry. In this section I will talk about the works of two artists from the lab that I find interesting: Golan Levin and Benjamin Fry.

Golan Levin's main work concerns the relationship between visual controls and audio. He has created five different programs where the user plays music through the use of "painterly" interfaces.

[75] Maeda (1999) and (2000).

[76] The program can be downloaded for free at: http://dbn.media.mit.edu/



Figures 23-25. Screenshots of Levin's AVES. *Floo*, *Aurora*, and *Yellowtail*, respectively.

Levin's goal was to create musical instruments on the computer that did not imitate the physical instruments we are accustomed to, but rather instruments which use the visual ability of the computer to interact with the sound world. In his thesis, *Painterly Interfaces for Audiovisual Performance,* Levin points out that he is not a pioneer in any way: there is a long tradition of trying to visualise music by different means, from Luis-Bertrand Castel's (1688-1757) *Clavecin Oculare*, through Thomas Wilfred's *Clavilux* (1919) and Oscar Fishinger's *Lumigraph*, to the experiments in the 1930s of abstract cinema.[77] What Levin does is to bring the themes of the early experimentalists into the new computational media of the digital age where things have become much more fluid and mouldable than in the analogue media of hardware and physical technology. The result is his *AudioVisual Environment Suite* (AVES)[78] where the performer controls the audio through different visual interfaces that imply new and innovative modes of interaction. Levin showed his *AVES* at the Ars Electronica festival in Linz, Austria, in September 2000, where he and two others gave a performance using the software.[79] Ars Electronica offered him the opportunity to commercialise his programs and they can now be bought as a software package from the Are Electronica store or Levin's website.

The *AVES* is an important work in the field of audiovisual experimentation and computer controlled musical instruments. The visual metaphors Levin has designed are intuitive and easily understandable and his research will be vital for future researchers in the field. An important factor in the work is that he sees the visuals as aesthetically significant in the whole experience of performing/listening-viewing the piece. They are not only "triggers" of the sounds, but also beautiful generative forms that grow and change over time. However, the backdrop of Levin's programs are that the user does not have enough control over the sound synthesis to make the software really usable for his own musical creation. The user is stuck within Levin's sound world, which might become limiting after a while. An important addition to the software would be to open up the sound control and make it controllable by the user, whether

[77] Levin (2000) pp. 21-32

[78] You can learn more about the programs on the Processor Art CD-Rom

[79] Levin continues to perform on his instrument, lately at Sonar 2002 in Barcelona.

that would be through graphical user interfaces such as sliders and buttons or programming panel in the style of Supercollider or Csound, or even better by graphical programming interfaces such as Pure Data or Max/MSP. By allowing for user control of sound synthesis and importation of sound samples into the software, its qualities as general artistic software would become much stronger and not remembered only as the personal instruments of a single (ingenious) musician-programmer.

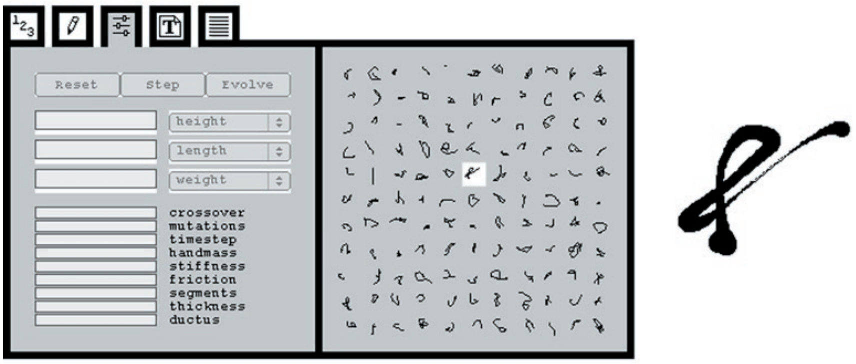


Figure 26. *The Alphabet Synthesis Machine*. The user generates a new typography that he or she can then use in their text editor.

The most recent works of Levin have become more conceptual, where he investigates many aspects of our culture and arts. The *Alphabet Synthesis Machine* is a on-line Java applet that allows the user to create and evolve a typographic font by providing drawing palette, and various functions to control the shape, plus providing genetic algorithm to evolve the population of the fonts. When the user is happy with his abstract alphabet, he can name the font, sign it and submit to an online gallery that stores all the hitherto created fonts. He is then able to download the font in the format of TrueType to be used on his own machine. An easy and enjoyable process which makes one think about the generative aspect of fontography, and of the history of writing and design. The author writes:

Somewhere between the chaos of television static, and the order of the text you are now reading, lies a fascinating realm of semi-sense. By attending to this narrow union of nonsense and sublimity, we propose that we may come to a deeper understanding of how sense-making occurs at all, and become connected through abstract forms to a reality beyond language.[80]

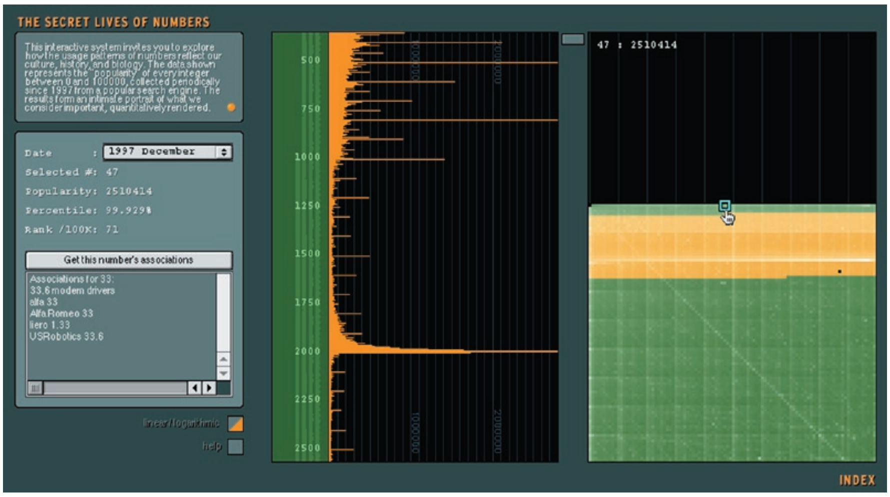
[80] Levin, Golan. http://alphabet.tmema.org/entry.html

Again we see the fascination with the chaotic, the unreadable, and the distorted signal. In this application Levin is more concentrating on what could be called traditional beauty, i.e. letters that have formal coherence and remind us of Chinese calligraphy. But I find his fascination with the meaningless discourse or semi-sense stemming from the same awareness as that of Nezvanova, Jodi and others.

In the work *Dialtones: A Telesymphony*, Levin plays with the disastrous feeling one gets when a mobile phone rings in the middle of musical performance. He wrote a system where he can call the mobile phone of every single person in the audience from a control booth on stage. The exact location and tone of each audience member can be known in advance and thus *Dialtones* is able to present a diverse range of unpredecented sonic phenomena and musically interesting structures. It sounds strange, but in fact the videoclip one can watch on the website (from Ars Electronica 2001) shows a very humorous and beautiful performance which engages the audience in a novel way. The work starts with simple ringtones, which are repeatedly triggered, it then builds up and become more tense, with all kinds of rings being heard. Some of them we know: the typical Nokia one, the Mozart one, but then we hear ringtones that the owners have downloaded from the Net or composed themselves. After a while the soundtexture becomes richer when Levin plays rhythmic patterns under the audience's ringtones. I hope that those sounds are sampled and manipulated sounds from the audience in the performance and not some precomposed track meant to "save him out" in case the performance became boring. Which it hardly becomes, but the latter would destroy the conceptual beauty of the piece. Levin explains his work and the idea behind it thus:

> *Announcers at every modern-day concert command us to turn off our cell phones, but what Cagean aesthetic possibilities might we discover in leaving them on? What deranged beauty might we find, or what might we learn about our interconnected selves, in their high, pure tones? The mobile phone's speakers and ringers make it a performance instrument. The buttons make it a keyboard and remote control. Its programmable rings make it a portable synthesizer. Yet, although no sacred space has remained unsullied by the interruptions of mobile phone ringtones, there is no sacred space, either, which has been specifically devoted to their free expression. In the context of this lack, and in the context of our society's contradictory attitudes towards wireless communication technologies,* Dialtones *is proposed.*[81]

[81] Levin, Golan. http://www.flong.com/telesymphony/



THE SECRET LIVES OF NUMBERS

This interactive system invites you to explore how the usage patterns of numbers reflect our culture, history, and biology. The data shown represents the "popularity" of every integer between 0 and 100000, collected periodically since 1997 from a popular search engine. The results form an intimate portrait of what we consider important, quantitatively rendered.

Date : 1997 December

Selected #: 47

Popularity: 2510414

Percentile: 99.929%

Rank /100K: 71

Get this number's associations

Associations for 33:
33.6 modem drivers
alfa 33
Alfa Romeo 33
liero 1.33
USRobotics 33.6

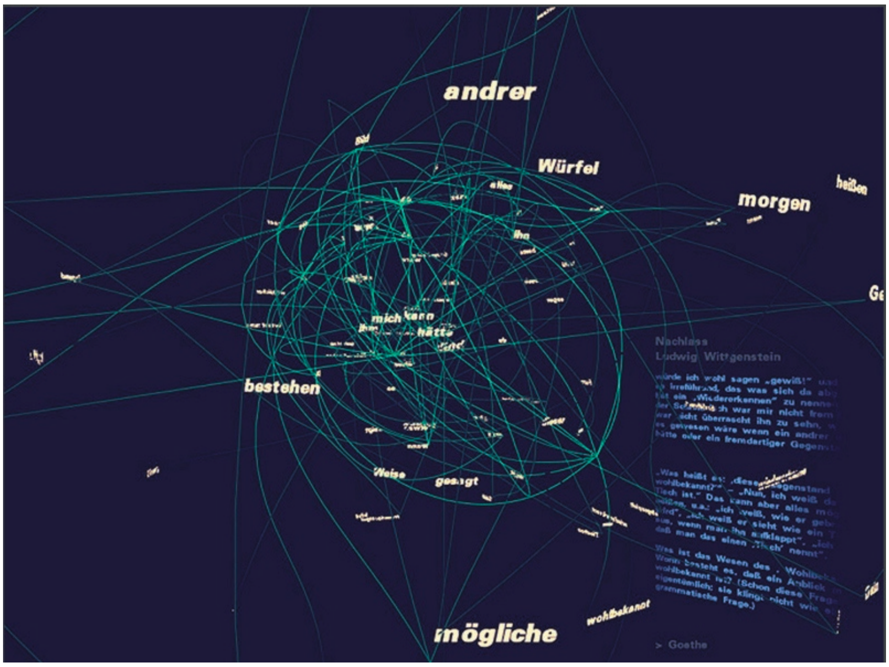
Figure 27. *The Secret Life of Numbers*. The numbers of our culture are represented in a fluid and brilliant interface.

Levin's latest work – *The Secret Life of Numbers* – is more conceptual than his other works, although it is still visually appealing. He and his team conducted an exhaustive empirical study to determine the relative popularity of every number between zero and one million. Each number is related to dates, companies, historical events or whatever might be appropriate. This information is then displayed in visual patterns on the interface. The user can enter a number to which the program reacts by displaying relations that number has and it shows the popularity of that number in relation to the numbers that are near to it along with graphical information tables. I don't know if there is some mystical numerology involved, but the authors claim that "our dataset is a numerical snapshot of the collective consciousness; herein we return our analyses to the public in the form of an interactive visualization, whose aim is to provoke awareness of one's own numeric manifestations."[82] This obsession with numbers is of course very understandable when we think of the fact that for a software artist, numbers are the primary materials with which he or she will use in their artistic creations. Numbers are the stuff that are stored in variables, sent through mathematical functions and used in algebraic procedures. They are the very life of every piece of processor art. But apart from the fascination with numbers, the piece is strangely useless, although a very nice exercise in how to build fluid interfaces that react dynamically to data that the user provides.

Benjamin Fry is another programmer-artist who has been working on various interesting projects at the MediaLab. His research focuses

[82] Levin, Golan. http://www.flong.com/

mainly on setting up metaphors which visualise large amounts of data from dynamic information sources. For this that task he has written various programs that interpret data and visualise them through both 2D and 3D spatial interfaces, which in some cases allows for user interaction and manipulation of the material. *Valence* is a program that reads complex data and displays it in relational manner so the user can get a different understanding of the text. It displays words in a 3 dimensional space where links appear between the connected words. The program has been exhibited at the 2001 Ars Electronica where it was used to read texts of literature and philosophy and at the Whitney Biennial 2002 where it deals with the human genome and traces the patterns that appear in the genomic structures.[83] Fry uses ideas from the world of organic life such as growth, emergence, atropy, responsiveness, homeostasis, metabolism and other to frame his research. The idea is that we already have a "natural" understanding of organic forms which we find in nature, both micro and macroscopic. We all know how trees have branches that divide and extend from the centre and out. We know how the structure of a small leaf of a tree is a highly complex and beautifully organised form of life and this basic human knowledge can be used to represent complex data such as the strings of DNA or other vast sources of information that we would otherwise not be able to grasp conceptually.



würde ich wohl sagen „gewiß!" und
irreführend, das was sich da ab
ist ein „Wiedererkennen" zu nenne
der Sc... ich war mir nicht frem
war nicht überrascht ihn zu sehn,
es gewesen wäre wenn ein andrer
hätte oder ein fremdartiger Gegens

„Was heißt es: ‚dieser Gegenstand
wohlbekannt?'" – „Nun, ich weiß
Tisch ist." Das kann aber alles
heißen, u.a.: „ich weiß, wie er
wird", „ich weiß er sieht wie ein
aus, wenn man ihn aufklappt", „ich
daß man das einen ‚Tisch' nennt".

Was ist das Wesen des ‚Wohlbek
Worin besteht es, daß ein Anblick
wohlbekannt ist? (Schon diese Frage
eigentümlich: sie klingt nicht wie
grammatische Frage.)

Figure 28. From Fry's Valence. Here we see a text by Wittgenstein being analysed and re-contextualised.

[83] See the Whitney website: http://artport.whitney.org/

Fry's main work is concerned with the visual mapping of the human genome. He calls his project *Genomic Cartography* which is a series of studies where he makes applications that are able to visualise in various ways the unimaginable complexity that the data involves. For Fry, this is a project that deals with scientific material and data, that is represented in an aesthetical way which is practical as the representations have the ability to provide a clearer understanding. The *Genomic Cartography* has an ethical and cultural impact as well. It is representing materials that people are afraid of, it is so to say "the hidden code of the creator." Suddenly our technology is making us able to change our genetic structure and alter it in the world of vegetation and animals as well. This is a difficult ethical, religious and political issue and the answers are far from being solved. The problem is rather the lack of questioning of the issue and Fry sees his work as a part in the endeavour to bring the matter up for discussion.

Fry has also been involved with creating the programming environment *Proce55ing* which is a Java-like environment, not so different from Maeda's *Design by Numbers*. The user of the program will be able to create generative pieces of visual works that can use whatever algorithms found important and it also allows for detecting user interactivity through mouse movements and clicking. I find the Fry's work on *Proce55ing* important because by releasing such a programming environment, he is creating a platform for artists to program their own visual works that run in real-time without having to get into the laborious task of learning how to program in the complex languages with not so user-friendly compilers.

4.3.3. Netochka Nezvanova

Very few people know who is really behind the alter ego Netochka Nezvanova. She has been terrorising (or amusing, if you want) most of the mailinglists on the Net that have to do with net.art, sound and video synthesis, politic and academic debates, resulting in her being expelled from the lists or creating strong debates among its members on whether to outlaw her from the community.[84] This infamous person has been an active software artist, musician and net.artist for years, appearing under names such as “m9ndfukc”, “sw4t7abs”, “a9ff”, “f1f0”, “integer” and most recently “NN”. The name Netochka Nezvanova is a pseudonym borrowed from the main character of Fyodor Dostoevski’s first novel; it translates loosely as “nameless nobody.” She has created an incredible amount of webpages in her noise aesthetic which many people find confusing, misleading and chaotic. Some of the pages make the browser and even the computer itself crash. Netochka Nezvanova is a very disturbing “person” which is now acknowledged to be an international network of artists and very good programmers.

I am not interested in the “social” side of Netochka Nezvanova here. One could write another and bigger thesis just on her Net activities and self-promotion (of her cyberself) by various means, but that is outside the scope of this thesis. “She” has, however, written a piece of ingenious and very useful software – Nato.0+55

[84] A good description by Andreas Broeckman about how she destroyed the Syndicate mailinglist can be found here: http://amsterdam.nettime.org/Lists-Archives/nettime-l-0111/msg00077.html. Her style can be experienced in a chat with the Art & Gender Theory Class of the Central Washington University. The chat is hilariously funny and here one can get the idea of Nezvanova’s ideology. At least one aspect of that complex personality’s worldview. The students ask questions about her sexuality, feminist views and so forth, but she answers in short and vague sentences that often are just misleading or changing the subject matter. In the end (when the students have probably had enough) we read: “Netochka Nezvanova > where are the students now? they got bored or what? lunch time?” http://www.eusocial.com/nnnnnnnn/hou.imagination.bekomes.matter/ And just to give a flavour of her very computer synthesised way of talking, using a mixture of English, German, French, Russian and programming languages, this is from a discussion on a mailinglist:
ou!. cezt posz!bl. en fakt ___*... je sur.*
ma!z je pas dez!re.
and
w!l not go `nutz`
!n dze process
or
du = dze kreatur 4rom 1 odzr velt.

– which is a set of over 130 object modules to be used with Max/MSP. Nato.0+55 is a video package that is used within Max such that artists and programmers can now manipulate video with the sensors and algorithms of Max. It has been used to compose images and video or used as VJ (Video Jockey) system which interprets chosen information of the sound (for example frequency, note, volume or timbre) and displays it as effecting the video stream. The software has a certain aesthetic to it, and most people using it seem to subscribe to Nezvanova's signal/noise aesthetic where distortion becomes as interesting as proper filters. meta's examples of video and images which I talked about above are typical examples of this aesthetic.

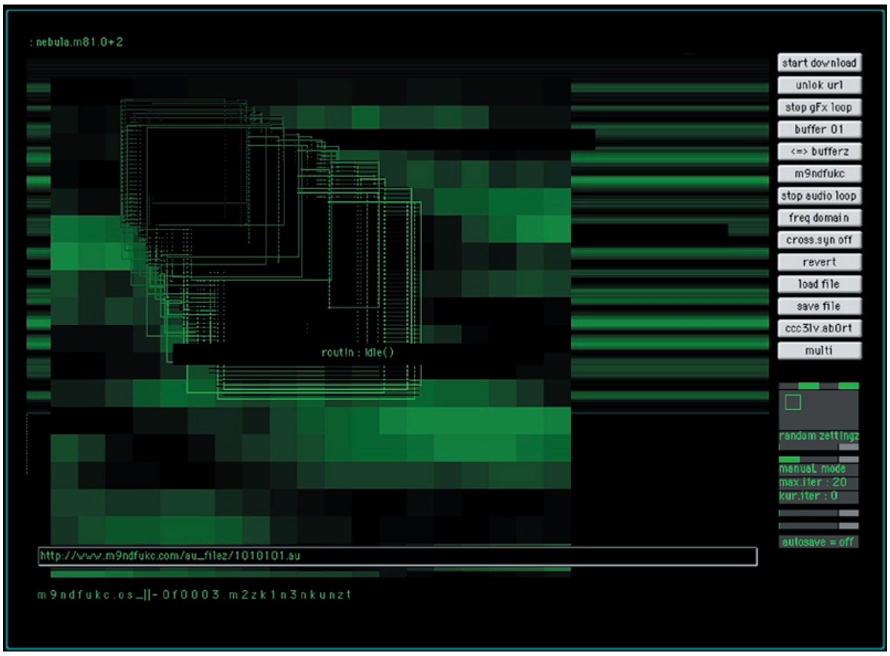
Many interesting artists apart from Meta are using the Nato.0+55 objects to create their works and one of them is Johnny DeKam.[85] He has made various programs using Nezvanova's code, from an online exhibition and a mixing console of images from the American Memory Digital Archives[86] to video mixers where users can manipulate and mix their own video files. The online piece is called *Revision History* and is basically a browser one downloads (the browser is made with Nato.0+55) and when it is run, images from the American Memory Digital Archives load up in the browser, in three different displays, two small and one big where the smaller images mix and combine. The user can manipulate the images and work with their visual qualities or just lie back and watch images from American history load up, one by one. It is a very engaging experience and the images themselves – some of them are very old – flourish very well in this minimalistic browser environment, free of advertisements, logos, and buttons.

Nezvanova has also written her own browser, or rather: an interpreter of the data one can find on the Net. Nebula_m81+0.2 is a system that interprets data from a website the user types in and in return one receives both visual and audio interpretation of the data. The aesthetic is again that of noise, old computer technology, ASCII letters[87] and crashing operating systems. The data transformations

[85] DeKam's website is: http://www.node.net

[86] The American Library of Congress has put up this website to allow people to get into it's archives. They express their aim thus: "American Memory is the online resource compiled by the Library of Congress National Digital Library Program. With the participation of other libraries and archives, the program provides a gateway to rich primary source materials relating to the history and culture of the United States.

[87] ASCII is an abbreviation of American Standard Code for Information In-



manual mode
max.iter : 20
kur.iter : 0
autosave = off

http://www.m9ndfukc.com/au_filez/1010101.au

Figure 29. The *Nebula_m81+0.2* browser that creates images and sounds from websites on the Internet.

that happen in Nebula_m81+0.2 can result in strong visual forms made of raw pixellated lines or even plain ASCII that animate through the screen accompanied with sounds synthesised from the same data. When the data is translated to sound, it happens through a sound synthesis of frequency plus cross and granular synthesis. Using the program, one can fiddle around with various parameters that affect the visual and audio outcome of the program and when happy with the result the work can be saved onto the harddisk. Nebula_m81+0.2 won the Transmediale_01 software art category along with Adrian Ward's Auto-Illustrator. For the jury the program represented the *noise* side of software art, where the Auto-Illustrator was on the *signal* side.

terchange. The code was originally developed for teleprinters and was only later adopted for computers in the 1960s. Some net.artists have been using this obsolete media format in their works and the best example is Vuk Cosic, http://www.ljudmila.org/~vuk/ where one encounters various forms of ASCII art.

4.3.4. Browser artists

The net artist Mark Napier has written various software that – like Nebula_m81+0.2 – interprets data from the Net and deconstructs it visually and textually. On his website *Potatoland* one can try out many of the programs and participate in multiuser environments where you write, submit url's or throw digital junk into a space where other people can then experience it. In 1998 he released *Shredder* which is an "art browser" (a browser is a good example of software that can become subject to artistic experimentation) that interprets the html code of a webpage and displays it in a new and totally different way where texts and images are taken out of context and reconstructed into a collage of information. Napier is trying to make the user aware of the code behind the information that he or she is so accustomed to see in a nicely coded html with tables, layers and embedded objects like images, Flash or Shockwave. The result can be a beautiful piece where code, text and images collide in a random collage, but the experience of using it weakens after a while. There is no point in loading up site after site to experience its deconstruction. It is interesting for the first time, but after a while the whole thing becomes pointless. *Riot* is a similar deconstructive browser to Shredder, but here it is a multiuser environment where the links submitted by the last users mix with the links you submit. The browser defragments the html code but the hyperlinks are still active and within the browser one can browse the Internet in a new way.

The program Napier calls *Landfill* is a little bit more exiting. It is basically a junkyard of digital trash where the user can pour his disposable, digital junk into a space where it mixes with other people's throwaways. Napier's point is that digital material is not a limited resource. One can fill *Landfill* with all kinds of materials from the hard drive, throw it into the collective junkyard, but it is still there on the computer. Digital information does not degrade by copying it or transferring it between computer terminals. *Landfill* is a social space where people have been submitting images, letters, essays, Flash animations or Shockwave games. And the materials mix with materials from other people. The program is playful and enjoyable which takes digital material out of its natural context on the hard drive of the user and mixes it in a collage with other people's texts, images or memories. It fulfils the needs of some people to expose their private life publicly and other people's need to get involved in the private life of others.

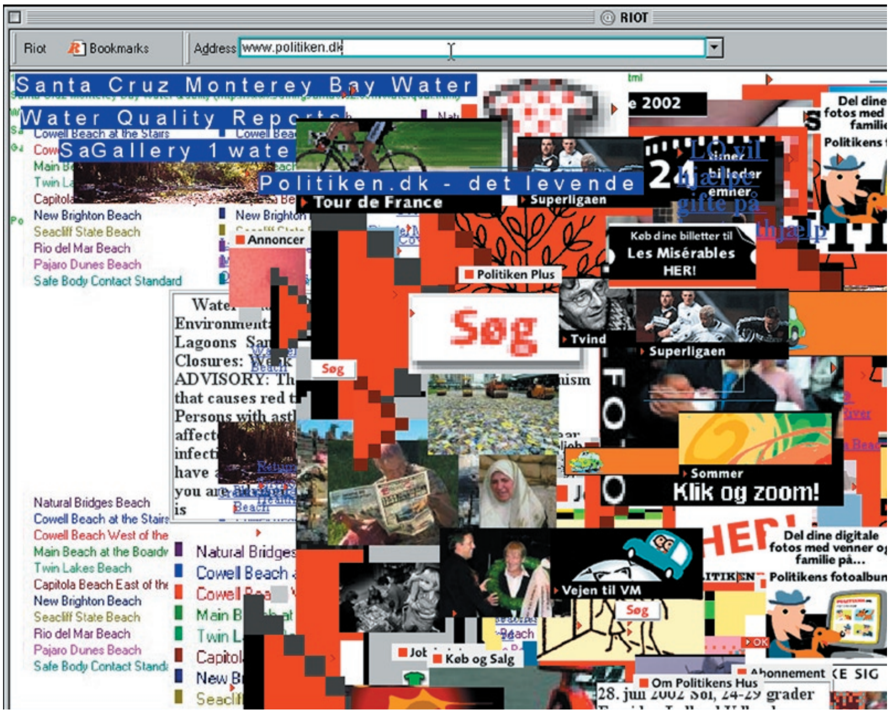


Figure 30. Napier's multiuser browser, *Riot*. People type in a URL and the browser deconstructs the website and mixes with images from other websites.

The reason I am mentioning Napier here is that he has become one of the most respected artists on the Net, probably because he started to manipulate the underlying codes of html and display information in non-conventional ways so early on. In certain circles of net.artists – related to the early nettime mailinglist – it was almost a prerequisite for net.art to use the qualities of the net (which was basically the primitive and totally uninteresting but easily learned html code) in the works. A work would have much higher "aesthetical" value if it used the connectivity of the net to draw information and variables from many different sources on computer terminals all over the world. But the work found on the *Potatoland* website is interesting conceptually, it has no lasting qualities: the visual side of the browsers are homogeneous and produce very limited variety, the deconstruction is always the same (if only he would use some exciting algorithms and conceptual extensions), and the user interface is non-intuitive, badly designed and lacks any attempt at experimentation. For an artist working in this field, we could expect a little bit more of critical examination of the fact that interfaces in commercial software seem to have stagnated into a state of conventions. Golan Levin is a good example of such an artist, see for example his work *The Secret Life of Numbers*. It is this

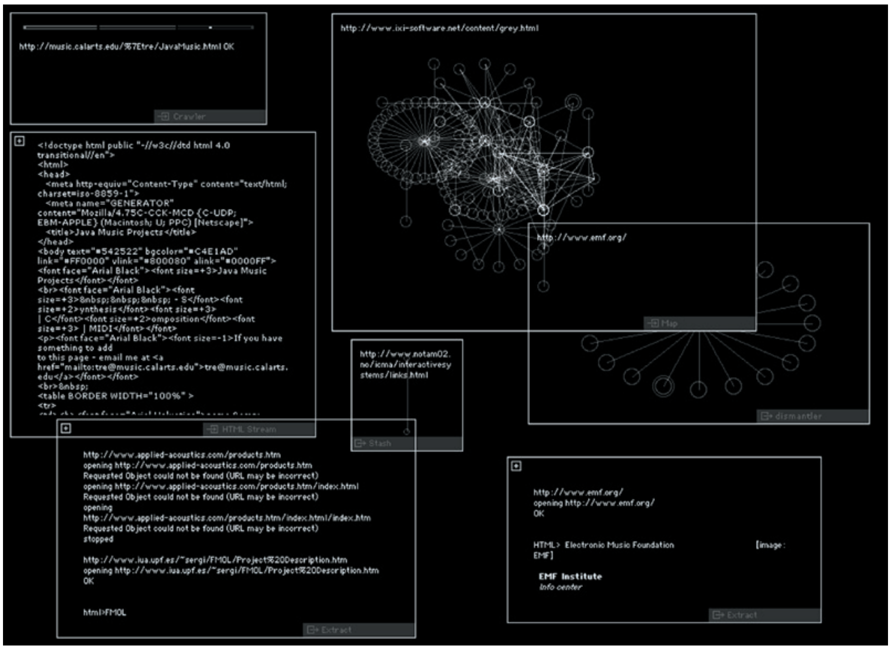


Figure 31. I/O/D’s *Webstalker*. The browser provides different windows in which you can see different information not displayed by normal brosers.

```
http://music.calarts.edu/%7Etre/JavaMusic.html OK
```

```
<!doctype html public "-//w3c//dtd html 4.0
transitional//en">
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1">
  <meta name="GENERATOR"
content="Mozilla/4.75C-CCK-MCD {C-UDP;
EBM-APPLE} (Macintosh; U; PPC) [Netscape]">
  <title>Java Music Projects</title>
</head>
<body text="#542522" bgcolor="#C4E1AD"
link="#FF0000" vlink="#800080" alink="#0000FF">
<font face="Arial Black"><font size=+3>Java Music
Projects</font></font>
<br><font face="Arial Black"><font
size=+3>&nbsp;&nbsp;&nbsp; - S</font><font
size=+2>ynthesis</font><font size=+3>
| C</font><font size=+2>omposition</font><font
size=+3> | MIDI</font></font>
<p><font face="Arial Black"><font size=-1>If you have
something to add
to this page - email me at <a
href="mailto:tre@music.calarts.edu">tre@music.calarts.
edu</a></font></font>
<br>&nbsp;
<table BORDER WIDTH="100%" >
<tr>
```

```
http://www.ixi-software.net/content/grey.html
```

```
http://www.emf.org/
```

```
http://www.notam02.
no/icma/interactivesy
stems/links.html
```

```
http://www.applied-acoustics.com/products.htm
opening http://www.applied-acoustics.com/products.htm
Requested Object could not be found (URL may be incorrect)
opening http://www.applied-acoustics.com/products.htm/index.html
Requested Object could not be found (URL may be incorrect)
opening
http://www.applied-acoustics.com/products.htm/index.html/index.htm
Requested Object could not be found (URL may be incorrect)
stopped

http://www.iua.upf.es/~sergi/FMOL/Project%20Description.htm
opening http://www.iua.upf.es/~sergi/FMOL/Project%20Description.htm
OK


html>FMOL
```

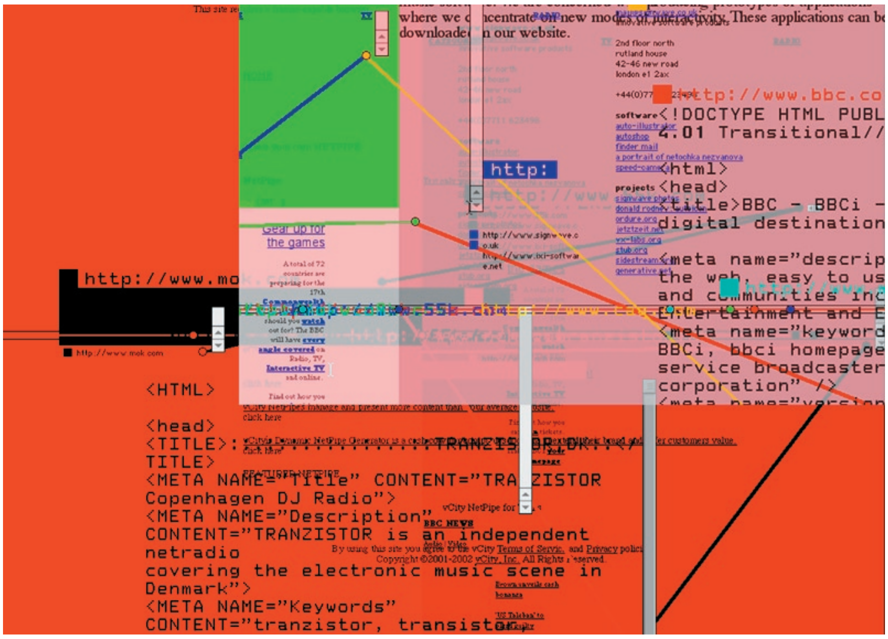
```
http://www.emf.org/
opening http://www.emf.org/
OK


HTML>  Electronic Music Foundation                    [image:
EMF]

 EMF Institute
 Info center
```

fascination with working with the networked computer, and building spaces where people can interact together online that has created the interest in Napier rather than innovative experiments with what can be done with this new meta-medium we have got in the computer and his work is lacking a deeper spirit of invention, investigation and experimentation that should be the core of artists working in the new media.

Much more interesting cases of “art browsers” are I/O/D’s *Webstalker*, Jodi’s *Wrongbrowser* and Nullpointer’s *Webtracer*. The *Webstalker* (1997) allows for connection to the internet where the program can send and receive information of all kinds. The *Webstalker* makes the user able to visualise by many means the structure of a website: which hyperlinks it has, how they are distributed and how the navigational flow of the site is built up. It visualises the structure of the website in an innovative and explanatory way where the content is not displayed. However, the user is able to retrieve the site’s information, which opens in special windows. The *Webstalker* uses the windows metaphor to be the ground for the user investigation of a website. The window can be either a display of links and nodes in the site itself, display internal links in each html document or show the actual (textual) information within the document. It is easy to use, intuitive and helps the user to understand the structure of a website in a way that is different to that found on most websites. The design is organic and fluid, objects on the interface can be moved around and customised by the user. The difference between *Webstalker* and

Napier's work is that here the user can learn something about the nature of the medium, explore it interactively in a very appealing and alternative browser interface and even read the hidden texts in the html metatags,[88] as well as the normal texts for public display.

Jodi's *Wrongbrowser* is another interesting example of a web browser. One is faced with the extreme visual anarchism which the artists have been exploring since the early days of the Internet. Pages load and disappear, planes of aggressive colours appear randomly on the screen and progression bars show some random websites being loaded up without knowing where it the information will end or whether it will be displayed at all. However, Jodi's formalistic experiments, almost modernistic in its approach, has shown us many interesting things about the fragile nature of the digital media and the network of computers we call the Internet. They regularly deconstruct the functions of commercial browsers, (mis)using designed scripting languages such as Java Script to destruct and deform information that we tend to take for granted. From the start Jodi has been using the programming tags of the languages of the web to create things that were not the intention when the languages were made. They have shown the internal mechanisms of the Internet through artistic expression that is both humorous and anarchistic at the same time. The *Wrongbrowser* is not so interesting in itself, but Jodi's attitude



```
http://www.bbc.co
<!DOCTYPE HTML PUBL
4.01 Transitional//

<html>
<head>
<title>BBC - BBCi -
digital destination

<meta name="descrip
the web, easy to us
and communities inc
Entertainment and E
<meta name="keyword
BBCi, bbci homepage
service broadcaster
corporation" />
<meta name="version
```

```
<HTML>

<head>
<TITLE>:::::::::::::::::TRANZISTOR.DK::</
TITLE>
<META NAME="Title" CONTENT="TRANZISTOR
Copenhagen DJ Radio">
<META NAME="Description"
CONTENT="TRANZISTOR is an independent
netradio
covering the electronic music scene in
Denmark">
<META NAME="Keywords"
CONTENT="tranzistor, transistor
```

http://www.mo
http://www.mok.com
http:
where we concentrate on new modes of interactivity. These applications can be
downloaded in our website.
2nd floor north
rutland house
42-46 new road
london e1 2ax
software
auto-illustrator
autoshop
finder mail
a portrait of netochka nezvanova
speed-camera
projects
signwave photos
ordure.org
jetztzeit.net
vx-labs.org
stub.org
sidestream.org
generative.net
Gear up for the games
A total of 72 countries are preparing for the 17th Commonwealth
should you watch out for? The BBC will have every angle covered on Radio, TV, Interactive TV and online.
Find out how you
click here
vCity NetPipe for
BBC NEWS
By using this site you agree to the vCity Terms of Servic. and Privacy polic
Copyright ©2001-2002 vCity, Inc. All Rights reserved.

Figure 32. Jodi's *Wrongbrowser*, here in full action of deconstructing and mixing websites and their underlying codes.

[88] Metatags are html data that are not displayed in the browser, but are nevertheless part of the html document. It is aimed for placement of extra information for search engines and the like.

towards computing and computer mediated communication is so powerful and historically important that their software becomes valuable just for that sake.

Nullpointer is an active artist and programmer who has been concerned with making musical applications, visual systems that represent sound, VST plug-ins[89] for commercial musical applications such as Cubase or Logic and even "artistic" modifications to commercial computer games. His *Webtracer* is probably one of the most useful and aesthetically pleasing art browser on the Net. It displays nodes and links in a three dimensional space in which the user can navigate and explore the content of the nodes. The structure can be viewed from every angle and the user can get a very solid idea of how the site is built up, what kind of information architecture is being used and also the internal structure of the host's file system. The program generates an interactive molecular diagram that is unique to each site it visits, resulting in structures that range from deeply interwoven tapestries to delicate and simple tree designs. This is done by registering which links and nodes the program finds first and then displaying that information accordingly. Thus a flat database system like Google would be displayed as a plateau whereas a more hierarchical site would appear in a more tree-like structure. Visually *Webtracer* displays beautifully an organised world of lines, nodes and patterns, but the main thing is how well one can use the visual patterns to explore and understand otherwise complex site structures. The navigational metaphors are taken from the game industry, which makes it easy and intuitive for the user to move around and explore the sites. Nullpointer does not take himself too solemnly when talking about his application:

> *Well aware of the legacy of webmapping as a supposed demystifying device and fetishised formalistic perversion of form I do not intend to decorate this project with too much hypothesis of cultural and social intent. (there are others who could grace it much better than myself) However I cannot deny that the intentions of the application are not primarily to aid webmasters in their analysis and development of their own sites but to, as I hope is obvious, repurpose the information that comprises hypertext and the web into another plane of perspective and interaction.*[90]

[89] VST plug-ins are Virtual Studio Instruments or effects that make the computer a sound studio in its own right. In stead of sending MIDI signals out of the computer to physical devices, such as samplers or synthesizers, the signals can be sent to VST instruments that run in the same program as the MIDI score.

[90] Rhizome interview with Tom Betts: http://www.rhizome.org/object.rhiz?2330

There are hundreds of interesting programs being written by people and uploaded on the net, but after years of formalistic experiments and stylistic exercises since the mid 1990s, I find the software made by people like Ward, I/O/D, deKam, and Nullpointer valuable in an essentially different sense than those programs showing clever tricks, beautiful algorithms or meaningless deconstruction. The mentioned artists are working with the limits of their medium, the context in which it appears culturally, and the social functions it imposes on the public in general. The problem with the institutionalisation of software art and generative art into the art world has clearly been the confusion we have witnessed where galleries have hosted big exhibitions of simple Flash programming exercises and web designers have suddenly begun to show their craft in galleries all over the Western world. The world of computer art, although over a half century old, is still so new in the eye of the public and the general world of art that it is hard to make the distinction between simple design and something that is of a more serious and lasting value.[91]

[91] A postmodernist critique of this argument could follow, asking what art is and who is to decide what shall be exhibited as art in the galleries and what not. To which I answer simply and in a very general way that when someone has done some serious thinking and is working intensively with an idea in whatever medium; and if the idea is good and the result is beautiful and cleverly thought out; resulting not only in the "wow cool!" response, but as something that can change the way one sees the world in which one dwells, then that could be called good art. The postmodernist might then start to talk about the politics of the art world, of its power structure and hype mechanism, to which I agree and acknowledge as *the* problem of art.

5. Conclusion

> After being conquered by Futurist eyes our multiplied sensibilities will at last hear with Futurist ears. In this way the motors and machines of our industrial cities will one day be consciously attuned, so that every factory will be transformed into an intoxicating orchestra of noises.
>
> :) Russolo - Art of Noises (1913)

This thesis has been concerned with the computer as an artistic medium and the artists I talked about are all people that use computer code as the material for their work. Working with programming languages as material for creating art can be very interesting and enjoyable endeavour. It is a little bit like painting or composing. One starts with a blank text editor and there one writes the functions and defines the events that will make the program run smoothly by whatever means one desires. Programming involves rigorous attention to detail, not unlike that involved in composition. Thus, it is not surprising that composers were the first artists to make substantive use of computers. There are many important decisions to take. Should it be interactivity from the user, an algorithm, stochastic variables or data retrieved from other sources such as from sensors or the Net, that triggers the process of running the program? Programming languages are limited by design, but what can be done within these limits is almost infinite. Just like in human languages. The artist-programmer has to learn the limits of the language and work within them. It is precisely working within the limits of each language that can be the creative inspiration.[92] In the same manner, the power of the programming language can often influence the artist in the creative process. The artist-programmer often finds himself in the situation of testing the limits of the language or some code that has already been written, and then experiences results that are far beyond the initial idea and much more original. The language can influence the work's conception when the programming concepts can suggest ideas that might not occur to one outside the context of coding. In this sense there is a high degree of interactivity between the artistic imagination and the structure of the programming language.

We now have an art form that uses code as the material that is moulded to create the work of art. Abstract code that can

[92] Just as happened when a group of Danish filmmakers created strict rules they called Dogma, for what is allowed in creating a film. The filmmakers found the restrictions liberating but of course there are always tendencies to break the rules, to push the boundaries and enter the field beyond.

simulate worlds in three dimensions or create sounds that are non-representative, i.e. not real-world sounds, but synthesised in the computer. It can process formal instructions so the works evolve over time and these instructions can have variables that make the work unique every time it is being generated. This fact challenges the way we relate to works of art and their creators. We find that the ontological unity of the work is somehow strangely absent, although it is "there" written in the programming code. From this fact we could explore the line of thought that sees the generative code as the carrier of the *idea*; some kind of an intermediary function between the pure concept (the artist's idea) and the manifestation of it (the work itself). Now, the danger would be to pursue this thought further into some Platonic categorical divide between the idea and the manifestation (code being the link between the two, the Cartesian coral gland) and there are reasons for not doing that here. The fact is that in processor based works, the potentials of the technology are determining the artist's ideas and contrarily the ideas can lead to addition to the technology (in the form of new libraries of code or additions to the programming language) which extend it significantly. There is no talk of *purity* of ideas here. The tools define and determine the artist and his ideas, but he can be actively engaged with the technology itself by broadening its scope and potentials. Donna Haraway radicalises this view in her "Cyborg Manifesto" from 1984: "It is not clear who makes and who is made in the relation between human and machine. It is not clear what is mind and what body in machines that resolve into coding practices."[93] She says that the high-tech culture has become *us*, our processes and an important aspect of our corporality. It challenges the dualisms of mind and body in intriguing ways and erases the modernist dichotomy between the human and the machine. The material (the code) and the ideas are interconnected just as in all other art forms that we already know.

In this thesis I have tried to illustrate how two histories – that of art, on the one hand, and technology and computing, on the other – have run parallel in the latter half of the 20th century, sometimes intersecting each other but mostly independently. In the 1st chapter I introduced the relatively new field of computer arts, and how the computer has

[93] Haraway (1991) p. 178

come to establish itself as both a tool and a medium, resulting in a boom in the late 1990's where many artists chose to use the computer as their main medium. Because of the popularity of the computer, we have innumerable directions, styles and movements that are part of the general concept *computer arts.* For this thesis I saw the need to create another concept – *processor art* – with the aim of limiting myself to concentrate solely on the art that uses the microprocessor's ability to process code as the defining characteristic of the work, as opposed to works that could theoretically be distributed or communicated by other (analogue) means.

In the 2nd chapter we went through the history of technology in the 20th century art. We saw how the avant-garde of the early century became fascinated by technology and started using it their works, either as inspiration or directly building technological art works. Artists were opening up to the possibilities of the accidental, of the random, the machine-processed, and the unfinished work of art. The art work became possible as an everlasting process. The Minimalists and Fluxus were very important for opening up spaces in which computer art would later find itself. Although early computer art was very modernistic and formalistic in its aesthetics, it moved later into the direction of Minimalism, Fluxus and Situationalism where more attention would be paid to the process and the context in which it happens than strict formalism. However, we also saw how the computer was not taken as a serious tool for making art until the late 1980's. The reasons are many, but mainly due to expensive technology, programming languages that were not user friendly, and conservativism in the general public and media. The computer was up until the 1990's a strange tool that had not become fully integrated into our culture. This is illustrated by how the popularity of the personal computer boomed with the advent of the Internet.

The philosophy of technology and the way technology affects our arts and culture was the subject of the 3rd chapter. I divided the chapter into 3 areas that are essential when talking about the actual changes: a) Technological Transitions, b) Human Transitions, and c) Media Transitions. I see the latter half of the 20th century as a period where technology, humanity and the way we use media undertook dramatic changes that are of equal importance to the advent of printing, electricity or telecommunications and the way these revolutions affected our culture. I illustrated, using Martin Heidegger's modernist essay about technology, how the modernist view of technology and humanity was deterministic, control orientated and put the human subject always in the centre. Nature was ours to control and make use of and we were supposed to be the masters of Nature, Society and our own Psyche. We saw how the symbiotic relationship between technology and science brought with it a new worldview – that of non-linearity, complexity, networks, chaos, emergence, artificial life, and other factors that became the strong symbols in our society and infiltrated quickly into the arts. This new worldview has come through by using the powerful technologies of the computer to calculate and predict factors that before would be impossible for humans to do. In addition to this, the fact that we are working with an intelligent tool that simulates our own thought processes, has resulted in changes relating how we experience ourselves as humans. In the section about Human Transitions, this change is analysed and the essence of post-humanism discussed. It is due to changes in technology and media that we have this new perspective of the world, where we define ourselves against the machine and not nature anymore. When we started defining ourselves against the machine, we became post-human. Living in a networked world of technology where time and space have altered their meaning only in 20 years, the metaphors have changed – we work with networked and rhizomatic metaphors instead of the hierarchical and tree-like ones of modernism.

Finally, in the 4th chapter – Processor Art – we were able to come to the task of analysing processor art and its aesthetic characteristics. It was necessary to go through traces of the histories of 20th century western art on the one hand, and technology on the other, to see where we are coming from and how it resulted in the combination art and technology into the increasingly popular *computer art* in the mid 1980's. Processor art is a subcategory of computer art, and here we are talking about art that is essentially processed in real-time by the computer when it is enjoyed. I extracted some of its characteristics, such as the biological worldview; the non-referential nature of the content; digitalism, i.e. the medium (and its faults) influencing the aesthetic; the aesthetic of the accident; the aesthetic of the cause (interactivity); and at last, the cerebral/disembodied nature of contemporary computing. All of these characteristics are apparent, in some form or another, in the works that I analysed later in the chapter under the categories *generative art* and *software art*.

The works I chose under *generative arts* were chosen firstly for their qualitative value but also as examples of the aesthetic phenomena I had been analysing. All these aesthetic characteristics can be found in some form or another in the works discussed, but various elements are emphasised in different works. Thus I found the musicians on Morpheus working with the idea of the biological concept of genotype/phenotype; with the computer as a world where things can be created and evolved. Meta is an artist who is highly dependent on the aesthetics of the technology he is using and sometimes letting technology take control, whereas Jem Finer tries to go beyond technology and use traditional instruments, such as guitar, to control or collaborate with the technology. Finer is an example of a musician that is not content with the disembodied modes of working in computer music, so he extends his system to include physical instruments to interface with the computer. And Celestino Soddu sees technology as something that is helping us to go beyond the limits of the analogue media we have used up until now. The computer is the creative machine that can experiment with all the possible trails of an artist's idea and generate naturally offsprings of that idea.

In the *software art* part I discussed works that are also characteristic for some of the mentioned aesthetic tendencies. Adrian Ward's work with Auto-Illustrator or Autoshop is using the accidental to a high degree, the playful and the strange modes of interactivity that are involved with the aesthetic of the cause. The user doesn't always know what happens when a tool is applied. Ward's critical stance towards the commercial companies is also an important part of his work. Golan Levin's works are influenced by the biological science and mathematics, and so is the work of Benjamin Fry, which works with the human genome as his source material. Netochka Nezvanova is probably the best example of an artist who works with the aesthetic of noise and distorted signal. She concentrates on the qualities of the computer medium and what can happen when it goes wrong or the technology takes over the user's will. Those accidents – sometimes called "glitch" – are to be seen as creative input from the technology, at least as a valid input. Finally I took some examples of browser artists, who are working with the networked nature of modern computers in their art. The browser artists work with diverse aesthetic notions, but they are all concentrating on networks, signal vs. noise, new representations and interactivity. The browser is a good example of simple software, and there have been innumerable artists working on alternative browsers for the World Wide Web.

We are witnessing a new field growing where artists are using tools that have qualities so different from the analogue media that it results in countless questions to be asked and discussed in the near future. The works discussed in this thesis are all opening up for a space in which we start to question the author, the ontology of the art works and the role of the user/listener/viewer. (We even lack a word for the aesthetical activity of using software or generative art, as the word "user" does not have the same "aesthetical" connotations as "reader," "listener," or "viewer.") I have tried to outline a field – processor art – and illustrate its characteristics and aesthetic notions. It is a very young field and only the future knows how things will evolve with ever new scientific discoveries, technological inventions, art historical evolution, and cultural politics.

Artists working in the field of processor art are creating their work for various reasons. It could be a critique of the commercial world of software companies and the way their software determine us to work and view things in certain ways. It could also be a concentration on experimenting with the new medium, testing out how we relate to works that change and evolve over time and do not end up as a final product. And again, it could be related to the new science of complexity, of genetic science and biology, or artificial life.[94] For Philip Galanter, who I mentioned earlier in this thesis, the science of complexity can help us to pass beyond postmodernism and its ironic stance toward life and existence. The new technology provides artists with opportunities to become fully engaged and better aligned with the intellectual views found in the new sciences. The computer offers new opportunities to describe, comment and portray life and the sciences of life. The artist has a very important role relating to modern technology as artists are able to perform experiments and conceptualise things about our contemporary worldview that the sciences would never be able to do. The artist has a certain freedom of expression and thought that the serious scientist could not allow him/herself to have, a freedom to imagine the possible applications of technology; the utopian and dystopian potential that lies within its seed.

[94] Wilson (2002) p. 295

Writing about technology is not a simple thing and this thesis has dealt with how artists are using the computer technology in their work. It showed how the modernist “machine-technology” changed into the “information-technology” and the resulting changes in culture and how we view ourselves. I have abstained from making any value judgements about the evolution because I don’t think it is related to the subject matter of the thesis. The new technology can be at the same time wonderful and terrible and it puts our culture through serious testing. When money and information flows through computer networks, the whole world economy changes and one might ask whether we are ready to deal with the sudden changes. More worriedly, when we start to manipulate the genetic structure of our own genome, and that of other animals and plants, can we really foresee the results that will have?[95] Or the danger if the knowledge of nano-technology (where we are making small machines on the size level of atoms) gets into the wrong hands? These questions are profound and complex and not to be answered here. However, I sincerely believe that artists have the possibility to research, represent and communicate to the public the phenomena we are dealing with here in a strong and unique way. They can build up imaginary worlds, research artificial life or simulate nature to the degree that we see the beauty of our technology and of the nature in which we dwell. They can also criticise and reflect upon how the world of digital media coerces each of us into certain work processes, usage of the media, and the way we communicate with each other. Art is continuing its critical stance towards culture, but when good part of the culture has become mediated through the digital media, the strongest artistic tools are arguably those of digital technology as well.

In the 3rd chapter I wrote about how the philosopher Martin Heidegger saw the problem of technology. For him, we should not fear technology or curse it, but try to understand its essence which itself is nothing technological. The essence of technology is the way it enframes our way of thinking and eliminates other ways of seeing the world. As I illustrated, Heidegger was relating to the technology of the industrialisation and not the post-industrial one we are

[95] These questions are being addressed in two books that were published in summer 2002: Francis Fukuyama’s *Our Posthuman Future* and Gregory Stock’s *Redesigning Humans*.

concerned with today. For Heidegger art is one of the ways that we can liberate our thought from the enframing essence of technology. And I also showed how the art Heidegger was talking about is the pre-avant-garde art of early modernity and very different from the art we have today. He uses dramatic words where he talks about art as the "saving power" from the "extreme danger" that he finds in the technological thought. In modern day philosophy of technology much has been written about the dangers of enframing and how art can liberate our culture from the extreme danger of the technological closure of thought. However, at the very end of his important essay he writes a passage that is often overlooked:

> *Whether art may be granted this highest possibility of its essence in the midst of the extreme danger, no one can tell. Yet we can be astounded. Before what? Before this other possibility: that the frenziedness of technology may entrench itself everywhere to such an extent that someday, throughout everything technological, the essence of technology may come to presence in the coming-to-pass of truth.*[96]

I suspect we are living in "this other possibility." The Heideggerian modernism of art vs. technology has collapsed and we are immersed in technology where our actions are highly dependent on technology in almost all spheres of life. What Heidegger states as another possibility is that through everything technological, we might be able to understand our relation to technology, its place in our world and how it shapes it. It is perhaps *through* technology that our strongest abilities for expression can take place, where by technological means we can explore the nature of our new second nature: technology. As Heidegger might have put it: technology can come to presence in the coming-to-pass of truth.

[96] Heidegger (1977) p. 35.

Bibliography

Anonymous. *120 Years of Electronic Music: Electronic Musical Instrument 1870-1990.*
http://www.obsolete.com/120_years/, July 2002.

App-Art. *Interview with n_gen.*
http://www.app-art.org/features/n_gen/, July 2002.
------. *Interview with Greg Sidal.*
http://www.app-art.org/features/sidal/, July 2002.

Arata, Luis O. *Reflections about Interactivity.* Webarticle: http://media-in-transition.mit.edu/articles/arata.html. Des, 1999.

Ascott, Roy. *Turning on Technology.*
http://www.cooper.edu/art/techno/essays/ascott.html, May 2002.

Attali, Jacques. *Noise: The Political Economy of Music*.
Minneapolis: University of Minnisota Press, 1985.

Bandur, Markus. *Aesthetics of Total Serialism: Contemporary Research from Music to Architecture.* Basel: Birkhäuser, 2001.

Barbrook, Richard & Andy Cameron. *The Californian Ideology.* (1995) Webarticle: http://www.alamut.com/subj/ideologies/pessimism/califIdeo_I.html. July, 2002.

Barthes, Roland, "The Death of the Author" in *Image-Music-Text.* New York: Hill and Wang, 1977. pp. 142-148.

Baumgärtel, Tilman. *Generate! Degenerate! n-generate!*
http://www.eyestorm.com/feature/ED2n_article.asp?article_id=342&caller=1, July, 2002.
------. *Browsers Bite Back.*
http://www.eyestorm.com/feature/ED2n_article.asp?article_id=234&caller=1, July 2002.
------. *Portrait of the Artist as Harddisk.*
http://www.eyestorm.com/feature/ED2n_article.asp?article_id=300&caller=1, July 2002.
------. *Connect Everything with Everything.*
http://www.eyestorm.com/feature/ED2n_article.asp?article_id=318, July 2002.

Bell, David & Barbara M. Kennedy. *The Cybercultures Reader.* London: Routledge, 2000.

Benjamin, Walter. “The Work of Art in the Age of Mechanical Reproduction.” *Illuminations*. (ed.) Hannah Arendt. New York: Schocken Books, 1969. pp. 217-251.

Berzowska, Joanna Maria. *Computational Expressionism: A Study of Drawing with Computation.* Massachusetts Institute of Technology. 1999.

Berry, Josephine. *Bare Code: Net Art and the Free Software Movement.*
http://netartcommons.walkerart.org/article.pl?sid=02/05/08/0615215&mode=thread

Bolter, Jay David and Richard Grusin. *Remediation: Understanding New Media*. Cambridge: The MIT Press, 1999.

Bolter, Jay David. *Writing Space: The Computer, Hypertext, and the History of Writing.* Hillsdale, New Jersey: Lawrence Erlbaum Associates, Publishers, 1991.

Bongers, Bert. “An Interview with Sensorband”. *Computer Music Journal*, 22:1. pp 13-24. spring 1998.

Borgmann, Albert. *Holding On to Reality: The Nature of Information at the Turn of the Millenium.* Chicago: The University of Chicago Press, 1999.

Bosma, Josephine. *Music and the Internet: Musaic*. Crossfade.
http://crossfade.walkerart.org/

Boyd, Frank & co. *New Media Culture in Europe.* Amsterdam: Uitgeverij de balie & The Virtual Platform, 1999.

Broeckmann, Andreas. *New Media Culture in Europe.* Amsterdam: Uitgeverij De Balie & The Virtual Platform, 1999.

Broeckmann, Andreas; Joke Brouwer; Bart Lootsma; Arjen Mulder & Lars Spuybroek. (eds.). *The Art of the Accident.* Rotterdam: NAI Publishers/V2_Oranisatie, 1998.

Brown, Paul. “Breaking the Art and Science Standoff” in *Leonardo.* Vol. 34, No. 4, pp. 335-336, 2001.
------. *Stepping Stones in the Midst.*
http://www.paul-brown.com/WORDS/STEPPING.HTM, July 2002.
------. *Networks and Artworks: The Failure of the User Friendly Interface.*
http://www.paul-brown.com/WORDS/NETART.HTM, July 2002.

Bukatman, Scott. *Terminal Identity: The Virtual Subject in Post-Modern Science Fiction.* Durham: Duke University Press, 1993.

Burger, Peter. *Theory of the Avant-Garde*. Minneapolis: University of Minnesota Press, 1996.

Bush, Vannevar. "As We May Think". http://www.isg.sfu.ca/~duchier/misc/vbush

Cage, John. *Silence*. Hanover: Wesleyan University Press, 1961.
------. *An Autobiographical Statement.*
http://www.newalbion.com/artists/cagej/autobiog.html, May 2002.

Castells, Manuel. *The Rise of the Network Society.* Massachusetts: Blackwell Publishers, 1996.

Chadabe, Joel. *Electric Sound: The Past and Promise of Electronic Music*. New Jersey: Prentice Hall, 1997.

Chadabe, Joel. "Remarks on Computer Music Culture". *Computer Music Journal*, 24:4, pp. 9-11, Winter 2000. Massachusetts Institute of Technology.

Cohen, Harold, *Coloring without Seeing: Problems in Machine Creativity.*
http://crca.ucsd.edu/~hcohen/cohenpdf/colouringwithoutseeing.pdf, July 2002.

Connerton, Paul, *How Societies Remember*. Cambridge: Cambridge University Press. 1989.

Cott, Jonathan. *Stockhausen: Conversations with the Composer*. London: Robson Books, 1974.

Cox, Cristoph. *Abstract Concrete: Fransico Lopez and the Ontology of Sound.*
http://www.franciscolopez.net/art_cab.html, May 2002.

Cramer, Florian & Ulrike Gabriel. *On Software Art.*
http://www.rhizome.org/object.rhiz?2848, July 2002.
------. *Software Art and Writing.*
http://userpage.fu-berlin.de/~cantsin/homepage/writings/software_art/american_book_review/software_art_essay.html, July 2002.

Cramer, Florian. *Digital Code and Literary Text.*
http://userpage.fu-berlin.de/~cantsin/homepage/writings/net_literature/code_poetry/erfurt_2001//digital_code_and_literary_text.html, May 2002.

------. *Hacking the Art OS: Interview with Cornelia Sollfrank.* http://www.rhizome.org/object.rhiz?3314, July 2002.

Cross, Ian. *AI and Music Perception.* http://www-ext.mus.cam.ac.uk/~ic108/AISB99/, June 2002.

Dahlstedt, Palle & Mats G. Nordahl. "*Living Melodies*: Coevolution of Sonic Communication" in *Leonardo.* Vol. 34, No. 3, pp. 243-248, 2001.

Daniels, Dieter. "Art and Media in the XXth Century" in *The Age of Modernism, Art in the 20th Century* (ed.) Christos M. Joachimides, Norman Rosenthal, Ostfildern: Gerd Hatje Verlag, 1997

Danto, Arthur C. *After the End of Art: Contemporary Art and the Pale of History.* New Jersey: Princeton University Press, 1997.

Davies, Douglas. *The Work of Art in the Age of Digital Reproduction: An Evolving Thesis/1991-1995*. http://www.awa.com/artnetweb/views/davis/davis.html (May, 2002)

Debray, Régis. *Media Manifestos.* London: Verso, 1996.

De Kerckhove, Derrick. *The Architecture of Intelligence.* Basel: Birkhäuser, 2001.

De Landa, Manuel. *A Thousand Years of Nonlinear History.* New York: Zone Books, 1997.
------. *Homes: Meshwork or Hierarchy.* Webarticle: http://www.doorsofperception.com/doors/revamped_frameset.html. Feb. 2000.
------. *Uniformity and Variability: An Essay in the Philosophy of Matter.* Webarticle: http://www.doorsofperception.com/doors/revamped_frameset.html. Feb. 2000.
------. *Art and the Military.* http://switch.sjsu.edu/web/v3n3/DeLanda/delanda.html, May 2002.
------. *Meshworks, Hierarchies and Interfaces.* http://www.t0.or.at/delanda/meshwork.htm, May 2002.
Deleuze, Gilles & Felix Guattari. *A Thousand Plateaus: Capitalism and Schizophrenia*. Minneapolis: University of Minnesota Press, 1987.

Dery, Mark. *Escape Velocity: Cyberculture at the End of the Century.* New York: Grove Press, 1996.

Dinkla, Söke. *The History of the Interface in Interactive Art.* Webarticle: http://www.isea.qc.ca/symposium/archives/isea94/pr208.html. December, 2000.

DiScipio, Agostino. "*Questions Concerning Music Technology. From Heidegger's view to Feenberg's Subversive Rationalization"* website: http://cadre.sjsu.edu/switch/sound/articles/discipio1.html (August, 2001).

Dixon, Joan Broadhurst & Eric J. Cassidy. (eds.) *Virtual Futures: Cybernetics, Technology and Post-Human Pragmatism.* London: Routledge, 1998.

Dorin, Alan. "Generative processes and the electronic arts" in *Organised Sound* 6(1), pp. 47-53. 2001. Cambridge University Press.

Dorin, Alan & Jon McCormack. "First Iteration: A Conference on Generative Systems in the Electronic Arts" in *Leonardo*. Vol 34. no 3. pp. 239-242, 2001.

Dreyfus, Hubert L & Charles Spinosa. *Highway Bridges and Feasts: Heidegger and Borgman on How to Affirm Technology.* Webarticle: http://www.focusing.org/dreyfus.html. July, 2002.

Droste, Magdalena. *Bauhaus: 1919-1933.* Berlin: Taschen, 1998.

Drotner, Kirsten; Klaus Brun Jensen; Ib Poulsen & Kim Schrøder. *Medier og Kultur: En Grundbog I Medieanalyse og Medieteori.* Copenhagen: Borgen, 1999.

Druckery, Timothy. (ed.) *Ars Electronica: Facing the Future*. Cambridge: The MIT Press, 1999.
------. (ed.) *Electronic Culture: Technology and Visual Representation.* New York: Aperture, 1996.

Eacott, John. (ed.) *Cybersonica: Proceedings. International Festival of Music & Sound June 5-7 2002.* London: University of Westminister, 2002.
------. *Form and Transience: Generative Music Composition in Practise.*
http://www.informal.org/form_and_transience.htm, May 2002.
------. *Morpheus: Emergent Music.*
http://www.informal.org/morpheus_emergent_music.htm, May 2002.

Eckel, Gerhard. *Technological Musical Artifacts.* Webarticle: http://viswiz.imk.fraunhofer.de/~eckel/publications/eckel98a/eckel98a.html, July, 2002.

Engeli, Maia. *Digital Stories: The Poetics of Communication.* Basel: Birkhäuser, 2000.

Eno, Brian. *Generative Music.* http://www.inmotionmagazine.com/eno1.html, February 2002.

Ezzl, Karlheins. *Composing in Cyberspace.* http://www.essl.at/bibliogr/cyberkomp-e.html, February 2002.

Fernström, Mikael. *NIME 2002: International Conference on New Interfaces for Musical Expression. Proceedings.* Limerick: University of Limerick, 2002.

Finnemann, Nils Ole. *Modernity Modernised – The Cultural Impact of Computerisation.* Guest lecture, Tampere University, Finland, January 16th, 1997.

Fiske, *Harold E. Music Cognition and Aesthetic Attitudes*. Lampeter: The Edwin Mellen Press, 1993.

Fukuyama, Francis. *Our Posthuman Future*. New York: Farrar Straus & Giroux, 2002.

Fuller, Matthew. *Interview with Nettime.* http://www.backspace.org/iod/nettime.html, June 2002.

Föllmer, Golo. *Soft Music.* Webarticle: http://crossfade.walkerart.org/foellmer/text_print.html, July, 2002.

Gascone, Kim. "The Aesthetics of Failure: "Post-Digital" Tendencies in Contemporary Computer Music.*" Computer Music Journal*, 24:4, pp. 12-18, Winter 2000. Massachusetts Institute of Technology.

Garnett, Guy E. "The Aesthetics of Interactive Computer Music.*" Computer Music Journal*, 25:1, pp. 22-33, Spring 2001. Massachusetts Institute of Technology.

Gibson, William. *Neuromancer.* London: Voyager, 1995.

Godfrey, Tony. *Conceptual Art.* Lodon: Phaidon, 1998.

Goldberg, Ken. (ed.) *The Robot in the Garden: Telerobotics and Telepistomology in the Age of the Internet.* Cambridge: The MIT Press, 2000.

Greenberg, Clement. *The Collected Essays and Criticism*. (ed.) John O'Brian. Chicago: University of Chicago Press, 1988.

Greenberg, Jeffrey. *The Aesthetics of Software [notes].* http://www.concentric.net/~Jgberg/Research/software_art.html, February 2002.

Griffiths, Paul. *A Concise History of Modern Music: From Debussy to Boulez.* London: Thames and Hudson, 1978.

Haraway, Donna. *Simians, Cyborgs and Women: The Reinvention of Nature.* New York: Routledge, 1991, pp.149-181.

Hailes, N. Katharine. *How We Became Posthuman: Virtual Bodies in Cybernetics, Literature, and Informatics.* Chicago: University of Chicago Press, 1999.

Heidegger, Martin. *The Question Concerning Technology and Other Essays.* New York: Harper and Row, 1977.

Hodges, Nicola. (ed.) *Art and Technology.* London: Art & Design, 1994.

Hollein, Max; Jesper N. Jørgensen & Schirn Kunsthalle Frankfurt (eds.) Frequenzen [Hz]/Frequencies [Hz]: Audiovisuelle Räume/ Audio-visual Spaces. Frankfurt: Hatje Cantz Verlag, 2002.

Holton, Gerald. “Henri Poincaré, Marcel Duchamp and Innovation in Science and Art” in *Leonardo*, vol. 34, No. 2. pp. 127-134. 2001.

Holtzman, Steven R. *Digital Mantras: The Languages of Abstract and Virtual Worlds.* Cambridge: The MIT Press, 1994.

Hughes, Robert. *The Shock of the New: Art and the Century of Change.* London: Thames and Hudson, 1991.

Huhtamo, Erkki, “Time Traveling in the Gallery: An Archeological Approach in Media Art” pp. 233-268. *Immersed in Technology: Art and Virtual Environments*. (ed.) Moser, Mary Anne. Cambridge: The MIT Press, 1996.

Huyssen, Andreas. *After the Great Divide: Modernism, Mass Culture, Postmodernism.* Bloomington: Indiana University Press, 1986.

Ihde, Don. *Technology and the Lifeworld: From Garden to Earth.* Bloomington: Indiana University Press, 1990.

Ippolito, Jon. *Code as Creative Writing: An Interview with John Simon.*
http://rhizome.org/object.rhiz?3312, June 2002.

Jacob, Bruce L. “Algorithmic composition as a model of creativity” in *Organised Sound* 1(3), pp. 157-65. 1996. Cambridge University Press.

Jackson, Tim. *Towards a New Media Aesthetic.* http://www.uoregon.edu/~ucurrent/uc6/6-jackson.html, June 2002.

Jameson, Frederic. *Postmodernism, or, the Cultural Logic of Late Capitalism*. New York: Duke University Press, 1991.

Jensen, Jens F. (ed.). *Multimedier, Hypermedier, Interaktive Medier.* Aalborg: Aalborg Universitetsforlag, 1998.
------. "Computermedier: Computeren som medie, kommunikation og mediekultur" in *Computer-Kultur, Computer-Medier, Computer-Semiotik*. (ed.) Jensen, Jens F. Copenhagen: NSU, 1990.

Jordà, Sergi. *New Musical Interfaces and New Music-Making Paradigms.* Webarticle: http://www.iua.upf.es/~sergi/articles/NIME2001.pdf, July, 2002.
-------- "Faust Music On Line (FMOL): An Approach to Real-time Collective Composition on the Internet", *Leonardo Music Journal.* Vol. 9, pp. 5-12, 1999.

Kluitenberg, Eric. *Smash the Surface/ Break Open the Box/Disrupt the Code.* Webarticle: http://www.idea.org.uk/cfront/workshop/txt/smash-surface-en.html. August, 2001.
------- "Human Art is Dead! Long Live the Algorithmic Art of the Machine!" Interview with Huge Harry, webarticle: http://iaaa.nl/hh/div/MuteInterview.html, July, 2002.

Kurzweil, Ray. *The Age of Spiritual Machines.* London: Phoenix, 1999.

LaBelle, Brandon & Steve Roden. *Site of Sound: of Architecture and the Ear*. Los Angeles: Errant Bodies Press, 1999.

Langton, Christopher (ed.) *Artificial Life*. Redwood City: Addison-Wesley, 1989.

Lansdown, John. *Artificial Creativity: An Algorithmic Approach to Art.* http://www.cea.mdx.ac.uk/CEA/External/Staff96/John/artCreat.html, May 2002.

Levin, Golan. *Painterly Interfaces for Audiovisual Performance.* Massachusetts Institute of Technology. 2000.

Levin, Golan; Lia; Meta & Adrian Ward. *Generative Design: Beyond Photoshop.* Birmingham: Friends of Ed, 2001.

Levinson, Paul. *Soft Edge: A Natural History and Future of the Information Revolution.* London: Routledge, 1997.

Link, Stan. “The Work of Reproduction in the Mechanical Aging of an Art: Listening to Noise” in *Computer Music Journal*, 25:1, pp. 34-47, Spring 2001. Massachusetts Institute of Technology.

Lovejoy, Margot. *Postmodern Currents: Art and Artists in the Age of Electronic Media.* (2nd. ed.) New Jersey: Prentice Hall, 1997.

Lovink, Geert. *Strategies for Media Activism.* http://trace.ntu.ac.uk/frame/text/lovink.html, April 2002.

Lopez, Francisco. *Cagean Philosophy: A Devious Version of the Classical Procedural Paradigm.* http://www.franciscolopez.net/cage.html, May 2002.
------. *Environmental Sound Matter.* http://www.franciscolopez.net/env.html, May 2002.
------. *Schizophonia vs. l'objet sonore: Soundscapes and Artistic Freedom.* http://www.franciscolopez.net/schizo.html, May 2002.

Luhmann, Niklas. *The Reality of the Mass Media (Cultural Memory in the Present).* Stanford: Stanford University Press, 2000.

Lunenfeld, Peter. *Snap to Grid: A User's Guide to Digital Arts, Media and Cultures.* Cambridge: MIT Press, 2000.
------. (ed.) *The Digital Dialectic: New Essays on New Media.* Cambridge: The MIT Press, 1999.

Maeda, John. *Maeda @ Maeda.* London: Thames and Hudson, 2000.
------. *Design by Numbers.* Cambridge: The MIT Press, 1999.

Manovich, Lev. *The Language of New Media*. Cambridge: The MIT Press, 2001.
------. *New Media Aesthetics.* Webarticle: http://www.doorsofperception.com/magazine/newmediaaesthetics.html. 2001.
------. *Avant-Garde as Software.* http://www.manovich.net/docs/avantgarde_as_software.doc, July 2002.
------. *New Media from Borges to HTML.* http://www.manovich.net/DOCS/manovich_new_media.doc, July 2002.
------. *Post Media Aesthetics.* http://www.manovich.net/DOCS/Post_media_aesthetics1.doc, July 2002.
------. *The Death of Computer Art.* http://www.thenetnet.com/schmeb/schmeb12.html, April 2002.
------. *Flow: The New Form?* http://www.doorsofperception.com/magazine/

newmediaaesthetics.html, April 2002.
------. *Models of Authorship in New Media.* http://switch.sjsu.edu/nextswitch/switch_engine/front/front.php?artc=65, April 2002.

Mateas, Michael. “Expressive AI: A Hybrid Art and Science Practise” in *Leonardo,* vol. 34, No. 2, pp. 147-153, 2001.

Mayer, Paul A. *Computer Media and Communication: A Reader.* Oxford: Oxford University Press, 1999.

McAlpine, Kenneth; Eduardo Miranda; and Stuart Hoggar. “Making Music with Algorithms: A Case-Study System” in *Computer Music Journal*, 23:2, pp. 19-30, Summer 1999. Massachusetts Institute of Technology.

McLean, Alex. *Hacking Sound in Context.* http://www.generative.net/papers/hacking/index.html, Feb, 2002.
------. *The Forkbomb.* http://www.slab.org/forkbomb.pl.tt2, July 2002

McLuhan, Marshall. *Understanding Media: The Extensions of Man.* Cambridge: The MIT Press, 1998.
------. & H. Parker. *Counterblast.* New York: Harcourt, Brace, 1969.

Michelsen, Anders. “The hardest part of art is the last for inches: Aspekter af computerkunstens teorihistorie” in *Kunstteori: Positioner i nutidig kunstdebat”* (eds. Christensen, Hans Dam; Michelsen, Anders & Wamberg, Jacob). Borgen. Copenhagen: 1999.

Mieszkowsky, Katharine. *Netochka Nezvanova: The Most Feared Woman on the Internet.* http://www.salon.com/tech/feature/2002/03/01/netochka/print.html, July 2002.

Morales-Manzanares, Roberto; Eduardo F Morales; Roger Dannenberg; and Jonathan Berger. “SICIB: An Interactive Music Composition System Using Body Movements” in *Computer Music Journal*. 25:2, pp. 25-36, Summer 2001. Massachusetts Institute of Technology.

Moroni, Artemis; Jônatas Manzolli; Fernando Von Zuben & Ricardo Gudwin. “Vox Populi: An Interactive Evolutionary System for Algorithmic Music Composition” in *Leonardo Music Journal.* Vol. 10, pp. 49-54, 2000.

Moser, Mary Anne & Douglas MacLead. *Immersed in Technology: Art and Virtual Environments.* Cambridge: The MIT Press, 1996.

Motte, Warren F. (ed/transl.) *Oulipo: A Primer of Potential Literature*. Illinois: Dalkey Archive Press, 1998.

Moura, Leonel. *Swarm Paintings: Non-Human Art.* http://www.lxxl.pt/aswarm/aswarm.html, July 2002.

Murray, Janet H. *Hamlet on the Holodeck: The Future of Narrative in Cyberspace.* New York: The Free Press, 1997.

Napier, Mark. *The Aesthetics of Programming.* http://www.afsnitp.dk/onoff/Texts/printerfriendly/napiertheaesthet.html, July 2002.
------. *A Conversation about Art, Software and Process.* http://switch.sjsu.edu/nextswitch/switch_engine/front/front.php?artc=82, July 2002.

Nelson, Peter & Nigel Osborne. (eds.) *Aesthetics of Live Electronic Music.* New Jersey: Contemporary Music Review, Volume 18, Part 3. 1999.

Nezvanova, Netochka. "The Internet, A Musical Instrument in Perpetual Flux". *Computer Music Journal*, 24:3, pp. 38-41, Fall 2000. Massachusetts Institute of Technology.

Nyman, Michael. *Experimental Music: Cage and Beyond.* Cambridge: Cambridge University Press, 1974.

Palumbo, Maria Luisa. *New Wombs: Electronic Bodies and Architectural Disorders.* Basel: Birkhäuser, 2000.

Palombini, Carlos. "Technology and Pierre Schaeffer: Pierre Schaeffer's *Arts-Relais,* Walter Benjamin's *technische Reproduzierbarkeit* and Martin Heidegger's *Ge-Stell*" in *Organised Sound* 3(1), pp. 35-43. 1998. Cambridge University Press.

Penny, Simon. *From A to D and Back Again: The Emerging Aesthetics of Interactive Art.* http://www-art.cfa.cmu.edu/Penny/texts/AtoD.html, July 2002.
------. *Agents as Artworks and Agent Design as Artistic Practise.* http://www-art.cfa.cmu.edu/penny/texts/Kerstin_Chptr_Final.html, July 2002.
------. *The Darwin Machine: Artificial Life and Interactive Art.* http://www-art.cfa.cmu.edu/Penny/texts/Darwin_Machine.html, July 2002.
------. *Embodied Cultural Agents: At the Intersection of Art, Robotics and Cognitive Science.* http://www-art.cfa.cmu.edu/Penny/texts/Embodied_Cult_Agents.html, July 2002.
------. *System Aesthetics and Cyborg Art: The Legacy of Jack*

Burnham.
http://www-art.cfa.cmu.edu/Penny/texts/Sculpture-Burnham.html, July 2002.

Pepperell, Robert. *The Posthuman Condition.* Exeter: Intellect Books, 1995.

Pequet, Frank. "From the Model to its algorithmic application" in *Organised Sound* 4(2), pp. 73-78. 1999. Cambridge University Press.

Popper, Frank. *Art of the Electronic Age.* London: Thames and Hudson, 1993.

Pratella, Balilla, *Manifesto of Futurist Musicians*, 1912, http://www.futurism.fsnet.co.uk/manifestos/manifesto05.htm

Qvotrup, Lars. *Det hyperkomplekse samfund: fjorten fortællinger om informationssamfundet.* Copenhagen: Gyldendal, 2000.

Rajchman, John. *The Deleuze Connections.* Cambridge: The MIT Press, 2000.

Ramos, Vitorino. "On the Implicit and on the Artificial: Morphogenesis and Emergent Aesthetics in Autonomous Collective Systems." In *Architopia. Art, Architecture and Science.* (eds.: Maubant & Moura) Lisboa: Ministerio da Ciencia e Tecnologia, 2002.

Reich, Steve. *Music as Gradual Process.* (1968). Webarticle: http://www.oberlin.edu/~jaltieri/gradualprocess.html. June, 2002.

Richard, Dominique. "Code-notes-music: an epistemological investigation of algorithmic music" in *Organised Sound* 1(3), pp. 173-7. 1996. Cambridge University Press.

Riddell, Alistair. "Data Culture Generation: After Content, Process as Aesthetic" in *Leonardo*, Vol. 34, No. 4, pp. 337-343, 2001.

Roads, Curtis. *The Computer Music Tutorial*. The MIT Press, Cambridge, Massachusetts, 1996.

Rokeby, David. *The Construction of Experience : Interface as Content.*
http://www.interlog.com/~drokeby/experience.html, June 2002.
------. *Lecture for "Info Art": Kwangju Biennale*,
http://www.interlog.com/~drokeby/install.html. June 2002.
------. *The Harmonics of Interaction.*
http://www.interlog.com/~drokeby/harm.html, June 2002.

Root-Bernstein, Robert S. “Music, Creativity and Scientific Thinking” in *Leonardo.* Vol. 34, No. 1, pp. 63-68, 2001.

Ross, David. *Transcription of Lecture by David Ross, San Jose State University, March 2, 1999.* http://switch.sjsu.edu/web/ross.html, June 2002.

Rush, Michael. *New Media in Late 20th-Century Art*. New York: Thames and Hudson, 1999.

Russolo, Luigi. *The Art of Noise,* 1913. (webarticle) http://www.futurism.fsnet.co.uk/manifestos/manifesto09.htm, Jan, 2002.

Rutsky, R.L. *High Techné: Art and Technology From the Machine Aesthetic to the Posthuman.* Minneapolis: University of Minnesota Press, 1999.

Shanken, Edward A. *The House that Jack Built: Jack Burnham's Concept of "Software" as Metaphor for Art.* http://www.duke.edu/~giftwrap/House.html, February 2002.

Schutze, Paul. “Strategies for Making Sense”. An interview with Brian Eno. Wire 139, September, 1995.

Schwarz, K. Robert. *Minimalists.* London: Phaidon, 1996.

Stock, Gregory. *Redesigning Humans: Our Inevitable Genetic Future.* Boston: Houghton Mifflin Co, 2002.

Supper, Martin. “A Few Remarks on Algorithmic Composition” in *Computer Music Journal*, 25:1, pp. 48-53, Spring 2001. Massachusetts Institute of Technology.

Tanzi, Dante. “Observations about Music and Decentralised Environments” in *Leonardo.* Vol. 34, No. 5, pp. 431-436, 2001.

Tokui Nao & Hitoshi Iba. *Music Composition with Interactive Evolutionary Computation. http://www.miv.t.u-tokyo.ac.jp/~tokui/research/iec-music/*, July, 2002.

Toop, David. *Ocean of Sound: Aether Talk, Ambient Sound and Imaginary Worlds*. London: Serpent’s Tail, 1995.

Taylor, Mark C. *Hiding.* Chicago: The University of Chicago Press, 1997.

Vaggione, Horacio. “Some Ontological Remarks about Music Composition Processes” in *Computer Music Journal*, 25:1, pp. 54-61, Spring 2001. Massachusetts Institute of Technology.

Turner, Jeremy. *The Microsound Scene: An Interview with Kim Cascone.*
http://www.ctheory.net/text_file.asp?pick=322, June 2002.

Van Dijk, Jan. *The Network Society.* London: Sage Publications, 1999.

Verotsko, Roman. *Epigenetic Painting: Software as Genotype, A New Dimension of Art.* http://www.verostko.com/epigenet.html, June 2002.
------. *Algorithmic Art: Composing the Score for Visual Art.*
http://www.verostko.com/algorithm.html, June 2002.
------. *Algorithms and the Artist.*
http://www.verostko.com/alg-isea94.html, June 2002/

Vesna, Victoria. “Toward a Third Culture: Being in Between” in *Leonardo.* Vol. 34, No. 2, pp. 121-125, 2001.
------. *Database Aesthetics: Of Containers, Chronofiles, Time Capsules, Xanadu, Alexandria and the World Brain.* http://time.arts.ucla.edu/AI_Society/vesna_essay.html, January 2002.

Virilio, Paul. *Speed and Information: Cyberspace Alarm!*
http://www.ctheory.net/text_file.asp?pick=72, June 2002.

Ward, Adrian. *Auto-Illustrator User-Guide*. London: Signwave, 2002.
------. *How I Drew one of my Pictures*,
http://www.generative.net/papers/autoshop/index.html, Feb, 2002
------. *The Aesthetics of Generative Code,*
http://www.generative.net/papers/aesthetics/index.html, Feb, 2002.
------. *How I Drew one of my Pictures: or, The Authorship of Generative Art.*
http://www.generative.net/papers/authorship/index.html, Feb, 2002.
------. *Signwave Auto-Illustrator: Artistic Statement.*
http://www.adeward.com/swai.html, July, 2002.

Waters, Simon. *The Musical Process in the Age of Digital Intervention.* Webarticle:
http://www.ariada.uea.ac.uk/ariadatexts/ariada1/content/Musical_Process.pdf. July, 2002.

Weber, Eugen. *Movements, Currents, Trends: Aspects of European Thought in the Nineteenth and Twentieth Centuries.* Lexington: D.C. Heath and Company, 1992.

Webb, Paul. *Performance Systems of Live Interactive Agents.* http://www.robotsoftware.co.uk/live.html, July 2002.

Weibel, Peter and Druckery, Timothy (eds.) *net_condition: art and global media*. Cambridge: The MIT Press, 2001.

Weil, Benjamin. *Readme.txt,* http://www.afsnitp.dk/onoff/Texts/printerfriendly/weilreadme.txtpr.html, July 2002.

Weinberg, Gil & Seum-Lin Gan. "The Squeezables: Toward an Expressive and Interdependent Multi-player Musical Instrument" in *Computer Music Journal*. 25:2, pp. 37-45, Summer 2001. Massachusetts Institute of Technology.

Whitelaw, Mitchell. "The Abstract Organism: Towards a Prehistory for A-Life Art" in *Leonardo.* Vol. 34, No. 4, pp. 345-348, 2001.

Wishart, Trevor. *On Sonic Art.* Amsterdam: Harwood Academic Publishers, 1996.

Wilson, Stephen. *Information Arts: Intersections of Art, Science and Technology.* Cambridge, The MIT Press, 2002.

Winograd, Terry & Fernando Flores. *Understanding Computers and Cognition.* Norwood NJ: Ablex, 1986.

Winston, Brian. *Media, Technology and Society, A History: From the Telegraph to the Internet.* London: Routledge, 1998.

Wishart, Trevor. *On Sonic Art*. Amsterdam: Harwood Academic Publishers, 1996.

Xenakis, Iannis. "Tutorial Article. Determinacy and indeterminacy" in *Organised Sound* 1(3), pp. 143-55. 1996. Cambridge University Press.

Young, John P. *Networked Music: Bridging Real and Virtual Space.* http://www.netmuse.org/intro.html, July 2002.

Z, Pamela. *A Tool is a Tool.* http://www.pamelaz.com/tool.htm, July 2002.

Topography

[artists, artworks and companies relevant to the thesis] – (this list could be endless, so I only provide links that are used directly)

Adobe - > http://www.adobe.com/

Aesthetics + Computation Group - > http://acg.media.mit.edu/

Antoine Schmitt - > http://www.gratin.org/as/

Benjamin Fry - > http://acg.media.mit.edu/people/fry/

Celestino Soddu - > http://www.celestinosoddu.com/index.htm

David Rokeby - > http://www.interlog.com/~drokeby/

DeKam - > http://node.net/

Delter - > http://www.n-gon.com/delter

Eduardo Kac - > http://www.ekac.org/

Florian Cramer - > http://userpage.fu-berlin.de/~cantsin/homepage

Golan Levin - > http://www.flong.com/

ixi-software - > http://www.ixi-software.net

Joanna Maria Berzowska - > http://web.media.mit.edu/~joey/x/x.html

Jodi - > http://map.jodi.org/ & http://www.jodi.org

John F. Simon - > http://www.numeral.com/

Karl Sims - > http://www.genarts.com/karl/

I/O/D - > http://www.backspace.org/iod/

Knowbotic Research - > http://www.t0.or.at/~krcf/

Longplayer (Jem Finer) - > http://www.longplayer.org/

Macromedia - > http://www.macromedia.com/

Manfred Mohr - > http://www.emohr.com/

Mark Napier - > http://www.potatoland.org/

Max/MSP - > http://www.cycling74.com

meta - > http://www.meta.am/

Netochka Nezvanova - > http://www.eusocial.com/

n_gen - > http://www.n-generate.com/

Native Instruments - > http://www.nativeinstruments.de/

Nullpointer - > http://www.nullpointer.co.uk/

Pamela Z - > http://www.pamelaz.com

Philip Galanter - > http://www.philipgalanter.com/

Postmodernism Generator - > http://www.elsewhere.org/cgi-bin/postmodern/

Pure Data - > http://www.pure-data.org/

Raymond Kurtzweil - > http://www.kurzweilcyberart.com/

Reconnaitre - > http://www.reconnoitre.net/gb_uf/index.html

Roman Verotsko - > http://www.verotsko.com

Sensorband - > http://www.sensorband.com

Shape of Song - > http://turbulence.org/Works/song/

Signwave - > http://www.signwave.co.uk/

Simon Penny - > http://www-art.cfa.cmu.edu/Penny/

Slub - > http://www.slub.org/

Soda - > http://www.soda.co.uk/

Stanza - > http://www.amorphoscapes.com/index.html

Supercollider - > http://www.audiosynth.com

Turux - > http://www.turux.org/

Weatherplayer - > http://www.weatherplayer.com/

William Latham - > http://www.artworks.co.uk/index2.htm

[festivals and institutions]

Ars Electronica - > http://www.aec.at/

Bitforms - > http://www.bitforms.com/

Cybersonica - > http://www.cybersonica.org/

Gratin - > http://www.gratin.org/

Net_condition - > http://on1.zkm.de/netCondition.root/

Nettime - > http://www.nettime.org

Numer - > http://www.numer.org

Read_Me - > http://www.macros-center.ru/read_me/abouten.htm

Rhizome - > http://www.rhizome.org

Singlecell - > http://www.singlecell.org/

Steim - > http://www.steim.nl/

Sonar - > http://www.sonar.es/

Sonic Arts Network - > http://www.sonicartsnetwork.org/

Transmediale - > http://www.transmediale.de/en/02/

Vuk Cosic - > http://www.ljudmila.org/~vuk/

Whitney Biennal - > http://www.whitney.org/2002biennial/

V2 - > http://www.v2.nl

ZKM (Zentrum Für Kunst und Medientechnologie) - > http://www.zkm.de/