Database Music

A History, Technology, and Aesthetics of the Database in Music Composition

by

Federico Nicolás Cámara Halac

A dissertation submitted in partial fulfillment

of the requirements for the degree of

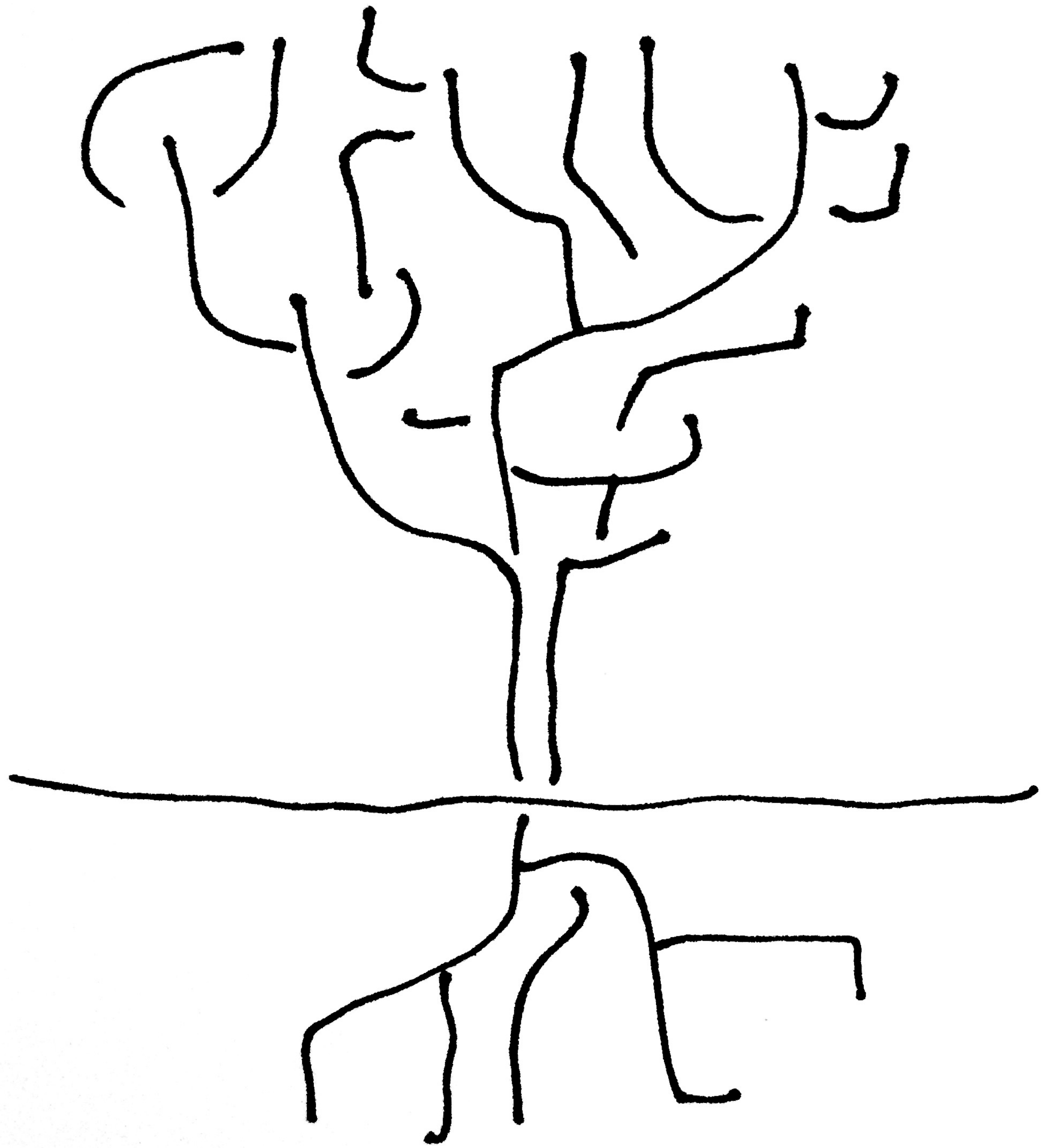
Doctor of Philosophy

Department of Music

New York University

May, 2019

Jaime Oliver La Rosa



Dedication

For my mother and father, who have always taught me to never give up with my research, even during the most difficult times. Also to my advisor, Jaime Oliver, without his help and continuous guidance, this would have never been possible. Finally to my loving wife, Aye, whose love and support helped me make it through the sleepless evenings.

Acknowledgements

I would like to thank my advisor, David Ledesma, for his role in inspiring this project, as well as his commitment to introspection, and to reflecting upon and exploring meaningful issues in clinical psychology. I am also indebted to committee members John Hilaire and Michael Douglas for their ongoing guidance and support, as well as their frequent feedback (often in the form of exceedingly prompt e-mail responses), at every stage of this project. This dissertation could not have come to fruition without the help of Del Aware and Barney Rubble, who offered balanced yet insightful, thought-provoking input. I am also everlastingly grateful to Jill Pullman, for always being available to listen and empathize, as well as to my husband John Doe, for his endless tolerance and his helping me maintain hope that I would indeed finish this project! I would also like to thank my parents, Paul and Mary Williamson, who inspired and nurtured my interest in observation and clinical judgement from a very young age. Finally, many thanks to all of the undergraduates who so patiently offered their time and clinical judgements.

Preface

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Abstract

The aim of my dissertation is to understand music composition as a database practice. To carry out this proposition, I will trace the conditions for the possibility of the above identity —i.e., composition = database—, and analyze the aesthetic implications this identity brings forth. In resonance with the extensive literature referring to the intersection between music and computers, I will situate this discussion around the broader question of technology's agency in art. Specifically, I will analyze this agency in terms of the relationship between image and sound. Finally, as an instantiation of the propositions above, I will develop an open-source library for multimedia composition that combines computer vision and timbre analysis algorithms to generate a database of descriptors, interpreting them as nodes in a network suitable for automated navigation.

This dissertation goes through the state of database art, the use of computers in music and art, the collaborative aspect surrounding computers, the immateriality that percolates through the arts as data, the terminological struggles in the definition of data-based media, the possibilities of new linkages between different media through data, the arbitrary world of the composer in the midst of an emergent, autopoietic, bottom-up art-world, the ubiquitous architecture enabling all of it, the resonating self in between, the non-human agency, the software communities, and the topology of the networked world. Data: database aesthetics Computers: memory, algorithm, and automation Immateriality: multimedia and new media I will briefly introduce and contextualize these key concepts upon which my dissertation will be organized. To begin with, the feature that must be addressed first in this discussion is what scientists, media theorists, artists, and computers

know as data.

The constantly changing technological, social, and global context of today's "data revolution" has placed data at the center of media research. Social media theorist José van Dijck claims that this revolution, or "datafication" of today —"transforming all things under the sun into a data format and thus quantifying them"— is at the "heart" of the networked world (van Dijck 2017:11). Thus, data (i.e., the plurality of datum as "something given" from the world) is accumulated in archives, or data-bases, and accessed for infinite purposes from anywhere, anytime (Lovejoy 2011:13). Data-driven practices have been indeed present in the emerging information society since the nineteenth century, as evidenced by the evolution of statistics (van Es; Shaefer 2016:13). However, the question of aesthetics in data-driven art has only recently been assessed, and questions regarding the agency of the database in art have yet to be fully considered, particularly in the field of music composition. According to artist Margot Lovejoy: "the hybrid and collaborative nature of art using digital technologies as a medium... questions established practices of institutions such as museums, funding organizations, and the art world at large" (Lovejoy 2011:26). Composers, in turn, have been working with computers —and data— since the beginning of the computing era. Therefore, I will analyze how this ubiquitous, non-human, data-based model of the computer itself shapes composition, which is a practice traditionally conceived to belong to single authors —composers— whose technique or aesthetic intuition is generally valued.

Before delving into a proper analysis of this data-based model, and into thorough investigations of its aesthetic consequences, I would like to take a detour and provide a glimpse of how data-driven practices have been conceptualized so far. Over the past 20 years the data-driven practice of art has been conceptualized under the term Database Aesthetics. One of the first proponents of the database as a cultural form is Lev Manovich with his (2001) The Language of New Media. In Lovejoy's words: " [Manovich] discusses the database as the center of the creative process in the computer age. He sees it as a new symbolic form [...] He speaks about modern media as a battleground between a system which relied first and foremost on the narrative form and the new culture based on the database." (Lovejoy 2004:168). In resonance with the "end of grand Narratives," Manovich considered the database as a "new way to structure our experience of ourselves and of the world." In the context of what he termed the emergence of the "metamedium of the digital computer" (Manovich 2001:6), he understood new media as digital media: "All existing media are translated into numerical data accessible for the computer. The result: graphics, moving images, sounds, shapes, spaces, and text become computable, that is, simply sets of computer data. In short, media become new media" (italics mine, p. 25). Put differently, for Manovich, the computer became useful only as a "remediation machine," a translator of old media into new —digital— media (Manovich 2017:58). Thus, the structured collection of media as data —the database— gained its cultural agency. Manovich suggested that the database encapsulated creative impulses that artists revealed by creating interfaces to databases as their artwork. However, I consider that Manovich's database is not the complete picture, since it remains on a certain level of analysis that, while certainly providing insight into digital practices, prevents further inquiries into deeper capabilities of the computer. These capabilities are dependent upon the computer's inherent structure, which in turn is only possible with John Von Neumann's notion of storage. I have so far proposed that the literature on data-driven art stands on the grounds of software as a given. In other words, while software is held as the gate to this data-driven art world, there is very little insight into the actual 'gate' itself. I will claim throughout this dissertation that the underlying architecture of the computer, which allows for the notion of software, is taken for granted. This model is what is known as the von Neumann architecture, which is an implementation by John Von Neumann of Alan Turing's theoretical Universal Computing Machine: "Inasmuch as the completed device will be a general-purpose computing machine it should contain certain main organs relating to arithmetic, memory-storage, control and connection with the human operator. It is intended that the machine be fully automatic in character, i.e. independent of the human operator after the computation starts" (von Neumann; Goldstine; Burks 1946:399).

I understand this model as follows:

world –> computation <– data –> memory –> world

In other words, in the Von Neumann model, the world is represented as data to be computed and stored: the datum is inscribed as discretized information, and traversed by a set of algorithmic —and automated— computations. The flow, represented here in the form of arrows, is what I consider the central property of data itself, and it is what I want to stress first of this model. I will argue that it is the Von Neumann model what predetermines this flow and, therefore, what needs to be analyzed when identifying music composition with database practices. This model is a fundamental aspect of the computer that still determines how we use it and what we can do when making music with it. The revolution of data cannot be understood properly without a closer analysis of the structure holding it. In a brief introduction on the computer's history, Manovich draws a "historical loop" with Konrad Zuse's (1936) computer prototype which used a 35mm movie film to store digits, and the fact that computers now make movies (Manovich 2001:26). Given that Manovich's intention is to "describe and understand the logic driving the development of the language of new media," (p. 7) he provides "general tendencies of a culture undergoing computerization" (p. 27) under a set of five principles: "numerical representation," "modularity," "automation," "variability," and "transcoding". What these "principles of new media" point to, in truth, is some aspects that are available by the underlying architecture of the digital computer (i.e., memory, computation, and data flow). What is important to note here, however, is the memory module of the computer, which is what enables this architecture in the first place and, in turn, Manovich's principles. On one hand, the storage unit of the computer allows to record information which is what constitutes Manovich's "new media". On the other, the same storage unit holds the algorithmia, that is, the functions, operations, or routines, which are used to access, process, etc., that information. Then, it is the flow of data (as instructions and information) with the processing unit what is at play, and what is mostly accesible by way of an interface (input and output devices). This model is what defines all computer practices, and in its center exists a different database of functions. I have proposed that it is the memory and its storing of instructions and information what enables the computer as such. This is certainly not new considering that hardware and software have traditionally formed the computer as we know it, and have developed somewhat equally over the years. However, what I argue here is that the model itself has remained conceptualized as a black box, and therefore it remained as such. I do not attempt here to enter into a discussion of whether this black box should or should not be opened, for that would take this dissertation into a different path. However, what I would like to point to is the fact that these instructions stored in memory, which constitute the algorithmic essence of digital logic, are in turn a database of functions. Therefore, the agency of the database in art cannot be considered without knowing this logical foundation: what do these databases represent if the case is that they do not constitute information of the world? What is different about them? How do they shape practices such as music composition? Before entering into this discussion, however, the automatic quality of the computing machine needs to be briefly mentioned.

Since any container of information can be considered a database, curator Christiane Paul proposes that the database, "[as a] structured collection of data [that] stands in the tradition of data containers, such as a book, a library, an archive [has] become an essential form of cultural organization and memory" (Vesna 2007:175-8). However, the process of traversing the database (the algorithm logic, e.g. a sorting process) is always automated. Furthermore, this automation is carried out through increasingly fast computations, providing us with "unprecedented" new aesthetic possibilities (Manovich 2002:217).

I have thus far introduced what I consider key concepts in the underlying structure of the computer (i.e., data, memory, algorithm, and automation) which constitute the von Neumann Architecture. Furthermore, I contextualized how data practices have been grounded upon a superficiality caused by the notion of black-box software, preventing deeper consideration of the capabilities of the computer. Finally, I have proposed that all computation depends on a foundational database of algorithms. I will now bring forth one aspect of digital practices that I consider to be one step further in the investigation of the essence of the aesthetic repercussions of the database in music composition. In an article referring to digital humanities, Katherine Hayles briefly comments on the cultural aspect of the database, as studied by Vesna and Manovich, with an emphasis on the collaborative aspect. In resonance with Sharon Daniel, Hayles claims the database allows "new kinds of relationships between a project's designer and her interlocutors" (Hayles 2012:55). Therefore, it is through the interface to the database that a tangible agency is granted to the participant, resulting in a collaboration where authorship is decentered; hence, as Manovich claimed, the interface itself became the artwork. Before analyzing in depth this new conceptualization of authorship within art, and music composition in particular, I would like to point to an underlying paradox here: the tangibility of the digital interface. Hayles portrays the database as "embodying different cognitive, technical, psychological, and artistic modalities" (italics mine, ibid 56). As I have suggested above, the database is simultaneously archival, creative, and aesthetic impulses, and a driving force in the work. However, the database and its elements are immaterial, consisting essentially of digital information (i.e., binary digits, or bits). What is more, despite the fact that storage is ultimately physical, data can easily move to and fro, from one storage space to another, immediately, in a state of constant flow: data is as immaterial as it is dynamic.

Therefore, since the artwork becomes the interface and thus it is available to touch, what constitutes and enables this embodiment? What is the nature of this immaterial agency? Furthermore, to what extent can this agency be understood in terms of the relationship between image and sound? In order to further advance in the analysis of what will be portrayed later on as a force, it is necessary to begin contextualizing the terminological discussions that are at play. In what follows I will underline some distinctions between multimedia and new media. In my dissertation, I will explore multimedia art as site to analyze technology's agency in art, with a focus on the relationship between image and sound. In art, multimedia generally refers to the integration of sonic and visual elements, often including text, within the context of a single work. Manovich considers cinema to be the "original modern 'multimedia'" (Manovich 2001:51).

Mark Hansen, in turn, criticizes Manovich's cinematic approach to new media in a tripartite way. First, for Hansen, new media extend beyond cinematic domination. Hansen considers Manovich's non-critical views of the Human-Computer Interface (e.g., its promotion of "immobility" on the part of the user) as an attitude that prevents the "exploration of unheeded or unprecedented alternatives" that digital media have to offer (Hansen 2006:34). Second, Manovich's cinematic metaphor itself is, for Hansen, not topologically suitable for digital image as such, since the latter is polymorphous: "lacking any inherent form or [geometric] enframing, data can be materialized in an almost limitless array of framings" (p. 34). Therefore, the aesthetic deployments of the digital image are thus restricted by the cinematic medium. Finally, Hansen claims that Manovich fails to provide "any account of the significant role accorded the body as the 'operator' of an alternative, post-cinematic interface with data' (italics mine, p. 44), therefore neglecting to properly consider the new possibilities for experience that digital media bring forth.

Jan Simons claims that the term multimedia has been restricted to the digital, computer-based environment distinguishing between multimediality ("a single information environment [that] offers the user simultaneous access to different sorts of media") with new media, which for him consists of "a repositioning and redefinition of old media: what Jay David Bolter and Richard Grusin call 'remediation"' (Simons 2002:236). Bolter and Grusin, in turn, base their concept of remediation by quoting Marshall Mcluhan (1964) Understanding Media: "the 'content' of any medium is always another medium" (Bolter; Grusin 2000:45). They argue that in being "the representation of a medium in another," remediation is central to new digital media (45). This is attuned to Manovich's description of old and new media. On the other hand, Kyong Chun argues that although it relies on computerization, new media is not just "digitized forms of other media (photography, video, text), but rather an interactive medium or form of distribution as independent as the information it relayed," and therefore, new media stands for "fluid, individualized connectivity, a medium to distribute control and freedom" (Kyong Chun 2016:1). In what preceded, I have briefly referenced the ubiquity of the use of computers, as evidenced by the recent focus on data in media studies and in the arts. I have proposed, however, that software has been held as the horizon throughout the literature on data-driven art. In contrast, and not without anticipating what this next section will reveal, I have pointed that composers have been working with data since the beginning of the computing era. As I have proposed in the introduction, music composition is in itself a database practice. Conversely, database practices can be thought of as composition practices. This is to say that they both constitute a construction process that is contingent upon the dynamic relationship between instruction and information. The Von Neumann model is what allows for this construction to occur in the case of the database, where the data is an immaterial and dynamic element in constant flow, automatically and algorithmically gathering in the form of information or instruction. In the case of composition, the vibrant and dynamic energy that resonates in the form of information, instruction and, ultimately, image is sound. This last condition is what I will develop and trace in the following section. Finally, the following exploration, in conjunction with Part I, will constitute the conceptual groundwork for the model that I will develop in the last section.

Sound: material, ILLIAC 1, MUSIC-N, stochastics, composition, and authorship Image: sonification, synchresis, resonance, and force Non-human Agency: actor-network theory Despite the fact that sound has become integral to new media works (Lovejoy 2011:18), visual approaches continue to dominate theorizations of database aesthetics. Therefore, there is a significant gap in the literature regarding the role of sound in new media. I will go through different situations in twentieth and twenty-first century music/sound art —such as improvisation, chance operations, stochastics, algorithmic composition, real-time composition, soundscape composition, etc.— in which composers and artists have consciously engaged with non-human agency in their work.

I have contextualized multimedia at the intersection of the visual and the sonic, and now I will focus on the sonic. There are several authors I will refer to subsequently for that matter: Timothy Murray, Brian Kane, Michel Chion, and Carla Scaletti. Before entering into this discussion of the role of sound and non-human agency, I would like to first bring back to surface the notion of composition and, particularly, the notion of musical material. In Edgar Varese's words:

“The computing machine is a marvelous invention and seems almost superhuman. But, in reality, it is as limited as the mind of the individual who feeds it material” (italics mine, Varese 1966:20). Data as musical material for composition is, for Varese the key, or “reality” of the composition. As with musical material, data is subject to, and depends on manipulation by the composer. Thus, in being instrumental to composition, data defines composition at least in two aspects: first, it constitutes what the composition is, and second, it delimits how the composer can compose. What I consider interesting about Varese’s quote, is that in an attempt to ameliorate the repercussions, and somehow demystify these machines, Varese grants them aesthetic agency: “like the computer, the machines we use for making music can only give back what we put into them” (idem). I claim what Varese’s words most strongly suggest is a need for the emphasis on the ‘we’. I consider this emphasis a reaction to the presence of the computer in music. Therefore, implied in his statement is the fact that computers do in fact shape music composition. Similarly, by indexing a lattice of frequencies and time intervals, Hiller and Isaacson, share authorship and agency with the ILLIAC 1.

Although without a direct mention of the Von Neumann model as such, L. A. Hiller and L. M. Isaacson (1955-7) composed the Illiac Suite with the ILLIAC 1 computer. In their account, they describe the computer’s architecture, its capability for logical operations, and immediately proceed to its use. In particular, Hiller and Isaacson used the Markov Chain Monte Carlo method to obtain an “ordering process in which specified musical elements are selected and arranged from an infinite variety of possibilities, i.e., from chaos” (Hiller; Isaacson 1959:16). Musical material was generated by statistical methods and probability matrices: far from considering sound as such, they had to simplify. By way of indexing pitch and time they developed several “experiments” which led to a score for string quartet. Despite the underlying discussions of the system (tonality, twelve-tonality, etc.), or even the possible impositions by the heads (or the heads of the heads) of their institution (i.e., the exploration of the potential of the Monte Carlo Method), they had to punch data for a computer. What I am trying to suggest here, is that Hiller and Isaacson’s compositional actions were shaped by the computer itself, and hence their music cannot be analyzed without the ILLIAC 1 itself, or without the Williams tubes in which their data shifted in and out from, or without the cards used for inputting the data in the first place. This relationship between the composer and the computer has gone under several changes and developments over the years, Max Mathews, an electrical engineer working in Bell Labs granted the composer and the computer a different form of mediation: he developed the first computer music language in the IBM 704. Mathews describes the architecture of this program in three stages of data flow —put simply, as reading, sorting, and executing— which are modeled in turn from three elements of music tradition: the score, the metronome, and the instrument. Thus, and by allowing for the synthesis of sound, his language brought the computer to a whole new sphere of potentiality. The data to be inputted (the material) became 'interpreted' by the computer, and so the type of instructions (as instructions for synthesis) changed. From this, two consequences emerged. First, instructions to generate materials were dependent upon the capacities of the language (FORTRAN first, then ultimately, C). Like in the case of the ILLIAC 1, material was generated. Second, what the MUSIC-N languages brought forth was a closer relationship between the acoustical result of the generated material, and thus the programmer, the computer, and the audio signal coming out of the speaker relationship became a much stronger one. Therefore, considering composition as database practice, in the MUSIC-N languages this identity is most evident. However, as sound became a much closer reality to the programmer and to the computer, the problem of composition became a heavier one for the composer, and this weight can be assessed in turn in the work of Xenakis.

Iannis Xenakis introduced stochastics as a reaction to what he termed "chance—improvisation" (Xenakis 1992:39). In a series of French publications between 1955-1965, he argued for precision, calculus, as something that would furnish "a powerful method for reasoning and enrichment of sonic processes." In a two-front criticism, chance was not calculated and improvisation was a substitution of the author, and a betrayal of the question of choice. Thus, in searching for meaning in music, Xenakis turned to probability distributions. However, later in his life he considered this inclination as a temporary measure: the role of the composer was equally present, “since,” as he explains, “each probability function has its own finality and therefore is not a nothing” (260). In other words, the stochastic process renders an actual thing, material, which needed to be given shape, form, etc., and using this material, in Horacio Vaggione’s words, is arbitrary: “It is always problematic to utilize global causal laws (e.g., stochastics) in music composition, if their automatism is not compensated by compositional choices concerning other levels of articulation. This is why Xenakis, after being concerned with Markov chains, subsequently adopted a consistent silence about his more recent compositional procedures, a silence not broken by his claim that he has introduced an ‘arbitrary’ (that is ‘composed’) manipulation of the data provided by his stochastic canons.” (Vaggione 1993:97) Arbitrariness is what best describes so far the role of the composer. In a similar key, Curtis Roads, while acknowledging the attractiveness of so-called bottom-up systems —which will resurface farther down this text— and opposing them to preplanned forms (i.e., top-down forms, like the sonata), argues for the freedom for the “creative composer”: “The bottom-up strategy can be fascinating, partly because its results cannot always be predicted in advance. On the other hand, why limit the scope of compositional decisions to a single time scale?” Roads, in resonance with Vaggione’s multi-scalar approach to time, holds the figure of the composer in opposition to the global, statistical processes upon which Granular Synthesis (or Xenakis, or Hiller and Isaacson’s work) is built. He continues: “To navigate the widest possible zones of creativity, the creative composer wants to float freely across time scale boundaries” (italics mine, Roads 2004:330).

In following this arbitrariness, the composer needs to choose, and hence selection and choice become the composer’s method. This method has always existed in music composition and it is certainly not the purpose of this dissertation to develop an account, let alone an investigation of this method as such. However, I argue that there is a contradiction regarding the object of selection, that is, the material: when the composer is faced with material, the composer must choose. What if this material is not something the composer can face, let alone choose? What if the composer is not anymore faced with material, but with an immaterial force that escapes selection? In other words, considering material as data, I argue, constitutes a different action on the part of the composer, and changes the aesthetic dimension of the composition.

I will now introduce the concepts of autopoiesis, emergence, and bottom-up, and set them in contrast to the notions of the composer and material thus far exposed. Alan Garfinkel exemplifies emergence as present in the formation of patterns of physical phenomena: “if a homogeneous fluid is subjected to sufficient heat or motion, forms develop” (Garfinkel 1987:182). Sharon Daniel, in turn, brings the notion of emergence into the social —and the art— world, when she considers the autopoietic collapse of the subject and system (Daniel 2007:144). Daniel explains that this collapse is generally understood in the arts —and similarly by Roads— as the bottom-up mentality. However, for Daniel this mentality serves as a model for rethinking authorship.

Addressing directly with the question of authorship in computer assisted composition, and embracing a bottom-up approach, composer George Lewis designed a program (i.e., composed Voyager) that “analyzes aspects of a human improvisor’s performance in real time.” This program uses the real-time analysis to “guide” the composition, and it “generates” complex responses both to the musician playing and to its own internal processes: “control of musical process is shared among players; interplayer communication takes place without necessarily involving a central authority. Local decisions taken by individual players percolate up to the global level, at which the overall form is maintained” (italics mine, Lewis 2000:33-7). This upwards percolation of the formation of form is what I consider an example of the bottom-up approach: based on local interactions (i.e., at the bottom level), the global form emerges. Furthermore, with Voyager, Lewis questions the taxonomy of the inherent categories with which we refer to music making (“composer,” “improviser,” and “culture”), by acknowledging formal agency (in the traditional sense of musical form) to a computer program that is to be improvised with, and he claims that this is the composition.

From the ILLIAC 1 to Voyager, I consider that the above delineation accounts for an adaptation on the part of the composer. The computer as such has changed, indeed, but its technical evolution may not be as significant to computing as the effect the computer has had in the field of music composition. I have suggested that composition is a single author practice, in which the composer's technique or aesthetic intuition is traditionally admired. This is no longer the case since the computer age: now the composer and the computer —and here I bring back the notion of the database— both share the weight of composition. Put differently, composition is imagined differently when computers are involved. At this point, I will begin articulating this imagination in the form of listening, and thus introduce sonification, synchresis, resonance and force as the last elements in this discussion of what I will understand as the imaginary composition.

Composer and programmer Carla Scaletti, defined sonification as "a mapping of numerically represented relations in some domain under study to relations in an acoustic domain for the purpose of interpreting, understanding, or communicating relations in the domain under study" (Scaletti 1992:224). Therefore, sonification is —like its visual counterpart, i.e., visualization— what is closest to what Vesna and Manovich have theorized of data-driven practices. However, the role of sound here diverges from that of music. While 'understanding' and 'usefulness' have been constitutive elements of 'intention' and 'purpose' in sonification as a practice, this is not the case for music, where signification is not such a crucial element in the matter. Notwithstanding, artist Andrea Polli, when referring to her sonification practices, claims that the database is neither a static subject "on which an artist projects meaning," nor material "clay" to be "transformed," but a "catalyzing factor in the conversation…In order for the expression of the data to be heard, we have to be listening" (Polli 2016:7). I therefore consider sonification to be a key element in the discussion of the intersection of the image and sound, since it is a data-driven sound practice that allows for linkages between the domain of the visual and the acoustic domain, as Scaletti's definition permits. This is to say, simply, that a digital image can be sonified and therefore brought to its acoustic presence. However, in what follows, this relationship —of image and sound— will be understood as taking place in experience, rather than in an already modelled sonificating map.

That is, instead of sonifying a distribution of data and thus forging the sonic from the data, the merging of the visual and the sonic happens in the body.

Based on Pierre Schaeffer's listening modes, Michel Chion's (1994) Audio-vision addresses the "contract" between sound and image. Primarily focused on cinema, Chion's text coined the term synchresis to describe "the forging of an immediate and necessary relationship between something one sees and something one hears" (Chion 1994:5). This principle of synchresis allows for what Chion calls the "added value," which constitutes sound's creation of a "definite impression" by way of providing "expressive and informative value" to the image (i.e., to what is inside the frame), hence "enriching" it. Therefore, while acknowledging the expressivity of sound and its informational power, Chion's language relegates the overall aesthetic agency to this enriched, cinematic image. This relationship between sound and image, however, leaves the realms of the screen —and of the cinematic medium itself—, and it is placed in between the eye and the ear. The underlying need for this relationship, moreover, is itself dependant upon the notion of signification. In what follows, this 'understanding' will take yet another turn with Brian Kane and his analysis of Jean Luc-Nancy's Listening.

In a recent article, Brian Kane considers Nancy's Listening as "the question of the subject. . . posed anew, outside of the horizon of the phenomenological subject" (Kane 2012:446). This statement raises a different kind of questioning and shifts the discussion of the in-between of the eye and the ear into the subject itself. Moving beyond Pierre Schaeffer's notion of signification, Kane explains, "Nancy calls his subject 'a resonant subject' because both the object and subject of listening, in his account, resonate." Nancy's resonance can be understood as "an infinite sending and re-sending [, and] the structure of the subject and of sense" (p. 445). What I will point to now of this resonance, for the purposes of my dissertation, is the kind of feedback loop it implies. The subject is seen as yet another empathetic node in its relationship with the world. I understand this idea as intrinsic to the relationship between music composition and database practice, namely in the sense that the composer and the computer constitute resonating nodes in themselves, and thus share aesthetic agency. The abovementioned concept of synchresis can be further expanded and criticized in tune with Kane's reading of Nancy: instead of an addition of value, synchresis can be understood as resonating synchresis, in which the visual and the sonic, as empathetic nodes, engage in vibrant relation with each other. Further, Timothy Murray brings forth another concept of Nancy that will also be analyzed throughout this dissertation, particularly in conjunction with the previously introduced immateriality of the database: the immaterial function of the image. Also quoting Nancy, Murray understands immateriality as "the expansion of 'sensation' at the expense of signification" (Murray 2011:139). Murray continues: "as the carrier of both 'intimate force' and 'intensity,' the image, even when musical, choreographic, or cinematic, does not so much 'represent' as 'activate' and 'intensify' representation and its 'touch.' It is this 'immaterial' force that lends the image its 'impalpable nature' and that constitutes that which it 'images"' (p. 149).

I have thus far provided a glimpse of a data-driven practice of music (sonification) which calls for notions of understanding and expression. I have briefly introduced Nancy's concepts of resonance and immateriality as read by Kane and Murray. Further, I have begun to reformulate Chion's synchresis as a way to understand the resonating aspect in the relationship between the eye and the ear. Furthermore, this resonating synchresis I claim that constitutes the essence of the identity of database and composition, computer and composer. These concepts will be developed throughout this dissertation, and I claim, what they begin to describe is the non-human agency the database has had in music composition. "We have actually three dimensions in music: horizontal, vertical, and dynamic swelling or decreasing, I shall add a fourth, sound projection —that feeling that sound is leaving us with no hope of being reflected back, a feeling akin to that aroused by beams of light sent forth by a powerful searchlight —for the ear as for the eye, that sense of projection, of a journey into space." (Varese 2004:18)

Varese's fourth dimension is a suitable metaphor today to question composition itself. His poetical notion of sound and light in constant movement into space, and away from ourselves, is comparable to the notion of materiality fading away into immateriality; the composer, like the composition, flows through the world collapsing with the database, forever changing music composition, and releasing a resonating and imaginary aesthetic force. In order to question composition itself, and to assess the aesthetic implications of this change, I will take a philosophical leap and, in consideration of the situations above, I will argue that what is at stake in the field of music composition today, through the use of technology at large, and of the database in particular, is the transformation of culturally dominant models of composition. Composition can no longer be thought of as a practice centralized in the composer. In thinking so, I argue, the composition is embedded in an opaque blob that dissuades apprehension. This is why I propose here that composition can be imagined differently. Bruno Latour's network theory can certainly help. Latour's network is in a sense —like sound, and like Varese's fourth dimension— pure expansion: "the surface 'in between' networks is either connected —but then the network is expanding— or non-existing" (Latour 1990:6). What is significant about this concept, moreover, is that it aims to replace spatial metaphors —high-low, in-out, far-close, global-local, etc.— with "associations and connections." For Latour, spatial metaphors preclude the study of "society-nature" (p. 6). I do not attempt to explain what he means by society-nature, for that would take this dissertation into a slightly different path. However, what I would like to point to here is the fact that the epistemology of the network, in my opinion, is a step further in the understanding of social practices, and particularly for this dissertation, music composition. Actor-network theory has been described as the "distribution of agency across a heterogeneous network of human and nonhuman actants" (Grusin 2015:xv). Thus, within a music composition, each actant brings forth its aesthetic agency. In connection with the previously introduced notion of resonance, therefore, these actants can also be thought of as resonating through every aspect of the music work. It is in this resonance that the figure of the author is expanded as far as we can listen. My purpose on this dissertation is to explore the aesthetic implications of this expansion, and to unveil this resonance as a new form of imaginary composition.

Since agency is shared among the entire network of "actants" in the composition (software, instrument, database, algorithm, composer, performer, etc.) how can we think of authorship in composition today? What is a composer? Second, the notion of musical material as data allows for our understanding of composition itself to no longer depend solely on the composer. This opens up the field to new kinds of collaboration; specifically, this opening helps to reduce transdisciplinary gaps, and to generate new linkages between the visual and the auditory. Third, the notion of data as immaterial has aesthetic resonances in the music work, and these resonances need to be further analyzed in closer detail. How is this force of imaginary composition different? Finally, much in the same way the computer's programs and architecture shape art and composition, how and to what extent have artists and composers influenced in turn the field of computer science? How does a music itself shape the computer? I can only invite the reader to continue this questioning, to further investigate this transdisciplinary feedback, and to imagine different aesthetic dimensions that the resonating network brings forth.

In what follows, I will briefly contextualize the proposed software library, and introduce its main functionalities. As of now, this model is at the very early stages of its development, and what will be presented here can be thought of indeed as an imprecise algorithm. This vagueness is in part intentional, since I consider that this dissertation will resonate through the resulting library, and therefore I have to develop hand in hand both the text and the model.

Multimedia Composition: open-source, code, model, steps, considerations

The library that I will develop will be available as open-source, meaning that it will have the capability to become a "collaborative production of software where the source code of how the system works is available for modification by any programmer." (Graham; Cook 2010:117). As such, it opens a space for "unconscious collaborations" and it reshapes composition as a process of "constant dialogue between people, media and technologies" (Oliver La Rosa 2012:61). However, even though an open-source community may seem utopic, or even anarchic at first sight, Graham and Cook warn that "collaboration may not be egalitarian; hierarchies of skill and time most certainly apply and are acknowledged by those inside the system." Attuned to the dynamics of socio-economic context and human relations, therefore, open-source software development carries similar tensions as in any collaborative system, where shadows of hierarchy still are projected upon the listening space. Despite these impending individualities, open-source communities develop in time, and eventually individuals are outlived by the community, which brings back the notion of a collective author into this software making world. The importance of software and its impact on media is expressed by Manovich in a rather blunt statement, several years after his Language was published: "If you want to escape our prison of software —or at least better understand what media is today —stop downloading apps created by others. Instead, learn to program —and teach it to your students" (Manovich 2015:206). This claim for programming as a form of liberation is now at the doorstep of art projects, institutions, universities, and particularly music composition itself. The composer-programmer identity emerged from the first computer music languages, and since then has grown exponentially. Open-source communities have added in this respect by their revealing of the source code, and their opening up of their discussions to non-programmer members of their community, that is, users. This gesture by the software development communities in general is what allows for a much larger user community to engage in programming and in turn collaborate and resonate back into the development itself. This is to say that the user is no longer an end point, but another node in the network. In what follows, I will draw from already existing open-source libraries to begin a delineation of the proposed model. I will mention the works of composers and programmers that are linked with the real-time music composition environment called Pure Data (Puckette 1997). In order to study the Physical and Perceptual Aspects of Percussive Timbre, Brent developed a library for Pure Data called timbreID. In this library, Brent includes "some of the most important sound descriptors" (Brent 2010:38). In turn, Villeret imported into Gem some functionalities of the broader OpenCV (Open Computer Vision) language, into a library called pix_opencv. All of these object-oriented libraries (pd, Gem, timbreID, pix_opencv, OpenCV) are open-source, and all are written in C and C++ languages. I will draw from these and in turn develop a new library. Since both of these libraries can render descriptors of their media (i.e., sound, in the case of timbreID, and image, in the case of pix_opencv and OpenCV in general), what I propose is a combination of such descriptors into navigable nodes that would exist in-between both media. A series of steps will be necessary.

Audio data is inputted: in the form of files, or (automatically) segmented audio stream. A database of audio descriptors is generated in A by process S:

S –> A

Image data is inputted: in the form of files or (dis-)continuous frames. Another database of image descriptors is generated in B by process T:

T –> B

AB are traversed to obtain a third database N by process P:

P –> AB

Thus, N will constitute the navigable nodes which can either be traversed to: generate an audio-visual trajectory, or recursively apply steps 1 and 2, i.e.:

P –> NE

where E is

P –> CD

and CD constitute two other audio and image databases, respectively.

In the case of [4.i], traversing Ncalls the original elements of ABand, therefore, results in an audio-visual stream that depends, reasonably, on P. This means that Pmust have variables that can be manually set, or that can change with another type of process, naturally in feedback with AB. I hypothesize that process P consists of calculating derivatives between both sets of data. However, I will experiment with different mathematical models of obtaining a third set out of two different ones, namely probability distributions, markov processes.

In the case of [4.ii], recursion between other databases raises the following question: once recursion Nhas been reached, and [4.i] is desired, to which database do we refer to when we call the original elements? Two possibilities arise:

A set of control variables is dedicated to narrowing or widening the scope of N N is used in reverse, that is: another process Q is used to generate audio and video material out of N.

In all [4], traversing can be naturally automated and algorithmically programmed to occur in innumerable ways. It is not the purpose of this model to exhaust these possibilities, only to present this database multimedia composition environment as a framework for data navigation. In being the process dedicated to bring the audio descriptor and the image descriptor database together in a new database, process Pconstitutes the core algorithmic navigation of this model. Similarly, process Q will constitute a crucial aspect of this framework since it will be generating new audio-visual material from the outcome of the previous algorithmic navigation process. Thus, Pand Q will both be subject to experimentation throughout and will constitute the core of the proposed model. Finally, while S (namely, timbreID) is already available, T, on the other hand will need to be fully developed. Since most pix_opencv outputs processed images (i.e., not the proper descriptors), additional development must be assigned to obtain descriptors out of these processes.

I have thus far proposed an overview of the state of database art, the use of computers in music and art, the collaborative aspect surrounding computers, the immateriality that percolates through the arts as data, the terminological struggles in the definition of data-based media, the possibilities of new linkages between different media through data, the arbitrary world of the composer in the midst of an emergent, autopoietic, bottom-up art-world, the ubiquitous architecture enabling all of it, the resonating self in between, the non-human agency, the software communities, and the topology of the networked world.

The database multimedia composition model here proposed is yet another node; this text, a leap in the beginning of a navigation between these concepts: a navigation without a ship, map, or space; an imperative drive to grasp the now.

Contents

Introduction

I am sitting in a room different from the one you are in now. I am recording the sound of my speaking voice and I am going to play it back into the room again and again until the resonant frequencies of the room reinforce themselves so that any semblance of my speech, with perhaps the exception of rhythm, is destroyed. What you will hear, then, are the natural resonant frequencies of the room articulated by speech. I regard this activity not so much as a demonstration of a physical fact, but more as a way to smooth out any irregularities my speech might have.[1]

I am sitting in a database

In this section I place my dissertation in relation to existing scholarship, scholars, artists, and developers working in the fields of music composition, computer science, affect, and ontology, with emphasis on the ubiquity of databases and one the need to reflect on their practice, particularly in relation to multimedia arts. There is a database everywhere, anytime, always already affecting our lives. The database is an agent in our aesthetic and political lives just as much as we are agents in their composition and performance.

Existing Scholarship

I provide a current panorama of Database Multimedia Composition today, mentioning relevant researches and artworks that touch or are related to the topic of my dissertation.

[1] Alvin Lucier. I Am Sitting In A Room. See: `https://en.wikipedia.org/wiki/I_Am_Sitting_in_a_Room`

The Differences of my Approach

I go further in the analyses of database practices in art by reaching beyond the historical, and technical description of the practice, to suggest that the agency of the database and its effects on sound, community, and the self, allows to surface new aspects of the aesthetics of the practice, and of its political implications.

Argument

My argument is that in order to analyze the extent to which music composition can be understood as a database practice, we need to reconceptualize the agency of the database in art. I claim that the aesthetic outcome and political implications of this agency can be understood by transversally exploring database and composition under the broader concepts of sound, self, and community.

Database and Sound

First, I contextualize and define database practices, reviewing the existing literature with emphasis on sound practices.

I conceptualize sound in terms of listening and resonance, in order to understand the agency of sound in database and composition practices.

Finally, I conceptualize the political implications of the practice of music composition as database performance, which I claim a form of a self-listening —a resounding activity, or *a listening that listens to itself*.

Database and the Self

In order to analyze transversally the concept of self in database and composition, I first develop a technical overview of the performatic activity of the database into what I call *databasing*.

Then, I analyze the agency and performance of both subjects: the human —i.e., the *databaser*, database performers such as artists, composers, programers, etc.— and the nonhuman —i.e., the *database*—, thus understanding the database as a subject in itself. Therefore, as a subject, I consider the self of the database both in terms of gender[2] and spectrality[3].

Finally, the self in database is then developed politically into new conceptualizations of collaboration and authorship.

Database and Community

In order to gain insight on the aspects in common between database and composition, and the qualities of the resulting community of databasers , I first provide a historical overview of database practices, focusing on examples of the use of database in early computer-assisted collaborations.

I then bring closer the gap between the human and the nonhuman by analyzing the database in terms of its resemblance to memory. Thus, thinking memory as a form of database practice proves to be a crucial aspect of how the activity of the composer and that of databasing are in constant resonance with each other.

Finally, I understand community in terms of the 'inoperative'[4]. I thus bring the inoperative aspect of communities into the sphere of the 'music object'[5] and the database itself. Finally, I emphasize on the 'anarchic' forces present on the music object and the database, by way of their resonance —understood as resemblance and resistance— to the concept of the 'archive'[6].

[2] I use Judith Butler's concept of the gendered self as projected subjectivity

[3] I use Jacques Derrida's definition of the spectrality of the archive

[4] 'Inoperative:' meaning 'not operative' or 'in negation of the operational.' I understand this term in relation to Jean-Luc Nancy's 1990 text *The Inoperative Community*, also translated from the french *désoevrement* as 'Unworking.'

[5] The 'music object' refers to both meanings of the word 'object': as a 'thing external to the thinking mind or subject' (i.e., *a music work*) and as a 'a goal or purpose' (i.e., *the purpose of music*)

[6] I use the concept of the 'Archive' as defined by Jacques Derrida with the 'archontic principle,' meaning both the origin and the rule, both the place where power is located and where it exercises its authority.

Chapter Structure

In this section I describe the aims and the chapter structure of this dissertation.

Principal and Secondary Aims

What do I intend to do with this text? Who is it aimed at? What do I expect from the reader? How does my theorization affect the practices of Composition, Multimedia, and Database? To what extent is my practical approach an instantiation of my theorization? What are its limitations? What do I expect from both my Database Model (Appendix A) and my Annotated Bibliography Model (Appendix B)?

The Chapter Structure

Chapter 1 serves to contextualize the database historically and technically. First I engage with the database from the point of view of media studies, as it as commented in the arts since the beginning of the 21st century. I then trace a history of the events which lead to the current database panorama, and refer to it as a database tree. In the end of this chapter, I trace the use of the database in relation to music, particularly in three fields based on computer-based sound: MIR, sonification, and composition. In Chapter 2, I dedicated to locate the aesthetic agency of the database from three points of departure: listening, memory, and performance. These three aspects relate sound, networks, memory and archives, in order to delineate the performativity of databasing. Thus, the agency of the database is seen at the intersection of the human and the nonhuman. The final chapter deals with the dynamics of databasing and composition. I engage with the political in database practices and question the established concepts behind music composition. Thus, I present a different conceptualization of the music work.

Chapter 1

Database Art

In order to define and contextualize database practices, I engage with the existing literature on data-driven art. Drawing mostly from media theory, I provide a sample of a variety of authors who have studied the use of databases in art. Specifically, I emphasize certain aspects of affect theory which relate to the intersection between the database and the body, in order to link database practice with sound and performance practices.

Therefore, in ‘databasing music,’ I describe different approaches to music practices —computer music, sonification, music information retrieval— and their interrelation with software design, to show how some of the major breakthroughs of these practices are related to changes in data structures. In the last section of the chapter, I describe all layers of the concept of the database, from lower —data structures— to higher —databases— levels, and describe the basic algorithmic designs in between. Specifically, I argue that all of these layers constitute what I call the performativity of the database, which is what is incorporated in the practice of database music.

1.1 The Database

In this section I engage with the existing literature on data-driven art. In relation to new media, Manovich first considered the Database to be a new 'symbolic form' of art. (Manovich 2001) The database became a term related to 'internet' and 'digital' art, and as such it was conceptualized in relation to interface design and interactivity. Hansen provided further insight to Database practices through theories of embodiment. (Hansen 2004) He claimed that since Manovich's theorizations did not take into account the Body as an active agent, human-computer interaction so far fell short of reaching its aesthetic potential. Vesna, Daniel and Lovejoy proposed a theorization of context within Database Art. (Vesna 2007b; Daniel 2007; Paul 2007; Klein 2007) In their theorizations of interactivity in Database Art, they reassessed authorship in Digital Art. These authors later provided a panoramic view of database practices within the digital art world, and arrived at the broader term "Database Aesthetics." Since then, studies on database aesthetics have only been present in the existing literature in terms of data practices.

1.1.1 New Media

Lev Manovich (Manovich 2001) is the first media historian to argue that the database occupies the center of the creative process in the computer age. In *The Language of New Media*, he identifies the emergence of new media with the development of the database. He argues that the amount of information that bloomed exponentially since the beginning of the 20th century demanded newer and more efficient media objects —databases— to store, classify, and access data. (ibid., p. 34) In his view, the world

> …appears to us as an endless and unstructured collection of images, texts, and other data records, it is only appropriate that we will be moved to model it as a data-base —but it is also appropriate that we would want to develop the poetics, aesthetics, and ethics of this database (ibid., p. 219).

The database, according to Manovich, became the content of new media works whose variable interfaces allowed the same content to be presented in different ways. He claims that this is the case because of the separation that exists between database and interface. In this way, Manovich argues, the notion of narrative is redefined as the user's trajectory through the database.[1] (Manovich 2001, p. 227) However, it is interesting to note how artist Graham Weinbren thinks of databases not as content, but as containers: "A database," he writes,

> ... does not present data: it contains data. The data must always be in an arrangement ... that gives the data its meaning. What can Manovich mean when he says that a database "presents a model of what the world is like"? How would a database have meaning?[2] (Weinbren 2007, pp. 67–9)

This difference is that of form and content, and, with this in mind, it is clear that the foundation of Manovich's argument is that the logic of the computer is transferred to culture at large.[3] (Manovich 2001, p. 235) Therefore, given that the form (in his words, literally "database form") is determined by the content, he describes the "ontology of the world according to a computer" (ibid., p. 223) with two types of computer objects: data structures and algorithms. He argues that these two objects form a symbiotic relationship which is projected onto the cultural sphere. The former, as databases, become ways of information access; the latter, as, for example, computer games, become a form of narrative: "while computer games do not follow [a] database logic, they appear to be ruled by another logic —that of an algorithm." (ibid., p. 222)

What this means, essentially, is that Manovich proposes a worldview where technology becomes the content, and thus, the form of art. As I will show further down this text, this statement stems from a technologically determined standpoint that dominated much of media theory in the second half of the 20th century (See 1.1.2). With this, I propose an embodied alternative where

[1] This trajectory is, technically, the traversing of the database, and it is one of the central aspects of the performativity of a database. I will return to this point at the end of this chapter.

[2] The question of meaning in database will be addressed in Chapter 3.

[3] Manovich supports this statement with the aesthetic that results from the ability of a computer to generate multiple variations. For example, the writing of computer programs to generate minimalist art. Likewise, in Manovich's view, computer art pioneer John Whitney's work *Catalog* is the application of the technological device of image variation. This technology, in turn, determined the proliferation of the use of computers as endless transforming (digital image) filters, as was the case with all the (MTV) music videos that followed Whitney's aesthetic throughout the 1990s. (Manovich 2001, p. 236)

databases have an aesthetic agency that does not cancel that of the human. In this sense, I interpret Weinbren's viewpoint as valid but incomplete, in the sense that it is not only the narrative[4] what gives meaning to databases. Furthermore, given the fact that both Manovich and Weinbren focus on visual arts, and thus, they base their discussion between database and narrative, I shift the focus from narrative to performance, arguing that it is performativity itself what provides meaning to database practice. Thus, by performing the database, that is, by the databaser engaging with the database tree (See 1.2.1), the self of the database appears in resonance with the self of the databaser, in a process which informs the database of its meaning. Therefore, in itself, I consider the database to have meaning only through this resonating performativity between the human and the nonhuman.

In order to reveal the extent to which the presence of the database has a radical effect on narrative, Manovich reverses the semiotic theory of syntagm and paradigm that governed the first half of the 20th century (Manovich 2001, p. 231). Quoting Ferdinand de Saussure from Roland Barthes' *Elements of Semiology*, Manovich explains that the syntagm is an instantiation of individual words as concrete elements in a sequence, and that the paradigm is the abstract notion of meaning that is drawn from the former; the former subjected to combination, the latter to substitution. Manovich claims that it is the paradigm what is made explicit with the material existence of the database —i.e., residing in a computer's memory— and that, since the narrative is just a set of links —i.e., pointers which refer to addresses on the computer's memory—, the syntagm is made implicit.

For example, consider the case of the typical timeline-view of a video editor[5]. Normally the user creates a session and *imports* files to working memory, creating a database of files —video files, in this case. Once this database is in working memory, the user places on a timeline the videos, cutting, and processing them at will, until a result is desired, and an *export* or a *render*

[4]Weinbren refers similarly to 'arrangement' and to narrative

[5]This example was used by Manovich in the late 1990s, and it is still valid today with most editing sofwares, i.e., any multimedia editor: audio, video, text, etc.

is made.[6] On the one hand, the timeline where the user places the videos is only a visualization of the set of links to the files. This structure exemplifies the Von Neumann architecture on which computers are constructed (See 1.2.2). In other words, the timeline is an editable graph that allows the user to locate in time the pointers to the elements on the database. This is what Manovich means by "a set of links," because the user is not handling the files themselves —as would be the case with an analog video editor, where the user cuts, pastes, the magnetic tape—, but the extremely abstract concept of memory pointers. Therefore, considering this as the syntagm —i.e., the instantiation of individual elements— would have to mean that the syntagm is de-materialized.

Inversely, considering the database of files as stored data on the computer's disk, Manovich comprehends it as materialized. Therefore, since the database comes to represent the set of choices, that is, the limits of the logic of subsitution, this is how Manovich considers the paradigm as materialized, which equates to saying that meaning itself is materialized, thus arriving to yet another of Manovich's arguments for the database as content, and as determination of form. While his attempt is valid, I argue that it is construed upon a false premise, essentially, that neither the pointers to the elements on disk, nor the stored data on the disk are material. Pointers are data which reference locations of other data in memory. Therefore, the consideration of materiality falls upon data istself, rendering this discussion of meaning null, if only considered as stored data. In turn, down this text I consider the distinction between data and information (See 1.1.3) as a foundation for a new way of thinking the database as neither explicit nor implicit, but as spectral (See 2.2.4). The discussion of meaning, then, acquires another level of complexity that requires data —as databases— to be resonating with the human.

Despite Manovich's technologically determined considerations of the database as form, he notes a fundamental aspect of the use of the database when he expresses that data need to be generated. (Manovich 2001, p. 224) In this sense, he begins to describe the actions that need to be

[6] 'Import,' 'export,' and 'render,' refer to processes that read from or write to the computer's disk.

performed around data. This is one side of what I call the performativity of the database:[7]

> Data creators have to collect data and organize it, or create it from scratch. Texts need to be written, photographs need to be taken, video and audio need to be recorded. Or they need to be digitized from already existing media... Once digitized, the data has to be cleaned up, organized, and indexed (Manovich 2001, p. 224).

He even goes further and proposes that this activity has become a "new cultural algorithm," (ibid., p. 225) which I reinterpret as:

```
function cultural_algorithm(world)
{
    database ← data ← media ← world
    return database
}
```

Following this line of thought, artist Victoria Vesna (Vesna 2007b) argues that creating a memory bank is a means of testifying to our existence. (ibid., p. 25) As Weinbren (Weinbren 2007) describes it, the terminology within filmmaking appears to have changed to fit in database practices:

> The first step in working with a database is the collection and assembly of the data... Sorting determines the sequence of presentation... 'cutting,' as 'editing,' loses its meaning, and 'sorting,' 'assembling,' and 'mapping' become more apt metaphors for the activity of composition (ibid., p. 71).

Furthermore, as Vesna describes, the architectural aspect of database design has an effect on the type of navigation that they predetermine:

> How one moves through physical space... is very much determined by the way an architect has conceived it... When navigating through various software 'containers' and inputting our data, we are in effect following the established parameters of information architecture (Vesna 2007b, p. 28).

[7] I continue developing this concept in the final section of this chapter.

While Manovich calls for an "info-aesthetics" (Manovich 2001, p. 217), as well as a poetics, and ethics of the database, his technologically determined argument prevents him from developing such an aesthetics, precisely because it is grounded upon a disembodied view of information. As I begin to suggest above —and will continue to do so in the following sections—, neither Manovich nor the following generation of media artists and theorists of *Database Aesthetics* (Vesna 2007a), could carry out an exhaustive account of the aesthetics in question. Manovich does emphasize that the resulting aesthetic —i.e., the database as form—, is a symptom of the uncritical use of database logic throughout the visual art world of the 1990s. However, he fails to provide a counter-argument to it because his own argument is supported by the same disembodied constructions that prevent him from including human agency in his account.

Therefore, in order to understand where Manovich and his followers fall short in their analysis of new media, I draw first on media theorist N. Katherine Hayles, who has developed an embodied criticism on the dominant theories of information and cybernetics —although her application was in the literary world. Then, I present media theorist Mark B. N. Hansen's criticism, who expanded Hayles' concepts into his consideration of the body, image, and affect in new media. My intention in this section is to prepare the grounds for an application in the world of digital sound practices the concepts of database and embodiment.

1.1.2 Bodiless Information

In *How We Became Posthuman*, media theorist N. Katherine Hayles (N. K. Hayles 1999) unearths the theoretical context upon which the posthuman has been constructed throughout the 20th century: cybernetics. She identifies three waves of cybernetics, each governed by different concepts[8]

[8]Respectively, the concepts of homeostasis —expressed in systems such as feedback loops—, reflexivity —or "the movement whereby that which has been used to generate a system [becomes] part of the system it generates," (N. K. Hayles 1999, p. 18) which then developed into autopoiesis [self-generation]—, and finally self-organization and the development of artificial intelligence —with genetic algorithms and cellular automata.

which help build particular structures[9] that defined the technologically determined and disembodied literature that was in vogue in the 1990s.

For example, Hayles draws four points of interest in which the argument for an embodied criticism of the posthuman can be focalized: the privilege of information pattern over material instantiation; the consideration of (Western) consciousness as an epiphenomenon secondary to the evolution of information; the notion of the body as prosthesis; and, thus, the derived premise that the human can be articulated by means of intelligent machines. (N. K. Hayles 1999, pp. 17–8) Most important, Hayles argues that all of these stages of cybernetics are possible and derive from mathematician Claude Shannon's formal definition of information as "a probability function with no dimensions, no materiality, and no necessary connection with meaning." (ibid., p. 18) Shannon's definition directly affects the concept of communication systems, and what it amounts to is that instead of a message, what is sent is a signal that needs to be encoded and decoded, and isolated from noise. (ibid., p. 33) Thus, information is understood as an immaterial pattern over randomness. Therefore, Hayles argues, information can be (wrongfully) considered as more important than material forms. The consequence of this is a technologically determined condition of virtuality that needs to be reconceptualized:

> By turning the technological determinism[10] of bodiless information [first-wave cybernetics], the cyborg [second-wave cybernetics], and the posthuman [third-wave cybernetics] into narratives about the negotiations that took place between particular people at particular times and places, I hope to replace a teleology of disembodiment with historically contingent stories about contests between competing factions, contests whose outcomes were far from obvious (ibid., p. 37).

By reconfiguring the concepts of body, consciousness, and technology as inherent to (post-) human life, Hayles argues for the impossibility of artificial intelligence to serve as a proxy for the human:

[9] Disembodied information, the cyborg, and the posthuman.

[10] Media theorist Mark Poster defines technological determinism as the "anxiety at the possibility of [the human mind's] diminution should these external [technological] objects rise up and threaten it." (Poster 2011, p. X) In other words, the fear or anxiety that the human is ultimately subjected to the power of technology.

> My dream is a version of the posthuman that embraces the possibilities of information technologies without being seduced by fantasies of unlimited power and disembodied immortality, that recognizes and celebrates finitude as a condition of human being, and that understands human life is embedded in a material world of great complexity, one on which we depend for our continued survival (N. K. Hayles 1999, p. 5).

Hayles objective is, then, to dismantle cybernetics from its (relative) assumptions and to provide insights into the field that question its major achievements over the years. By doing so, she opens the field for new considerations of the body within cybernetics, and by extension, of the body in new media. While her work is concentrated in the literary narratives that were built in parallel with cybernetics, she leaves incursions in new media for other media theorists. This is where Mark B. N. Hansen comes in.

1.1.3 Embodiment

Manovich's notion of database and interface results, as we have seen, in the conception of the interface to the (art) work. In turn, what this results in, as media theorist Mark Hansen noted, (Hansen 2004) is the disembodied concept of the "image-interface" to information. Hansen locates the source of this disembodied conception that prevented Manovich's account of new media from carrying out his above-mentioned proposal of an "info-aesthetics" (Manovich 2001, p. 217), in Manovich's implicit —but nontheless evident— premise of the overarching dominance of cinema in contemporary culture.[11] For Hansen, this assumption results in a "disturbing linearity [with] hints of technical determinism" (Hansen 2004, p. 36).

For example, Manovich argues that the dominance of cinema is the reason why new media insist on sequence and language-like narratives. He claims that discretization of modern media has its origins in the Industrial Revolution, when the factory system replaced artisan labor (Manovich 2001, p. 30). Specifically, he describes the two principles of Ford's assembly line: the standardization of parts and the separation of the production process into simple, repetitive, and

[11] Media theorist Alexander Galloway calls this Manovich's 'dirty little secret.' I will expand on this later in the text.

sequential sets of activities. His claim is that cinema suffered a standardization process in terms of *resolution*,[12] which was determined by the industries that massively produced the electronic devices used for recording and playing.

In this sense, he claims that the internal structure of the assembly line conditions how we see and listen to a multimedia work. On the other hand, the separation of production into smaller, repetitive tasks is related to the operations that computers are well suited for performing. Among these operations, which will be covered in detail further on, Manovich distinguishes between low-level and high-level automation processes. While low-level operations can be image processing algorithms, such as color filters; physical modelling techniques, such as displaying fire or waterfalls on screen; and mathematical modelling techniques, such as generating swarm-like motion; for Manovich, high-level processes depend on the computer's semantic ability to understand media objects, as in the case of artificial intelligence within computer games.

In other words —somewhat Kittlerian, as we will see later on—, the technological devices driven by industrial forces largely condition both the way in which the art-form is not only recieved, but produced, and reproduced, and, as an extension of this, the aesthetics of cinema are conditioned and shaped by the technology supporting it.

An aesthetic and technic convergence that extends to the definition of cinema itself, and percolates through all of new media as a consequence. However, in order to speak of new media, Manovich distinguishes two types of deterministic trends. On the one hand, mass-standardization and reproducibility of media —i.e., the "logic of the factory" (Manovich 2001, p. 30)—, which was taken under scrutiny by Walter Benjamin, and later the philosophers of the culture industry comes to shape the form and evolution of cinema. On the other, the logic of post-industrial society, which favors individual customization, is what governs new media development.

Manovich claims that new media development and data access preponderance are reflec-

[12]Resolution in terms of image —dimensions, frames per second, aspect ratio— and audio —bit depth, sampling rate, number of channels.

tions of the logic of post-industrial society, which favors, for example, individual customization. For Manovich, as we have seen, the logic of new media is no longer that of the factory, but that of the interface. And this is due to the internal role of the database. Through the interface to a database, the user is given access to multiplicities of narrative, information; the user is granted the power of the database, making in Manovich's eyes the database an icon of postmodern art.

What Manovich misses, however, is the ontological significance of the digital because, as Hansen suggests, for Manovich the user of new media assumes the passive position of a traditional cinema audience. In opposition to this passivity of the body, and following Hayles' embodiment critique of informatics (N. K. Hayles 1999), Hansen provides great insight into the aesthetics of new media, particularly in his consideration of the image in new media. In a general sense, Hansen describes images as something that emerges out of the complex relationship between the body and some sort of sensory stimulus.

In radical disagreement with Manovich, Hansen considers that the image has become a process which gives form to information, and that this process needs to be understood in terms of the body as a filtering and creative agent in its construction. Drawing from Henri Bergson's theory of perception, and in resonance with cognitive science, Hansen defines the function of the body as a filtering apparatus. Under this conception, the body acts on and creates images by subtracting "from the universe of images" (Hansen 2004, p. 3). In other words, through this filtering activity, the body is empowered with "strongly creative capacities." (ibid., p. 4)

Hansen draws from process philosophy to furnish his focus on the coevolution of embodied cognition. He emphasizes on affectivity —what he considers "the capacity of the body to experience itself as 'more than itself'" [Hansen quote from (Thacker 2006, pp. 266–7)]— as a creative force to achieve the 'new' in new media. Eugene Thacker (ibid.) writes in his review of Hansen's text, that one of Hansen's aims is to "rethink embodiment in radical ways, ways that are uncannily 'nonhuman'" (ibid., pp. 266–7). Therefore, by engaging with the intersection of body and data, Hansen attempts to release the digital image from its frame, claiming, for example, that "there is no longer anything materially linking the content of the image with its frame." (Hansen 2004, p. 8) In this sense the data holding images is polymorphous, i.e., "lacking any inherent form or enframing.[13]" (ibid., p. 36) Furthermore, in Hansen's redefinition of the image out of the frame, and back into the body, he locates the source of Manovich's problem: his failure to recognize the body within his argument because his cinematic metaphor does not allow him to do so. With a reconstitution of the importance of embodied experience, Hansen locates in the body's virtuality the capacity to create images within itself, the affective potentials of the 'new' in new media.

In order to understand why Manovich and his followers reveal so forcefully the anxiety that is characteristic of this irrelevant (human) agency, in the next section I analyze Hansen's assessment of another media theorist, Friedrich Kittler.

1.1.4 Convergence and Framing

Hansen argues that media theorist Friedrich Kittler's concept of 'digital convergence' —in which the bodily resonance of media becomes obsolete in the face of absolute digital information storage— results in —and depends on the premise that— the human, as far as it is constituted in terms of its sense perception, becomes a "dependent variable." (Hansen 2002, p. 59) For example, Hansen writes in relation to Kittler's *Gramophone, Film, Typewriter*:[14]

> Whether it is the celluloid inscriptions of film, or the plastic inscriptions of phonography, or the discrete impressions of typewriting, media function as *databases from which bodies are constructed*, and the thresholds governing their capacities for registration continue to be set, however indirectly, by the limits of (human) sense perception (ibid., p. 62) (Italics mine).

What is important to note here is the role of the database. On the one hand, in the case of physical ('old') media, the database is the place from which a body is constructed; the constraints of such body are delimited by human perception. On the other hand, in the case of digital ('new')

[13] I return to the concept of enframing later in this text.

[14] Kittler's work was published in German in 1986, and it was translated to English in 1999 by Winthrop-Young and Wutz

media, the database fulfills the role of what Hansen —referring to Kittler's digital convergence— defines as an "absolute system of information storage." (Hansen 2002, p. 63) Therefore, since the bodily constraints of human perception are no longer in question, the human becomes secondary to the rule of information. From this condition of human subordination, Kittler's technological determinism acquires a different connotation: what is central and determinant in Kittler's view is, therefore, the economic interest of capitalist institutions. Because these institutions control the development of technology, they dominate the consumer, and this is why, for Kittler, digital technology cannot be separated from history, or military planning. The following passage reflects the extent to which Kittler's thesis is essentially grounded in Adorno's and Horkheimer's arguments in the *Dialectics of Enlightenment* (1944):

> A mere 'byproduct' of pleasure, entertainment is a hangover from the media epoch: a function that caters to our (soon to become obsolescent) need for imaginary materialization through technology, but that, as Kittler argues most forcefully in his consideration of computer software, serves as a diversion to keep us ignorant of the operative level at which information, and hence reality, is programmed (ibid., p. 59).

In order to contest Kittler's digital convergence, Hansen, following Hayles, thus explores the extent to which Kittler's theory is grounded on the dominant strand of cybernetics and Shannon's mathematical theory of information. (ibid., p. 63) Therefore, Hansen's goal is not only to demarcate the contingent nature upon which Kittler's thesis is built, but also to provide insight into parallel theories of information —especially British physicist Donald M. MacKay's— that would re-embody Kittler's posthumanism.[15]

For Hansen, MacKay's argument works against the separation of information and meaning, and it is grounded on a notion of "framing." Hansen describes that the

> activity in the receiver's internal structure generates symbolic structures that serve to frame stimuli and thus to *in-form* information: this activity converts regularities in the

[15]This is why Hansen later considers Lev Manovich's analysis of new media —as well as some of Kittler's analysis— to be extremely partial, in the sense that they focus on the formalist and technical aspects of programming and computation.(Hansen 2006, p. 264)

flux of stimuli into *patterns* of information (Hansen 2002, p. 76).

Therefore, the activity of framing, according to Hansen, must be differentiated from that of "observation." In this way, "information remains meaningless in the absence of a (human) framer," (ibid., p. 77) and framing becomes a resonance of the (bodily) singularity of the receiver. Quoting MacKay's *Information, Mechanism, Meaning* (1969), the meaning of a message

> . . . can be fully represented only in terms of the full basic-symbol complex defined by all the elementary responses evoked. These may include visceral responses and hormonal secretions and what have you. . . an organism probably includes in its elementary conceptual alphabet (its catalogue of basic symbols) all the elementary internal acts of response to the environment which have acquired a sufficiently high probabilistic status, and not merely those for which verbal projections have been found (ibid., p. 78).

It is with this conception of framing that Hansen describes precisely that information always requires a frame:

> . . . this framing function is ultimately correlated with the meaning-constituting and actualizing capacity of (human) embodiment. . . the digital image, precisely because it explodes the (cinematic) frame, can be said to expose the dependence of this frame (and all other media-supported or technically embodied frames) on the framing activity of the human organism (ibid., pp. 89–90).

Therefore, in the presence of convergence, framing is understood as the evidence that the human cannot be rendered a dependent variable. To the contrary, the framing function of the human body is the possibility condition for the digital to become information. The frame, as Hansen describes, is the human body filtering images from the world, and creating a virtual image that gives form to data. Furthermore, since the frame needs to happen as a relation, and thus, it is the temporal intantiation of a process, I would like to focus now on the relation itself. This is to say, given that Hansen has proved that there is no information without embodiment, in the digital world there is no information without data, and there is no possibility of framing without a data structure and its implementation linked to some sort of interface.

What the interface means in this context, however, is simply a way for the human to "request" and to "get" some data out of a larger pool of data. The interface itself may be of any shape, and nowadays there are conferences such as , for example, dedicated to the creation of interfaces for music. If you consider a raw, text based interface —what is known as a computer terminal—, a user inputs letter combinations with constitute commands for the computer to run. The moment a command line is entered —e.g., the user hits "enter" or "return"—, the interface interprets the inputted line searching for its correspoding command. If it finds it, it executes it, ortherwise it returns a default error that no command was found. Therefore, on this very simple action of typing a word and entering it, there is a structure that supports querying —of a function, in this case— and either the execution of the function or a message reporting an error, for the user to effectively get the initial request. This basic interface serves to exemplify a very simple situation of framing. For example, running the command <ls> —or <DIR> on Windows command prompt—, will return a list of elements that are visible for the user in the current directory. How can the framing function take place within this very simple and routinary task? The use of this command is so simple that any user with a keyboard can type those two or three letters and then hit enter, just as if using a typewriter. What is difficult to interpret, however routinary a command of such sort it might seem to avid computer programmers, is the result that is returned. For every line of returned text, there is an entry with the name of the file or directory available within the directory where the command was executed. This means that the user can see what files and folders are inside in any specific location of the directory tree of the computer. Without seeing anything, and without doing anything other than requesting, the user has to get that indeed the names of the files correspond to the files that are in the current path. Therefore, the way for the command to be of any use to the user, the user has to incorporate the image returned by the computer. This image is none other than the internal path structure, or directory tree, which depending on the operating system would have different shapes, implementations, etc., but in essence would remain as a data structure with an input/output interface.

Thus, when the path structure of the computer is framed by the user is not only when data is effectively converted into information, also when an image of the computer itself is created, and therefore a relationship between the human and the digital nonhuman is established. Of course, this is just one example of many, but the specificities of paths and files, if understood properly in terms of addresses and memory, may bring the database itself to a completely different level, that is, understanding of the databasing in terms of memory. However, before delving into the interconnections that exist between the concepts of database and memory, a few more words on the development and theorizations of database practices need to be mentioned. Specially, the work of the later generation artists and theorists drawing from Manovich's understanding of new media.

1.1.5 Database Aesthetics

Digital art practices in general, and internet art in particular, have reshaped the way art is being produced. As Manovich had reflected of the art in the 1990s, and even more prominently through the 2000s with the increasing presence of personal computers at lower prices, data access has become an artistically fruitful concept. In this sense, and Manovich has expressed this on numerous ocasions, the internet is a place of unlimited access, or a database in continuous and exponential growth, which reconfigures the grounds on which art has traditionally been built on. An example of these reconfigurations in art is the case of media artist Sharon Daniel and her collaborative approach. Drawing intensively from Katherine Hayles' posthumanist critique, Daniel raises questions about authority and politics in collaborative art, and particularly, in database art (Daniel 2007, p. 177).

Two aspects of Daniels' work will be commented on here: her association of data with the social sphere, and her concept of mapping.

For Daniels, algorithms such as cellular automata constitute "models for rethinking representation and authorship" (ibid., p. 146). Since cellular automata are systems that reveal emergent (global) behavior from very simple rules set on the local level. This means that given a large set of automata, each automaton changes states in time according to its surrounding automata, resulting on a grand motion on the macro level that cannot be easily predicted from the predicates of the system itself. As a consequence, the social metaphor assigned to cellular automata is a reconfigured hierarchy that is not governed from the top to the bottom, but from the bottom-up. This, for Daniels, is an example of an expanded conception of authorship that precludes any top-down decision on the system —i.e., removes the idea of authority itself—, and instead enables a concept of "cultural democracy," which Daniels links to an aesthetics of dignity (Daniel 2007, p. 159). Brought to collaborative multimedia art where participants engage with the work itself at every stage, changing and determining the future states of the artwork, this concept of democracy is held against possible harmful uses of database practices that would continue with an imperialist —i.e., top-down— line of cultural domination and subordination.

From a different perspective, writer Norman Klein (Klein 2007) notes a shift in the role of the database narrative reader. He argues that since the reader gets immersed in data, "the reader evolves pleasantly into the author " (ibid., p. 93).[16] However, he immediately takes this back: "finally, instead of an ending, the reader imagines herself about to start writing" (ibid., p. 93). In contrast to Daniel's logic of a decentralized bottom-up concept of authority, Klein's argument leads to the conclusion that with the imposition of database logic to artworks both roles of the reader/participant and the writer/author fade into each other and vanish. On one hand, the reader turns into the author as the data-driven narrative unfolds by gradually replacing the passivity of the figure of the reader with the activity of database navigation —which comes to represent the action of the writer. On the other hand, since there is no actual writing —only the imagination of the beginning of writing—, the database comes to trump writing itself.

Under this reader- and writer-less scenario, the database is the sole aesthetic agent to which the human is subjected. I argue that the technological determinism that is still at play here

[16]The fact that Klein is a writer himself may attest to why he classifies this shift as evolutive.

can be disarticulated by simply reconfiguring the notion of writing with databases, as yet another aspect of database performance. Klein fails to develop his writing metaphor because it is grounded in the same Manovichian fallacy of the image-interface that Hansen noted.[17] If Hansen is correct in defining the image as a process which in-forms information, then Klein's reader would be not a passive cog in the database logic, but an active agent in its unfolding. In this way, the reader's role does not evolve into the author's: there is a co-evolution of the notion of writing with the database technology.

On her account of mapping as a form of intersubjective communication, Daniels describes the cybernetic bug Hansen had found in Manovichs' text: the difference between data and information. As Hayles and Hansen remark, information needs to be embodied (in-formed, given form). Daniel begins her text with the definition of the latin word "data" in singular: "datum: something given" (Daniel 2007, p. 177). Extending this definition to information equates to a reductionist view of information as a bodiless fluid going from sender to receiver. Understanding data as information isolates the human as a residual element in the circuit —as in Kittler's view. This technologically determined anxiety is present in two authors who refer to new media as visualizations: Norman Klein (Klein 2007, p. 99) and curator Christiane Paul (Paul 2007, p. 99) both equate new media with data visualization. Such an extremist view of new media suffers from identifying the "new" in new media with the "digital" on the one hand, and with "visual" media on the other, and it is built upon the Manovichian anxiety that led him to convey the image-interface relationship that precludes the creative agency of the human body in the process of information. To the contrary, in Daniel's description of mapping, a plurality of givens is given form (mapped) visually. Thus, intersubjective communication, for Daniel, involves two (human) subjects faced with visualized data. While mapping is an agent in the formation of data, it is yet another stage in the process of information itself, one that engages with the relation in between body and data.

[17]Furthermore, in Klein's words (Klein 2007, pp. 86–8), the human is a slave to data, and as a consequence the human is economically colonized and psychologically invaded by the evolving force of computers, information, or technology in general.

Therefore, the actual center where information takes place in any process of communication is itself the body. This is also evident in the etimology of the word ‘data’ that Daniels points to. The word ‘datum’ —from its latin roots— is itself the past participle of the verb ‘to give.’ The plural nown ‘data’ is therefore the condensed result of a previous action of giving. Thus, nothing can be data unless it is given. However, what constitutes the ‘giving’ in this context? Simply put, data is first something not ‘given’ but necessarily ‘taken’ from the world; only after it was indeed taken, it becomes that which was given from the world. In this reversal, what makes things even more complicated is the concept of ‘getting.’ To get something means to understand or incorporate something. Therefore, when one ‘gets’ data, one is informing it, that is, creating the necessary virtuality for information to emerge out of data. This terminological incursion serves to shed some light on the distinction that Daniels points to with her description of mapping and visualization. In fact, the concept of mapping itself, coming from cartography, depends on translating geographic to graphic data in such a way that a spatial image can be created within users, for their locational needs.

1.1.6 Open Source

For media theorist Alexander R. Galloway (Galloway 2011), the semiological reversal of syntagm and paradignm is one among several other positive aspects of Manovich’s text that have become “common parlance” in new media discourse throughout the years. (ibid., p. 379) Therefore, for Galloway, Manovich’s account is still valid in 2011, but only if taken in hindsight as the outcome of the first wave of new media theorists. Furthermore, Galloway identifies in Manovich’s argument an often criticised point: the political. Galloway claims that Manovich’s “abdication of the political” in favor of a semiological incursion into software is rooted on a post-communist-era intellectual danger on Manovich’s end when it comes to form, poetics and aesthetics. For Galloway, this means that any “critique of state-driven ideology” on Manovich’s end is thus placed on hold. As a result, Manovich embraced the concept and industry of cinema wholly as the first new media and, thus, provided the necessary structure to sustain his argument. Galloway calls this Manovich's "dirty little secret." (Galloway 2011, p. 379)

However informative Galloway's revision of Manovich's text is, even after ten years of the publication of the latter, Galloway's text still presents hints of sympathetic approval of Manovich's disembodied stance. This is better exemplified in the way Galloway defines the notion of open source development. For him, open source is a "communicative artifice" (ibid., p. 383) which can be identified with a neoliberal impulse he calls the "California ideology." (ibid., p. 377) What this ideology consist of is the liberation of mankind, which he traces to the social movements of the 1960s. Galloway draws a teleology of freedom that starts from liberating capital, desire, and finally information itself. Following the Shannon-Weaver's —disembodied— notion of information, he writes: "information wants to be free." (ibid., p. 378) Within this understanding of open source development, it makes sense for Galloway to equate open source culture with "the freedom to connect to technical images." (ibid., p. 383) Therefore, by considering it a "migration into a new way of structuring information and material resources" (ibid., p. 384) and, in thinking this way, by reducing it to yet another communication system, Galloway strips meaning out of open source culture. The posthuman consequence of this semantic void in Galloway's open source is, naturally, what Hansen calls the Kittlerian optoelectronic view of the world. In this view, and as we have seen initially with Hayles' account, information no longer needs the human or, rather, the human itself is a residual and dependent variable of the evolution of an overarching information system.

Like I believe Hayles would, I consider this description of the open source world as a nightmare. Through open source communities there is a new way to understand posthuman agency. Therefore, in what follows, I bring into the picture some concepts from the *Database Aesthetics* (Vesna 2007a) generation, and propose ways in which posthuman agency is understood.

1.2 Databasing

This is why I now focus on Database practices before entering the 'age of information' (McLuhan 1964, Chion 1994, Manovich 2001, Simons 2002, Vesna 2007, Paul 2008, van Dijck 2017). I define two precursors of the database as 'statistics' and 'archival practices.' First, I contextualize briefly the practice of statistics, and describe the notions of 'sampling' and 'data collection,' in order to provide a glance on what I consider the origin of data-based practices. Then, I begin relating the concept of the Archive —as presented by Derrida (Archive Fever, 1995)—, so as to deepen the understanding of the principles involved in archival practices. I describe the main technical concepts behind Database Navigation and provide use cases from both appendices A and B, the former relating to joint image and audio databases, and the latter to text databases. I then reflect on the quality of this navigation in relation to the type of navigation and results that they obtain.

1.2.1 Databasing

> The first step in working with a database is the collection and assembly of the data. . . Sorting determines the sequence of presentation, while filtering gives rules for admission into the set presented. . . only a portion of the material goes through this process. . . resulting in a database that is a subset of the "shot material" database. Editing is selecting from the database and sequencing the selections. . . For a filmmaker, database thinking is liberation—one is freed to let the material breathe.[...] To go further: for a filmmaker the term "cutting," as "editing," loses its meaning, and "sorting," "assembling," and "mapping" become more apt metaphors for the activity of composition (Weinbren 2007, p. 71).

Weinbren describes data collection and assambly as the first steps in working with databases. I have previously mentioned Manovich's insight on what I call the performativity of the database. Manovich treated the selection process as a fundamental step in database practice, one that as a filmaker, he discovered to be a major change in filmmaking. Weinbren breaks this selection process into further steps that are more revealing in terms of what I call databasing. These terms are sorting and filtering. With this new terminology, Weinbren points out to a linguistic shift from 'editing' and 'cutting' to'sorting,' 'assembling' and 'mapping.' This linguistic shift is significant in the sense that it announces the practice that is 'under' the filmmaker: databasing. Databasing is the practice of the database, all the elements that lead to its performativity, which involve the repeated thinking and acting through a database.

This performativity is repeated from higher to lower levels of software, but it is present all the way through. For example, on a higher level, a user can select and import files from their computer into the editing software environment. This action already is enabled by a very complex lower layer that is built for bridging the software with the computer's memory. Within this lower layer, files are generally copied —depending on the software— into working memory for non-destructive editing. The copying itself can be performed in many ways, but generally it involves a combination of programming routines. It generally consists of two sets of routines, one for memory allocation and another one for traversing and writing to disk. The details of these routines are of no importance for this discussion.[18] However, this very simple action preceeds most editing software available: for most users, this is the condition to start working on any project using computers. In digital music composition contexts, much of the work is done using audio recordings (i.e., samples), a technique that goes back to the early musique concrète days, where sounds where tape manipulation and selection was the innovative aspect of sound manipulation. However, since any digital media is in essence a data structure, it can be generated using algorithms instead of analog-to-digital converters. This applies to all media, and it relates to Manovich's five characteristics of digital media: numerical represetation, modularity, automation, variability, and transcoding. An audio sample is a data structure holding numbers (numerical representation), which can be called many times in different contexts (modularity), to be read or generated by iter-

[18]As an example, consider these two sets of routines that are called for the purpose of copying a file into working memory. The first set of routines are dedicated to (1) opening the file for reading, (2) using the file's header information to (3) get the file size, so that (4) another file is opened for writing, and (5) the corresponding file size is allocated in memory. The second set of routines are geared towards data traversing of the original file, and storing it on the copy. This task is of course automated, and operates on the bit-level, generally as fast as the can manage.

ative processes (automation) either for reproduction or for editing (variability), and which can be interpreted as audio —as it was intended to—, or as any other media type (variability). Furthermore, these characteristics of the digital serve to support —to an extent that will be questioned—Manovich's notion that the "data structures and algorithms are two halves of the ontology of the world according to a computer" (Manovich 2001, p. 223). In other words, for Manovich the computer 'sees' or 'understands' the world by way of a conjunction of these two halves. To the extent that computers are essentially binary constructions, and that the world is something that exists as a collection of images —i.e., images as in strictly sonic or visual constructions—, Manovich's claim can be considered true. However, the problem rises if we consider that the computer as a tool to understand the world, and therefore not as a world-making technology. The world understood with computers is not only one that is presented in binary terms, it is one constructed upon a specific set of data structures. As we have seen with music software —and this can be extended to the history of software development in general—, data structures constitute a fundamental research focus point. What this means is that with better and more efficient data structures, software demands fewer computer resources and runs faster, and is therefore better. Thus, since it can be said that software development is essentially data structure development, the world that is shown through the computer is a constantly accellerating stream of bits. This is to say, at every step of software development, efficiency increments the rate of command execution, and the result is an accelleration of efficiency. In other words, the speed at which the world is shown and interpreted by computers becomes faster in every software iteration. Of course, not all software develops its data structures on every release, but it is a reality that, for example, music software such as Supercollider, Pure Data, RtCMIX, among others, have emerged into the computer music scene with an introduction of a novel data structure, or have introduced changes within their own structures over the years for the purpose of efficiency.

This accelleration is neither linear nor centralized, however geared it may seem towards convergence; it constitutes, especially in open source software development, the negation of Manovich's above quoted claim. For example, consider a programmer —or team of programmers— writing source code for a computer music software. The multiple possibilities of data structure development that are set forth in this situation is only limited by the creativity of the programmer. This is to say that even if the programming language used by the program does not enable the necessary resources from the start, not only other languages can be used, languages can be created for the purpose of such tasks. This is the case of the music software examples created above, where new higher languages were created as interfaces to lower languages like C or C++. This is the case of many sound synthesis libraries written in C or C++ which provide functions and structures for programmers to use not only in music production or analysis —as is the case with Paul Koonce's , or Tzanetakis' Marsyas [19], among many others—, but in software production, that is, as a means for programmers to include a synthesis library within their games, applications, instruments, or any other sound-making software —as is the case with libpd[20], stk[21], among others. Furthermore, the fact that programmers are limited only by their own creativity, defines a fundamental aspect of the performativity that I am describing here. This aspect relates to the capacity to design data structures that reconfigure the way the computer-generated world is constituted. In this way, the programmer feeds back into the computer a notion of world that is then returned by the computer's performance. In this world-generating loop of software development, the ontology of the world as seen by a computer is a reflection of the programming community that is making the software. Thus, through open source development, computers return a multiply-sourced way of world-making. Therefore, Manovich's idea can be easily debunked from two perspectives. On the one hand, the way in which Manovich conveys the world as seen by a computer reflects on Kittler's convergence, discussed above, which implies a reduction of media to digital media and, by definition, a disembodied world-view where the human is an extraneous and residual element. It is no surprise that ten years after writing The Language of New Media, Manovich's warning turn

[19] `https://github.com/marsyas/marsyas`
[20] `https://github.com/libpd`
[21] `https://github.com/thestk/stk`

into a more serious command:

> If you want to escape our prison of software —or at least better understand what media is today—stop downloading apps created by others. Instead, learn to program —and teach it to your students (Manovich 2015, p. 206).

Taking into account the embodied approach, the data stored within data structures cannot be considered information before the instance of incorporation that is essential to any action or performance. Therefore, a world can only appear if it appears to the body. As Hansen pointed out, image creation, or world creation, is not necessarily in contact with the reality that surrounds the body (or the reality of the body), but it is a result of the embodiment of a virtuality that is inherent to our senses, and our proprioception. Therefore, the world is a virtuality that is constructed with our senses, and our body, and as such it is unique to the instance of virtuality. Therefore, the idea of an ontology of the world, that is, a sense of being in the world can only occur through the body and its senses, and this is dependent on the brain's capacity to create the neccessary virtuality for that world to come into being. Under these considerations, data structures turn into very efficient storage devices that have no relation to worlds in themselves, but that are the condition for the possibility of world creating with computers. If it were true that data structures are themselves a representation of the world, then there would be no reason for the world to be there, since it can be endlessly reproduced with incresing efficiency, leaving humans completely out of the picture.

On the other hand, the fact that a community of developers is in charge of designing and implementing data structures and algorithms for computer software makes the idea of 'world' as a singularity come to terms to its own structure, that is, the structure of an idea. This is to say that given that each developer has their own body, and their own capacity for virtuality and world-making, the notion of world making within computers is an imposition. Thus, each program routine or structure is a result of a feedback network that links the history of software development with the practice at hand for which the software is being designed.

For example, when thinking of how to design an oscillator within computer music software development, not only digital signal processing techniques and programming language capabilities are taken into account, also other factors are in play. These are, for instance, previous routine implementations; different types of user interactions; interfaces for which the code will be built; computer hardware differences; and so on. These micro decisions at the code writing level have a tremendous resonance not only in the efficiency of the running code, also in the acoustic and aesthetic results.

What is important to note here, is that these interrelations of what is *already there* in software development can be though of as resonances colliding their way into stability; a stability that emerges not only as a 'stable release' of the code, but also as the condensed multiplicity of worlds that is displaced into a software package. Therefore, far from being an ontology of the world, data structures are world-making and world-revealing devices that engage with our own capacity for virtuality, and thus they are nodes in our world-making networks.

Data structures are the turning point of the history of the database. Their appearance enabled the performance of automated algorithms. This means that within the history of computer technology, the data structures designed by Jon Von Neumann (von Neumann and Burks 1946) proved to be the measure by which computer science was to develop throughout the 20th century. This is how Neumann and his team implemented Alan Turing's original concept for a general-purpose computing machine, that is, what became known as the Von Neuman Architecture:

> Inasmuch as the completed device will be a general-purpose computing machine it should contain certain main organs relating to arithmetic, memory-storage, control and connection with the human operator. It is intended that the machine be fully automatic in character, i.e. independent of the human operator after the computation starts (ibid., p. 1).

In this introductory statement that preceeded their discussion of the electronic computing instrumnet, Neumann et al defined three elements that were to be essential to the computer: automation, memory, and control. All of these elements were considered in this paper as goals for the construction of the electronic hardware that would support them. With the Von Neumann Architecture and its data-based mode, a turning point in the history of databasing was to develop. In the following section, I traverse the history of the database in computer music, in order to find use cases which exemplify instances in which the database is made to be 'sounded,' that is, performed, instead of simply 'read' as an automated and plain computer output. I consider this difference crucial to the concept of databasing that I am developing throught this chapter. Furthermore, the performance and sounding of the database needs to be explained in terms of the existing relationship between memory and the database. For example, in the quote cited above, and as it is present in their 1946 report, the authors focus on memory-storage in their description, expanding on the needs for memory storage at great lenghts. Memory and the relation to the database will be further developed down this text (See 2.2). However, before commenting on Neumann's report, I would like to comment on yet another mention of memory that will later prove valuable. This consists on Manovich's brief introduction of the history of the computer. Manovich's attempt to draw a "historical loop" between the paralell histories of computer and cinema. This loop resides in between Konrad Zuse's 1936 computer prototype —which used a 35mm movie film to store digits—, and the fact that computers now make digital movies (Manovich 2001, p. 26). What is important to note here, however, is the fact that the loop resides within —and is only possible by— the storage device. Of course, memory and storage technology are two concepts that cannot be taken lighly, not only in themselves but also in their interrelation and feedback. Therefore, a more detailed incursion in this relationship will be carried out further down this text. For now, it will suffice to focus on the technology device at hand, that is, the memory module of the computer.

The memory module, or 'organ' in Neumann's terms, is what enables the computer's architecture in the first place. On one hand, the storage unit of the computer allows data to be written and erased in different locations and times. On the other, data itself can be not only values to be used during computation, but actualy the algorithmia itself, that is, the commands —functions, operations, routines, etc.— which are used to access and process data for computation. Thus, it is the interaction of commands and computes what define the data flow inside a the computer. This model is what defines all computer practices, and it is only possible by the data structure holding the data, that is, by the database —from the simplest to the most complex sense of the word.

Neumann explained the need for a storage device in terms of partial differential equation solving. Without delving into the structure of the solutions of these equations, it will be enough to mention that any next value of the solution would depend on the present value. Therefore, when iterating through every step of the solution, the function in charge of solving the equation needs to access the present value, change it, output the next value, and finally update the present value with the outputted result (i.e., the next value). This is what it would look like in pseudocode:

```
present ← 0
next ← 0
iteration {
        output ← next ← function() ← present
        present ← next
}
```

Therefore, in order to provide such solutions, Neumann proposed that:

> Not only must the memory have sufficient room to store these intermediary data but there must be provision whereby these data can later be removed (von Neumann and Burks 1946, p. 3).

In other words, the definition of computer memory is narrowed down to automated ways of inputting, outputting and removing data. Notice, however, that this definition is also effective in a terminology coming from writing, that is, from yet another storage technology: reading, writing and erasing data. Thus, the notion of storage technology when it applies to computers or scripture, varies in a matter of scale. In other words, the difference between writing down a digit, and storing it on a computer is essentially one of scale. In both cases, the number on the piece of paper or the number on some range of the memory slot, what is stored depends on a reading process for it to be of any use at all. Reading —and by reading, I mean not only being able to discern its shape, also to convey a certain meaning that would come of the interaction between the shape, the background, and the overal social context surrounding the paper— a number on a paper can lead to some form of communication, and reading a value on a computer memory can result in the solution of a partial differential equation. However, in essence, it is human memory being displaced into a technological device, that is, a form of *hypomnesis*. Therefore, given the spatial displacement enabled data structures as *hypomnesis* —i.e., the movement from the human to the nonhuman—, to what extent has this displacemente also happened temporally? In other words, how does data structures, in their condition of representing memory outside the human, affect human temporality? The answer can be located in the accelerated rate at which computations have become available, as more efficient hardware and software integrations became available over the years. For exampl,et the 40-bit long 4000 numbers that Neumann et al were aiming at for their storage-memory unit —which was more than plenty for the computational purposes required at the time— represents around 16 Kilobytes, something which today might seem absurd in comparison to current computer storage capabilities that can be found in the case of cloud computing. Furthermore, the accelerated rate not only applies to storage itself, it goes hand in hand with the rate at which computer operations are carried out.

These considerations, however obvious they might seem, touch on the relation between memory and time, and help us reflect on the resonances that occur in between the two when the performance of databases is at hand. In other words, database practice has direct effects on temporality and on memory. Therefore, when designing computer software for art, the way in which data is structured, together with the speed and design of data flow, has significant effects on the temporality of art altogether as a practice. I have proposed that it is the memory and its storing of instructions and information what enables the computer as such. This is certainly not new considering that hardware and software have traditionally formed the computer as we know it, and have developed somewhat equally over the years. However, what I argue here is that since the commands and computes stored in memory —as data— constitute the ontology of the digital, their understanding as a central aspec of database performance leads to a specific consideration of the agency of the database in art. Since any data container can be considered a database, curator Christiane Paul proposes that the database:

> . . . is now commonly understood as a computerized record-keeping system. . . it is essentially a structured collection of data that stands in the tradition of "data containers" such as a book, a library, an archive, or Wunderkammer (Paul 2007, p. 95).

In this sense, Paul suggests that databases are simply an instance of data collection, which in turn is related, as I have hinted above, to writing. However, the fact that the item 'book' is in itself a structured collection of words —with page numbers, indeces, references to other books, etc.— does not preclude the fact that books themselves constitute structured collections called libraries. Extending this concept to all objects, any gathering of things into one place, that is, the practice of collecting, has its resonances on database performance. This is to say that however distant Numismatics[22] is from making a party playlist on Itunes, both of these practices rely on gathering elements of some similar characteristics in one storage device that will be used for display in one form or another. In this sense of collecting, computers become immediately —and quite transparently— useful for the average end user. This means that even if an average user uses the 21st century computer as a typewriter, chances are that text files will gradually accumulate into the database that constitutes the collection of texts written or gathered by the user. And since nowadays computers —and most mobile devices are of course included in this category— exists in constant network connection with the internet, the most extreme case of data collection is taking place with every device having data packages sent back and forth with, for example, Google and Facebook servers. I dont mean by this that a case can be built on the paranoid, panopticon-like grounds of gubernamental surveillance.[23] I argue that if the internet can be considered a place where data is being collected —the use of passive voice is intentional, not to depict an external force, but to

[22] `https://en.wikipedia.org/wiki/Coin_collecting`

[23] There are however cases on which data from Facebook has been used on political campaigns, shifting the discussion of democracy into another kind of democracy that exists in the silent and constant voting that takes place in the exchange of data packets between clients and servers

emphasize on the plurality of data collectors that each user represents—, then any consideration of database performance has to relate to networks as such. This is to say that data and network structures are related, at least to the extent that the latter depend on the former. Furthermore, what the internet data network represents —and this is the central characteristic of databases that Pauls suggested above—, is the concept of the archive. This is why I propose that understanding the relationships between archives and netorks will explain the type of agency that can be achieved by database performance.

1.2.2 Programming

The computer's capacity for data processing and storage is inherent in their constitution. Conversely, a computer without such capacity would cease to be defined as a computer. This capacity that revolves around data has different levels of complexity. In this section, I describe these levels of complexity as a database tree, starting from basic data structures to the more complex database systems.

The ground on which the tree is built is the Von Neumann architecture, that is, the interconnected memory and central processing units and their supporting hardware. As a consequence of different ways of making hardware, that is, different companies, there are different kinds of soil that result in different root layout —for example, different operating systems. The roots of the tree are the data structures, that provide flow between stored data and the above-ground components. The below-ground level is accessed through machine and assembly code, which constitutes the core of low-level programming languages and, to a certain extent, are humanly un-readable: the world of bits. Above the ground, readability by humans is the main feature.

Thus, the trunk of the database tree is composed of programming languages: these go from lowest —ground-level, close to machine code— to highest levels —languages built on top of other languages. These language layers, after they reach a certain level of complexity, begin to form branches that, while being separated from each other, are linked to the same trunk and to the same root. To this category belong programs with simple —mostly text-based— interfaces, such as Bash, C, etc., on one end and, on the other C++, python, and more complex programs built on the former: Pure Data, Supercollider, R, octave, etc. This is a way for the more avid user aquainted with programming languages to interact with the tree as a whole, by way of input and output that does not require the photosynthetic qualities of leaves —explained below. Thus, this is not only how branches are built, but also how they are hacked.

Before continuing, one thing must be noted about this database tree metaphor, which relates to the concept of modularity. The database tree only takes the form of a tree once it is instantiated as a software and when it is run. This is to say that the database tree unfolds every time it is opened, and in this unfolding it exists the possibility of dynamically adapting to different grounds —i.e., programmers can set conditions (a.k.a. macros) so that their programs can compile to different compilers, hardwares, operating systems, etc. Therefore, these database trees have as their main feature the capacity to unfold their roots in different directions upon demand.

When programming languages branch out into more specific, more complex, and more high level branches, the general use of the computer as such is constituted. Most often, these programs come with complex graphical user interfaces which come to be the leaves of the tree. Leaves are a way for the end user to interact with the tree —i.e., where input and output resides, very much like the photosynthesis that takes place within the leaves. These programs may be multimedia editors (Adobe Creative Suite, Microsoft Office, etc), browsers (Chrome, Firefox, Internet Explorer, Safari), mobile apps, etc. These leaves also include database managemet systems (DBMS), which constitute the most traditional business software for data processing and editing: MySQL, PostgreSQL, NoSQL, Oracle, etc. These systems implement different data models (See Table 1.1), whose two fundamental features are their structural specifications of data within a data definition language (DDL) and a data manipulation language (DML) for accessing and updating data (Serge Abiteboul, Hull, and Vianu 1995, p. 4).

An important feature of trees is their network capabilities. Networks can be established both within the same tree —connecting leaves between themselves, or between other branches, or other roots— or within other trees. For example, software can establish a network between its graphical interface an its core program —as is the case with Pure Data, for example. Another example would be the way in which DBMS interact with data: the MySQL database model allows the user to load a data set in working (i.e., RAM) memory, and establishes a connection between the opened memory and the input/output mechanisms.

On another level, networks of trees constitute data streams running in between trees by way of a internet protocols (IP) and a client-server type of relation. This is to say, for example, that while one tree can serve as data storage and processing repository, other trees (clients) can connect to the server tree and request data or processing of data from it. This is the essence of the internet and all the communication services that it enables, such as email services and social networking sites. This is why, for example, both Pure Data and MySQL can have their respective core program and data sets in one computer, and their interfaces on a different one.

Furthermore, combined with computer clusters —gigantic server architectures made out of multiple processing and memory units joined together—, these network protocols form what is known as cloud computing. For examples, most universities provide clusters for data processing —e.g., NYU's Prince cluster— that can be accessed from remote locations. Another use of this type of clusters is cloud storage services —e.g., Google Drive, ICloud, OneDrive, Dropbox, etc.— which are generally used as a way to store and share data.

1.2.3 Database Models

Year	Model	Designer	Instances
1959	Hierarchical		
1960s	Network; Navigational	; ; ;	; ; ; ; ;
1960s	Deductive	J. Minker; L. Kuhns	
1960s	Non-Relational	; ;	; ; ; ;
1970s	Relational	E.F. Codd; P. Chen (1976)	; ; ; ;
1975	Semantic	U.S. Air force; J.H. ter Bekke (1991)	
1980	Graph	; ;	; Oracle Spatial and Graph; ; Amazon Neptune; ;
1985	Object	Brown University; Texas Instruments; Bell Labs;	GemStone (Smalltalk); Gbase (LISP);
1990s	Semi-structured	W3C	XML;
1995	In-Memory	Oracle; Sybase; SAP; Exasol AG; VMWare	TimesTen; ; ; EXASolution; WebDNA

Table 1.1: Database model development timeline with examples.

The central aspect of database management systems (DBMS) is the database model. In essence, all DBMS share the same function: provide access to a database.[24] Database models have been thought of as collections of conceptual tools to represent real-world entities and their relationships (Angles and Gutierrez 2008, p. 1). Building upon the work of E. F. Codd —an IBM programmer who revolutionized the field of database systems with Relational databases—, Angles and Gutierrez name the three most important aspects a data model should address: data structuring, description, and access. Therefore, any data model "consists of three components: a set of data structure types, a set of operators..., and a set of integrity rules" (ibid., p. 2).

The first component consists of the core aspect of the model: data structures. These are the building blocks upon which the entire database model is designed. A data structure is simply a way to organize data so that a set of element operations are possible, such as ADD, REMOVE, GET, SET, FIND, etc. Data structures can be thought of in two ways: on one side, they are thought of as interfaces, or better, as *abstract data types*:

> An interface tells us nothing about how the data structure implements these operations; it only provides a list of supported operations along with specifications about what types of arguments each operation accepts and the value returned by each operation (P. Morin 2019, p. 18).

In other words, the abstract data type is the idea of the structure, and the possibilities that this idea sets forth. On the other hand, these abstract data types are then implemented in code, which is how the speed and efficiency of the data structure is physically evaluated. An implementation of this sort includes

> ...the internal representation of the data structure as well as the definitions of the algorithms that implement the operations supported by the data structure (ibid., p. 18).

In this sense, data structures have constituted a focal research point in the database and

[24] The qualities of this access, however, have a direct influcence on the quality and success of the business to which the DBMS is applied to. For example, if the database system in charge of airline reservations is not immediately updating, the results would be detrimental for the company —e.g., empty seats not being filled— or for the customers —e.g., double-booking.

computer science communities.[25] The second component —the operators— constitute

In scholar Serge Abiteboul's words, a database model

> ...provides the means for specifying particular data structures, for constraining the data sets associated with these structures, and for manipulating the data (Serge Abiteboul, Hull, and Vianu 1995, p. 28).

Thus, in order to arrive at a specific structure that would enable constraints to the model, database designers create two types of languages. On one hand, they device a data definition language (DDL) which is dedicated to the particular tailoring of data types —commonly known as 'typecasting' in C, a process by which the programmer establishes the kinds of variables that will be used within a program, e.g., 'integers,' 'floats,' 'strings,' etc. By limiting the input type, the programmer introduces constraints to the database, which allow for a more efficient and less error prone data management. On the other hand, in order to manipulate the data, programmers generally device yet another language, known as the data manipulation language (DML). These two languages are common to most database systems developed to date.

For Abiteboul, the most prominent models are Graph, Network, Semantic, Object, Hierarchical, and Relational (See 1.1). Each model differs greatly in its structuring, definition, and maintenance. Furthermore, the evolution of these models has gone hand in hand with both hardware improvement and with the rd growth of data sets over the years, to the point that many databases are simply available online.[26] The following section contains a brief description of the main database models developed throughout the second half of the XXth century until today.

[25]For more on data structures, see (P. Morin 2019)

[26]For a list of online databases, see: `https://en.wikipedia.org/wiki/List_of_online_databases`

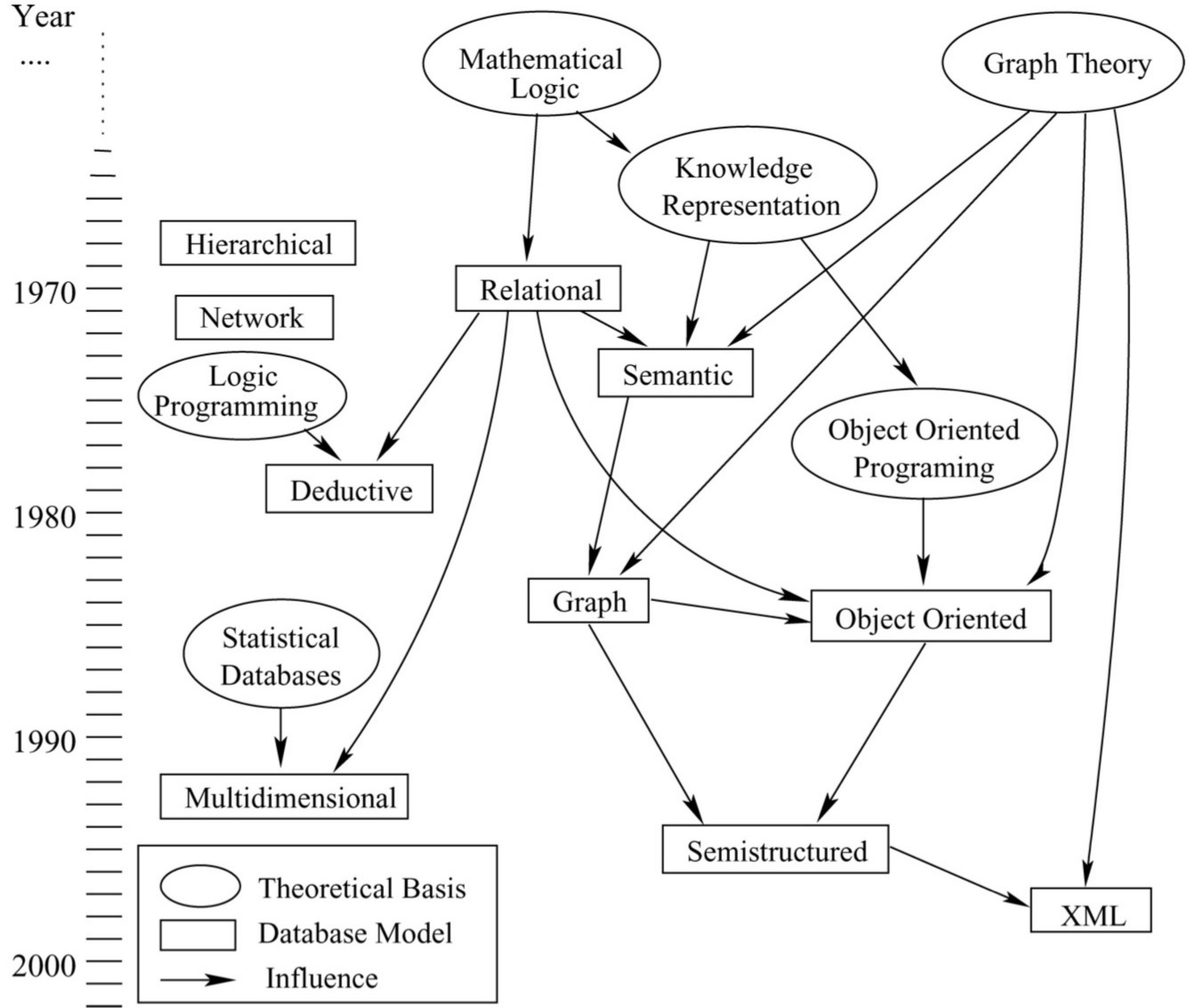


Figure 1.1: A graph showing the evolution of data models taken from (Angles and Gutierrez 2008, p. 2): "Rectangles denote database models, arrows indicate influences, and circles denote theoretical developments. A time-line in years is shown on the left (based on a diagram by A. O. Mendelzon)"

Hierarchical

This is the first database model ever created. It was created by IBM during the early 1960s, in conjunction with two other American manufacturing conglomerates (Rockwell and Caterpillar) for NASA's Project Apollo. The result of this project was IBM's Information Management System (IMS).[27] This model is sometimes called a tree structure, where each branch of the tree constitutes a hierarchical path from the root node to the end node (Long et al. 2000). Nodes, or records —understood as collections of single-value fields— are interconnected by way of these paths, or links. Records can have type definitions, which determine the fields it contains. As a rule of this structure, a (child) record can be linked in one direction to only one (parent) record and in another direction, to one or more (child) records. This structure stems from a root record, which is the initial (parent-less) record that begins any traversing of the structure for retrieval. The relationship paradigm of this model is one-to-many.

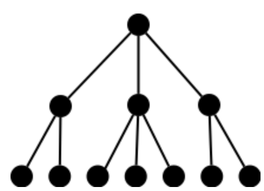


Figure 1.2: Diagram of the hierarchical model

Their use was eclipsed by the relational model during the 1980s, but it resurfaced through relational-type implementations of hierarchical models, and with the appearance of Semi-structured model in the late 1990s. Most path structures within computer file systems implement a hierarchical model. For example, not only Microsoft's filing system, also the Windows Registry constitutes a hierarchical database.[28] Programming and software in general are thought as hierarchic structures. For examples, the terminology 'higher-' and 'lower-level' corresponds to a hierarchical model of thinking code execution: the higher the command, the more child processes it calls,

[27] https://en.wikipedia.org/wiki/IBM_Information_Management_System

[28] https://docs.microsoft.com/en-us/windows/desktop/SysInfo/structure-of-the-registry

computations it encompasses, and resources it demands; the lower the command, the least child processes it calls, computations it encompasses, and resources it demands.

Network

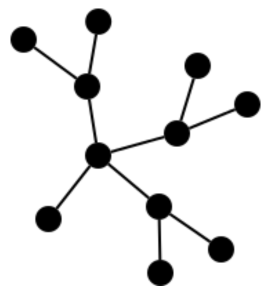


Figure 1.3: Diagram of the network model

Invented by Charles Bachman in 1959 and published by the Conference on Data Systems Languages Consortium ([29]), the network model is a way of representing objects as nodes in a graph whose relationships can be represented as arcs. The programming language was designed for the implementation of network databases. The nodes in these networks are known as ‘records,’ and their relationships form ‘sets’ that have one-to-many relationships in between records, that is, one ‘owner’ and multiple ‘members.’ Their main feature was that these relationships are not bounded to any hierarchical or lattice-like structures, providing a more natural way of record relation. Structurally, the each node has an identity —i.e., a database key— which corresponds to the pointer to the physical address of the record on disk. These keys can thus be used to implement linked lists and trees for record navigation, allowing for very fast retrieval speeds due to the interlocking of the physical implementation and the internal logic of node identity and access. Despite the fact that Relational databases came to overshadow network models, some DBMSs today —such as Unisys OS 2200[30]— enable a networked view of its contents.

[29] `https://en.wikipedia.org/wiki/\acrshort{codasyl}`
[30] `https://en.wikipedia.org/wiki/Unisys_OS_2200_databases`

Navigational

> My proposition today is that it is time for the application programmer to abandon the memory-centered view, and to accept the challenge and opportunity of navigation within an *n*-dimensional data space (Bachman 1973, p. 657).

This is a class or type disk-based database systems —in contrast to magnetic tape or punched card systems— that enabled a different way of thinking database navigation. Charles Bachman (ibid.) was the first to conceptualize this class, which was implemented in the network model (See 1.2.3) of his Integrated Data Store (IDS), developed for General Electric. The concept behind this navigation is that of thinking data records and attributes as *n*-dimensional space. This means that a database can be traversed not only by accessing the first element —i.e., the head of a list, as in a linked list— and then moving sequentially to the 'next' record, or by using database keys —i.e., a pointer to a record— and primary data keys —i.e., a unique identifier assigned to each record by hashing or indexing. Bachman's programmer-as-navigator could create secondary data keys, sets made out of these secondary keys, and navigate through those sets starting from any of its members. In other words, given a database with records and attributes, all atributes can become a new dimension (i.e., a new set of primary data keys used for navigation), making retrieval times much more efficient. Therefore, navigating through a database within this paradigm becomes following record relationships —i.e., data strcucture sets— instead of record order in storage.

With this navigational paradigm, a new level of abstraction was given to database management systems which, while not arriving at the level of complexity that Relational models got to with relational algebra operations, resulted in better and more efficient database retrieval. In this sense, the main difference between navigational and relational databases can be seen from the way in which users formulate queries: while in the latter users specify what needs to be found in terms of a logical comparison between record sets, in the former users specify what steps need to be made in order to arrive at a certain record. While navigational databases are closely related to the network model, the **??** model is also a structure that can be paired with this navigational paradigm. For example, the Document Object Model (DOM) is an that provides access to the different elements of a webpage, for example, using Javascript to traverse an HTML document. In general a DOM contains a hierarchic structure, but this structure can be access using this navigational paradigm. During the 1980s, navigational databases were eclipsed by the relational model, but after the 1990s, they re-emerged as, for example, Non-Relational databases.

Relational

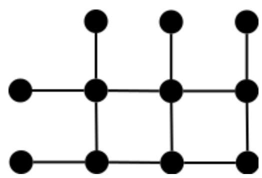


Figure 1.4: Diagram of the relational model

The relational model was first designed by E. F. Codd (Codd 1970; Codd 1972). The main feature of relational databases is a table-like organization of the data, together with a separation between the physical level of data storage and the abstracted language, allowing users to achieve highly complex data manipulations by way of a simple query language based on relational algebra. Data is placed into uniquely identified rows —also called records or tuplets— which can have multiple columns —also called fields or attributes. A table thus becomes a relation, that is, a set of tuples with the same attributes. Relationships can be established in between different tables and the logical interaction of instances and values. Deductive database are advanced forms of relational databases (Ramakrishnan and Ullman 1995).

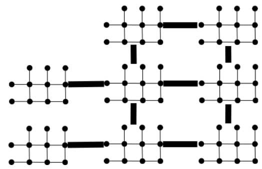


Figure 1.5: Diagram of the extended-relational model

Two models stem from the relational model: the extended-relational (a.k.a. RM/T) and the entity-relationship (ER) models. The former was designed by Codd as an improvement to his first model, and the latter was designed by Peter Chen in 1976.

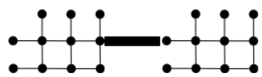


Figure 1.6: Diagram of the entity-relationship model

Within the entity-relational model, two entities —i.e., relations, or tables— are connected by a relationship —i.e., a link— of the type ‘is a,’ establishing a more complex way of joining relations that provide meaningful insight on the way data is organized. In this sense, entities can be thought of as ‘nouns,’ while links or relationships can be thought of as ‘verbs.’ In essence, both relationships and entitites constitute ‘relations’ in themselves —i.e., in the same sense that they contain rows and attributes—, but the complex structures that derive from the types of relationships proposed by Chen can lead to clearer entity sets that convey meaning of the data in a more natural —i.e., language-like— way.[31]

Semantic

Semantic Databases developed in the 1970s by the US Air Force, and a model was deviced by Bekke (Bekke 1991) which was implemented in the Xplain DBMS[32]. Two types of semantic databases come to represent the Hierarchical and Network models.

[31] For more information on relational database management systems, see `https://en.wikipedia.org/wiki/Comparison_of_relational_database_management_systems`

[32] `http://www.jhterbekke.net/XplainDBMS.html`

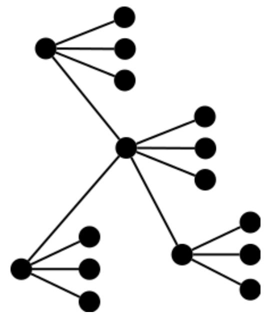


Figure 1.7: Diagram of the semantic network model

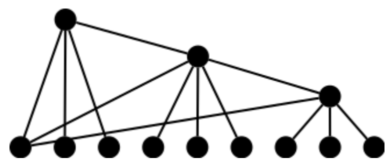


Figure 1.8: Diagram of the semantic hierarchical model

Non-Relational

This is a more generall class of database models where the internal structure is other than the tabular kind that the Relational presents. The NoSQL acronym tends to define these databases — i.e., meaning 'non-' or 'not-only-' sql—, since SQL is the most commonly used language used for querying and updating relational databases. Within this class or group of non-relational models, some examples can be:

- Key-Value databases, which are centered on associative arrays (hash tables): such as dictionaries.
- Semi-structured databases, also called document-oriented databases: such as XML, YAML, JSON, etc.
- Graph databases

- mixed graph models: such as the way in which the World Wide Web convention (W3C) structures websites, with a URL as a 'name' and their content as a 'graph'
- Object databases
- mixed models, among others.

Graph

> Graph database models can be defined as those in which data structures for the schema and instances are modeled as graphs or generalizations of them, and data manipulation is expressed by graph-oriented operations and type constructors (Angles and Gutierrez 2008).

In their survey of graph modelled databases, Angles and Gutierrez (ibid.) date the beginning of graph databases to the early 1980s, in conjunction with Object.

In-Memory

A DBMS relying on main memory (RAM), contrasted with others that rely on disk storage mechanism. They are faster and well suited for data analysis, but since they depend on main memory, they are subjected to RAM's volatility —i.e., non persistence, e.g., in the presence of a power failure. In cloud computing terms, in-memory databases (IMDBs) would enter the realm of 'hot' data storage —for fast retrieval and modification—, in contrast to 'cold' data storage —for less frequent update.[33]

Object

[33] For more information on in-memory databases, see `http://web.archive.org/web/20121009191558/http://slashdot.org/topic/datacenter/the-rise-of-in-memory-databases/`

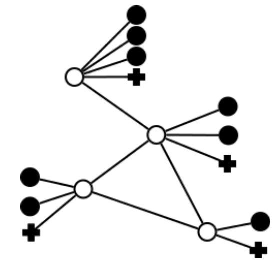


Figure 1.9: Diagram of the object model

These databases combine the object-oriented programming paradigm with database concepts. On one side, each record is treated as an object, with capability to store variables —i.e., attributes— and methods —i.e., functions or routines that the object can perform. This way, when an object is instantiated in the form of a record, all the attributes and methods become available to itself —and to other objects, provided these are setup in a 'public' way—, and so different interactions can occur throughout the database. Some programming languages are directly object-oriented, from which certain databases were created (See 1.1). From 2004, the open source community has been developing open source object databases that are easily accesible in several object-oriented languages.

Semi-structured

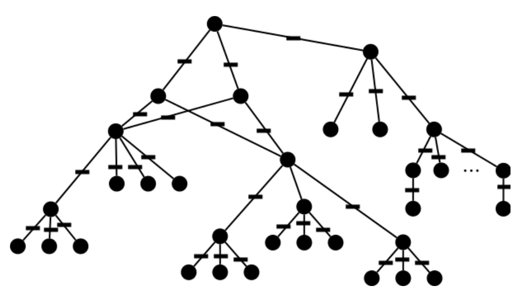


Figure 1.10: Diagram of the semi-structured model

> We call here semi-structured data this data that is (from a particular viewpoint) neither raw data nor strictly typed, i.e., not table-oriented as in a relational model or sorted-graph as in object databases. (S. Abiteboul 1996)

Abiteboul (S. Abiteboul 1996) comments that given the amount of data that has grown in non-standard structures, a new way of accessing data has emerged. Furthermore, access to data can take place from a variety of different platforms —i.e., browsers, query languages, application-specific interfaces, etc.—, making the process of obtaining useful information increasingly more difficult since these platforms call for specificallyt tailored methods and languages. Abiteboul claims, therefore, that first there is a need to extract the non-standard structure from the data, so that it can be traversed afterwards. These databases constitute the semi-structured model. Some examples of this model include XML databases, JSON files, YAML files, among others.[34] A known database of this kind is the Internet Movie Database (IMDB).[35]

1.3 Databasing Sound

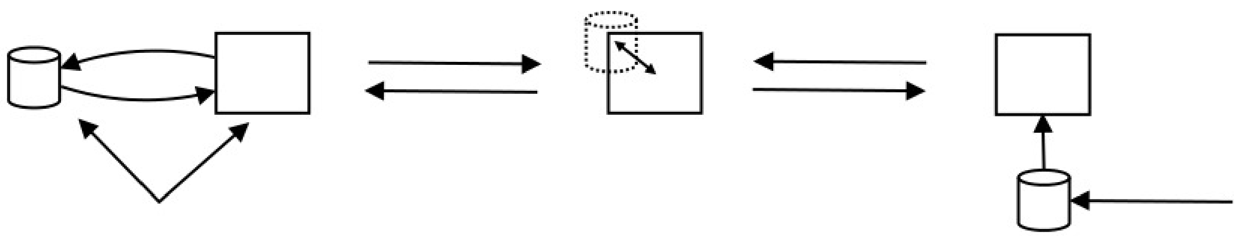


Figure 1.11: The metaphorical positioning of the database in relation to computer music practices. The arrows between databases (cilinders) and computers (squares) represent data flow. Left: the database is 'visibly next' to the computer, as is the case with Music Information Retrieval (MIR); the two bottom arrows indicate the intervention of the human operator. Right: the database is 'visibly below' the computer as is the case with Sonification; the database feeds the computer from an external source (right arrow). Middle: the database is 'invisibly behind' the computer, within the softwares used for (and as) music works. The arrows between MIR and composition, and between composition and sonification, represent interdisciplinary feedback.

So far I have reviewed Manovich's declaration of the database as a new creative center in new media art, and I have taken Hansen's criticism on Manovich's disembodied arguments, iden-

[34] For more infomration on semi-structured databases, see (Buneman 1997)

[35] `https://www.imdb.com/`

tifying in Manovich a silent allegiance to Kittler's posthumanism. This interlocking between the identification of data as information and —as a consequence of this— surrendering human agency to technology, has been overlooked by the next generation of writers who followed Manovich's call for 'info-aesthetics' in their 'database aesthetics.' My intention in this section, rather than being focused on where these artists and theorists of the aesthetics of the database have gone wrong, it is centered, however, on the question of how database practices have been considered in new media over the years. While Hansen's emphasis on visual and virtual art is well justified within his text,[36] this is not the case within the collection gathered in *Database Aesthetics*. Therefore, what I intend to describe next is the missing element in the reviewed literature of the database, namely, a theory of the use of databases in relation to sound.

The database has been present in the music literature as the silent partner since the first computers were used to make music. For a figure representing the position of the database in relation to music practices involving computers (See 1.11). Through the 1990s the use of computers —and databases— can be found in diverse music fields such as computer assisted composition (CAC), electroacoustic music, computer music, sonification, music information retrieval (MIR).

For now, I use the words 'database' and 'computer' somewhat interchangeably. I will provide a more acute definition of the database at the end of this chapter. However, the decision is not random, since the database is itself the condition of possibility of the computer. What does this lead to when we can speak of database music? Is computer music —or, better, all music made with computers— database music?

1.3.1 Music Information Retrieval

In music information retrieval (MIR), the database is in front of the programmer, and has a crucial position in the field, but in a very different way. This practice has been present in academia for

[36]He goes through great pains to prove his theory of body and affect within new media while dedicating himself solely to what he considers 'the most disembodied' of the senses (Hansen 2004, p. 31).

a while[37], most generally within electrical engineering departments. The objective of MIR is the analysis of sound signals in order to obtain useful information. Therefore, out of an audio file which contains millions of audio samples[38], a database of a handful of feature descriptors[39] is obtained in order to identify the file. Thus, by reducing the dimensionality of data-points to a smaller number, MIR defines an easily navigable feature space.

Information Space

The space of navigation is often referred to as 'information space,' which essentially consists of a database of feature descriptors that can be navigated. The purpose of this navigation, for MIR is still being explored. However, this type of database navigation has been used to obtain information to perform automatic tasks such as categorization (for recommendation systems), for track separation or instrument recognition, for automatic transcription, among other uses. What this feature space consists of, however, is a *data-set* —as it is commonly referred to in the literature— holding an identification tags —i.e., a set of 'id's— of a certain audio file, and its description vector. Over the 18 years of the ISMIR conference, for example, about 30 databases of this sort have been publicly created.[40]

Applications of MIR

Among the many analytical uses of MIR over the years, I would like to point to the Semantic Hi-Fi system, under development for at IRCAM under the supervision of Hugues Vinet (Vinet 2005). It

[37] See for example the International Society for Music Information Retrieval (ISMIR), founded in 2000

[38] Depending on the sample rate of a recording (usually 44.1 or 48 kHz, that is, samples per second)

[39] A descriptor is stored data that identifies other stored data, and in this case, a feature descriptor relates to the value(s) of a certain characteristic (spectral centroid, brightness, etc.) of the analyzed audio file. Descriptors will be developed at the end of this chapter.

[40] For further reference, the following citations point to different audio databases which have been created over the years: (Goto et al. 2002; Goto et al. 2003; Wüst and Celma 2004; Maxwell and Eigenfeldt 2008; Bertin-Mahieux et al. 2011; Karaosmanoglu 2012; Jaimovich, Ortiz, et al. 2012; Mital and Grierson 2013; Bortz, Jaimovich, and Knapp 2015; Jaimovich and Knapp 2015; Nort, Jarvis, and Palumbo 2016; Defferrard et al. 2017; Vigliensoni and Fujinaga 2017; Meseguer-Brocal, Cohen-Hadria, and Peeters 2018; Donahue, Mao, and McAuley 2018; Xi et al. 2018; Wilkins et al. 2018)

consisted of a database system aimed for content-based querying of audio files, which would enable DJ's to browse through files, apply beat-synchronized transitions between them, and many other automated tasks. The project lasted for over 3 years, and stemmed from the research of another previous project by Vinet at IRCAM, the CUIDADO project (Content-based Unified Interfaces and Descriptors for Audio/music Databases available Online) (Vinet, Herrera, and Pachet 2002a; Vinet, Herrera, and Pachet 2002b).

Although an emphasis in open source database creation has gained momentum in recent years (Fonseca et al. 2017), databases are mostly created under restricted licences. Some of the uses that MIR has given for the database have been:

- to create digital libraries (Dunn 2000)
- to store actual music notation (Good 2000)
- for audio classification and clustering (C. Yang 2001; Homburg et al. 2005)
- for the evaluation of multiple-source, fundamental frequency estimation algorithms (Yeh, Bogaards, and Röbel 2007)
- to describe performance expression (Hashida, Matsui, and Katayose 2008)
- for genre recognition and classification (Xu, Zang, and J. Yang 2005; Jr., Koerich, and Kaestner 2008; Sanden, Befus, and Zahng 2010)
- for structural analysis (J. B. L. Smith et al. 2011)
- for contextual music listening pattern detection using social media (Hauger et al. 2013)
- to train models for phoneme detection (Proutskova et al. 2012)
- for schenkerian analysis (Kirlin 2014)

- for tonal music analysis using generative theory of tonal music (GTTM) (Hamanaka, Hirata, and Tojo 2014)
- for counterpoint analysis (Antila and Cumming 2014)
- for emotion recognition and color associations in the listener (Pesek et al. 2014)
- for melody extraction (Bittner et al. 2014)
- for harmonic analysis (Devaney et al. 2015)
- for the evaluation of tempo estimation and key detection algorithms (Knees et al. 2015)
- for orchestration (Crestel et al. 2017)
- for computational musicology[41] (Parada-Cabaleiro et al. 2017)
- for training and evaluating chord transcription algorithms (Eremenko et al. 2018)
- for multi-instrument recognition (Humphrey, Durand, and McFee 2018)

1.3.2 Sonification

To a certain extent, the database is the ground floor of sonification, or what allowed sonification to emerge as a practice:[42]

> Sonification. . . seeks to translate relationships in data or information into sound(s) that exploit the auditory perceptual abilities of human beings such that the data relationships are comprehensible (Walker and Nees 2011, p. 9).

[41] Within musicology, an early use of the database can be found in the digitization of the Bridgman file by Hèléne Charnassé in 1980 (Charnass 1980), which consisted of the extensive annotations made by Madame Bridgman in the Paris Bibliothéque Nationale, of Renaissance polyphonic music from 1420-1520

[42] *The Sonification Handbook* (Hermann, Hunt, and Neuhoff 2011) continues to be the most recent and comprehensive reference in the field.

The ‘data’ that is sonified is very likely to be digital[43], which means that data needs to be stored in a structured way for fast access by computers. Therefore, in order for sonification to take place, the data needs to be anchored in a database. According to Walker and Nees, there are three types of sonifications: event-based[44], model-based[45], and continuous[46].

Sonifying the Database

All of the above mentioned ways of sonifying data depend on pre-existing data —i.e., data before sound—, and thus, these practices depend on databases: on the interaction between databases, and on their traversing. This dependency on databases notwithstanding, sonification literature has been focused mainly on the interaction between technics and the body. What this means is that the developments in sonification have always been guided by principles of psychoacoustics, and thus shaped by the limitations of the human body’s capacity to listen.

In this sense, sonification is

> . . . a relatively recent subset of auditory displays. As in any information system[,] an auditory display offers a relay between the information source and the information receiver. . . In the case of an auditory display, the data of interest are conveyed to the human listener through sound (Walker and Nees 2011, p. 10).

What sonification as a practice has taken into account is the auditory system’s evolved ability to extract biologically relevant information from the complex acoustic world.[47] What this emphasis on sound perception and cognition mainly attests to is the fact that there is no one-

[43] While there are cases where sonification is entirely analog (e.g., see the first sonification tool ever created: the Geiger counter), I will focus on the most recent use cases of sonification.

[44] This is also referred as ‘parameter mapping,’ meaning that some dimensions of the data are mapped or translated into some parameters of sound (frequency, periodicity, density, etc.). I use the word ‘some’ to represent the necessity of data dimensionality reduction. Parameter mapping is most useful to represent changes in the data (Walker and Nees 2011, p. 16).

[45] Model-based sonification is useful for what is known as ‘data exploration.’ The sonification is driven by the interaction of the user with a virtual model of the data (ibid., p. 17).

[46] This is also known as ‘audification,’ and it consists of directly translating waveforms of periodic data into sound (ibid., p. 17).

[47] See for example Carlile’s chapter on psychoacoustics (Carlile 2011).

to-one correspondence between sound parameters (e.g., frequency, amplitude, spectral content, etc.) and how these are perceived (e.g., pitch, loudness, timbre, etc.). Therefore, the success of a sonification is a result of the play between, on the one hand a rigid link between data and sound, and on the other, the perceived acoustic relations. From these acoustic relations, then, information can be obtained of the data. In other words, in sonification practices there is no communication unless the data has been acoustically shaped, and perceived as information (or better, perceived *into* formation) by the listener. In what follows, I present some instances of sonification practices as described by their authors.

Parameter mapping

David Rossiter and Wai-Yin Ng (Rossiter and Ng 1996) sonified the Dow Jones financial stock market data with Csound. Since the Csound program depends on two separate files —i.e., an *orchestra* file holding the code for the music synthesis instruments, and a *score* file holding the code for the sequences of parameters for the instruments—, they implemented another program to control the data flow. Within this second program, the Csound score was automatically generated based on a 'configuration' file which was used to map the 'data file' —holding the stock market data— as it was read —in separate window frames— into the Csound-formatted score.[48]

Artistic sonification

Judy Klein composed in 1998 a piece called *The Wolves of Bays Mountain*, using a set of recordings she took along the Bays Mountain Park in Kingsport, Tennessee, for a period of six months. In this period she researched the sonic activity of a pack of wolves, and in her recordings she achieved a level of intimacy with the pack that, as a result, translated into a strong animal rights activism. Therefore, her compositional choice was to treat the sound file in a non-destructive and

[48] Other examples of stock market sonification include Ciardi's set of tools for downloading and sonifying real-time data (Ciardi 2004), and Ian Whalley's research on telematic performance (see Affective Sonification).

non-intrusive way, thus analyzing spectral contours of extremely precise frequency bandwidths of the data and resynthesizing into the soundscape in almost unnoticeable ways.[49]

Natasha Barrett (Barrett 2000) composed an electroacoustic work called ‘Viva La Selva’[50] using 14-hour long recordings taken from a forest[51], with an array of four microphones. From these recordings, she extracted location —by difference in arrival time— and timestamps —by manual logging— of different animal sounds, and long-term energy distribution in various frequency bands —to describe various environmental sounds such as airplanes, wind, insects, etc. While the spatio-temporal data of the animal sounds was used for sound spatialization of sounds within the electroacoustic work, the long-term energy distribution was scaled down to 20 minutes so as to constitute the form of the piece.

Bob L. Sturm (Sturm 2002) sonified ocean wave conditions of the US Pacific coast obtained by the Coastal Data Information Program (CDIP) since 1975. The database until 2002 contained over 50 gigabytes (GB) of spectral and directional content of the wave-driven motions at the location of the sensing buoys. By scaling to hearable range and then performing an Inverse Fourier Transform (IFT) of the data, Sturm composed a piece called ‘Pacific Pulse,’ on which frequency sweeps indicate storms beginnings (rising) and endings (falling).

More recent examples of artistic sonification include Nichols et al (Nichols et al. 2014) with their sonification of river data as a multimedia collaboration. Falk Morawitz (Morawitz 2016) used molecular sonification in his piece https://soundcloud.com/falk-morawitz/spin-dynamics-stereo-reduction, by two audification processes —direct audification and via a straightforward additive synthesis process— applied to the Human Metabolome Database (HMDB)[52], a database holding nuclear magnetic resonance (NMR) spectroscopies of molecules.

[49] From a lecture given by Judy Klein at New York University’s Waverly Project, on February 2nd, 2017.
[50] `http://www.natashabarrett.org/viva.html`
[51] A biological field station called ‘La Suerte’ in Costa Rica
[52] `http://www.hmdb.ca/`

Sonification Software

The following software was selected from the many sonification tools built over the years.[53]

Originally intended for sonification purposes, SonART (Ben-Tal et al. 2002) was an open-source platform that enabled users to map parameters to sound synthesis, and later (W. Yeo et al. 2004) to obtain cross-correlated image and sound synthesis. In other words, users were able to easily translate a database into sound parameters, or image and sound data into one another. The program acted in a modular way, that is, it was networked with other software via OSC connections. This software enabled Berger and Seung Yeo (W. S. Yeo and Berger 2005) to generate novel image sonifications, by combining the *scanning* —data sonified in a fixed, non-modifiable order— and *probing* —sonification data points are selected by the user— methods into one interface.

Cadiz et al (Cádiz et al. 2015) proposed a sonification approach based on statistical descriptors of regions of interest (ROI) selected from medical images. In their study, they focused on enhancing breast cancer symptom detection in mammograms by mapping statistical descriptors —e.g., mean, minimum, maximum, standard deviation, kurtosis, skewness, etc.— to different synthesis techniques in various ways. They then surveyed the usefulness and pleasantness of the sonifications to different subjects in order to better adjust the technique to the task. What is novel of their approach is on the creative use of statistical curves obtained from pixel distributions within computer music techniques.

In his computer aided data-driven composition (CADDC) environment called DataPlayer —programmed as a standalone MAX/MSP application—, Nardelli (Nardelli 2015) sonified data from the AFLOWLIB[54] —a database of material compounds for which he also designed a separate Python script to map data parameters to MIDI values. His sonification intent was aimed towards data navigation, by means of a unique mapping that would convey an overall trend —i.e.,

[53] For further reference, see: http://www.icad.org/websiteV2.0/Conferences/ICAD96/proc96/lodha.htm software. Pauletto and Hung (Pauletto and Hunt 2004) designed The Sonification Toolkit. Joao Menzenes () designed https://www.joaomenezes.net/sondata using MAX/MSP.

[54] `http://aflowlib.org/`

a gist— of each material compound. Furthermore, this environment allowed for artistic remixing and exploration of the sonification procedures, simultaneously touching on the scientific use and the expressive value of databases.

Hamilton et al (Fox, Stewart, and Rob Hamilton 2017) deviced *madBPM*, a data-ingestion engine suitable for database perceptualization, i.e. sonification and visualization. This modular C++ software platform enables data loading from CSV (Comma-Separated Values) files, multiple mapping via tagging, several traversing algorithms and units, and networked connectivity to SuperCollider —for sound output— and OpenFrameworks[55] —for visual output. Their approach is innovative since they provide features for database behaviors. By 'behavior' they mean ways of structuring, traversing and perceptualizing the database. These behaviors define the dual purpose of the software: finding relationships among the inputted data and interpreting them artistically. Furthermore,

> . . . by asking users to explicitly define the data parsing and meta structure (program-level logic), the platform is flexible enough to allow work with both un- or pre- processed datasets, structured or unstructured data, or multiple forms of data segmentation and tagging. At the current state of the project, these objects are still defined in C++ source code and compiled into the platform (ibid., p. 504).

In this sense, users can structure and re-structure potentially any type of data set. However, in order to design new behavior objects the user needs to implement them in the source code and compile them. Thus, besides real-time data streaming and networking functionality, in their future work the authors aim at designing a Domain Specific Language (DSL) that would enable extending the functionality of these behaviors in real-time.

Sonification Installations

Mark Ballora et al (Ballora et al. 2010) sonified a database of HTTP requests at Penn State's Center for Network Centric Cognition and Information Fusion. This database contained entries with four

[55] OpenFrameworks is a C++ program for visuals: `https://openframeworks.cc`

fields —timestamp, location (latitude-longitude), IP address, and response type— which they used as time and spatialization (for the first two), and different pitch and timbre representations, from the more concrete —IP values mapped to frequency— to the more abstract: IP mapped as a formant filter, and as a high-pass filter for a brown noise generator. Their aim was to build a soundscape with different but simultaneous sonifications of the data, so that the resulting work could run as in the background generating a pleasing —but informative— experience.[56]

Schlei and Yoshikane (Schlei and Yoshikane 2016) proposed a novel way to generate waveforms by populating an array using vertex data obtained from the GPU (Graphics Processing Unit). In order to carry this out, they used the https://developer.apple.com/documentation/metal —Apple's built-in framework to interface with the GPU—, and intervened on the processing pipeline to output CPU (Central Processing Unit) accessible data. Thus, the audio engine running on the CPU was able to interpret as waveforms the values of the vertex and fragment shaders —i.e., the position data related to a rendered shape and the pixel values respective to its display, respectively. Therefore, they obtained simultaneous visualization and audification of the rendered three dimensional shape. In their installation https://vimeo.com/167646306, they used the generated waveforms as a database, composing each waveform together with their visual generators as a collage.

More recent installations include PerMagnus Lindorg's sonification of real-time earthquake data as a sound sculpture (Lindborg 2017). Within *Pacific Bell Tower*, Lindborg used data from the IRIS[57] Data Services, which transmits seismographic data packets updated every 30 minutes from multiple observation sites. He spatialized using coordinates of the events using a four-speaker array located at the center of the gallery space, and mapped the rest of the data to FM synthesis parameters. Camara Halac et al (Camara Halac, Delgadino, and Simonelli 2017; Camara Halac 2018b) designed an installation based on face-tracking and real-time sonification of spectral

[56]Ballora's interface and sonification design stems from his PhD dissertation on cardiac rate sonification (Ballora 2000).

[57]`https://www.iris.edu`

features present in both pixel information containing the face, and the x and y coordinates of the moving data points of the face mesh used for tracking.

Model-based sonification

One example of model-based sonification is the *Data Listening Space* project by Vogel and other members of the QCD-Audio[58] project (Vogt et al. 2012) at the Institute of Electronic Music and Acoustics (IEM) of the University of Music and Performing Arts, Graz. Within this installation, they proposed a three dimensional, navigable space holding a Monte Carlo simulation of the theory of quantum electrodynamics (QED). Within this 'lattice QED,' a walking participant holding sensors —x,y, and z coordinates— could explore the simulated data by way of sonification.

Affective Sonification

Ian Whalley (Whalley 2014) carried out an embodied approach to data sonification in his research on telematic electroacoustic music. He claims that although sonification is well established as a practice, the role of the human body when it concerns data streams has been traditionally left out.

In this sense, telematic performance, which is for Whalley an expression of the extent to which human life is mediated by data streams, suffers from a (disembodied) limitation. Whalley's solution consists in emphasizing on the micro-gestural quality of music performance. Thus, he designed parameter mappings for sonification of real-time stock market data basing parameter constraints on actual human performers —musicians who were used as baseline for an affective sonification mapping.

[58] `https://qcd-audio.at`

1.3.3 Computer Music

In the field of computer music, the database is behind the music software. In other words, because computer music programs are built on programming languages that use data structures, the composer using the program is indirectly —but nonetheless still thinking in terms of— data structures. The question is then, What differentiates this indirect use of the underlying structure of the database in composition with other sound practices using computers? In order to address this, I will go through a set of articles that show how composer programmers have shaped music concepts within programming concepts. My argument is that given that the technical constraints a software presents are dependant on programming decisions, and in turn, that these choices are a function of the structure of the programming language, the overall result is a change in the performativity of the composer. In other words, the way in which the composer plays, interacts, creates, and performs, is reconfigured by the structures of the database. However, in no way this means that the performativity of the composer is technologically determined. What I am stressing here is that the factor that weights in this network of constraints is the contingency of the programmer's choice. Therefore, what the database behind the software stands for is the plethora of possibilities that can stem from its structuring.

SSSP and the question of complexity

For example, consider William Buxton's work on the Structured Sound Synthesis Project (SSSP) in Toronto (Buxton, Reeves, et al. 1978). The SSSP was an interdisciplinary venture tackling "problems and benefits arising from the use of computers in musical composition" (Buxton, Fedorkow, et al. 1978, p. 472). A central aspect of this project was, however, human-computer interaction (HCI) —a field which was in its very early stages in 1978—[59] and music composition became a subject of study. Buxton's concern in this project was to reduce the cognitive burden of the

[59] William (Bill) Buxton is now considered a pioneer in HCI, and he is now a major figure in the Microsoft Research department.

composer:

> . . . the composer should simply not have to memorize a large number of commands, the sequence in which they may be called, or the order in which their arguments must be specified" (Buxton, Fedorkow, et al. 1978, p. 474).

In other words, by reducing the amount of information that is given to the composer at the time of interacting with a computer, the composer can concentrate better on making music. Therefore, Buxton continues, the organization of the program is divided into four tasks: definition of a palette of timbres definition of the pitch-time structure the orchestration —placing the previously chosen timbre in the pitch-time structure and the performance —the note or score playback. In this simple but very concise structure, the SSSP delimited the scope of action of the composer. Their innovations in the project have had enormous resonances in computer science —still today the concept of reducing cognitive burden of the user is dominant in Buxton's view—(Buxton 2016b), and many of the features of the human-computer interaction involved in the program are still used today.[60]

Black-boxing Programs

What is important to note here, however, is that the main purpose of the programmers is to make the software work in such a way that it becomes invisible to the user. This is also known as a black-box approach. What this approach enables is what media theorist Vílem Flusser (Flusser 2011) described as our capacity to envision. It consists of the power to visualize beyond the surface of the image —e.g., a television screen—, and to bring the technical image into a concrete state —as opposed to the abstract state of the electrons in a cathode ray tube—, that is, to make out of the technical image an experience. Therefore, he argues, "if we are asking about the power to envision, we must let the black box remain —cybernetically— black" (ibid., p. 35). This concept,

[60]For a detailed overview of the project, see (Buxton, Patel, et al. 1980), and to get a visual idea of the interface, see (Buxton 2016a).

then, of the black box is the possibility condition for envisioning to take place. If we recall in the previous section, Hansen makes a divergent point in claiming that it is the virtuality that is inherent in the human body's creative potential towards images what marks our interaction with new media. However, these two different views are far from being mutually exclusive, and quite the contrary, they can be considered complementary. Therefore, on one hand there is the technical device, whose multidimensionality is as complex as it is hidden from the envisioner. On the other, the human body with its capacity to create and embody. Flusser's point is, however, paradoxical, since the same black-box is what condemns and frees the envisioner to a state of superficiality.[61] However, in a very acute statement, Flusser writes: "envisioners press buttons to inform, in the strictest sense of that word, namely, to make something improbable out of possibilities" (Flusser 2011, p. 37). In other words, Flusser's text in 1985 justifies the invisibility of the technological device in favor of its most useful consequence, that is, its ability to make the user create something 'out of possibilities.'

CAMP and the vanilla synthesizer

The SSSP lasted until 1982 due to lack of funding. In the mid-1980s the code was re-written due to its many hardware dependencies, in order for the research it had advanced to continue under the Helicon Systems Computer Assisted Music Project (CAMP) (Free and Vytas 1986). John Free (Free 1987), one of the researchers of CAMP, developed SCRIVA (Buxton's notation software in SSSP) into extensible data structures for music notation, arguing that

> ...music data structures must be general enough so that as many styles of music as possible may be represented. This implies that the data structures (or the application's interface to them) should not enforce a musical model (such as equal temperament) that is inappropriate for the musical task at hand (ibid., p. 318).

Free's emphasis on generality extended onto the problem of hardware-independent soft-

[61] "The envisioner's superficiality, to which the apparatus has *condemned* him and for which the apparatus has *freed* him, unleashes a wholly unanticipated power of invention" (Flusser 2011, p. 37).

ware, and led him to imagine a ‘vanilla synthesizer,’ a computer environment that started from his development of a *Music Configuration Database* for the CAMP environment (Free and Vytas 1988). This database consisted of an intermediate program between the physical MIDI[62] input devices (such as the Yamaha DX7, Casio CZ101, etc.) and the computers, so that the channel management and the control mapping was automated by a simple query of the database by the user. In other words, with simple commands, the user was in immediate control of the diverse timbre world offered by various synthesizers made by different manufacturers, each holding an individual set-up that would be otherwise tedious and time-consuming for the user to re-arrange each time. Although John Free’s research was discontinued, this flexibility that stems from his notion of generality of use and simplicity of application, is a common practice in the programming world. Therefore, in a similar way to Buxton’s approach, the user can simply connect to different devices without the burden of thinking how to do so. However, Free’s approach is not entirely black-boxed, since the database is open to modification by a specific set of commands provided to the user.

Enter Linked Lists

One important technical shift in the use of data structures came with the concept of linked lists. A linked list is a data structure which contains not only information fields, but also pointers to the preceding and the following data record, in order to maintain the order of a sequence of elements. This functionality was only available after a feature was added to the FORTRAN ’77 programming language, in 1977, and later it became integrated in the C programming language (Kernighan 1978). Composer Charles Ames (Ames 1985) used linked lists to represent melodies within an automated composition system. Already in the programs developed during the SSSP years, linked lists were used because they were written in the C programming language.

[62]MIDI (Musical Instrument Digital Interface) is a communication protocol between synthesizers and computers.

Notating scores

Other programming approaches for music notation software were developed during the 1980s, specifically using the DARMS[63] notation project. The DARMS project started in 1963 by Stefan Bauer-Mengelberg and it is one of the first programming languages for music engraving (Brinkman 1983; Erickson 1975). Peter J. Clements (Clements 1980) joined together the DARMS data structures with those used in Max Mathews' synthesis program MUSIC V. However, Clements' attempt was not as successful as Leeland Smith's famous SCORE program (L. Smith 1972), which is one of the earliest music engraving softwares still in use today by major publishing houses (Selfridge-Field 1997). The core of Leeland Smith's program was a sort of character scanner, which interpreted rhythmically complex musical input into MUSIC V output. Thus, it was an intermediary between music notation and a computer music synthesizer. In the 1970s, with the appearance of vector graphics[64], SCORE shifted into music printing, and after the appearance of the PostScript format in the 1980s, the program was well established for a commercial release in 1989.

Computer Scores

Alexander Brinkman (Brinkman 1981) modelled Smith's input format into *Score-11*, adapting it to the MUSIC-11 program —Barry Vercoe's development of MUSIC-IV which later grew into the still widely used Csound. Written in Pascal, *Score-11* uses circular linked lists which are traversed by an interpreter, and result in an output formatted for MUSIC-11. Thus, the user creates a text file with blocks dedicated to individual instruments, and specifies parameters —such as rhythm, pitch, movement (glissandi, crescendo), amplitude, etc.— that are then re-formatted to fit the less musically-oriented notation of the MUSIC-*N* type software. The purpose of the software, then is to reduce some of the programming burden that MUSIC-11 demands, for example:

> . . . a crescendo over several hundred very short notes requires several hundred differ-

[63] DARMS stands for 'Digital Alternate Representation of Musical Scores.'
[64] Vector graphics are computer graphics images that are defined in terms of two dimensional points.

> ent amplitude values representing the increasing volume. Typing in several hundred note statements each with a slightly larger amplitude number would take forever. If the computer could be instructed to gradually increase the amplitude value over twenty seconds then life would be much simpler. This is the type of job that a note list pre-processor is designed to accomplish.(Brinkman 1982)

Brinkman emphasized, as well, on the program's extensibility by users, and it inspired the current *nGen* program, by Mikel Kuehn, which is written in C, and consists of the version of Brinkman's program for the currently available Csound (McCurdy et al. 2015). Brinkman, however, went on to designing an interpreter —or 'scanner,' as he named it— for the DARMS language (Brinkman 1983), which was useful for obtaining computable data structures for automated music analysis (Brinkman 1984). Similarly, J. Stephen Dydo (Dydo 1987) worked on an interface to the DARMS language: the *Note Processor*, which became one of the earliest commercially available music notation systems.[65]

N*u*tation

Another approach was carried out at CCRMA, when Glendon Diener (G. Diener 1988; G. Diener 1989) deviced a "pure structure" devoted to the "hierarchical organization of musical objects into musical scores" (G. Diener 1988, p. 184). This *TTree* data structure, as he named it, stemmed from his PhD thesis on formal languages in music theory (G. Diener 1985), and it was based in the hierarchic structures of the SSSP project. The change Diener introduced to these structures was their capability of sustaining links between not only the previous and the next data records, but to the 'parent' or 'child' data records to which it was related. This is known as 'inheritance,' and it enabled "any event in the [structure] to communicate with any other event" (G. Diener 1988, p. 188). While Diener implemented this data structure in the object-oriented programming language Smalltalk, he later developed it into *Nutation* (G. R. Diener 1992), a visual program-

[65] However, the structures of Dydo's software were not publicly released when he presented it at the ICMC in 1987. His software was commercially available in the early 1990s at a significantly lower price than other notation software, namely, *Finale*, which is still available today by MakeMusic, Inc. (Skinner 1990a; Skinner 1990b).

ming environment for music notation. *Nutation* was written in Objective-C[66], and it combined the previously developed *TTree* structure with *Glyphs* —small snippets of the screen that display a symbol—, and a music synthesis toolkit —the *Music Kit*— that the NeXT computer provided. This resulted in an extremely malleable composition environment, which enabled sonic feedback to the composer, very fast manipulation, at the cost of limiting the sonic feedback to a predefined and hardware-dependant set of digital instruments. Furthermore, what notation software is most often criticised for is precisely the way in which sonic feedback often comes to stand for music performance by humans.

Theoretical Performance

Consider, for example, Leeland Smith's closing statements in his presentation of SCORE:

> It is not a "performer's" instrument, but rather a "musician's" instrument. Theoretically, any performance, clearly conceived in the mind, can be realized on [the computer]. (L. Smith 1972, p. 14)

It is indeed a fact that computers can offer automated tasks to an unimaginable extent, but to translate this into music composition —and, thus, music performance— results more often than not in a disembodied conception of music composition. In other words, a computer generated stream of notes that is available by the algorithms that are enabled by these data structures may result in completely impossible physical tasks given to the performer. Therefore, envisioning goes beyond embodiment.[67] It can be argued, however, that precisely by pushing the limits of bodily skills it is how further developments in musical performance techniques can be achieved. Nonetheless, what I am stressing here is the extent to which music composition can be reconfigured by the possibilities data structures have brought to the field.

[66]Objective-C is a programming language combining the Smalltalk messaging system and the C programming language, which enables an object-oriented approach to the latter.

[67]The same can be said about music listening: the generated output can have internal structures that go beyond our perceptual skills.

Real-time data structures

Composers and programmers in the 1980s began focusing, however, on real-time performance of computer music.[68] In a joint venture between MIT and IRCAM, the software MAX was being developed by Miller Puckette. Within this software, Puckette's emphasis was on a different aspect of the use of database: time and its scheduling. For example, the main concern in the design of the software is located in the following idea regarding complexity:

> …complexity must never appear in the dealings between objects, only within them. Three other features currently in vogue seem unnecessary. First, there is no point in having a built-in notion of hierarchy; it is usually a hindrance. Second, I would drop the idea of continuously-running processes; they create overhead and anything they do can be done better through I/O [i.e., input, output] related timing. Third, there should be few defaults. Rather than hide complexity I would keep it visible as an incentive to avoid it altogether (M. Puckette 1986, p. 43).

In yet another turn of the screw of Buxton's concept of reducing the user's cognitive burden, Puckette keeps complexity "visible" —within the concept of the programming *object*. Furthermore, he removes the notion of hierarchical programming —which was also present in Buxton's view—, and proposes an on-the-spot, light, and ephemeral —i.e., discontinuous processes– programming practice. In other words, the structure of the database was placed *horizontally*, that is, in the time axis, and Puckette's efforts were dedicated to optimizing the handling of the internal timing of the processes:

> The scheduler keeps the runnable-message pool in the form of a separate queue for each latency. The scheduler always sends the first message in the lowest-latency nonempty queue. When the associated method returns the scheduler sends another message and so on. The only situation in which we need to interrupt a method before it is done is when I/O (including the clock) causes a lower-latency message to appear…In this case the scheduler causes a software interrupt to occur by pushing a new stack frame onto the stack and executing the lower-latency method. When this method returns …we pop the stack back to the prior frame at latency d_2 and resume the associated method (ibid., p. 46).

[68]For example, note a curious paragraph published at the ICMC in 1981 stating that a real-time version of MUSIC-11 was "near completion" by group at MIT (M. Puckette, Vercoe, and Stautner 1981).

What this means, is that the concept of linked lists is used to keep track of the order of processes that are run, and that each process is scheduled according to its temporality (i.e., its ‘latency’). Thus, the entire network of processes that can be run is maintained in a dynamic list (i.e., the ‘stack’) that can be changed at any time by adding or removing elements (i.e., ’pushing’ or ’popping’). The way in which these processes (or ‘methods’) are called is by messages that can be sent (inputted or outputted, i.e. I/O) by the user or objects themselves. This way of thinking programming in terms of objects is known as object-oriented programming, and it was already in use in the MUSIC-N programs. However, in this case, the scheduling system that Puckette implemented consisted in a ground-breaking application of the technique that changed the real-time computer music performance scene:

> “Rather than a programming environment, Max is fundamentally a system for scheduling real-time tasks and managing intercommunication between them” (M. Puckette 2002a).

Pure Data

Because of this emphasis on process, when Puckette developed Pure Data (M. S. Puckette 1997), he made it so data structures were a more accessible feature for the user to define and edit. To this day this Max paradigm (M. Puckette 2002a) can still be found in both widely used programs MAX/MSP (Zicarelli 1998) and Pure Data. What is important to note here, moreover, is the emphasis on Puckette’s end to provide a musical instrument without stylistic constraints:

> The design of Max goes to great lengths to avoid imposing a stylistic bias on the musician’s output. To return to the piano analogy, although pianos might impose constraints on the composer or pianist, a wide variety of styles can be expressed through it. To the musician, the piano is a vehicle of empowerment, not constraint (M. Puckette 2002a).

Puckette, therefore, aims to a certain stylistic ‘neutrality,’ which he represents by the way in which the user opens the program: a blank page: “no staves, time or key signatures, not even a notion of ’note,’ and certainly none of instrumental ’voice’ or ’sequence”’ (ibid.). While acknowledging that even the ’blank page’ is a culturally loaded symbol referring to the use of paper in Western Art Music (much in the same way that it is favoring complexity in the above described design), Puckette reconfigured computer music design, composition, and performance with great conscience on the way in which the structure of the program resonates aesthetically.

Pd Data Structures

Puckette also implemented a feature within Pure Data to enable the creation of graphic scores for electronic music (M. Puckette 2002b). In order to do this, he contextualized his research with the SSSP original project to include graphic scores within the composition environment, the Animal project by Lindemann and de Cecco —which allowed users to “graphically draw pictures which define complex data objects” (Lindemann 1990)—, and three other cases of graphic scores used to model electroacoustic music —i.e., Stockhausen, Yuasa, and Xenakis. Puckette’s data structures derive from those of the C programming language, and they can be used in relation to any type of data. Following his notion of neutrality within musical style, he resolved in avoiding graphic predeterminations, namely, the fact that the x-value within the graphical canvas is left unlinked with time, without intervening in the user’s conception of time. Despite this significant programming decision, Puckette provided the user with a sorting function, “on the assumption that users might often want to use Pd data collections as x-ordered sequences” (M. Puckette 2002b).

RtCMIX and the heap

In the same ICMC conference of 1997, two object-oriented languages were presented. Brad Garton and David Topper presented RtCMIX (Garton and Topper 1997), a real-time version of the CMIX program by Paul Lansky. What they described as innovative in this project is, in a similar way to the data structures for time management that Puckette presented, the scheduling capabilities of the program. In contrast to the CMIX language, which assumes a non-realtime access of objects, the

> RTcmix event scheduling is accomplished through a binary tree, priority-queue dynamic heap. . . The heap[69] is also designed to do "scheduling-on-the-fly," allowing notes to be scheduled at run-time (usually triggered by an external event, such as a MIDI note being depressed) (Garton and Topper 1997).

What this means, virtually, is that the real-time problem became a scheduling problem of computing tasks. However, in the case of RtCMIX, the instruments that are instantiated 'on-the-fly' can also establish their own TCP/IP connection sockets[70] in order to allow for networked access to the individual synthesizers (ibid.).

OpenMusic

The other object-oriented language was a visual programming environment for non-realtime composition called OpenMusic (Assayag et al. 1997). While not a synthesis engine, the strength of this Lisp-based graphic language developed as a collaboration at IRCAM, resides in its ability to provide the composer access to a variety of sound analysis tools for composition (Bresson and Agon 2004; Bresson and Agon 2010), as well as the possibility to generate algorithmic streams that output directly into a traditionally notated score.[71]

Kyma

Another powerful example is the case of an object-oriented language for music composition is Kyma (Scaletti 1987). It was developed by Carla Scaletti at CERL in 1987, and it was intended to use as an interactive non-realtime composition environment with the Platypus digital signal processor. Scaletti's language was hierarchical in its structure, which —as I have described above—

[69] "In computer science, a heap is a specialized tree-based data structure which is essentially an almost complete tree that satisfies the heap property: if P is a parent node of C, then the key (the value) of P is either greater than or equal to (in a max heap) or less than or equal to (in a min heap) the key of C." From `https://en.wikipedia.org/wiki/Heap_(data_structure)`

[70] A TCP/IP connection is an internet protocol that enables data to be transferred between two 'sockets,' or ports, that is, the basic elements to establish a network.

[71] For example, OpenMusic introduced the concept of a *maquette*, which is a graphic canvas upon which a heterogenous set of elements as varied as audio waveforms, scores, or piano-roll type notation can be displayed.

enabled the data records to be linked, that is, to be grouped vertically and horizontally. These data structures formed together objects, enabling the composer to treat as objects any set of sounds within the composition, starting from the composition itself as an object. In such a way:

> . . . the composer could create a "sound universe," endow the sound objects in this universe with certain properties and relationships, and explore this universe in a logically consistent way (Scaletti 1987, p. 50).

Therefore, and given the "vast amounts of data required for sound synthesis" (ibid., p. 50), the overall concept of Kyma was to fit the conjunction of timbre creation and temporal event lists into the same traversable database that the program proposed. What is important, however, is Scaletti's intention. In much the same way as Puckette, for Scaletti "the language itself would not impose notational or stylistic preconceptions" (ibid., p. 50).

Audacity and the Sequence

An example of an effective use of data structures was implemented within the sound editing software Audacity by Dominic Mazzoni and Roger Dannenberg (Mazzoni and Dannenberg 2001). Their innovation was evidenced in the very fast processing times achieved when large audio files were structured as sequences of arrays of small lengths. This approach stemmed from Crowley's research on text sequences.[72]

SuperCollider and nodes

The literature on computer music software for composition alone would extend beyond the scope of this dissertation.[73] However, there are important features to acknowledge from yet another

[72] In Crowley's view, a "linked list and an array are the two obvious data structures for a sequence. Neither is suitable for a general purpose text editor (a linked list takes up too much memory and an array is too slow because it requires too much data movement) but they provide useful base cases on which to build more complex sequence data structures" (Crowley 1998).

[73] For further reference in other sound synthesis data structures, see: the Diphone synthesis program (Rodet, Depalle, and Poirot 1988; Caraty, Richard, and Rodet 1989; Depalle et al. 1993; Rodet and Lefèvre 1996; Rodet and Lefèvre 1997); FORMES (Boynton et al. 1986); the Otkinshi system (Osaka, Sakakibara, and Hikichi 2002). For an overview

object-oriented language for music synthesis that was developed at the end of the 1990s, namely, James McCarntney's SuperCollider (McCartney 1996; McCartney 1998). This high-level language provides the user with a different paradigm to handle audio processes, their ordering, and —most importantly— their switching. The innovation that this language implemented is the "garbage collection" of each process. McCartney took the hierarchic structure of the object-oriented paradigm and defined 'nodes' in a tree-like structure, each with its own capability of nesting groups of other nodes, but most importantly, with its own initial and expiration times. In other words, in contrast to the constantly running audio processes that Pure Data and MAX/MSP, SuperCollider only consumes CPU resources whenever it needs to. This economy of resources that is enabled by the data structure of the language, together with its high-level quality, make it quite remarkable.

1.3.4 Applications

It is important to point to, at this point, that one of the central concepts of the object-oriented programming is extensibility. The list of objects that can be added to the main program tends to grow exponentially as a function of its use. Furthermore, for open-source programs like Pure Data and Supercollider the case is more extreme, since every user has access to the source code. Thus, a list covering all extensions would require a research project of its own. However, I would like to focus on those extensions that enable further and more specific use of databases in the context of music composition. In addition to program extensions, I will also provide some examples — among the many that exist— of the artistic possibilities that the type of data structure access that is provided to composers by means of computer music software sets forth. The following examples provide a glimpse of the major trends.

of existing audio software up to 2004, see Xamat's PhD Dissertation, the chapter titled: "Environments for Audio and Music Processing" (Amatriain 2004). See also the Integra project (Bullock and Coccioli 2009; Bullock, Beattie, and Turner 2011)

Synthesis

Diemo Schwarz developed the concept of data-driven concatenative synthesis in his PhD thesis at IRCAM (Schwarz 2000; Schwarz 2003) Scwharz later contextualized information space as a musical instrument (Schwarz 2012).

. . . more is coming on concatenative synthesis . . .

Ryoho Kobayashi (Kobayashi 2003) used a database of Short-Time Fourier Transform (STFT) analysed sounds in a very original way. Upon calculating the distances between the results of these analysis he was able to define a database of similarity between his original database which he then re-synthesized. Kawahara (Kawahara, Banno, and Morise 2004) demonstrated the morphing techniques of his software STRAIGHT (Katayose and Kawahara 1999) by analysing the RWC Database, which will be mentioned below.

Christopher Ariza (Ariza 2003) was able to implement a model for heterophonic texture by pitch-tracking the highly ornamented music of the Csángó[74] music into a database that enabled him to present a data structure of the ornament. While not strictly used as computer synthesis, his implementation of analysis and subsequent algorithmic rule extraction can be thought of as a form of analysis-based sound generation.

Navigation

Insook Choi[75] designed an interactive installation at the Dorsky Gallery in NYC, in which she prototyped a "sensory information retrieval system where the acquisition of information is an acquisition of an experience" (Choi, Zheng, and Chen 2000). By creating what she termed a *sensorial network* made out of a database of *semantic units* describing speeches by famous leaders (e.g., Roosevelt, JFK, among others), she enabled participants of the installation to explore the

[74] "The Csángó, in some cases a Szekler ethnic group, are found in eastern Transylvania (Kalotaszeg), the Gyimes valley, and Moldavia" (Ariza 2003).

[75] `http://insookchoi.com`

information space of the database of audio recordings by walking. A motion tracking system using computer vision enabled the users to traverse the database, not in an event-triggering fashion (i.e., interpreting space as a boolean switch for each audio file), but in such a way that the sound was modulated as a function of the different 'clouds' of pixel data where gradually changing values as participants moved across the sensing area.[76] Furthermore, she included *histeresis*[77] within the system, enabling condition-dependent events to occur as participants' interaction lasted longer.[78]

Robert Hamilton (Robert Hamilton 2006) used bioinformatic data taken from galvanic skin sensors attached to a cellist's toes within a live performance environment. The Galvanic Skin Response (GSR) activity was correlated with intervallic distance between adjacent musical notes in a score —i.e., a database of cell nodes: short fragments of pitch phrases— previously written by the composer. However, such score, according to Hamilton, acts as a "filter for the autonomic control signals generated by the performer" (ibid., p. 601). What this means is that the music fragment database, involuntarily navigated by the performer, becomes a parameter against which the live-generated —and voluntarily performed— score is eventually built upon.

Loviscach (Loviscach 2008) proposed data mining of sound libraries for music synthesizers in order to obtain statistical analysis that was used for the creation of intelligent parameter settings. Christian Frisson (Frisson et al. 2010) managed to navigate a database by way of similarities between the elements using a very fast interface prototyped in Pure Data. Frisson's PhD dissertation (Frisson 2015) provides an overview of multimedia browsing by similarity, and his approach in building a new multimedia manager is centered on open-source tools for audio and video information retrieval. Cartwright and Pardo (Cartwright and Pardo 2014) implemented querying of a database of computer synthesizer parameter presets by vocal input, enabling users to mimic sounds with their voices in order to obtain the parameter setup needed for the synthetic instrument

[76]"Pixels do not switch on and off, they fade in and out forming clusters in the 2D camera plane according to the degree of movement projected from the corresponding floor positions" (Choi, Zheng, and Chen 2000).

[77]Histeresis is the recorded history of the observer's interaction with a system.

[78]A video of the installation can be seen here: `https://vimeo.com/23086026`

to generate a similar sound.

Park et al (Park et al. 2010) created an interface called *COMPath*[79] (Composition Path) which enabled users to draw paths on a map, sonifying data points which represented information —e.g., traffic, weather, culture events, etc.— along the points of the path. They used major commercial web services —e.g, Amazon, Google, Yahoo, etc.— offering public APIs (Application Programming Interfaces) so that each geo-location inputted on the path acted as queries. The returned data from these services was then mixed together —what they refer to as *data mashup*— and mapped with a virtual synthesized via MIDI.

William Brent's research on timbre analysis (Brent 2010b) developed into a timbre description library for Pure Data called *timbreID* (Brent 2010a). Within this library users are able to analyze sound files using most available timbre descriptors. In his dissertation, for example, he explored the relationships between perceptual dimensions of percussive timbre and those obtained by the timbre description algorithms in *timbreID*. He concluded that in order to identify percussion timbres, the Bark Frequency Cepstral Coefficient (BFCC) performed on multiple successive analysis windows was the most efficient way to do so. Thus, a database of percussive sounds could be navigated by performing this type of analysis, enabling a fast and cheap —i.e., not CPU-expensive— content-based querying. Brent's *timbreID* library, however, since it enables users not only to analyze sounds but to gather descriptors into clusters within a database, allows for much more applications, one of them being concatenative synthesis (explained below).

Performance

Melucci and Orio (Melucci and Orio 1999) proposed a musical information retrieval system aimed at query-by-content navigation of a musical collection based on melodic segmentation. In their research, they implemented a Local Boundaries Detection Model (LBDM) to perform the automatic segmentation of melodic information taken from MIDI files of Baroque, Classic and Romantic

[79] `https://sihwapark.com/COMPath`

music. They then proceeded to normalize and index the melodic phrases into separate files for querying.

Norman and Amatriain (Norman and Amatriain 2007) developed a performance oriented interface called *Data Jockey*. It consisted of a software with an integrated capability to generate databases of audio file descriptors —together with beat recognition and synchronization— that the user could query by content, tags or metadata. Nakamoto and Kuhara (Nakamoto and Kuhara 2007) deviced a networked performance system using a MySQL database to store and retrieve vocal parts enabling users to sing together in canon form. Price and Rebelo (Price and Rebelo 2008) developed an installation that consisted of an interface to a database of percussive sounds. In their project, they used a MAX/MSP library called *net.loadbang-SQL* to query and import data for the communication with SQL Databases. This database contained information from some audio descriptors such as The brightness (spectral centroid), noisiness, and loudness of the beginning of each analyzed sound file. In this way, the users were able to navigate a bank of percussion timbres based on basic spectral content.

Price and Rebelo had based their research on the concepts set forth by the Semantic Hi-Fi project (described above), and on Casey and Grierson's *SoundSpotter* (Casey and Grierson 2007), the latter a software system dedicate to real-time matching of audio or video input. The underlying concept of *SoundSpotter* was that of audio input as control, that is, given the spectral similarities between the incoming signal from a microphone, a database of audio segments was navigated. In other words, the user selected fragments of a large bank of audio files by means of sound. This approach, paired with a very low latency engine that, provided real-time capabilities. The innovation of their approach resided in a joint use of a similarity matching technique called audio shingling[80] with log-frequency cepstral coefficients (LFCC) for pitch information.

[80]"Audio Shingling is a technique for similarity matching that concatenates audio feature vectors into a sequence of vectors, and matches the entire sequence" (Casey and Grierson 2007). " Shingles are a popular way to detect duplicate web pages and to look for copies of images. Shingles are one way to determine if a new web page discovered by a web crawl is already in the database" (Casey and Slaney 2006).

Benjamin Carey (Carey 2012) integrated a database within a real-time improvisation system designed in MAX/MSP. Liu et al (Liu et al. 2013) created an audiovisual environment for live data exploration. They implemented simultaneous sonifications and visualizations of networked database queries made by participants using iOS devices. Carthach, Jorda and Herrera (Nuanàin, Jordà, and Herrera 2016), for example, implemented a model for real-time rhythmic concatenative synthesis. Vogl and Knees (Vogl and Knees 2017) trained a computer model using a database of commercially available drum rhythm patterns in order to provide an intelligent algorithm for the variation of drum patterns intended for electronic dance music (EDM).

Gesture

Schoner et al (Schöner et al. 1998) proposed an intermediate synthesis technique between sampling and physical modeling, by training a compute model with sensor data of a violin performer paired with its respective sound signal. The resulting data-driven inference was carried out by Cluster-Weighted Modeling (CWM), a technique "based on probability density estimation of a joint set of feature (control) and target (spectral/audio) data" (ibid.). Kawahara's above mentioned STRAIGHT system (Katayose and Kawahara 1999) utilizes sensor data —such as breath input, head movement, and finger positioning— so as to enhance the sound analysis engine of the system, thus achieving more degrees of freedom when carrying out the sound synthesis. At CCRMA, Serafin et al (Serafin et al. 2001) managed to invert the concept of physical modeling by estimating violin bow position, pressure, and speed using Linear Predictive Coding (LPC) coefficients of violin audio recordings.

Schloss and Driessen (W. Schloss et al. 2001) used audio analysis to obtain gesture features from the non-audio signals obtained from the Radio Drum[81] —such as peak detection for determining mallet strokes. Later, Jones, Lagrange, and Schloss (Jones, Lagrange, and W. A.

[81] "The Radio Drum is a three-dimensional controller that has been in existence in various forms since its original development at Bell Labs in the late 1980's" (W. Schloss et al. 2001).

Schloss 2007) created a dataset of hand drumming gestures using data from a two-dimensional pressure sensor that could be compared to the membrane of a drum. Their intention was to provide physical model designers with a collection of six techniques of hand drumming, recorded as matrices at a slow rate —100 Hz, the maximum rate of the sensor— suitable for non-realtime synthesis by way of interpolation into a waveguide[82] mesh.

Andrew Schmeder (Schmeder 2009), stemming from the research at CNMAT[83] on the Open Sound Control (OSC) format, proposed a PostgreSQL database for efficient storage and retrieval of gestural data aimed for real-time application. Schmeder's paper presents a technical overview of database practices, focusing on relational databases. I will comment on his research at the end of this chapter.

Young and Deshmane (Young and Deshmane 2007) created a web-accessible database of gestural and audio data concerning violin bow strokes. Hochenbaum, Kapur and Wright (Hochenbaum, Kapur, and Wright 2010) developed a gestural and audio joint database that enabled identification of a given performer between a group of performers, gaining insight on musical performance itself. Caramiaux, Bevilacqua and Schnell (Caramiaux, Bevilacqua, and Schnell 2011) succeeding in proposing a query-by-content type of database navigation by way of gestural input. They used gesture-to-sound matching techniques based on the similarities of temporal evolution between the gesture query and the sound target. Garcia et al (García et al. 2011) constructed a multimodal database of sound and sensor data —e.g., blowing pressure on a recorder— for the purpose of designing a synthesis model for recorders with different blowing profiles. Visi et al (Visi et al. 2017) also designed a multimodal database, using sensor information from listening subjects who were asked to move as if creating the sound, resulting in an innovative way to train a computer model for gestural-based synthesis.

[82]Digital waveguides are an efficient model for physical modeling of wave propagation
[83]Center for New Music and Audio Technologies, Berkeley

Resource Sharing

Important research has been done in file formats suitable for data interchange. For example, the SDIF (Sound Description Interchange Format) and the GDIF (Gesture Description Interchange Format), widely used in audio analysis software like SPEAR, OpenMusic (Bresson and Agon 2004), among others. Bresson and Schumacher (ibid.) used SDIF format for the encoding and interchange of spatialization data. Bullock and Frisk (Bullock and Frisk 2009) implemented within their Integra[84] project a data format for sharing between different multimedia environments. Based on previous work on file formats[85], they developed the IXD format which is capable of containing sequences, tags and meta-data, and presets. Their argument for an XML[86] format resided in the semantic richness that can be allocated in opposition to the binary format only readable by machines.

Roberts et al (Roberts et al. 2014) implemented within Gibber —a real-time live coding web-based environment— a centralized database for the storage and quick access of digital instruments that can be prototyped in the environment. Taylor et al (Taylor et al. 2014) also implemented a centralized database within their platform for mobile-device performance, so that user-created interfaces could be saved and shared. Xambó et al (Xamb et al. 2018) provided a live coding system within SuperCollider to enable access to Creative Commons (CC) sound databases —such as Freesound— or user-made databases, enabling content-based querying by pitch or tapping.

[84] See footnote: 73

[85] Important references for their research were the following file formats: SDIF; GDIF; MetriXML (developed by Amatriain in (Amatriain 2004)); and the SMIL language (Synchronized Multimedia Integration Language).

[86] XML is a human-readable, eXtensible Markup Language very similar to HTML.

Chapter 2

Database Aesthetics

Delineating the agency of the Database in the practice of music composition, I discuss the aesthetics of Database Music, developing the concepts of listening, memory, and performance.

First, I analyze the extent to which the Database can be a listening subject which promotes illusions of style and authority. I consider style and authority as central aspects of the sphere of aesthetic agency of the Database. I then focus on a form of collective 'listening' and I arrive at my conception of the Database as an inherently deterministic system. This system is shaped as a network of nonhuman agents, whose 'resonance' is fundamental to its definition. I use this resonant network to further analyze the agency of the Database, in terms of how authorial qualities percolate through the network. I use Jean-Luc Nancy's ontology of sound to understand how the database can be a listening subject. In Brian Kane's reading (Gratton and M.-E. Morin 2015, pp. 143–144) of Nancy's work (Nancy 2007), he presents the this ontology —i.e., what Nancy calls *resonance*—, considering it as a process constitutive of a phenomenology of the self.

Second, in order to narrow the gap between human and nonhuman agency, I assess the extent to which computer memory resembles human memory. On the one hand, I compare memory and writing with digital information storing, and thus arrive at databasing as a form of memory. On the other hand, I consider archives as collective memory, which serves to to explain how the

Database can also be a form of collective memory.

Finally, I focus on the performativity of the database. On the one hand, I claim that the database is gendered. I argue that the notion of 'style' is what promotes the illusion of a gendered subject in the Database. I argue that since both the performance and the directionality of the 'styling process' remain strictly on the virtual skin of the database, the database's authorial subject, like the gendered self, remains in the spectrum of the illusory. On the other hand, I claim that the limit of the Database resides on its performativity. I consider the technical aspects of databases and define computer systems as networks of interconnected-but-independent databases. This definition serves to extend the performatic limit of databases to computers, and therefore to link the performance of the database to the performance of the computer. My goal in this final section is to lead the way to the connection between Database performance and Music Composition: the performatic limit of the Database is also the limit of Music Composition.

2.1 Listening Database

In this chapter, I analyze resonant networks in order to assess the extent to which databasing can be reconfigured by listening, and viceversa. The following questions will be revised: To what extent is the listening subject present within database music? How is the notion of listening subject reconfigured by way of the database? To what extent can the database be thought of as a listening subject, and, if so, to what extent does the agency of the database as listening subject resonate aesthetically?

In section one, I delineate Jean-Luc Nancy's ontology of sound in order to present the database as a resonant subject in itself.

In section two and three, given the multiplicity of factors that are in play when databases enter into the process of music listening, I focus on the links that exist between Nancy's resonance and Bruno Latour's actor-network theory (Latour 1990; Latour 1993), arriving at the concept of a resonant network. Then, a distinction between sound and networks is made, and the concept of the work of actors is introduced.

Finally, I understand resonant networks in terms of community. Since the notion of in-operativity is closely related to that of community —and the exposure of selves—, this relation of selves can also be understood as the resonating force that unfolds hand in hand with the performativity of the network. Thus, the expansion of the network, the propagation of sound, and the exposure of selves, can be connected to each other with a force of inoperativity.

2.1.1 The Resonance of a Return

In this section, I attempt to understand the aesthetic agency of databases, and, thus, how the concept of database music can be contextualized within an overall current of phenomenological thought. Therefore, my aim here is to understand how the agency of databases can be grounded within the agency of sound, linking an embodied theory of listening (See 1.1.3) with a phenomenology of the self.

Nancy's 'sonorous presence' is given by the combination of both the listening body as being part of the medium —medium understood as the transmission medium upon which sound propagates, i.e., a space filled with gas, liquid or solid particles matter—, and the body's sense perception linked to that medium. Because all bodies are part of media, from the above can be said that sound reaches, enters, and traverses bodies within media. Thus, for Nancy, sound immerses all bodies within itself, that is, sound includes all bodies within its oscillation, making listeners vibrate as if the singular listener —or, the listening subject— was always already part of sound. This is what Kane means when he refers to Nancy's concept of the presence of sound. Furthermore, because all bodies are linked singularly with their senses to media, sound is filtered differently and uniquely within each body. In this sense, any listener is a singularity in the presence of sound, and sound itself acts as a presence that exposes —in an instant— the in-between-ness that mutually exposes listeners to each other. The present —i.e., the temporal noun— of the physicality of sound is, in a sense, expanded in space and time within this aspect of Nancy's concept of listening. In itself, this sonorous presence is given by two physical qualities of sound.

The first quality is the instant of attack, which is the exact moment when a sound arrives and simultaneously leaves: the instantaneous appearance of sound within space or, more accurately, within a given transmission medium. A central aspect of this instantaneous motion is a combination of the speed of sound with its attenuation rate —i.e., how fast it travels and fades away. Physically, sound propagation is conditioned by the qualities of the medium, for example, the combination of density, pressure, temperature, and motion —affecting its speed—, and viscocity —affecting its attenuation rate. This means not only that within hot and humid climates sound will move slower, or that if there is wind blowing in the same direction of a sound it will travel faster; it also means that the listening body, because it is a medium, changes how sound moves inside of itself, adding yet another level to the body's filtering capacity —a layer that would exist in between the sense perception and the body's exterior.

The second quality is the spreading and expansion of pressure waves through space. The propagation of this oscillatory motion of particles passes not only through every reachable part of the space, it also traverses through the listener's body. Within this propagation, waves change direction by way of reflection or refraction, and they fade out by way of attenuation. This means that waves are affected in different ways according to different media, some being more reflective —such as, generally, interiors, or a reverberation room—, some being less reflective —such as, generally, exteriors, or an anechoic chamber. However, since the listener's body refracts, reflects, and attenuates waves, the singular filter that is the body itself changes wave propagation not only for itself —i.e., for its own listening experience—, but for the listening experience of other subjects in different places in space. This explains why empty concert halls sound different —i.e., more reverberant— than filled concert halls.

What is important to note here is that these qualities of the sonorous presence point the listener to the presence of sound itself, as a sensing experience in itself, and not to what it signifies. As Kane writes, to be listening in the sonorous present constitutes "a mode of listening that exposes itself to sense" (Gratton and M.-E. Morin 2015, pp. 143–144). In other words, this exposure is that of the body to itself sensing, inasmuch as it makes the body's sense perception appear —or, better, emerge— out of itself. This means that in the sonorous present, the listening body emerges as a creation of itself as sense. Therefore, within the process of listening the sonorous present, the first image is that of the body —if we consider this term following Hansen's concept of the image as creative process resulting out of the filtering capacity of the body (See 1.1.3). In this sense, the body is given form by itself, or, it is self-in-formed during the sonorous present.

Furthermore, within this self reference of the body to itself, and since media —as matter— is the always already of the possibility condition of listening, the resulting structure of the listening experience as such is that of infinite referals and deferals. This structure comprises Nancy's reading of Derrida's *différance*, one of the latter's most iconic concepts.[1]

On one hand, deferrals constitute the temporally deferred or delayed qualities of the different waves of not only stimuli, but also referrals themselves. Thus, they relate to differences in time that emerge throughout listening. On the other, given that for Nancy, meaning "is made of a totality of referrals: from sign to a thing, from a state of things to a quality, from a subject to another subject or to itself, all simultaneously" (Nancy 2007, pp. 4–9), referrals constitute the different points of reference or links between, for example, the listening body and itself. This is to say that, inasmuch as the sensing body is aware of its own multiplicity of sense, this multiplicity is itself dislocated in space. Thus, the distances in both time and space, between all points of the listening experience —of the singular listener and the plurality of listeners— create a web of references that is instantiated in the moment of listening. Furthermore, Nancy continues, since sound "is also made of referrals: it spreads in space, where it resounds while still resounding 'in

[1]For a commentary on this concept, see (Gratton and M.-E. Morin 2015, pp. 71–72); for Derrida's original essay on the matter, see (Derrida 1978; Derrida 1982)

me" (Nancy 2007, pp. 4–9), this understanding of *différance* is also related to the properties of sound itself, resulting in a process that intertwines sense and signification, along with an emergent self coming out of a resonating plurality.

Within this understanding of *différance* as the structure that emerges out of the pulsation sound is capable of stirring by way of listening, the condition of repetition, that is, of oscillation, can be comprehended as circular. This circularity explains the 'infinite' quality that Nancy gives to the structure. To put this differently, the constant oscillatory motion that is present throughout the listening experience has the structure of a feedback circuit, or a loop. For example, I listen to myself as resonant subject, while creating meaning from a certain quality of a sound; in simultaneity with another, who is also creating itself while giving meaning to sound waves; and the vibrating link in between ourselves is also, simultaneously, changing the way we are listening, every singularity being exposed to each other, that is, in touch with each other, resonating, in the process of listening. Within this loop, not only sound enters and leaves the body, also the body-image emerges —referred or referenced— and re-emerges —deferred or delayed for later— to itself. Therefore, this infinite emergence and re-emergence becomes the presence of the self, in the sense that the self becomes present to itself, not in a way that would imply some outward appearance of a substance, as if the self were some originary essence that appears in resonance. Quite the contrary, for Nancy, the phenomenality of the self comes as a result of the resonance, in the form of a return. In other words, the self is something that appears to itself, at the limit of itself, as sound. This is why, for Nancy, to be listening [*être à l'écoute*] is the state of the resonant subject engaging with an approach to self. This approach is neither to the self, not to the self of another, but to the structure of resonance as understood in terms of a relationship to self from itself:

> To be listening is thus to enter into tension and to be on the lookout for a relation to self: *not* it should be emphasized, a relationship to 'me' (the supposedly given subject), or to the 'self' or the other (the speaker, the musician, also supposedly given, with his subjectivity), but to the *relationship in self*, so to speak, as it forms a 'self' or a 'to itself' in general, and if something like that ever does reach the end of its formation.

> Consecuently, listening is passing over to the register of presence to self, it being understood that the 'self' is precisely nothing available (substantial or subsistent) to which one can be 'present,' but precisely the resonance of a return [*renvoi*] (Nancy 2007, p. 12)

The above phrase "if something like that ever does reach the end of its formation" points precisely to the infinite resonant process against which Nancy builds his concept of listening, but specifically, it is how he sets forth an image of a self coming from within this ontology of sound.[2]

In the following section, I assess the extent to which databasing can be reconfigured by listening, and viceversa, focusing on some Derridean links that exist between Nancy's resonance and Bruno Latour's actor-network theory.

2.1.2 Resonant Network

It is safe to wonder that Nancy's resonance points to instrumental music listening, specially because of his operatic examples. However, the case of listening to music with computers —i.e., not only computer or electroacoustic music, but also any playback system involving computers— can also be understood together with Nancy's resonance. The same principles apply, namely because of the practice of sound recording and reproduction, and how it was developed to replicate wave conditions by way of tranducers. In other words, considering modern transducers and digital technology only, sound waves can be reconstructed in such a way that the illusion of sources that are physically absent in one place, can be felt as if they were exciting the medium and body without much complications. This fact alone brings out a plethora of concepts that have been widely accounted for in the literature. However, in order to account for the presence of the database within listening, I focus here on one way of understanding Nancy's resonance that comes from Bruno Latour's actor-network theory.

[2]In this sense, Nancy is one of the first philosophers of the self to propose such a theorization of the self as resonance, extending his speculations to, for example, considering if the philososophical truth —since Kant— could be something listened to, as opposed to something seen: "...shouldn't truth 'itself,' as transitivity and incessant transition of a continual coming and going, be listened to rather than seen? (Nancy 2007, p. 4)"

Consider, for example, an acousmatic concert in which one of the music works is made with violin sounds —say, pre-recorded violin samples. Thus, when this violin begins playing sounds, an illusion may very well begin to emerge, that is, as listeners, we can imagine a violin player. Furthermore, if the imaginary player continues to play sounds and move them in space, this illusion continues in the direction of physical —but illusory– motion *in-space*, that is, a virtual perception of an actual violin, and an actual violin player. Therefore, this virtuality may very well project itself throughout the complete music work, thus grounding the music work on an imaginary force that is only alive because of the listener's virtuality. The ghostly qualities of this force will be addressed further down this text (See 2.2.4). Most presently is the fact that this magic show — happening in front and because of the listener's body-sensing mind— can be understood in terms of a resonating linkage between the human and the nonhuman, that is, a network of interconnected objects that refer to each other. What this network amounts to is twofold.

First, the network is yet another instantiation Nancy's ontology of sound. This is to say that the ilusory violin, however imaginary, can be listened to, and, thus, it can enter into an infinite game of resonance. In this way, the virtuality of the human mind, by way of listening, engages with the sonorous presence to the point of not only embodying a violin player altogether, but also emerging as resonant subject, thus exposing itself to itself just as well as to the virtual self of the violin. If this can be considered true, it follows that singularity, in Nancy's comprehension of the self, is not only an instance coming out of —and because of— a plurality of selves (Gratton and M.-E. Morin 2015, p. 71), but also a virtuality that is inherent in our brain's capacity to create images. The self, as image of itself, is, thus, a virtuality that comes out of the process of interaction with others, just as much images are processes of creation by a sensing-filtering body, and just as much a sound arrives and leaves from the body as an ongoing oscillatory process of *différance* in the duration of a listening experience.

Second, the network can be thought of according to Bruno Latour (Latour 1990; Latour 1993). This network is one made of interconnected frames of reference, that is, semiotic actors and their accounts of each other. In this sense, the network is comprised entirely of motion and activity:

> ...there is an actor whose definition of the world outlines, traces, delineate, limn, describe, shadow forth, inscroll, file, list, record, mark, or tag a trajectory that is called a network. *No net exists independently of the very act of tracing it, and no tracing is done by an actor exterior to the net. A network is not a thing but the recorded movement of a thing* (Latour 1990, p. 14). (Italics mine)

This performativity of the semiotic actors is the network's very own movement reflected back onto itself. This is to say that the network itself is the set of links that are being established as nodes. Therefore, these nodes are in constant reference to each other, and, thus, they engage in an ongoing motion of meaning. Within this universe —of what Latour calls 'ontological hybrids', i.e., world-making *etceteras*—, only meaning and connectivity is present, and there is nothing that falls outside the network. The network encompasses its own actors and its own performance, as its very defining gesture. What this amounts to is that performativity is the fundamental motion of the network —although there is no foundation, i.e., no up, no down, etc. This activity is why Latour takes the network to be "not a thing but the recorded movement of a thing" (ibid., p. 14).

Latour's actor-network theory, therefore, is a way of thinking the complex ways in which things are connected between each other. It is an image of a world made out of ontological hybrids which constitute the bare nodes on a de-centralized web of meaning. These actors/hybrids build their frames of reference by way of navigating from node to node, thus traversing a network. Within this network, the emphasis is placed on performativity and, thus, on movement. Furthermore, the very presence of the network is only evidenced by way of motion, outside of which there simply is nothing. Considering this concept of the network as one grounded on movement, and, precisely, on the movement of meaning, there exists the possibility of thinking this movement under the scope of Nancy's ontology of sound. This is to say, given that sound is also composed of the oscillatory motion of particles in space, and, given that this structure of sound is that of *différence* —in the sense of a structure of infinite referral and deferral—, then Latour's phrase can be reformulated as follows: the network is not a thing, but resonating thing.

In what does this resonating network consist of? What does this network bring to the idea of database music? Considering the concept of a resonating network of humans and nonhumans, the database not only enters as a node in the network, as a semiotic actor, it enters as a resonating subject in its own right. If semiotic actors in Latour's network are in constant reference to each other, it can be said that they are in infinite resonance with each other, that is, in a permanent state of vibration, or, simply, listening. Therefore, these listening actors are not only human, that is, they are listening things that engage with resonance just as much as Nancy's resonant subject. This is the crucial leap that comes out of the idea of a resonant network: the moment the nonhuman in the network is comprehended as resonating, it is the moment that they engage with an approach to self. This is to say, for the purposes of my argument in this section, that when the database of samples used by our virtual violin begins to sound, it is not only being listened to by the audience in the hall; it also begins to listen to itself.

This listening to itself of the database is what constitutes the possibility condition of the database as a resonant subject. This is not to say that listening subjects are indifferent to whatever sounds, in the sense that no matter what is sounding, a resonant subject will emerge out of the process of listening. If this is true, then, any sounding *whatever* would have no agency in the listening experience whatsoever. However, when considering the database as a resonant subject in itself, with all the forces of its own virtuality swiming towards the creation of this virtual self, then, the database is granted with an agency that is both creative, aesthetic. In other words, the database engages within an emergent community of the human and the nohuman. Our virtual violin, made out of stored samples in a computer's working memory, makes the network resonate while resonating itself to itself.

2.1.3 The Work of Actors

> The surface 'in between' networks is either connected —but then the network is expanding— or non-existing. Literally, a network has no outside. (Latour 1990, p. 6)

Latour refers to the in-between-ness of networks as something that can only exist when a connection is established. In other words, the structure of referrals and deferrals, of which the resonant network is built, only exists within a feedback-like motion, and there is nothing outside this motion. Further, Latour sustains that if this network effectively exists, it is, by definition, expanding. This is to say that, if it is understood as a feedback network, then it is a positive feedback loop with infinite growth capabilities.

Connection as expansion is the mode of being of this resonant network. It only grows through its nodes and its connections, and it is in constant growth since it is in constant expansion. What this means is that, given the network's capacity to self-propagate by way of these edges in between the nodes, the opposite is quite paradoxical: the notion of a network unlinking itself until extintion is utterly impossible. Although it would be rational to think this way, given that Latour's actor theory has this notion of network —which, in fact, came only later, and much to his resistance, as a way to think of his theory in a graphical way—, the case of the network would imply that once a connection is disconnected, then, yet another connection must refer to the unlinked status of the affected nodes, which, in turn, would unfold a new set of connections into new and unexpected directions. Therefore, a very acute distinction with sound arises: while the network is expanding in redundancy, that is, in complete overwhelming of itself, and in an ongoing multiplicity of self-reference, sound can be understood as a system that, by its very same propagation, attenuates towards an imperceptible threshold. Notwithstanding the infinitesimal motion of media —which prevents conceptualizing sound as fading into nothingness—, the embodied presence of sound does fade away into a metaphorical nothingess —or, if you will, yet another deferred sonorous presence.

Therefore, given this distinction, How can the concept of a resonating network be anything other than a contradiction? In order to analyze this contradiction, a few more things need to be mentioned. First of all, given the physicality of sound —i.e., sound as something that is actually there, reaching and resonating your body, etc.—, it can be posed that Latour's network is the inverse, that is, a conceptualization. This means that while objects are there, the network is not, since it is a semiotic excercise of thought that can expand infinitely, like language, or any other systematization, or articulation of objects in the world (See 2.2.2). However, Latour is very clear about distinguishing between this way of thinking that prevents the network to be an actual object in the world. It is true that the notion of network is, on the one hand, and to a certain extent, mathematical. This is to say that, the spatial metaphors that are generally used —i.e., "close and far, up and down, local and global, inside and outside"— come to be replaced by "associations and connections" (Latour 1990, p. 6).

For example, distance and proximity are two —geographically based— concepts that do not exists within the notion of networks, but come out as an "effect" of "enormous supplementary work" done by actors. Thus, while I am typing these letters, the distance from my eyes to the screen is nothing compared to the distance between my fingers and the stored characters in some cluster of the cloud service I'm using. However, these characters are there, somehow closer than the screen. This is the type of work that actors do in the network, that is, not only I am bringing into my own proximity those characters, the cloud computing service is displaying them on my screen. Because of my own capacity for virtuality, I can actually forget this distance, place it out of the equation, or, if you will, think of it as a black box. Simultaneously, the black box is in constant motion, that is, in a dynamic state that allows itself to disappear —to a certain extent, because its presence is felt by way of, say, server latency, waiting for database requests, etc. Thus, this proximity is, as Latour writes, an 'effect' that comes out of each node's 'work.'

Since this notion of work is embedded in the interaction of actors, and, furthermore, since I am trying to understand the network as a resonating network, what constitutes this work in terms of resonance? In other words, how can this notion of work as activity be considered together with Nancy's ontology of sound? However crucial the distinction between network and resonance is, given that motion is inseparable from both concepts, the resonant network and, furthermore, the extent to which its nodes have aesthetic agency in the particular case of music with databases, can be understood in terms of a deepened analysis of work. Work as activity, motion, oscillation, propagation, reverberation, feedback loops, etc., all of these come to represent the type of infinite dynamism of the mode of being of the resonant network. In turn, what this activity or mode of being amounts to is an unfolding of itself, that is, an *un*work.

2.1.4 The Unworking Network

> Community necessarily takes place in what Blanchot has called 'unworking,' referring to that which, before or beyond the work, withdraws from the work, and which, no longer having to do either with production or with completion, encounters interruption, fragmentation, suspension. Community is made of the interruption of singularities, or of the suspension that singular beings are. Community is not the work of singular beings, nor can it claim them as it works, just as communication is not a work or even an operation of singular beings, for community is simply their being —their being suspended upon its limit. *Communication is the unworking of work that is social, economic, technical, and institutional* (Nancy 1991, p. 31). (Italics mine)

The idea of *un*working —also *in*operative— relates to Nancy's previous text (ibid.), on which he developed the idea of *in*operativity as the grounds for a theory of 'community.' Therefore, these two terms —inoperativity and community— are already intertwined, and I will describe them in the following paragraphs. Furthermore, given that Nancy thinks of community as an exposure of selves towards one another, by understanding this movement of unworking of the resonant network, I intend to arrive at a similar notion of community, one that can emerge out of thinking the network as resonant and, thus, one that is closely linked to the ontology of sound. In turn, this idea of community serves as a means to explain the extent to which database music can be considered the performatic aspect of, precisely, a community.

Borrowing from Maurice Blanchot's concept of *desoevrement*, Nancy sets forth a concept of community that is based and depends on the notion of the inoperative. Nancy's argument for the possibility of community to occur is grounded on a reconfiguration of the concept of work. Work within a community, understood by Nancy, comes to be a form of withdrawal. This is to say that in a community, withdrawal is a form of interruption and suspension of both production and product. Therefore, if work is understood as the means by which a community arrives at the production of any given product, then withdrawal comes to place this system of production into a halt. If the process of production was thus interrupted, the system of production leading to complete formation —or any idea of wholeness, or completeness— is truncated and never completed. Therefore, within this logic of production, if community itself is thought of as product, that is, as the objective of the community itself —the process which composes itself as process—, then the concept of community is dismantled and broken. In other words, when community is thought of as work, the work of community can never be achieved, and community —as its product— can never exist. Thus, Nancy's conclusion is that community can never result out of a work, but it is only something that unfolds as unworking. In this sense, the system of production is rendered inviable for a relation to exist between selves, and only an inoperative force can arrive at the common exposure of selves.

Inoperativity can be compared to resonance, that is, if the distance between the resonant subject and the sonorous presence is thought of as being suspended on its limit. As I have described above (See 2.1.1), Nancy's concept of resonance has two dimensions, one relating to sense —i.e., sonorous presence, or the body sensing itself— and the other to signification —i.e., the structure of infinite referrals and deferrals. On the one hand, the sonic medium encompasses the concept of resonance with those of exposure, immersion, and openness. By allowing listeners to be exposed to one another, in contact with themselves through the medium, resonance exposes a liminality. This is also understood in terms of the reaching of the limits of the listening body. The listening subject, while becoming space —that is, while it begins to resonate as medium in itself—, it opens the resonant subject, exposing it to a limit which comes as a form of sharing, exchange, and ultimately community. This space of community is an exposure to "partitions of sense that are not regulated by any signifying dogma" (Gratton and M.-E. Morin 2015, pp. 162–164). What this means is that, by opening, resonance becomes fragmentation, allowing, thus, for the liminality upon which singularities can be brought to contact with each other. This contact is a form of touch, in the sense that touching permits being in common with the other. This is Nancy's take on community, and it is exemplified by resonance, that is, by his ontology of sound.

Therefore, given the fact that the ontology of sound points to the distance between sense and signification, and, thus, to the emergence of a resonant subject during the sonorous presence, this distance can be thought of as suspended at the limit. A limit, in the sense that it constitutes a edge between two objects. Using Latour's terms, these objects are none other than actors themselves, their edge being their connectivity. Using Nancy's conceptualization of community, this limit exposes selves to themselves and to one another. Therefore, by understanding resonant networks in terms of community, the result is a resonant self-exposure of the human and nonhuman at the limit. Thus, the instance of inoperativity: because of this (resonant) suspension, there is no possibility for completion —only expansion—, and within this quality of incompleteness —which relates to suspension, but also to fracture, fragility, instability, unpredictability—, is how the notion of a product can never be realized; or, better, how the notion of product is evidenced as an abstract conceptualization.

Since there is nothing outside of community, and, since there is only exposure, propagation, repetition, and expansion, there is room for thinking of the resonant network as an inoperative agent of community itself. What this amounts to is that the human and the nonhuman resonate as community, that is, they unfold their relations towards each other, suspending themselves in between one another. This in-between-ness is not to mean a gap between selves, but the very connectedness, that is, the same network of associations and referals that exists between selves. To put this differently, this liminality can be thought of as a skin, not in the sense of a layer that separates the interior form the exterior of a body, or, for that matter, as a surface under which or over which two selves can connect. This skin is not a surface, but a texture; it is not a layer, but an interweaving of minuscular threads that, in their own locality are fragile, but in their state of being weaved to one another, expand into a redundancy of fragilities that prevent concepts such as unity, concentration, or purity, to enter into the picture. This constitutes Latour's "material resistance argument," for example, one that refers to the heterogeneous, disseminated, and careful "plaiting of weak ties" (Latour 1990, p. 3).

The resistance of the skin represents, thus, the resistance of connectivity itself, that is, the resistance of a community. Therefore, by substituting, on one hand, the human with the database performer, that is, the database*r* and, on the other, the nonhuman with the database, the resonant network, as an instantiation of a process of unworking, enables the thought of a *database community* to emerge. With this concept of database community, the notion of database music can be further understood as a hybridly social and communicative event. This is to say that, following Latour's hybridity of objects in his understanding of society (ibid., p. 2), database music is a social practice that comes as a result of databasers and databases in resonance with each other. Furthermore, database music is an instance of community, in the sense that it is an event of communication, understanding the communicative as "the unworking of work that is social, economic, technical, and institutional" (Nancy 1991, p. 31). Therefore, the database community emerges as —but also, from, through, during, etc.— the unworking of databasers and databases, that is, the human and nonhuman. Finally, database community can be considered as a skin upon which the human and the nonhuman resonate, which amount to —but never finalize in— a musical event, that is, an instance of database music.

2.2 Memory

There is yet another substitution that can be ammended to Latour's definition of the network. If the network is 'recorded movement,' that is, a trace, a trajectory, this means that its existance is evidenced by way of not only motion in itself, in the sense that the very same oscillatory motion of reference in between the nodes creates its defining gesture. It is also the case that the recollection of movement constitutes an structurally inseparable part of the definition of the network. Therefore, in this chapter, I understand the database in its relation with memory, understanding memory from three points of view: the human, the nonhuman, and the spectral.

In section one, I analyze memory as process of embodiment and relate it to resonant networks.

In section two, I analyze the concept of the archive and its relation to the database, specifically, to databasing as a collective form of memory. In this sense, the nonhuman comes as an instantiation of memory outside the human. I assess the extent to which resonant networks can be considered under the scope of the concept of the archive.

In section three, I understand the dynamics of resonant networks spectrally, that is, as an expression of a force, or a power, that comes out of the spectrality that results out of the resonance of the human and the nonhuman. By considering the spectrality of the database, I asses the extent of its aesthetic agency in terms of a *haunting* force. Therefore, if database music is indeed haunted by the specter of the database, how does this affect the aesthetic result? If the ghost of the database can be understood as an image of the nonhuman, that is, an image emerging out of the plurality of memory, then, to what extent can it be considered a singularity, a self in itself? Finally, if, as databasers, we are engaged with this force in creative action, how does the music made with databases sound, and what is making it?

2.2.1 Interlude: Embodied Memory

> I suspect, nevertheless, that he was not very capable of thought. To think is to forget differences, to generalize, to abstract. (Borges 1942, p. 2)

The importance of memory —and forgetfulness— can be represented by Jorge Luis Borges's famous 1942 text, *Funes, the memorious* (ibid.). Due to an unfortunate accident, the young Irineo Funes was —"blessed or cursed" as Hayles points out (K. Hayles 1993, p. 156)— with an ability to "remember every sensation and thought in all its particularity and uniqueness " (ibid.). A blessing, since a capacity to remember with great detail is certainly a virtue and a useful resource for life in general; a curse, because he was unable to forget. Throughout the years, he became condemned to absolute memory, and so to its consequence, insomnia: he was secluded in a dark and enclosed space so as not to perceive the world.[3] Hayles focuses on one aspect of the story, namely, the fact that Funes invented —and begun performing— the infinite task of naming all integers, that is, of giving a unique name —and sometimes, lastname— to each number without any sequential reference. According to how Hayles describes it, by carrying out his number scheme, Funes epitomizes the impossibilities that disembodiment brings forth, to the point that:

> If embodiment could be articulated separately from the body —an impossibility for several reasons, not least because articulation systematizes and normalizes experiences in the act of naming them— it would be like Funes's numbers, *a froth of discrete utterances registering the continuous and infinite play of difference* (K. Hayles 1993, pp. 156–159) (Italics mine).

Therefore, the point that she is touching is that of the limits and fragility of embodied memory. In that Manovichian world which "appears to us as an endless and unstructured collection of images, texts, and other data records" (Manovich 2001, p. 219), this idea would be perfectly viable. Indeed, data banks have already been growing exponentially much in the same way as Borges' 1942 character's mind was aiming at. This capability of accumulation without the need of erasure is enabled by the database structure inherent in computers.[4] However, the distinction that Hayles presents —which has been discussed before (See 1.1.2)— is crucial: data is not information because information needs to be embodied. Therefore, on one hand, a disembodied data bank can have all the uniqueness and difference that is available by the sum of all cloud computing and storage to date; however, on the other hand, an embodied memory is only available by the human capacity to forget.

Studies in cognitive psychology might have something to add here. In Wessel and

[3] Within this fictional universe, the only way for him to sleep was to imagine the opaqueness of an unknowable future…

[4] In fact, the demand when it comes to computers is less its ability to erase —or even compress data— than storage space, a hardware-dependent commodity that has circulated ever since Von Neumann's architecture came into the picture (See 1.2.2).

Moulds's commentary (Wessel and Moulds 2008) on Paul Connerton's *Seven Types of Forgetting*, the authors consider forgetting to be the "failure" of certain search processes on account of an inability to recall information from memory. While pointing to current psychological models of memory which consider forgetting to be an adaptive and functional activity, the authors acknowledge the mystery of certain aspects of memory: "In human memory, it is unclear what really happens to old, disused or deliberately ignored memory traces —they might be retrievable, they might be lost, but no-one can tell" (ibid., p. 292)

Thus, considering this distinction between embodied and disembodied memory, I propose an imaginary experiment, one that I believe was missing from Borges' short story, however utterly fantastic his writing was.

One thing that can be read from the story is that, in order to seclude himself from perceiving the world, Irineo stayed in the dark. This is how he cancelled light, a quite powerful stimuli if memory-space is to be optimized —for the purpose of, say, getting some sleep.[5] However, there is little to no mention of the sonic environment in which Funes was embedded —probably in the outskirst of the quiet Uruguayan city of Fray Bentos. In fact, the only sonic references are focused on the narrator's perspective, referring to Funes' high-pitched and —due to his being in the darkness— acousmatic voice.[6] Therefore, focusing on Funes' listening, by locking himself inside a room he would have managed to attenuate sound waves coming in from outside. Notwithstanding his isolation —or, better, his self-inprisonment—, sound waves are actually very difficult to cancel. An interesting experiment would have been to have John Cage take Irineo to an anechoic chamber and ask him what he can remember then. From Cage's own experience, we can guess that Funes would effectively remember his own sounding body. However, it is very unlikely —but nontheless possible— that Borges was aware of American acoustician Leo Beranek's research for the

[5]For example, one of Irineo's concerns was to reduce the amount of memories on a single day, which he downsized to about seventy thousand…

[6]This acousmatic quality of Funes' voice will not be touched here, but it is indeed a good point of departure for an essay.

US Army during World War II, that is, when the first anechoic chamber was built.[7] Furthermore, even if he managed to isolate himself perfectly from the world, cancelling perception altogether, Funes would have been with his memories, which are not discrete, but continuous iterations of the world he had accumulated over the years. What this means is that all the sounds he had listened to would be available to his imagination.

As far as we can learn from the narrator, while *smell* is referenced to in the story, *sound* was completely out of Funes' concerns. Therefore, the question is how would the world sound for Irineo Funes? The task is not difficult to imagine: the world would be inscribed in poor Irineo's memory in such an infinitely continuous way that each fraction of wave oscillation would be different, unique, leaving no space for repetition of any kind. All sounds would be listened completetly, with every infinitesimal fraction of oscillation of the waves pointing to the most utterly complete scope of imaginable references. It would not be inaccurate to compare this type of memory saturation with CPU saturation, for example, the way a computer would —even the most gigantic multi-core imagined—, if it was commanded to compute, with accuracy, the wave equation. In this complete state of listening, there would be no possibility for thought, no processing of any kind, only infinite accumulation and storage.

In this sense, an infinitesimal incorporation of sound is unthinkable. This is not to be confused with the infinite structure of referrals and deferrals that difference is made of. The problem here —and this is evident in the story itself— is in the intersection of the finite with the infinite. While the structure of sound itself is infinte, in the sense that it is an ongoing process —i.e., circular loop— of difference, the singularity of the listening subject is finite: its limit exists as its possibility condition. This is to say that, given such an ontology of sound, the emergence of a resonant subject occurs *at* the limit, that is, at the liminality of an exposure to itself, to others, to waves, etc. In the case of the story, while Funes' nonhuman qualities correspond to dynamics of the infinite, his human body is just like any other that was thrown in the world. Therefore, no mat-

[7] https://en.wikipedia.org/wiki/Leo_Beranek

ter what the narrator has the reader believe, Funes is not deprived of this liminality. Throughout the story, Funes' liminality grows more and more evidently, all the way until the end —an ending that, despite all his nonhumanly infinite qualities, is, nontheless —be it a blessing or a curse—, utterly human.

Less than focusing on literary analysis, in this interlude I take the concept of an absolute memory within a human being to be at an intersection between disembodied theories of information and, precisely, the concept of an embodied memory. The aim is to differentiate between human and nonhuman in terms of memory and databases, so as to provide a link between Latour's 'recorded movement' of the network, and Derrida's concept of the archive.

2.2.2 The Effraction of the Trace

> Lo cierto es que vivimos postergando todo lo postergable; tal vez todos sabemos profundamente que somos inmortales y que tarde o temprano, todo hombre hará todas las cosas y sabrá todo (Borges 1942).

> All these differences in the production of the trace may be reinterpreted as moments of deferring. . . Is it not already death at the origin of life which can defend itself against death only through an *economy* of death, through deferment, repetition, reserve? (Derrida 1978, p. 202)

I would like to take a brief, but necessary, psychoanalitical detour, so as to set some context for the concept of memory. For this purpose, I take Derrida's reading of Freud's conceptualization of memory as a starting point towards distinguishing between human and nonhuman memory.

According to Derrida (ibid.), Freud understood memory as the essence of the psyche: "Memory, thus, is not a psychical property among others; it is the very essence of the psyche: resistance, and precisely, thereby, an opening to the effraction of the trace" (ibid., p. 201). In this sense, the perceived world enters as a force of effraction into the resisting unconscious, resulting in the inscription of a trace, that is, a memory. By definition, this is a violent process of impression —i.e., pressing *in*—, which is only possible by the notion of resistance. Thus, in order for something to leave a mark, there has to be some force acting against an other. In this case, one force is the world with its constantly changing images; another, the unconscious with its resistance to change. In this sense, Freud describes memory as breaching, or better, a state of being opened to this effraction. Memories are inscribed with incredible violence, one that can define the extent to which the inscription can be recalled later on, for instance, the case of repression, which is not forgetfulness but the deferred case of a force of containment. Furthermore, this breaching can be equally thought of as a fracture, hence the notion of effraction, which speaks of a the violent rupture. Despite, however, this violent image of path-breaking, Derrida writes of memory as an opening, in a paradoxical game between the resistance of the unconscious and the unavoidable —i.e., unclosing— state of being innocently ready for the next perception.

To complicate things a bit further, Derrida points to one issue in Freud's opposing forces: the case where resisting forces meet equally strong forces, which would result in a paralysis of memory. Therefore, he proposes that it is "the difference between breaches which is the true origin of memory, and thus of the psyche" (Derrida 1978, p. 201). In other words, Derrida reconfigures the psyche with the interplay of *différance* (See 2.1.1), that is, he understands pathbreaking only in conjunction with, and in terms of, a structure of infinite referrals and defferals, of the spacing and delaying of traces:

> Trace as memory is not a pure breaching that might be reappropriated at any time as simple presence; it is rather the ungraspable and invisible difference between breaches (ibid., p. 201).

Therefore, what does this notion of memory as breaching and *différance* bring to the concept of databases? In order to answer this question, the distinction between human and nonhuman memory must be made. So far, with Derrida's psychoanalitical incursion, the essence of the psyche —i.e., human memory— is breaching —but also, tracing, inscription, writing, etc.— and *différance* —i.e., the space and temporal dislocation of traces. Therefore, a link between human and nonhuman can be established: writing.

As Derrida points out, writing as *hypomnesis* —i.e., as an externalization of memory— has been considered since Plato (Derrida 1978, p. 221). However —and this is what concerns the relation with the nonhuman—, Freud's metaphor for the perceptual system found yet another place: the Mystic Pad. There were many derivations of this device, but it consists in something like a writeable surface board. When you write on it, a thin layer on top sticks to a sort of wax on the back, thus leaving a trace; when you lift the upper layer, the traces vanish completely. This special device came to represent, for Freud, the structure of the perceptual apparatus itself, or as Derrida says, it is where "the psychical is caught up in an apparatus, and what is written will be more readily represented as a part extracted from the apparatus and 'materialized'"(ibid., p. 222). Therefore, this new —Freud writes about this in 1925— hypomnesic device allowed Freud to shift from considering regular writable surfaces —such as a piece of paper—, to a combination of resisting textures in a device which allowed for "a perpetually available innocence and an infinite reserve of traces" (ibid., p. 223).

Derrida, however, reconceptualized this differently. The crucial distinction he finds on this writing device is the notion of erasure, but most pressingly, that of repetition. In this sense, traces "produce the space of their inscription only by acceding to the period of their erasure" (ibid., p. 226). This means that tracing constitutes a process whose only possibility is its own negation, that is, its own undoing of itself. Furthermore, these traces are, from the first moment, that is, from the beginning of the process of impression, "constituted by the double force of repetition and erasure, legibility and illegibility" (ibid., p. 226). Hence, the spatio-temporal duality of the concept of *différance*, that is, the interplay between spacing and deferrments, which comes to be the "work of memory" itself (ibid., p. 226).

The structure of memory, therefore, can be comprehended as pathbreaking and *différance*. This allows for the concepts of resistance, referrals, and deferrals to enter into its definition. Furthermore, the externalization of memory comes as an instantiation of the very structure of not only our perceptual machine, but also of the structure of the psyche itself. This hypomnesis allows for a materialization of the psyche that is constituted as a tool, that is, as an apparatus to be handled. With this conceptualization of memory as writing, Derrida reconfigures the notion of the self of writing, that is, of an author:

> The 'subject' of writing does not exist if we mean by that some sovereign solitude of the author. *The subject of writing is a system of relations between strata*: the Mystic Pad, the psyche, society, the world. Within that scene, on that stage, the punctual simplicity of the classical subject is not to be found. In order to describe the structure, it is not enough to recall that one always writes for someone; and the oppositions sender-receiver, code-message, etc., remain extremely coarse instruments. We would search the 'public' in vain for the first reader: i.e., the first author of a work (Derrida 1978, p. 227) (Italics mine).

Therefore, when memory is thought of as writing, the classical notion of self begins to disappear, opening up the space for the nonhuman. In what sense can this disappearance of the self be accounted for if we substitute writing for databasing? How does the databasing self emerges in this non-origin of the originary moment of writing that Derrida describes above? How is this "system of relations between strata" to be understood in terms of the work of databasers? This is how a further step into the conceptualization of memory can be of aid, one that considers memory as resonance, and writing as databasing.

I have already described above how resonance and memory constitute processes of *différance*, that is, how the structure of sound, on the one hand, and traces, on the other, relate to spatiality and temporality. This is to say that, while the infinite situations of reference create a space of multiple situations, connections, associations, etc., which constitute an instance of signification, simultaneously, the ongoing process of deferral creates the oscillatory condition of perception, which constitutes the infinite return, the repetition, and the reverberation of a self. This self, however, is not an essential one, that is, it is not a substance, but it is understood as a resonating subject, *the resonance of a return*, which, like Derrida's "subject of writing", does not exist in itself, and only appears as a result of a resonating "system of relations between strata."

Therefore, if this resonating self can only emerge out of this system —or better, network— of relations, then, every resonating point of the network must be considered as an agent in the constitution of a self. This means not only that the database, in this particular case of database music, becomes an agent of self-hood —and, for that matter, an agent of authorship relating to the resulting work of database music. It also means that, as an instance of hypomnesis, that is, as technology that externalizes memory, the database appropriates the qualities relating to memory itself that were described above: breaching and *différance*. Thus, not only can the nonhuman be reconceptualized within these qualities, also the human itself becomes reconfigured when faced upon the situation that memory, as breaching and *différance* can also be found in the nonhuman. Therefore, given that the nonhuman and the human are engaged in this resonating network that cancels the classical notion of self, then, distinguishing between the human and the nonhuman cannot be carried out in classical terms, that is, in terms of substance, essence, etc., since they do not apply anymore. The only thing that remains is, basically, the body, the singular instance of a resonating skin, whether it be of a human or a nonhuman, of a databaser or a database; a skin, but also an open resistance to the forces of change.

In what follows, I consider Derrida's *Archive Fever* (Derrida and Prenowitz 1995), where he sets forth the concept of the archive rooting it in Freud's psychoanalysis.

2.2.3 The Archontic Principle

For Derrida, the archive cannot be reduced to memory, it is is neither a form of rememoration nor a "conscious reserve," but simply a place where "the origin speaks by itself" (ibid., p. 60). In this definition, the word 'origin' refers to the greek root of the word 'archive' —*arkhē*—, whose meaning is twofold: on one hand, its definition is topological —i.e., refering to space; on the other, it is nomological —i.e., refering to the law. In the topological sense, *arkhē* is related to the greek word *arkheion*, which, in turn, refres to “a house, a domicile, an address, the residence of the superior magistrates, the *archons*, those who commanded” (Derrida and Prenowitz 1995, p. 9). In the nomological sense, *arkhē* refers to the ruling, to authority, to the command or commandment, i.e., the law. Thus, the magistrates —or *archons*— are those who “have the power to interpret the archives,” and they indeed reside inside this place called *archeion* (ibid., p. 9). Therefore, considering the fact that these magistrates actually reside in place of the archive, Derrida refers to an “institutional passage from the private to the public” which is constitutive of the archive itself (ibid., p. 9).

From this definition of the archive, what is alluded is the very structure of civilization itself, that is, of government and legislation. Going further into this aspect of the concept of the archive would extend the limits of this text.[8] Therefore, I would like to focus on one principle that Derrida described as belonging to the concept of the archive, what he calls the *archontic* principle.

The archontic principle is a type of authority that the archive projects, which can be understood as powerful dictating rule that is before anything else. Hence, its categorization as principle, which is also related to the origin (e.g., the latin root *principium* which refers to the beginning) and to the figure of the ruler (e.g., principal, prince, etc.). For Derrida, this principle is constitutive of all words deriving from *archē*, which is why he recomprehends the archive, for example, as being patriarchic. Since the archontic principle is present in the etymology of the word ‘patriarchy’ —which refers, likewise, to ‘lineage,’ ‘father,’ and ‘I rule’—, therefore, he argues that the archontic is embedded first with a sense of filiation, that is, with fatherhood and the relationship between father and child. Thus, the archontic is “paternal and patriarchic” (ibid., p. 60).

What follows is that this patriarchy is grounded in a domicile —i.e., a house— or an institution —i.e., a family. Within this domiciliation or institutionalization is from where rules

[8]Derrida, with this conceptualization of the archive, focused on the exponential growth of archives that spawned during the 20th century, defining it as an impulse, or better, as a ‘fever’ and a drive. Thus, he proceeded to perform a psychoanalisis of this symptomatic condition of mid-1990 society, beginning with the archivization of the house of the *father* of psychoanalisis, Sigmund Freud.

are prescribed, and where the ruling takes place. In the case of the patriarchic concept of family, the father is the ruler; in the case of civilization, the governor is the ruler. Therefore, this is what Derrida means when he performs a psychoanalisis of the archive: the topo-nomological aspect of the archive comes precisely from a —strategically male-centered— concept of fatherhood. In this sense, Derrida discovers that the archontic has the form of Freud's oedipus complex. An oedipus complex constitutes a desire of the child's unconscious hatred towards a parent. It is itself based in Sophocle's drama *Oedipus Rex*, in which such desire results in eventual parricide. Derrida notes that this complex has the form of an infinite loop between father and child, which is bound to repeat itself, just as is the case with Freud's definition of the complex, which, according to the greek drama, is articulated by the figure of the parricide. In this sense, Derrida claims that Freud "has shown how this archontic, that is, paternal and patriarchic, principle only posited itself to repeat itself and returned to re-posit itself only in the parricide" (Derrida and Prenowitz 1995, p. 60).

What is the case when this patriarchic principle is found on archives? I mentioned above the passage from the private to the public that Derrida recognized in the concept of the archive. This fact resonates with the plurality that is condensed in the place of the archive, which is opposed to the singularity that is condensed in the plurality of memory. In other words, while archives are a singular place where documents —made by a plurality of authors, that is, by different people, technologies, etc— are singularly kept, in the case of human memory, the reverse occurs. While the psyche is singular, that is, while there is only 'myself' remembering, the difference between each trace of memory is indeed a plurality. While I remember, the memory I rememorate is composed out the set of differences of traces that I have effracted in the moment of remembering.

The following pseudocode may help in this distinction:

```
function make_an_archive {
	public  ← PRIVATE
```

```
        singular ← PLURAL
        place ← COMMUNITY
        archive ← place ← singular ← public
        return archive
        }

function make_a_memory {
        private ← PUBLIC
        plural ← SINGULAR
        trace ← SELF
        memory ← trace ← plural ← private
        return memory
}
```

For example, the case of making an archive can be exemplified with the passing of a law or the signing of a contract. Derrida speaks of the act of consignation as constitutive of the archontic principle, since it is an act of "gathering together signs" (Derrida and Prenowitz 1995, p. 10), as is the case of the signatures on a bill or a contract. One's signature is private, therefore, the moment one signs a document one makes it public. Furthermore, in making a contract, multiple signatures need to be gathered into the same space: the plurality of the community needs to be condensed in the singularity of the document. Finally, the document itself comes to be a place, but it itself needs yet another place where it is stored, which comes to be the archive. What this act of consignation entails is, precisely, the archontic principle, since the contract itself now becomes an authoritative presence that emerged out of this function of making an archive. The structure of this presence will be discussed in the following section (See 2.2.4). For now, let us continue with the case of making a memory. The moment one perceives the world, one makes it private, inasmuch as one engages with remembering it. When one begin remembering, oneself begins the process of pathbreaking and effraction of traces —and of forgetting, as I will continue to develop below–, which results in the plurality of the condition of traces, that is, on the difference between traces constitutive of memory.

So, given that if this constitution of archive and memory are indeed reflections of each other, that is, one being the reverse of the other, how is the archontic present in the making of a memory? What would a reversed archontic principle look like? How would it act? These questions deserve a few paragraphs because they will explain the relationship between database, memory, and archive.

What is important to note here, before continuing, is that just as memory is in a state of fracture and rupture, an archive is also in such state of discontinuity, and thus it is this condition of being in the form of disconnected gaps that makes memories and archives so alike.[9] However, while the archive is indeed patriarchic, memory, on the other hand —and in terms of the archontic principle— can only be described as an-archic. While there is indeed oneself remembering, there is no possibility of ruling over the plurality of traces that one has in memory. There is no place to go to remember things, one may try to trick the body into feeling similar things as when one was remembering so as to make a memory resurface, but in the end, memories remember themselves. One has no control over neither remembering nor forgetting. Memories forget themselves.

In the fantastic case of Funes (See 2.2.1), for example, forgetting is a mysterious lack that cannot be reduced to a simple loss of information. It is also a necessary condition for memory to be considered as trace, given that the moment a trace begins is when its own erasure begins. Thus, every memory comes with its own forgetting mechanism, a mechanism that is triggered on its own, and, further, that it is still not fully understood (Wessel and Moulds 2008, p. 292). However, in the case of an archive —or, what some (perhaps wrongfully) refer to as 'collective' memory—, the opposite occurs: data loss consists of plain and simple erasure, that is, a destruction that cannot be considered an instance of forgetting:

> In collectivistic memory, where the database has a tangible form, it is more apparent that permanent loss is a possibility. In archives, ink may fade, paper may crumble and entire files may end up in the shredder (ibid., p. 292).

[9]Spieker notes is that of discontinuity and rupture: "Like all kinds of data banks, [the archive] 'forms relationships not on the basis of causes and effects, but through networks"' (Ernst 2013, p. 113). From these two qualities of archives —i.e., filtering and fracture—, their ressemblance to memory can be drawn.

Since writing can function as a link between human and the nonhuman memory (See 2.2.2), several instances of the metaphor of writing need to be explained in relation to databasing. For example, not only programming languages imply writing in terms of symbols —generally words and characters—, also data structures appropriate the concepts of writing and erasing. Furthermore, in resonance to the Derridean trace, erasure is embedded in the structure of writing. This is to say, that —to bring it into different, object-oriented programming words— C++ classes include within their own data structure a call to their `destructor`. This means that, whether explicitly or implicitly, all classes —i.e., all data structures which correspond to instantiated objects— have a way to self-erase, or self-destruct after the object is no longer needed.[10] This self-destruction means, precisely, releasing the object's resources, that is, to free the physical memory space that it has occupied throughout its lifetime.

This comparison between `destructors` and forgetting serves as a starting point to determine the extent to which computer memory —i.e., data structure handling in restricted, discrete space— can be thought of as human memory itself, and, if so, the extent to which it also constitutes an instance of an-archic structure. (The first thing that comes to mind is an an-archic program written in C++ with self-destructing classes that fire at will, in complete unpredictability, rendering the software utterly anarchic, but utterly useless.) Therefore, there has to be a different way to think of an-archy in software.

In what follows, I will address the extent to which databases can be understood in terms of memory and anarchy, taking German media theorist Wolfgang Ernst's concept of the *anarchoarchive* —which he grounds in Kittlerian, digital convergence— to a different level.

2.2.4 The Spectral Database

> The structure of the archive is *spectral*. It is spectral *a priori*: neither present nor absent "in the flesh," neither visible nor invisible, a trace always referring to another whose eyes can never be met... (Derrida and Prenowitz 1995, p. 54)

[10] `https://en.cppreference.com/w/cpp/language/destructor`

Despite the multiplicity of elements that constitute software programming —such as compiler instructions, hardware estipulations, collaboration platforms, etc.—, writing, that is, the concepts of inscription and erasure are still in place. Because of this, the Derridean trace can still be applied to programming. This fact can be linked further back to the conception of the computer itself, as it was previously discussed (See 1.2.1). In his architecture, Neumann proposes that the storage unit of the computer allowed data to be written and erased in different locations and times. In fact, he was following Turing's conceptualization of the *a-machine* —i.e., the *Turing machine*—, which was a mathematical model for computation, that can be represented by a symbol scanner and an infinite tape, where the scanner gets, sets, or unsets a symbol on the tape, and the tape moves to the next slot accordingly. Therefore, these setting and unsetting movements represent inscription and erasure, to the point that, as Kittler notes, quoted by Ernst: "the two most important directing signals which link the central processing unit of the computer to external memory are being called READ and WRITE" (Ernst 2013, p. 131).

However, an important distinction needs to be made here. While I am arguing for the similarities that exist between memory and database, Ernst proposes that databases —i.e., what he refers to as digital an-archives— enter as a form of replacement. This is to say that, given the convergence of all media into digital media, 'reading' is rendered null, giving way to mathematical processes that come to interpret the data. In Ernst's words, "signal processing replaces *pure* reading" (ibid., p. 130) (Italics mine). This statement is only valid within the Kittlerian, disembodied worldview (See 1.1.4), which would also allow to convey the following: signal processing replaces *listening*. Thus, in such a world where data structures and algorithms perform our own capacities to create and, by doing so, remove our bodies from ourselves, there would be no need for neither reading nor writing. Since, if the database reads itself, it also writes itself, and, in the case of music, if it listens to itself, it also sounds itself, removing listening altogether. Therefore, in order to arrive at the point of resonance that enables both the nonhuman and the human to coexist, a reconceptualization of the anarchic in relation to databases needs to be made; specifically, one that is grounded not on substitution, but on difference, and one that explains how authority enters into the sphere of databases.

In his conceptualization of the *anarchoarchive*, Ernst (Ernst 2013) opposes technical recording with symbolic transcription. Given the fact that a microphone captures the entire sonic environment, this involuntary memory —of "past acoustic, not intended for tradition: a noisy memory" (ibid., p. 174)— comes to be a form of anarchic archive, or *an-archive*. Therefore, for example, he claims that Bela Bartok's transcriptions to musical notation of Milman Parry's Serbian epic song recordings[11] becomes an archivization process, that is, a process by which symbolic transcription leads to an ordered archive, i.e., a score.[12]

In Ernst's sense, a sound file storing samples of the recorded world is anarchic because of the unfiltered —i.e., 'pure'— way in which the world is inscribed by the recorder. He bases this idea on another media theorist, Sven Spieker, who claims that, following Derrida's conceptualization of the archive, a central feature of archives is not memory, but a need to "discard, erase, eliminate" that which is not intended for archivization (ibid., p. 113). In other words, the possibility condition of an archive is filtering, or better: framing. The problem is that, the moment Ernst deposits on the recording technology the dissapearance of the body, he dislocates framing altogether, and thus is how the concept of 'pure' recording comes in. Therefore, if for Ernst, Parry's audio recordings are anarchic, this is because they are constructed upon a Kittlerian idea of purity that leaves the human out of the equation (See 1.1.4).

If we consider the body's capacity to create images from the world of images, then framing is also a possibility condition for human memory. In fact, we have seen with the Derridean trace that it is a resisting force what first enables the violent rupture of traces, and thus, this force can also be considered as an instance of framing itself. Resistance of the psyche as creativity tak-

[11] `https://mpc.chs.harvard.edu/`

[12] Another example Ernst provides of the anarchive is the Internet itself: " [The Internet] is a collection not just of unforeseen texts but of sound and images as well, an anarchive of sensory data for which no genuine archival culture has been developed so far in the occident" (Ernst 2013, p. 139).

ing action as the moment of framing. In the an-archic memory graph that I am conceptualizing here, framing can be thought of as the edge (the arrow) that thus connects the public to the private. Furthermore, framing can also be inversely considered as the arrow between the private and the public in the case of archives, since, for example, in the case of a contract, only a limited amount of signatures are needed, thus a frame needs to be an active part of the process of archivization.

Therefore, given that framing is in between the public and the private, in order to consider the case of audio recording as either memory or archive, these two categories, privacy and publicness need to be taken into account in relation to nonhumans. In other words, given that the world can be considered public, because it is in a constant state of availability at any time, then, a microphone can be considered an actor of privacy, since it deprives some sounds from being recorded —e.g., because it has not been designed for certain frequency bands— while recording other sounds. Therefore, if transducers have a filtering capacity, they engage with the passage from the public to the private —and viceversa, since both speakers and microphones are transducers. Consider, for example, Jonathan Sterne's definition of the 'tympanic' function of transducers, as quoted by Cathy Van Eyck in her PhD dissertation on microphones and loudspeakers:

> As Sterne describes clearly: "every apparatus of sound reproduction has a tympanic function at precisely the point where it turns sound into something else (usually electric current) and when it turns something else into sound. Microphones and speakers are transducers; they turn sound into other things, and they turn other things into sound." (Eck 2013, p. 107)

Therefore, considering these nonhuman tympans as actors of privacy and publicness, audio reproduction technology can be compared to the structure of memory and archives. For example, audio recording can be conceived as a form of memory, only to the extent that its inscription is singular yet reproducible. This is because a magnetic tape or a hard drive cannot be considered in the same (plural) way as Derridean traces, since once data is stored, it exists only once in discrete, limited space, unless copied to another location, in which case it becomes a duplicate, an identical (in-different) clone of itself. Furthermore, the case of audio playback can be a form of archive, since it consists of the passage from the private —the stored air pressure waves— to the public —the reproduced air pressure waves in space—, only to the extent that the latter is not considered space as such, but resonance within space. In this sense, Ernst' consideration of the Parry's recordings as anarchic needs to be reconfigured.

Given that these nonhuman tympans constitute the limit between sonic —but also tactile (Eck 2013, p. 223)— world and the binary world of databases, their comparison to memory and archives renders a hybrid object: one that, on the one side, becomes a private and singular 'trace', and, on the other, becomes a public, resonant 'space.' Thus, databases represent neither a trace nor a space. At this point it is important to address the quote at the beginning of this section, that is, the structure of the archontic that Derrida assigned to archives: spectrality. Derrida claims that, addressing a phantom is a "transaction of signs and values, but also of some familial domesticity" (Derrida and Prenowitz 1995, p. 55), meaning, on the one hand, that in the uncanny encounter with a ghost, there is familiarity, that is, there emerge feelings of what is known to be close to us, but also that which composes the authority of that closeness. Therefore, embedded in this familiarity is the archontic, the oedipal, etc., and thus the expression of power that this apparition brings forth. On the other hand, the familiar is also related to an economy —Derrida points to the greek root *oikos*, meaning 'house'—, that is, to the passing through (trans-action) of signs, but also de translation —or better, the 'transduction' of things. This is why Derrida considers any encounter with the spectral to be an instance of addressing, that is of transaction. Finally, what this uncanniness of the ghost entails is nothing other than the haunting itself, as Derrida writes: "haunting implies places, a habitation, and always a haunted house" (ibid., p. 55). To bring back our database, the hybridity that the database projects when compared to memory or archives points to a certain uncanniness, that is, precisely to the hauntedness that comes from its spectrality. Furthermore, it can be argued that considering the database in such way is plain and simple delusion, that is, insanity at its best. Not surprisingly, this is the point exactly; within this delusion exists truth:

> . . . it resists and *returns*, as such, as the spectral truth of delusion or of hauntedness. It *returns*, it belongs, it comes down to spectral truth. Delusion or insanity, hauntedness is not only haunted by this or that ghost. . . but by the specter of the truth which has been thus repressed. The truth is spectral, and this is its part of truth which is irreducible by explanation. . . (Derrida and Prenowitz 1995, pp. 54–56)

Therefore, given that Derrida considers the structure of the archive to be indeed spectral, I can bring this spectrality to the database itself, and consider it just as spectral but in a different sense. The spectrality of the database comes not from its hauntedness, that is, it is not domiciliated since, as humans, we have no access to data space (address space) directly with our bodies. As humans, we cannot engage in transaction with the specter directly: we need transducers. In the case of audio playback, we need loudspeakers because we need to create, with our bodies, the image that was recorded in the first place. In consequence, while embodying stored data, we embody its specter. Therefore, the hauntedness needs to be 'transduced' into space, and with this transduction the agency of the database can be effectively felt. Likewise, the uncanniness of the case of audio recording can be felt as the ghost that will 'remember us,' because of this computer 'memory' that will keep (to its own privacy) the sonic environment that attests to our presence in space —because we sing, we make sounds, etc. Hence, the haunting that resides in-memory, together with the stored data and pointers.

In recognizing this presence, which is the archontic specter of the database, comes the recognition of the agency of the database itself, and, furthermore, of the quality of the aesthetic experience that we encounter whenever there is a database in art. As Timothy Morton (Morton 2013) writes in relation to what he calls hyperobjects:

> Recognition of the uncanny nonhuman must by definition first consist of a terrifying glimpse of ghosts, a glimpse that makes one's physicality resonate (suggesting the Latin *horreo*, I bristle): as Adorno says, the primordial aesthetic experience is goose bumps. Yet this is precisely the aesthetic experience of the hyperobject, which can only be detected as a ghostly spectrality that comes in and out of phase with normalized human spacetime. (ibid., p. 169)

To the point that databases are spectral, they can be considered hyperobjects in Morton's sense, and thus, agents translating through aesthetic networks. If we consider the ‘recorded movement of a thing‘ with which Latour identified his concept of the network, databases are agents not only of recording, also of motility, and of thing-hood itself. This is to say that, in the constitution of networks, the spectrality of the database expands in multiple directions. Bringing back the illusory violin of the acousmatic concert I mentioned earlier (See 2.1.2), it exemplified Hansen’s concept of the creation of images our brain’s capacity for virtuality has, that is, our imagination. The sound of a violin can be recorded —but also synthesized— into the privacy of a database. Then, it can be played back with loudspeakers located in such a way that they emulate the location of an actual violin player. As a result, the listener could very likely imagine a physically present violin in the room, that is, a ghost. This ghost comes in as the phantom of a human player; of the violin itself; of the histories and traditions that those two elements bring forth; of the presence of the nonhuman that the database implies; of the privacy that is not human but that it is still uncannily private; of the plurality that is embedded in the construction of databases; of the hauntedness of the archontic that the above sets forth; and so on. In this way, the spectrality of the database attests to its relation to memory and archives, and, thus, to its aesthetic resonance within our experience.

In what follows, a crucial aspect of the database will be addressed, one that defines the condition of possibility of this hauntedness to be indeed instantiated into appearance, not only in the form of authority, as I have shown in the case of its archontic presence, but also in the form of style, and most important, gender: the performativity of the database.

2.3 Performance

In this section, I draw from performance and gender studies to analyze database practice as a performative activity. I use Judith Butler’s concept of gender (Butler 1988) to analyze the extent to which authority in database practices can be understood in terms of style. The databaser, which is the human subject in this case, begins resonating with the database, the nonhuman subject, in a form of feedback loop. This resonant loop is only possible through the performance of the database. I thus locate the origin of the database as listening subject the moment its performance begins. In this moment of performance, both human and nonhuman listening subjects are resounded upon their limit. I consider this limit to be a surface in between the human and the nonhuman, and I like to think of it as the 'skin' of the database. I argue that this skin of the database is the possibility condition for illusions of style —as in the style that is visible in one's clothes, or in one's decoration of the skin— and authority —as in the appearance of a subject who has some sort of power: the subject of the listening database. I thus analyze these illusions as belonging to the sphere of Agency that the Database presents, and assess the extent to which they affect the aesthetics of Database practice.

2.3.1 Gendered Database

> Gender is not passively scripted on the body, and neither is it determined by nature, language, the symbolic, or the overwhelming history of patriarchy. Gender is what is put on, invariably, under constraint, daily and incessantly, with anxiety and pleasure, but if this continuous act is mistaken for a natural or linguistic given, power is relinquished to expand the cultural field bodily through subversive performances of various kinds (Butler 1988, p. 531)

Following Butler's distinction between expressiveness —a self from inside-out— and performativeness —an illusory self, strictly outside, on the surface—, I claim that, given that databasing is the performative condition of databases, the database is gendered. In other words, in the performativity of databasing resides the possibility for the skin of the database to emerge. This skin is the spectral surface of the illusory gendered self that the database projects. As a consequence, on this spectral skin, which is the limit upon which resonance between the human and the nonhuman takes place, is where the inscription of a style can be made. This is to say that, defined as a repetition of acts, style is a form of embodiment that is ascribed to databases. With this stylistic definition of the self of the database, the culturally and historically mediated possibilities come to instantiate an authorial subject, one that is uncanny not only, as I have described above, because of its nonhuman spectrality, also because of its gendered appearanace which comes to redefine our own social categories such as reality itself.

First of all, how is a definition of gender of any use to database practice? Butler sets forth a critical genealogy of gender which relies on a "phenomenological set of presuppositions, most important among them the expanded conception of an 'act' which is both socially shared and historically constituted, and which is performative..." (Butler 1988, p. 530). Furthermore, the difficulty Butler recognizes in this view of gender, is that "we need to think a world in which acts, gestures, the visual body, the clothed body, the various physical attributesusually associated with gender, *express nothing*" (ibid., p. 530). Therefore, how can the database itself be conceived in this terms of performativity, to the point that, while having the capacity to store millions of data, a database can in deed express nothing?

Butler defined gender identity as a historical situation, distinguishing between physiological facticity of the body —i.e., as sex— and the cultural signification of such facticity in terms of gender. Added to this distance between body and identity, Butler speaks of the body as performative process of embodying cultural and historical possibilities. These possibilities, which are delimited by historical conventions, are thus materialized on the body, and, in this way, Butler argues that "one does one's body differently from one's contemporaries and from one's embodied predecessors and successors" (ibid., p. 521) Therefore, the body comes to be a "historical situation" that results from the performativity of embodiment itself. In other words, the actions related to what Butler calls the "structures of embodiment" constitute an ontological sphere of present participles, such as 'doing,' 'dramatizing,' and 'reproducing.' Futhermore, what this structure of embodiment entails is the constitution of not only gender, but also style. For example, in her definition of *woman*, Butler writes:

> to be a woman is to have *become* a woman, to compel the body to conform to an historical idea of 'woman,' to induce the body to become a cultural sign, to materi-

> alize oneself in obedience to an historically delimited possibility, and to do this as a sustained and repeated corporeal project.

In this sense, since gender is constituted temporally, it is necessarily historical. Consequently, far from being a prescribed given, the constitution of gender on the body is itself a result of mediated history, that is, a creative act of interpretation and reinterpretation that reveals itself on the body, not as an expression that comes from within, but as the sedimented layerers that deposit themselves in time. Furthermore, within this notion of temporality there is the need for repetition, but a repetition that is susceptible to breakage, or what Butler refers to as "subversive repetition" (Butler 1988, p. 520). In being a temporal identity which reveals itself through a "stylized repetition of acts" that take place on the body —i.e, in the mundane instantiation of bodily gestures, movements, and enactments—, gender constitutes an "illusion" of a gendered self. In the ontology of databasing, repetition comes to be a crucial aspect of the temporality of the practice, one that ressembles the gender performativity. Furthermore, the performativity of gender is necessarily discontinuous, and with this discontinuity exists the possibility of gender transformation. This is to say, in databasing terms, that the performativity of the database, if understood as representative of this discontinuity, or better, fluidity, can appear in a plurality of styles.

The database is a collection of facts. It is facticity itself. This is what Butler's constituted gender can teach about databases: in performing the database, the database appears, like gender, as a historical situation. Its body is felt, but neither as the database body —as if the materiality of the computer's architecture could come as a proxy for the nonhuman body—, nor as the extension of the embodying databaser, that is, as a prosthesis that expands the databaser in an expressive way. The body of the database emerges as a phantom, as spectrality itself, and it is this nonhuman presence that engages in the publicness of performative acts. The specter of the database must not be understood spiritually, or as a *deus ex machina*, or as a soul, or singularity that begins to act *as* human and, by extension, superceeds human in being better at it, thus replacing the human. It is simply a nonhuman fabrication of selfhood: there, around, making its way through the rupture of the permanent condition of performativity to which we —humans and nonhumans— are phenomenologically bound. That is the how the style of the database appears. This nonhuman self, like Butler's gendered self, is equally 'outside,' that is, "constituted in social discourse" (Butler 1988, p. 528). This is to say that the skin of the database is open for perception outside of itself, and in fact, nothing of the database can be considered expressive. Inside the database there is literally nothing but zeros and ones, nothing but data; in the same way, nothing is inside of the body but flesh, bones, and veins. When considered as internal, inherent, or essential, the classical notion of the self, in its heteronormativity, is seen as a "publically regulated and sanctioned form of essence fabrication" (ibid., p. 528). This is to say that, in this state of being fabricated, expressivity serves as the foundation for what Butler refers to as the 'punitive' aspects of wrong gender performance. In this sense, the social quality of acts that fall outside the regulated binary gender construction work their way into punishing the body, incarcerating it, severing it, as is, for example the famous case of Turing himself:

> Turing's later embroilment with the police and court system over the question of his homosexuality played out, in a different key, the assumptions embodied in the Turing test. His conviction and the court-ordered hormone treatments for his homosexuality tragically demonstrated the importance of *doing* over *saying* in the coercive order of a homophobic society with the power to enforce its will upon the bodies of its citizens. (N. K. Hayles 1999, p. xii)

2.3.2 Towards the Limits

Databasing, insofar as it is considered in terms of its performativity, is a function of what Nancy calls 'exposition.' This exposition is the appearance of a limit, that is, the finitud of a singularity. With this limit, which is instantiated in the (public) moment of the performative act, is how communication comes to emerge as that which is in common among singularities. This is to say that, because it is itself nothing, that is, "neither a ground, nor an essence, nor a substance," Nancy considers finitud to just appear: "it presents itself, it exposes itself, and thus it exists as communication." His emphasize on communication as exposure or finitud —i.e., as limit— marks a crucial distinction on the concept of community iself. For Nancy, as I have described above (See 2.1.4), community cannot come from an instance of work: it emerges as an instance of, precisely, the communicative action which is "the unworking of a work that is social, economic, technical, and institutional"(Nancy 1991, p. 31). In other words, the performativity of this communicative act, the publicness and dramatic qualities with which it unfolds, result in exposure. This is why, in te case of databasing, what is exposed at the limit of its performance is the finitud of the database itself. Throught this exposition of the limit is how the singularity of the database can be communicated; or, better, throught this exposure is how communication expands our concept of community towards one that includes the database as an agent of community itself.

As Nancy writes, "A singular being does not emerge or rise up against the background of a chaotic, undifferentiated identity of beings" ibid., p. 28. This is to say that, like Butler's gendered self, there is no substance within singularity. The appearance of a database as finitud comes not from an originary *archē* which would impose its archontic power as is the case of the feverish condition Derrida diagnosed in archives. Further, the limit of the database is not instantiated out of the one —i.e., a "unitary assumption" ibid., p. 28, the wholeness of a single 'one.' Finally, this finitud does not come from intentionality or any essentialist notions: it simply appears, as a ghost, or, as Nancy claims, as skin:

> . . . as finitude itself: at the end (or at the beginning), with the contact of the *skin* (or the heart) of another singular being, at the confines of the same singularity that is, as such, always other, always shared, always exposed. . . ibid., p. 28 (Italics mine).

Now what would this skin, understood as the spectral limit of the database, mean in terms of gender? How does this skin resonate aesthetically in database music? The limit of the database, as performative, spectral skin, is the condition for community to emerges between the human and the nonhuman. This means that the agency locus of the database needs to be placed precisely on its skin, because it is what becomes public of itself. In other words, given that this skin is available to the perception of others, it becomes touchable, it reaches our own limit as databasers. By doing so, that is, by exposing our own limits to ourselves and to each other, the database changes our definition, or, better, delimits the extent of our own singularity.

However, this does not mean that it stands in the way of our performativity, or worse, that it precludes or determines ourselves. If this were true, we would be once again subject to technical determinism, essence fabrication, etc., and falling out of considering any possibility of community between anything that is not human —or, more accurately, anything that is not 'man.' As Nancy rightfully claims, " it is not obvious that the community of singularities is limited to 'man"' Nancy 1991, p. 28. Thus, the fact that the skin of the database changes our own skin simply means that we are already in communication with it, that is, in community, and also in a state of resonance with it. This is the function of the skin of the database: like the skin of a drum, or the skin of a loudspeaker, the skin of the database resonates with our own skin, engaging the resonant body with the resonant spectrality. This is the community of resonance, the community of the resonant network, which has no purpose, no intentionality behind, no essence; only appearance and motility, performance and repetition: an activity that is the unworking of which community is composed.

To what extend is the skin of the database resounding in database music? How does this sonic apparition manifests itself? What conditions need to be met for it to be available for the listening experience? These questions can be answered with a closer inspection of the structure of databases. I have described above the different database models that were consolidated over the years (See 1.2.3). By no means the database models have converged into one; quite the opposite, these models tend to reside next to each other, that is, either within a single database system or within an interconnected network system that enables network communication between different database systems. With this plurality of the model, databasers have access to the many features that each model offers, focusing on those features that are suitable for their needs.

In the case of a computer music software like Pure Data, for example, this is certainly the case. While not technically a database system, Pure Data comprises (internally) a limited amount of data structures that are, nontheless, different between each other. These structures are, in turn, arrays and linked lists built as a layer of the C programming language. In terms of database models, Pd is mostly hierarchical when it comes to, for example, canvases —i.e., the windowing system that has a 'root', and multiple 'subcanvases' that can be (almost) infinitely nested. These canvases, while being hierarchic, can be navigated, that is, traversed as in the navigational model, either for a specific keyword —i.e., a query from the 'find' menu—, or, most importantly, for order of execution when it comes to signal processing. Pure Data is also object-oriented, since it depends on class instantiation to function. Every internal and external is a class made of C data structures, with its own methods, that can be loaded in-memory at run time and instantiated any time afterwards. Furthermore, the ".pd" file, which is designed visually by connecting boxes with cords, is literally a directed graph, where objects are nodes and edges are assigned to a node's *iolet*[13]. Furthermore, Pure Data is already a networked environment, since in order to effectively 'patch' using the graphical interface, a network is established between Pure Data instance —i.e., the running instance of the compiled C program— and the Tcl/Tk interface. Added to this, the network capacity that Pure Data comes with, that is, the "pdsend" and "pdreceive" objects that support creation of endless TCP/IP connections, literally exploding the concept of a hierarchical patch into the non-hierarchic, networked model. Among other things, this means that —and this is a common warning that Pure Data developers have to announce—[14] if you open a listening port and share your port number, anyone can connect to that port, without any restriction whatsoever. Finally, the inherent opennes of the source code enables programmers not only to create and load externals (in C, C++), also to change the program itself —that is, in being open, Pure Data prevents any definition to reach completion; or, for that matter, any paragraph such as the one you just read from being complete.[15]

[13]Programming 'slang' for inlet/outlet.

[14]Miller Puckette suggested this during an open discussion at `PdCon16`

[15]In fact, Pure Data is just an example, and several other open and non-open source computer music sofware come to represent such a plethora of models for the user. This is what makes the skin of the database something that reaches any databaser, or, better, any computer user that has ever moved a mouse…

The point I am aiming at in this section is that the skin of the database is as fluid as the constitution of gender, and if this is true, then the fluidity of Pure Data itself comes to represent the constitution of gender through the performativity of databasers. This is why it is as difficult to pinpoint what the skin of the database is, as it is easy to find yourself in resonance with. By resonating in such a performative way, databasers approach the limit of the database. However, this does not mean that they reach it: approaching the limit means, precisely, that it never arrives. Thus emerges the difficulty, because, like Butler's constitution of gender, the meaning of this skin does not come in form of an expression, that is, it "expresses nothing" (Butler 1988, p. 530). It is plain and simple appearance itself, or, to reconfigure the meaning of the name of the software, it is 'pure' skin, insofar as 'pure' is taken to mean 'sheer,' that is, not at all related to an essentialist, or classical notion of purity of any kind.

2.3.3 Contingencies of Style

> . . . style, supplementing timbre, tends to repeat the event of pure presence, the singularity of the source present in what it produces, supposing again that the unity of a timbre —immediately it is identifiable— ever has the purity of an event.
>
> . . . The timbre of my voice, the style of my writing are that which for (a) me never will have been present. I neither hear nor recognize the timbre of my voice. If my style marks itself, it is only on a surface which remains invisible and illegible for me. (Derrida 1982, p. 296)

'Style' comes from the latin *stilus*, meaning a sharp object with which you can write: it is a writing tool. Its meaning extends through writing to the manner in which the writing is carried out: the variations and oscillations of the pen and of the text itself, hence resulting in the style of a certain text, or, for that matter, a programming style, or even the style of an author. Beyond writing, style becomes the way in which the body moves, how it looks, whether it is human or nonhuman: the style of a music work, the style of a composer; and beyond, the style of an entire musical period, thus extending style in time and space. Most important, for the purposes of this section, is to note that style is a manifestation of the singular, that is, in the sense that style does not lend itself to duplication, and, provided that it happens as the apparition of an event, it exposes singularity as such. In this sense, style is comparable to the voice —specifically, to the sound of the voice— of a certain author. That is to say, style and timbre understood equally as the presence of the singular: the signature that comes with the unique and irreproducible timbral quality; what in signal processing terms may be approached —but not reached— as the "timbre stamp" of a sound.[16] In this sense, Derrida writes:

> In its irreplaceable quality, the timbre of the voice marks the event of language. By virtue of this fact, timbre has greater import than the *forms* of signs and the *content* of meaning. In any event, timbre cannot be summarized by form and content, since at the very least they share the capacity to be repeated, to be imitated in their identity as objects, that is, in their ideality (Derrida 1982, p. 296).

The case in point is that the style of databasing is of singular emergence, but only insofar as databasing is not separated from performativity. An unperformed database comes to represent the disembodied 'base', that is, the spatially ordered set of computer hardware together with the software routines that it embedds, at the most 'basic' level of instantiation. A foundation upon which the database tree can be performed. The 'base' in database comes as a stage for databasing itself: a stage without performance is an empty stage, extension of space itself, and such disembodied space resides in the plane of ideas. Databasing, on the other hand, projects its own style as a result of its performance, and through this projection comes the exposure of its skin, its singularity that is also its voice, and its sound. The "stylistic repetition of acts" in the dramatic case of the gendered database is now revealed as style itself, and this will be clarified in what follows.

Ths skin of the database, which is also its sound as it is resonating like the tympan, comes as a singularity in the event of performance, but it is, in fact, the result of a complex web of interrelations, that is, a historical situation that interweaves a plurality. This is the case of the networked condition of databasing, or to use Terry Winograd's words —quoted by Horacio

[16]"Timbre stamp" —also known as "vocoders"— is a type of Fourier-based filter in which "the spectrum of one sound is used to derive a filter for another" (M. Puckette 2007).

Vaggione—, the "complex systems" (Vaggione 2001) of which the human operator is just another component.[17] As part of these complex systems, composers perform the skin of the database, which appears in every resonant node of the network. In this skin, which unfolds in the duration of the performative act, what is exposed as its singularity is the ruggedness of the traces of which it is composed; the discontinuities it has by its constitution of itself as programming style, as music style, as timbre, as sign. The length of this skin —or, its surface, its duration— can only be estimated because there is simply no possibility of rendering it complete. In its fractured state, because of its fractal qualities, it points to infinity as well as to chaos itself. In this sense, databasing means participating in the infinite, taking a small part of the infinite, or better, performing the infinite within the limits of our own embodiment. Likewise, databasing refers to a state of being in chaos, or better, of embodying chaos itself as the contingent situtation that is resonance within the frayed spatio-temporal configuration of networks.

Considering databasing as chaotic systems brings yet another aspect to the contingencies of style that I am pursuing in this section. For instance, given that this style can be considered as a emergent singularity of databasing, this singularity can be considered as well deterministic. In mathematics, determinism refers to the capacity to predict results, specifically, by solving differential equations. This is the case of dynamic systems studied within chaos theory, for example, the case of the Lorenz system of equations.[18] Dynamic systems are defined, among other principles, by their capacity to render very different and quite unpredictable results by minimal changes on its initial conditions, despite the fact that they are indeed deterministic systems. Considering, thus, databasing as a dynamic system, on the one hand, two performances can be exactly the same if given the same initial conditions and states. This is the case, for example, of the performance of fixed media (digital) works, which —at least at the sample level— every bit of it is exactly the

[17]Max Mathews refers to the human operator simply as a 'cooperator', see (M. V. Mathews 1963)

[18]The 'lorenz attractor' is a system of differential equations discovered by Edward N. Lorenz in 1963, following experiments on weather conditions prediction. The attractor is most famously recognized by the butterfly-like appearance of its visualization, which is also related to the concept of the 'butterfly effect.' See `https://en.wikipedia.org/wiki/Lorenz_system`

same of the original bit that was rendered by the composer. This is the most extreme case of utmost determinism, which is given, precisely, by the metaphorical stage of the 'basis' upon which databasing is build.

However, identifying predictability in this way means falling in a cybernetic trap, of which Hayles already warned about when considering Turing's Test. Hayles reads Turing's test as a game, precisely in the sense that in order to enter into the dynamics that the game proposes, you are already part of the outcome that the game was intended to predict: in Turing's case, the moment you enter into the disembodied place where the screen is the only thing you see, you are already a cyborg, and the definition of the human and the nonhuman is already layed out in principle. In this case of deterministic databasing, in identifying the fidelity of data storage with fidelity of performance, you are already removing the human out of the concert stage —and the question of performance altogether—, leaving only the idealist and romatinc notion of the work of art in its pure, objective state. Therefore, in order to allow for the skin to emerge, that is, in order for the style of databasing to emerge, one has to consider not only the actual staging of performance, also the staging of listening itself, which is the possibility condition for the resonating subject of the database to emerge as the communicative apparition of a skin. This is the chaotic state, the contingency of style, the unpredictable agency of the unfolding that takes place in the moment of databasing. Because of the apparition of this contingent style, we are faced, or better, touched, in contact, exposed to aesthetic experience as such.

This aesthetic experience, it must be noted, is of a nature that slips through the cracks of traditional conceptualizations of the work of music as a result of stylistic (I use *stylistic* to be provocative here) or stipulated constraints on the part of the composer, or stochastic procedures. For example, composer Horacio Vaggione goes to great lengths to prove, that, indeed the musical work affirms itself as singularity, in this particular sense that its rules are only prescribed from (and to) itself from within, and always in a "action-perception loop" with the composer. What Vaggione is arguing against is the tendency of formalized musical processes that had reshaped the role of the composer working with computers: "the role of the composer here is not one of setting a mechanism and watching it run, but one of setting the conditions that will allow him or her to perform musical actions" (Vaggione 2001, p. 58).[19]

There is a fundamental concern that needs to be addressed in relation to the contingency of style, which relates to Vaggione's constitution of the singular thing that is the musical work. Style, in Vaggione's words, comes to represent the reified status of the rules within a work, insofar as this reification is taken for the starting point from which to compose, and not the result of a composed thing. Consider this quote from an earlier text:

> Here lies what seems to be one of the sources of confusion regarding the nature of music composition processes: on the one hand, we must make as careful a distinction as possible between the collective rules and the composer's own constraints; on the other, this distinction seems irrelevant [because] any primitive (coming from a common practice or postulated ad hoc) is to be considered as a part of what is to be composed, *in order to produce a musical work affirming itself as a singularity, beyond an exercise in style*. Adorno was of course conscious of this dialectic: his statement about sound material considered not as something "given" but as a "result" of a musical thesis clearly points to this fact. (ibid., p. 59) (Italics mine)

Despite the fact that distinguishing between rules and constrains, that is, between socially and historically established canons —as stylistic conventions—, and locally established postulates to be carried out by the composer —as constraints—, is crucial in defining style in databasing, at the same time both ends collapse into the realm of arbitrariness that composition is made of, for,

[19]"I am not denying the utility of these procedures, in that they are used by scientists looking for regularities. A composer, however, knows how to generate singular events, and how to articulate them in bigger and bigger chunks without losing the control of the singularities: this is what Myhill calls stylistic coherence, and Kramer syntax. This is why it is always problematic to utilize global causal laws (e.g., stochastics) in music composition, if their automatism is not compensated by compositional choices concerning other levels of articulation. This is why Xenakis, after being concerned with Markov chains, subsequently adopted a consistent silence about his more recent compositional procedures, a silence not broken by his claim that he has introduced an "arbitrary" (that is "composed") manipulation of the data provided by his stochastic canons (Xenakis 1979)" (Vaggione 1993, p. 97). Vaggione, continuing the tradition of Max Mathews and Jean-Claude Risset, advocates for a considering computer assisted composition as an instance of micro-time articulation: "the role of the computer in the shaping of musical time —which must be viewed as being the matter of computer-aided composition— needs to be further worked out, from the temporal behaviour of the partials of a spectrum, to the multiple temporalities involved in a complex sound-object, to the macro-time of the (causal and/or emergent, lineal and/or non-lineal) global form, without forgetting all the fractional dimensions in between, nor the structural relationships established at each level by interaction with all other levels.(ibid., p. 103)"

if any "primitive" of the composition is indeed to be considered "part of what is to be composed," then style itself becomes a result. This is what Vaggione means by "beyond an *exercise* in style:" it is not an exercise in the sense of a draft, in the military context of training, "practice for the sake of training."[20] Style is not an exercise because it cannot be operative in the sense that it is considered a product of work. That is to say, given that style is the contingent skin that emerges out of the performative action of databasing, if it were considered productively, that is, in terms of a targetted object towards work is oriented, then it would only result in closure, in a closed object, and thus, it cannot be considered contingent, precisely because it is stipulated from the start as a law to which every composable element abides. This is the case of, in Vaggione's sense, formalized processes which act as global laws, and in which —comparing it with a marching army following the one-two directive: "no singularities —and hence no specific formal (relational) properties— are here allowed" (Vaggione 1993, p. 101)

However, if style is understood as inoperative, that is, as the unfolding of the performative act of composition, which implies equally the 'composability' of all primitives, then, its contingency appears in the form of exposure, not as a closed object, but as an unclosed object, some *thing* that is exposed and bound to exposure; a thing that exposes us in the same resonance of its touch. Like the marks on our skin, like its wounds; like the cracks of an old house, like debris, wreckages, or any form of residual mark that is the evidence of an event; with forensic intimacy, the contingent style of a musical unwork reveals itself in communication. This is what connects aesthetic experience of style with forensic (musical) analysis as well as with an encounter with the spectral. Furthermore, this is how the spectral itself cannot be but a result of the inoperative, of that which escapes the limits of the work —e.g., the constraints[21]—, and that which, by releasing

[20] `https://en.wiktionary.org/wiki/exercise`

[21] Vaggione writes about constraints: "I use the expression 'constraint' in the sense of its etymology: limit, condition, force, and, by extension, definition of the degrees of freedom assumed by an actor in a given situation within self-imposed boundaries. In this broader sense, the composer's constraints are specific assumptions about musical relationships: multi-level assumptions that can be in some cases translated into finite computable functions (algorithms), and in other cases satisfied only by means of the composer's interaction (performance). Constraints are embedded at every level in the 'world' posited in the musical work"

itself from itself, returns to itself like the timbre of the voice. This voice of the unwork is what is "invisible" to it, or better, it is what it can never listen, and in being hidden or silenced from itself is how it becomes available for listening, what begins listening at the first staging of the waves.

How, then, can the first sign of the database skin be considered as a self? How is the figure of an 'author' —i.e., databaser, composer— still at play after Vaggione's compositional approach, where both author and composer are simultaneously exploded as network, but reified back into the name, by the notion of 'work'? I don't attempt to provide an answer to this enigma, but in what follows, I can perhaps approach the name of the composer not by its work, but from a very different, illusory perspective of authority.

2.3.4 A Specter of Authority

> Gender is instituted through the stylization of the body and, hence, must be understood as the mundane way in which bodily gestures, movements, and enactments of various kinds *constitute the illusion of an abiding gendered self* (Butler 1988, p. 519) (Italics mine).

However composable all Vaggonian primitives can be, the structure of the database tree itself is so vast that any attempt to comprehend it as a whole would extend it even further. This is the condition of networks (See 2.1.2). However, by no means this determines, neither the extent of the performativity of databasing, nor the agency of the human itself. Quite the contrary, expansion throught the network can be considered as the trace of the author, or better, the shadow of an illusion: the elonging of the shadows of a spectral author. Further, with the performativity of databasing, the databaser is bound to be an incomplete whole, meaning that no extension ever reaches an end.[22]

The infinitude in this fractality of databasing, however, is at some point reified in a figure or a name. This figure is the place where authority is condensed, and it responds to traditional

[22]For example, that which what escapes the databaser in Pure Data: the programming language C; or, even databasing in C: compiler instructions; etc.

—essentialist— conceptualizations of the romantic author which, despite the many attempts during 20th century,[23] are still in effect today, specifically in the field of music composition. It is not the purpose of this section to criticize this excellent tradition, namely because I don't consider it relevant for the purposes of databasing. That is to say, in the case of databasing, such figure of an essential author is simply dislocated —i.e., forced upon the structure of the network— and anachronic —i.e., a temporality set against the temporality of networks—, and in so being, focusing on it would be missing the point. Databasing, as resonant performativity already exists beyond this traditional figure of the author. However, in its spectrality —stemming from the archontic (See 2.2.3)—, authority can be seen as the illusory resonance of an author. It is this illusion that I attempt to address here, this ghost which haunts music composition.

Then, it is not a coincidence that I bring here the name 'Vaggione,' at least to the extent that he, as composer and author, but also as his own spectral voice in his music and his writings, can exemplify the figure of authority in composition. It is not a coincidence because it is a name that I have created over the years, as I am sure there are as many Vaggiones as there are grains in his music. The Vaggione that I have created is one that haunts me personally, because we both come from the same place (Córdoba, Argentina), attended the same university (National University of Córdoba), facts that, for my own situation as young composer, resonated deeply in the music and research that I pursued over the years.[24] Further, this discussion of authority of the name Vaggione does not intend neither to criticize, nor to deconstruct the name itself, it is only an approach to to the spectrality of the author.

Consider how different styles, as expressions of compositional authority, are used in some cases of algorithmic formalization, for example, in David Cope's *Experiments in Musical Intelligence*. This is a case of intentional stylization by way of databases, that is, by transcripted

---

[23] See for example Roland Barthe's 1967 *Death of the Author*, or Michel Foucault's 1969 text *What is an author?*, both of them commented on in (Daniel 2007).

[24] As I have described above (See 2.2.4), Derrida claims that addressing phantoms is a transaction that is familial and domestic; thus is how I feel when thinking of the name Vaggione.

MIDI data from a given composers' score corpus. Written in the functional programming language LISP, EMI's focus is "style imitation" in order to assist the composer when in front of a "composing block," provoking the "author into almost immediate action. Any blank moments along the way are immediately filled by simple queries…" (Cope 1987, p. 38). This is how composer (and PacIOOS Data System Engineer) John A. Maurer IV describes it:[25]

> [Artificial Intelligence] systems have the further capacity of defining their own grammar —or, in essence, a capacity to 'learn.' An example of this is David Cope's system called Experiments in Musical Intelligence (EMI)…[It] is based on a large database of style descriptions, or rules, of different compositional strategies [, and it] has the capactiy to create its own grammar and database of rules, which the computer itself deduces based on several scores from a specific composer's work that are input to it. EMI has been used to automatically compose music that evokes already somewhat successfully the styles of Bach, Mozart, Bartók, Brahms, Joplin, and many others (IV 1999, p. 3).

In this case, the database of MIDI-transcribed scores comes to represent the internal relations between composed elements, at least in the case of common music practice, from which probabilistic generalizations are drawn. This is what the 'intelligence' would 'deduce' of the database, and thus how the 'grammar' of a composer can be indeed formalized (Maurer's words). Cope's work on this matter, therefore, is aimed at these generalizations which, in turn, reify in a formalized style the authority of the composer. In other words, Cope's is an analytical approach using databases —similarly to the case of music information retrieval (See 1.3.1)—, aimed at a re-synthesis not of sounds but of style altogether: a rule-based re-composition. The case in point, in relation to this formalized approach, is not only that by performing this type of machine learning techniques databasing becomes at once detached from the human, it also replicates a notion of the idealized work of music. Since, even if the data was inputted by Bach himself, the output would not only be entirely deterministic, it would be the result of globally defined operativity. Hence, it would be impossible to think of this formalization of style as an instance of resonance, for example,

[25] `https://ccrma.stanford.edu/~blackrse/`

since there is no moment of resonance at all, no exposure between the nonhuman and the human, no communication.

Precisely these, among other more traditional cases of formalization —which can be found, to a certain point (a clearly delimited and well commented one in the literature) in the work of Lejaren Hiller and Iannis Xenakis—, is what Vaggione is addressing when he proposes the equal role of the "informal" or the "craftmanship" of the composer using computers. In a very different case of the use of databases, consider Roads' account of Vaggione's workflow when composing the work *SHALL*:

> These involved arranging microsounds[26] using a sound mixing program with a graphical time-line interface. He would load a *catalog of pre-edited microsound* into the program's library then select items and paste them onto a track at specific points on the timeline running from left to right accross the screen. By pasting a single particle multiple times, it became a sound entity of a higher temporal order. Each paste operation was like a stroke of a brush in a painting, adding a touch more color over the blank space of the canvas. In this case, *the collection of microsounds in the library can be thought of as a palette*. Since the program allowed the user to zoom in or out in time, the composer could paste and edit on different time scales. The program offered multiple simultaneous tracks on which to paste, permitting a rich interplay of microevents (Curtis Roads 2001, pp. 313–314) (Italics mine).

While this workflow is only representative of certain aspect of the piece in question, it does serve as an example of his concept of craftmanship. Makis Solomos also call this 'direct' or 'manual' action on the side of the composer, and quotes Vaggione: "to write music 'manually', note by note, partial by partial, or grain by grain, is an approach proper to a composer, and he should not be embarrassed about using this aspect of *his craftsmanship*"' (Solomos 2005, p. 3) (Italics mine). Beyond the obvious gender bias in this last sentence, I would like to refer to 'craftsmanship' instead as 'artistry,' keeping its signification to hand-made art, but also relating it with articulation, one of Vaggione's crucial concepts. Further, this 'hand', as Solomos very well points out, is not to be understood as being without the tool —in this case, the 'mouse'— that it needs

[26]The word 'microsound' refers to sonic events shaped below the threshold of the 'note.' See (Curtis Roads 2001)

to use in order to precisely locate sounds on the timeline interface.[27] This call for artistry stemms from the radical formalisms that governed computer-assisted composition in the early stages of computer music.[28] In resonance with the formalist backdrop, Vaggione built his terminology, not in opposition, but in the spirit of reconfiguring computer-assisted composition from an embodied stance coming from outside information theory.[29] Therefore, the focus, instead of being on the rule-based programming of formalization processes alone —i.e., instead of residing on the keyboard—, the artistry of the composer resides in the use of the mouse. The timeline (i.e., sequence) interface (See 1.1.1) workflow depends on the mouse pointer. It also depends on the software's capacity to store soundfiles in memory (See 1.3.3) and, in Vaggione's interes for micro-time composition, on the zooming capabilities of the software, that is, so as to 'descend' into micro-time (Vaggione's word).

So if the presence of the hand of the composer is evidenced by the trajectory or the trace of the pointer, that is, by its scripture (Vaggione also uses *écriture* to refer to manually composed chunks), to what extent is the shape of this trajectory of the pointer —and also the historial of clicks, the drag-n-drop motions, etc.— the spectral presence of the author? To what extent is this mouse —that is, the point of the pointer, the writing device, the 'stilus'— that with which we resonate as listeners, and thus how we perceive the marks in the authorial skin of the music? The Vaggionian singularity-based approach to authority outlined above, embeds composers and computers in a complex system, allowing for the world of music with computers to be a hybrid one. This is how the specter of the author coexists with the specter of the database, and thus, how databasing and composition reveal themselves to be instances of performativity that resonate

[27]"It is clear that this 'hand,' is not necessarily working without the extension of a computer's mouse (Solomos 2005, p. 4).

[28]"During its protracted beginnings, computer-assisted composition was dominated by the ideal of automatic music – work by Hiller, Barbaud, Xenakis or König comes to mind" (Solomos 2005, p. 3)

[29]This stance is not only evident in Vaggione's writings and music, it is, to a debatable extent, a point of departure to think of a branch of Argentinian electroacoustic identity that developed in France; an identity that reveals singularly in the work of Beatriz Ferreyra, and in the spatial music of Elsa Justel. For an approach to Justel's timeline-based spatialization techniques, see (Camara Halac 2018a).

aesthetically through the work of music.

Now I have arrived at a point of articulation —the work—, which in Vaggione's terms comes to mean the operativity of the composer, but which, in the following chapter, I will attempt to distinguish from the operativity of the piece of music itself. Vaggione's term comes from computer science, and I can link it with Max Mathews' concept of the human as 'co-operator' with machine 'co-operators' in relation to computer music. This operativity coming from humans and nonhumans is not into question; however, when aesthetic experience is constructed upon operational standards —e.g. in the expresion "this (music) works"—, then comes an issue that is not only related to aesthetics, it enters the realm of the political.

Chapter 3

Database Politics

In search of understanding the political in Database and Composition practices, I question the established concept of music composition and arrive to new definitions of the music work, practice, and authorship.

First, I consider the concepts developed in the previous chapter to understand Music Composition as Database Performance. I propose that the ontology of Composition needs to be redefined in terms of the agency of the Database. My goal in this section is to reveal that the Database agency, when contextualized within Music Composition, has the form and the politics of a music listening to itself.

Second, I use Nancy's concept of inoperativity to redefine the music object. I argue that the inoperativity of the listening experience, which resides on the delay between sense and sensuality, provides insight on the type of unworking that affects music composition. I thus redefine the outcome of music composition as the *severed music object*, emphasizing its inoperative status of suspension, withdrawal, and its inherent state non-completeness. I then consider how this state of suspension of the severed music object can be analyzed in terms of a Community of artists, database performers, composers, etc., mutually exposed to each other (Nancy 1991). Therefore, in order to understand the dynamics of this transversal community of Database and Composition, I analyze the paradox of anarchy and reflect on the consequences of both the anarchic and the inoperative in Database and Composition practices.

Finally, I present my view on collaboration, and propose a redefinition of the term uprooting it from the traditional union of forces forming a whole. I claim that the new form of collaboration can be understood as a form of collective, or *trans-inoperation*, consisting in the mutual exposure of the limits of singular, performing beings. As a consequence of this form of collective inoperance, I claim that a new politics of authorship needs to be analyzed, particularly in terms of the spectral in the Database. I question the power of this illusory figure in terms of the effectiveness of the archontic principle that is present in *trans-inoperant* works of art. I believe the specter of the author loses the sensuality and the sense of the listening subjects in state of trans-inoperance, and thus the power of the author ceases to take place.

3.1 Rethinking Composition

The concepts exposed in Database Aesthetics affect and reconfigure transversally the practices of composition and databasing. Traditionally, music composition was considered a single author practice, in which the composer's technique or aesthetic intuition is the sole agent, romanticizing the artist as an "involuntary vessel through which inspiration flows" (Born 1995). As I have outlined in , this is no longer the case, since understood in terms of its resonance and of its performativity, composition explodes the name of the composer, leaving as many spectral remains of its trace as can be imagined. Conversely, databasing is already embedded in a networked structure that only allows partial and temporary allocation of authors (databaser), since in the structural database tree exist multi-authored branches that renew themselves, outgrowing themselves in perpetual difference and instability. The notions of stability and authority can only be related to snapshots in the history of a software. However, the institutional quality of both databasing and composition is still at play, namely in the many cases of proprietary software and in the composer's name, that is, in the commercial release and the objectification of music work. Less than focusing on general criticism, in this section I argue that, since the agency of the database reveals itself as aesthetic experience, then it is the dynamics of this agency need to be addressed. I claim that this agency, when contextualized within music composition, specifically composing with computers, it has the form and the politics of a music listening to itself.

3.1.1 Performing the Database

How does thinking of database music affect the practice of music composition? One would have to begin at the origin, start at the beginning of composition, that is, at the moment of performative action that I am calling databasing. Identifying music composition with databasing would mean to interpret both practices under the scope of computer practices. That is, not just databasing, but computer-based databasing; and, likewise, not just composition, but computer-based composition.

In today's composition and databasing practices, the probabilities of a composer or a databaser working without computers are very slim. Databasing or composition outside the digital seems rather fictional. However, the very image of a 'composer,' which traditionally stems from romantic standards, is already outside the world of computers. This image of composing can be painted as follows: the composer at work, quietly on a desk with pen and paper, transcribing, arranging, making parts, drawing line after line, dot after dot, notating instructions for the performance of an imagined music. Where is the computer in this image of composition? Certainly, placing a computer on this idyllic desk would be anachronic and obtrusive; anachronic, since the romantic quality of the scene would point to the fact that personal desktop computers were not available until late in the 20th century; obtrusive, in the sense that it would attempt against this composer, by interfering with the 'ethereal' link between imagination and notation.

This reification of the composer already precludes not only the digital, also the many technological devices that have entered music composition over the years, such as tape recorders, or electronics in general. These technological devices have redefined the composer in many ways. For instance, as explained throughout Georgina Born's ethnography of IRCAM (Born 1995), by way of institutionalization, which more often than not resulted in hierarchical structurings of work dynamics that were coated with false notions of collaboration.[1] At least in the particular case of IRCAM throughout the 1980s, the inequalities of social, economical, and political status among technicians and composers within the institution became privately evident. Knowing how to use computers and knowing how to compose comprised two irreconcilable poles in the institutional structure. One would be tempted to link this irreconciliation to the (extreme) reification of the image of the (composer) name Pierre Boulez. The obscure dynamics behind this reification, however privately and secretly kept they were within the institution —hence Born's mysterious (but telling) anonymization of anyone but Boulez on her transcriptions—, can be nonetheless seen as the shadow of the more general specter of the music maker that, at least socially, has been considered as an outsider, marginalized, but simultaneously an integrator of society itself.[2]

Converseley, this shadow might also be that of the computer itself, the structural presence of a fictional intelligence constructed upon first wave cybernetics. That is to say, precisely because the computer projects an insurmountable power that comes from its calculations, the human is inevitably bound to be subordinate, and with this subordination comes the subordination of the composer, and the end of music. This hyperbolic reaction would explain the need for privacy and secrecy of information —the 'undocumented' 'oral culture' of Born's IRCAM—, as well as the (preventive?) reversal of the human-computer subordination that was evidenced in the social strata of the institution.

[1]For example, see the chapter titled "Aporias: Technological and Social Problems around Production" (Born 1995, p. 252), where Born describes these internal hierarchies in terms of 'superuser' password knowledge, source code access and software licences. An interesting case is the 'manual' modifiations to IRCAM's architectural space: "workers concocted their various informal ways of protecting privacy and retaining secrecy: blocking the glass walls of their studies, working at night to prevent others knowing what they were doing or even whether they were working at all" (ibid., p. 272)

[2]For example, consider Jacques Attali's placement of the musician within civilization as "simultaneously excluded (relegated to a place near the bottom of the social hierarchy) and superhuman (the genius, the adored and deified star). Simultaneously a separator and an integrator" (Attali 2009, p. 12)

In any case, coming back to the composition world of today, a composer without computers cannot be imagined, but this is not due to the practice of composition itself. My argument here is that in any given situation, it is hardly possible to imagine a human without computers at all. This is what media studies has to teach us about the posthuman condition in which we hybridly live, where humans and technology, humans and nonhumans, unfold as interminably networked traces. At the risk of drawing a straw-man out of this computer-less composer, it is very unlikely in today's world to imagine a composer that has not googled 'clarinet multiphonics' for more than a few YouTube tutorials on the topic. The same can be said for digitized music listening, which, in order to escape it, one has to go to great —cult-like— lenghts to do so: going to instrumental performances, getting a vynil record or a tape player, etc. A composer without computers today would need to whisper the score to the performers who would, in turn, play by ear.[3] The compose should also whisper invitations to a few neighbors to be part of the audience. The composer should also demand no recordings whatsover while performing for an audience that has been kindly reminded not to bring their cellphones. Even then, the concert would need to take place on an amphitheater —to avoid architectural networks, and, say, —, before the sun sets —to avoid electricity networks altogether while we are at it—, away from the city —a car driving by would be unforgivable—, so far away that we would, in fact, need to bring non-perishables for the pilgrimage —and even then, packaging networks or agriculture networks would be almost impossible to avoid.

And this is precisely the point: in attempting to avoid it, the pilgrimage exists not in space, but in time, and thus it enters into the realm of fiction. The same can be applied to the overloaded case of a composer totally *with* computers, that is, a 'computer composer,' that would not need the human to write music. Such a computer, rooted on information theory have been commented on before (See 1.1.2), particularly by Hayles' analysis of the evolution of 20th century cybernetics as reflected on fictional works (N. K. Hayles 1999). Therefore, composing with or

[3] 'By ear', in the sense that they would need to play from memory, since no printed score would exist, for even if the composer wrote the parts, the score would have to be inscribed on a paper —and somewhere along the way in today's paper networks there is at least one computer.

without computers cannot be seen as poles on a continuum upon which the name of the composer writes and rewrites itself. Composing cannot be separated from computers, because the human cannot be separated from the non.

The same applies to databasing itself (See 1.2.1). Removing computers altogether from databasing takes us to the world of libraries, encyclopedias, collectors, gatherers. Most important, it takes us to the place databasing occupies within society: to museums, but also to the dynamics of civilization, to church, put simply: to institutionalization itself. That is to say, it relates performance with the archontic, with the oedipal drive to re-place (See 2.2.3), to an infinite return that the structure of the archive imposes upon us. So, if we imagine a computer-less census, we'd have to picture a gatherer of names walking around town, asking out loud for each person's name and place of residence. Getting rid of networks which might have computers —as in the case of the above painted computer-less composer—, it is clear that the only suitable person for the job would be Irineo Funes (See 2.2.1), and the only possible storage medium would be his memory. Hopefully, the reader would consdider this resort to hyperbolic fictions less as a means of justification of the hybrid condition of composition and databasing, and more as an absurd parenthesis that brings nothing of criticism to the —still valid— efforts of working 'outside' the digital. These efforts are not questioned in regards of their validity, only in terms of their definition, which, for the purposes of this text, is understood as built upon organicist notions of the human: the human as the one, indivisible, complete whole. These notions are precisely those that help reify the image of the composer in its essentiality.

With these distinctions in mind, if we consider databasing as the instance of computer performativity, then it follows that composition and databasing have not only much in common, but they project similar structures that need to be addressed.

3.1.2 Working Composition

Consider Peter Szendy (Szendy 2008), and his discussion on Scheonberg's modern organicism and what he calls "the modernist regime of listening:"

> And the listening in question here is not that of a given listener, or of a category of listeners one has to take into account; it is rather structural listening in Adorno's sense —or even, beyond Adorno, a *listening without listener in which the work listens to itself*. ... we hear an organicist concept of the work being strongly articulated [by Schoenberg] (where the work is a whole that doesn't allow any cuts) and a regime of listening whose ultimate, ideal aim is the absorption or resorption of the listener in the work. A listener who is somewhat distracted, inattentive, who would skip over a few tracks daydreaming —*such a listener could fall away like a dead limb. Useless*. Bringing nothing to the great corpus of the work. This organicism, in the radical (or structural) tendency that Schoenberg gives it, forms the cornerstone of the construction of a modernist regime of listening (ibid., p. 127) (Italics mine).

Here is where 'listening' and 'work' collapse into each other, and it is how the problem of the musc work can be articulated. If this articulation is successful, the image of the composer and the practice of composition can be understood differently. By extension, databasing will be reconfigured as well, and the database —as that which is a product (or work) of databasing— will be seen differently.

In what does this articulation of the problem of the music work consist of? First of all, why is it a problem? As Szendy suggests with the metaphor of the the self-amputation of the listener, we —as the body of listeners, or better, the listening body— would be severed. Put differently, listening itself would be delineated from outside itself, that is, with the presence of an object —the music work— that, in its interest of perfecting, polishing, and thus giving it a 'finish,' would shape it and reshape it until an ideal listening was achieved. The work would work out listening as work. This is the point: the moment the work of music begins to act as 'work' itself, its listening is worked out as well, not by the physicality of the waves in media, nor by the virtuality of perception, but by the concept of 'work' itself. The "modern regime of listening" played this through and through, shaping its listeners into an idealized listener, that is, a measured listening, developed into different degrees of listening —like a *gradus ad auscultare*— existing beyond any psychoacoustic measuring.

From this shaping of listening by the work, that is, from the working activity that is performed by the work itself, the presence of the work as an object can be traced. That is to say, the work of the music work can be considered as the work of an agent in the composition network. Thus, the music-work-as-object and the listening-as-object would become nodes in the network. In the modernist regime, the ruler is work-node, that is, all that can be identified with listening-node is arrived at by subordinating the relations to the work-node, and by restricting the directionality of this relation to being $\boxed{\texttt{work}} \rightarrow \boxed{\texttt{listening}}$, with the only exception of the extremely cultivated case of composers, which revert the arrow. Furthermore, this exception is not so much an exception, as the prescription of the rule —and of ruling— itself, since it is this reversal what enables the structure in the first place: the ruling of the exception.

This is what occurs when the work begins to listen to itself. Which is radically different from the case of a *listening* that listens to itself. Jean-Luc Nancy, in the foreword to Szendy's text, considers that "music places us outside of ourselves" when he writes:

> . . . what listens to itself is not just what resounds in the self and what rebounds to the self: this same movement, and this very movement, places it outside of self and makes its rebound overflow (Szendy 2008, p. xii).

As I have explained before using Nancy's concept of the resonance of a return (See 2.1.1), listening is a relation to self. Only in a permanent state of overload, redundancy, or excess, this relation is represented by the 'arrow' or the 'edge' in the composition network that I have outlined above: $\boxed{\texttt{work}} \leftarrow \rightarrow \boxed{\texttt{listening}}$. In thinking listening this way, the concept of the work is relieved of its duties, discharged, fired, it becomes unemployed. The regime of listening becomes a listening space, but a space not of equality: a space of difference. Within this space where difference resonates, the music work no longer 'works', it 'unworks' (See 2.1.4). That is to say, the relations between the different resonating points in the composition network expose themselves in a state of suspension, or interruption, creating space with the space of their own incompleteness, by the fractality of their fracture. Thus, inoperativity is creation, it is techne, but it is a creativity that is necessarily indefinite, incomplete: the moment it becomes a thing it begins to work in the realm of the 'archi'; the moment it remains suspended upon its limit, it unworks in negation of the 'archi'. One is tempted to place this inoperativity in utopia, in the very instance of the non-place itself, but then one would forget what is already 'there', the fluid medium, as well as gravity itself, which was until recent studies, thought of as unrelated to sound.[4] One would be tempted, equally, to place this inoperativity outside temporality itself, but then one forgets forgetfulness itself. Inoperativity is within the resonating space of an always.

What constitutes, then, that moment when the music work becomes a work? How is it possible for the work to become a thing, for the object to become the ruler, for the regime to be built on the first place, if the resonating space is already an inoperative space, interrupted and suspended? I would like to revert Szendy's metaphor of the amputated limb, and propose that it is the music work itself what is amputated, what falls off, the moment that it becomes a finished thing. Thus, just like the modern (unattentive) listener falls in the uselessness of its excess, sound 'outside' the work is cut from the work, fading out in the uselessness of its excess. Like the human in Kittler's digitally converged apocalypse, redundancy is out of the question, it is left at the gates of the magestic concert hall, with the rest of the (useless) humans: it is literally and conceptually placed outside architecture itself. The created work, in its essential nature of being a cohesive, coherent whole, separates itself from the world of mechanical waves, and forms the one and only work: the piece of music. A 'piece' not because it is in itself incomplete, but because it is the piece of the whole of the work of the composer.

I would like to add to this worldview another concept brought by Szendy, that of 'absorption.' He claims that it is the absorption of the listener in the work what is the ultimate aim of this

[4] In a recent study, sound itself proven to make (tiny) graviational fields: "We show that, in fact, sound waves do carry mass —in particular, gravitational mass. This implies that a sound wave not only is affected by gravity but also generates a tiny gravitational field, an aspect not appreciated thus far" (Esposito, Krichevsky, and Nicolis 2019).

modern regime of listening. Not surprisingly, 'absorption' is the key concept in Iannis Xenakis' narrative of the degradation of Western Music's "outside-time structures," in the 1967 article "Towards a Metamusic."[5] However, Xenakis' narrative contextualizes his sieve theory, deviced as a means to "establish for the first time an axiomatic system, and to bring forth a formalization which will unify the ancient past, the present, and the future" (Xenakis 1992, p. 182). Further, Xenakis formulated this theory with computers in mind, that is, with its concrete application in computer programs, under the subtitle 'suprastructures':

> Moreover, in the immediate future we shall witness the exploration of this theory and its widespread use with the help of computers, for it is entirely mechanizable. Then, in a subsequent stage, there will be a study of partially ordered structures, such as are to be found in the classification of timbres, for example, by means of lattice or graph techniques (ibid., p. 200).

Therefore, considering the organicity of the work in Xenakis's overly modern gesture towards unity, metastructure, and mechanization, which was built in reaction to the "poison that is discharged into our ears" as he witnessed the "industrialization of music [that] already floods our ears in many public places, shops, radio, TV, and airlines, the world over" (ibid., p. 200), how can these notions of inoperativity be found together within architecture? How would this archi-techne be designed? Would it still be the product of the 'archi'? Where is the poison that Xenakis –the architect, the composer— was identifying with 'industrialized' music? Is it a product of modernity itself, as the working that listened to itself to the point of working out the Xenakis-listener-node to the extreme?

[5] "We can see a phenomenon of absorption of the ancient enharmonic by the diatonic. This must have taken place during the first centuries of Christianity, as part of the Church fathers' struggle against paganism and certain of its manifestations in the arts. . . " Later, refering to larger structural groupings: "this phenomenon of absorption is comparable to that of the scales (or modes) of the Renaissance by the major diatonic scale, which perpetuates the ancient syntonon diatonic. . . " Finally, "one can observe the phenomenon of the absorption of imperfect octaves by the perfect octave by virtue of the basic rules of consonance" (Xenakis 1992, pp. 189–190). The final stage of this process of absorption and degradation comes with atonalism, which "practically abandoned all outside-time structure" (ibid., p. 193).

3.1.3 The Composer as Navigator

> I am motivated to present this architecture, which is linked to antiquity and doubtless to other cultures, because it is an elegant and lively witness to what I have tried to define as an outside-time category, *algebra*, or structure of music, as opposed to its other two categories, in-time and temporal (Xenakis 1992, p. 192) (Italics mine).

> With [the relational] model any formatted data base is viewed as a collection of time-varying relations of assorted degrees... this collection is called a *relational algebra*... a query language could be directly based on it... The primary purpose of [relational] algebra is to provide a collection of operations on relations of all degrees... suitable for selecting data from a relational data base (Codd 1972, pp. 1–5) (Italics mine).

If we consider pitches as an outside-time (relational) database, one way of understanding Xenakis' sieve theory is as a query method, for which E.F. Codd's Relational model —and its subsequent — would fit perfectly. The nature of this consideration stems from the application of algebra as a programmable selection mechanism or simply, filters. Both concepts (sieves and relational algebra) have a common link, which is, not surprisingly, the computer itself, and not just any computer, the IBM-7090. While Xenakis' experiments were carried out on the IBM-7090 mainframe computer located at IBM-France in Paris, Codd himself worked at the IBM Research Laboratory in San Jose, California. Among other things, the IBM-7090 computer was used in the computation of the first 100,000 digits of π (Shanks and W.jun. Wrench 1962), Roger Shepard's computation of the homonymous 'shepard' tone (N. Shepard 1964), Alexander Hurwitz's computation of the 19th and 20th mersenne prime numbers,[6] and Peter Sellers' plot-twisting moment in Stanley Kubrick's "Dr. Strangelove or: How I Learned to Stop Worrying and Love the Bomb."[7] Most important, the computer used the programming language FORTRAN IV, as can be seen at the end of the chapter "Free Stochastic Music by Computer" (Xenakis 1992, p. 145), where the FORTRAN routines for Xenakis' 1962 work "Atrées (ST/10-3 060962)" are printed entirely.[8] Xe-

[6] `https://www.mersenne.org/primes/`

[7] `https://en.wikipedia.org/wiki/Dr._Strangelove#/media/File:Dr._Strangelove_-_Group_Captain_Lionel_Mandrake.png`

[8] Interestingly, given that Christopher Ariza finds Xenaki's sieves code unusable (Ariza 2005, p. 1), chances are that the printed code for the ST/10-3 composition is likewise useless.

nakis's work on sieves came a few years after his experiments on the IBM-7090, and his sieves program was written in Basic and then in C. However, the experience with FORTRAN IV at the IBM-7090 serves nontheless as a common ancestor to both Xenakis and Codd.

(Before continuing, a sound synthesis parenthesis must be opened. While Xenakis praised the speed at which the IBM-7090 could perform computations, Max Mathews (M. V. Mathews 1963), then director of the Behavioral Research Laboratory, Bell Telephone Laboratories, wrote:[9]

> A high-speed machine such as the I.B.M. 7090, using the programs described later in this article, can compute *only* about 5000 numbers per second when generating a reasonably complex sound. However, the numbers can be temporarily stored on one of the computer's digital magnetic tapes, and this tape can subsequently be replayed at rates up to 30,000 numbers per second (each number being a 12-bit binary number) (ibid., p. 553).

Mathews' concern for speed was grounded on the need to achieve sound synthesis, which meant fast computations of the sample theorem. Initially, the first synthesized sound was obtained in 1957, with the (assembly-code written) MUSIC 1 program with the IBM-704 (a predecessor of the IBM-7090). Later, when Bell Labs obtained the IBM-7094 —which "was a very, very effective machine" (C. Roads and M. Mathews 1980, p. 16)—, and in combination with the (then) widely available FORTRAN compiler, Mathews could develop the MUSIC I program, into MUSIC V, which became the first portable computer music language designed for computer music synthesis.[10] I will close this parentheis, now without returning to this discussion in a following section (See 3.1.4))

Xenakis' and Codd's papers came out around the same time: Xenakis' english publication of "Towards a Metamusic" was in 1970, Codd's papers were published in 1970 and 1972. While sieve theory was aimed at providing a plethora of computable sets (or relations) of pitches,

[9]Beyond a mention of "communication studies" by Mathews et al in New Jersey

[10]As an example, I would refer the reader to James Tenney's work from 1962 "Five Stochastic Studies," which can be found on his YouTube account: `https://www.youtube.com/channel/UCEzSaoPnxCJVzXxA9obuRWg/videos`. Roads, while interviewing Matthews recalls this piece to be named "Noise Studies" (C. Roads and M. Mathews 1980, p. 18), which fades out the reference to Xenakis' music.

according to different temperings of the smallest displacement unit and the selected value for the modulo operator, Codd's relational algebra was meant the internal structure of a query language for selecting elements based on their relations. Both of these can be considered algebraic abstractions of a selection process. In the case of Xenakis, the abstraction was one held outside-time. This meant that the composer could make a snapshot, or a tomography of the pitch space in order to analyze it, extrapolating structural relations. In Codd's case, the abstraction was spatial: the query language would be separated from the database itself, allowing a distance between a 'backend' and a 'frontend,' allowing databasers to perform queries without worrying about internal data structures, memory allocation, since these operations would occur in the background. Both methods came as an extension of freedom on the human operator: by black-boxing hardware-specific programming, the human operator could device any kind of algebraic queries, thus operating at a higher level of abstraction, enabling a less problematic kind of envisioning. Converseley, Xenakis writes:

> Freed from tedious calculations, the composer is able to devote himself to the general problems that the new musical form poses, and to explore the nooks and crannies of this form while *modifying the values of the input data...* (Xenakis 1992, p. 144) (Italics mine)

Therefore, the composer delegates to the computer the minutiae of iterative computations, that is, precisely what the computer is better at than the human. As a result, in Xenakis' view, the composer becomes a pilot:[11]

> With the aid of electronic computers the composer becomes a sort of pilot: he presses the buttons, introduces coordinates, and supervises the controls of *a cosmic vessel sailing in the space of sound*, across sonic constellations and galaxies that he could formerly glimpse only as a distant dream. Now he can explore them at his ease, seated in an armchair (ibid., p. 144) (Italics mine).

Codd's and Xenakis' propositions were abstractions deeply rooted in and contextualized

[11] Note here, too, the navigational metaphor that computers are generally linked to. The early 70s also gave rise to the Navigational model, by Charles Bachman, who also wrote the famous article "The Programmer as Navigator" (Bachman 1973) (See 1.2.3)

against a backdrop of their own fields. Xenakis wrote against the current state of Western Music with its “degradation of outside-time structures”, the “followers of information theory” and the “intuitionists.” Codd wrote against the previously developed Hierarchical and Network database models. Most important, these tools and their development had the human operator’s considerations in mind. The composer, like the databaser, would engage in a rudimentary and limited, but still present, feedback process at the *input* level. That is to say, unless rewriting the code, which consisted in a very long process combining punch cards and magnetic tapes, the composer and the databaser could change the input several times, achieving different outputs in a matter of hours.[12] For example, queries made on the relational model would appear on screen at a very fast rate, thus enabling better tuning of the input in relation to a wanted output. Likewise, the composer could modify the input values to highly complex calculations that would otherwise take a long time, or be error prone. The limitation, of course, is the level of intervention with the code itself, which the overall circuitry would itself complicate; criticism on account of this shortcoming of the circuit would thus be rendered anachronic.

3.1.4 The Database as Performer

I would like to take an improvisation detour that would make Xenakis fall off his armchair, but not as an architectural prank which would involve removing the armchair before he sits down, tired after having pressed all those buttons. Xenakis’ fall would be contemplated agains the spirit of the later discusisons on interaction that came out of George Lewis’ work[13], which would imply a paradigmatic shift in the activity of the composer in (networked) relation to the computer. That is to say, Xenakis’ metaphor of the computer as pilot, would be turned upside down, altogether reconfiguring the navigational metaphor: the ship begins to navigate the navigator, or simply, the

[12] As a reference, the computation of the first 100,000 values of π took about eight and a half hours (Shanks and W.jun. Wrench 1962).

[13] I am refering here to the music improvisation work/system called *Voyaguer* in the early 1990s, and to some of Lewis’ subsequent writings (Rowe et al. 1993; G. E. Lewis 1999; G. Lewis 2000).

computer turned databaser. Lewis called his approach “a improvisational, nonhierarchical, subject-subject model of discourse, rather than a stimulus/response setup” (G. E. Lewis 1999, p. 104).

It is now pertintent to bring back Mathews’ philosophy of the “computer as musical instrument” (M. V. Mathews 1963), particularly in relation to Lewis’ design of *Voyager*. At the end of his introduction of his MUSIC V, Mathews writes:

> So far I have described use of the computer solely as a musical instrument. The composer writes one line of parameters for each note he wishes played and hence has complete control of the note. He is omnipotent, except for lack of control over the noise produced by the random-number unit generators. *Here a minor liberty is allowed the computer* (ibid., p. 557) (Italics mine).[14]

Mathews describes the architecture of MUSIC V in three stages of data flow —put simply, as reading, sorting, and executing— which are modeled in turn from three elements of music tradition: the score, the metronome, and the instrument. Input data was interpreted by the computer and resulted in synthesis. From this, two consequences emerged. First, instructions to generate materials were dependent upon the capacities of the language (FORTRAN IV). Second, what the MUSIC-N languages brought forth was a closer relationship between the acoustical result of the generated material, and thus the composer and the computer were closely related by the (rather) rapidly computed audio signal coming out of the speaker. This architecture, however, is built on the concept of the computer as an instrument, which the composer performs by providing it score. Mathews goes further at the end of this article in 1963, and projects the computer itself as composer, mentioning earlier work by Hiller and Isaacson (1957) —considering it as an extreme in which “the computer can be ginven a set of rules, plus a random-number generator, and can simply be *turned on* to generater any amount of music” (ibid., p. 557)—, among a few other examples.

However, the limitations of computer capabilities precluded more complex conceptualizations of the type of interactivity between computer and composer. In fact, interactivity became

[14]This ‘minor liberty’ is also present in *Voyager*’s ‘embedded indeterminacy.’ In both of these cases, this degree of freedom is provided by the random-number generator, see **??**

a form of negotiation of the composer, one between interest, cost, and work; a very different negotiation than Lewis' sonic negotiation between computer and improvisor.[15] In *Voyager*, as the composer himself claims, "the computer system is not an instrument, and therefore cannot be controlled by a performer. Rather, the system is a multi-instrumental player with its own instrument" (G. E. Lewis 1999, p. 103). This means that the composer intentionally relinquishes control of the structure of the piece, to the system itself. He achieves this by means of a different paradigm of interaction: the computer stores features during the course of the performance, which are then averaged over time, and which serve as 'guides' for the sonic outcome on the part of the computer. As a result, the computer becomes an 'improvisation partner,' and by this, the complexity of the program itself increases exponentially.[16]

This notion of interactivity differs greatly from Xenakis' (modern) composer. He is sitting quietly in his armchair (in 1962) pressing buttons. By pressing them and inputting certain values, he controls the output, since he knows beforehand the internal mechanisms that are embedded in the software. This image of the modern composer in front of computer technology can also be found in, for example, Edgar Varèse: "The computing machine is a marvelous invention and seems almost superhuman. But, in reality, it is as limited as the mind of the individual who feeds it material" (Varese 2004, p. 20). Varèse's words, however, refer to the creative limit that a computer might have, which is always a function of the input (and, by extension, of material itself). Furthermore, in relation to electronic technology, he writes: "like the computer, the machines we use for making music can only give back what we put into them" (ibid., p. 20). Therefore, from

[15]"Complexity of the instrument-unit is paid for both in terms of [very costly] computer time and in terms of the number of parameters the composer must supply for each note. In general, the complicated instrument-units produce the most interesting sounds, and the composer must make his own compromise between interest, cost, and work" (M. V. Mathews 1963, p. 555). "There is no built-in hierarchy of human leader/computer follower, no 'veto' buttons, pedals, or cues. All communication between the system and the improvisor takes place sonically. A performance of Voyager is in a very real sense the result of a process of negotiation between the computer and the improvisor" (G. E. Lewis 1999, p. 104).

[16]This is why, in response to Lewis' criticism of the patching paradigm, Miller Puckette responds: "If you wish your computer to be more than just a musical instrument —if you want it to be an improvisation partner, for instance— you need a programming lanuage. One thing people in this situation might want to do is write external C procedure" (Rowe et al. 1993, p. 8)

these images of Varese-composer and Xenakis-composer, two axioms can be extrapolated: first, that composers do not loose control of the output; second, that the way to interact with computers is precisely telling them what and when to do it, so that the user is in total operative control. It is against these two axioms of computers and composition that Lewis' work in the late 1980s and 1990s can be contextualized. More precisely, it is because of the anachronic presence of the modern 'eurocentric' composer, and of its 'popularity' among computer music history, that Lewis brings into surface the question of interactivity. Placing into perspective by commenting on the social and cultural environment of computer music of the late 1980s, he writes:

> 'interaction' in computer music has moved from being considered the province of kooks and charlatans (I'm proud to have been one of those), to a position where composers now feel obliged to 'go interactive' in order to stay abreast of newer developments in the field (Rowe et al. 1993, p. 11).

However, the way in which interactivity was considered in the 'interactive' music made with was, for Lewis, determined by a fundamental feature of program —the 'trigger'—, which, in turn, was grounded on a more general programming concept: the conception of the patching window as a digital equivalent to the analog synthesizer's patching mechanism, where graphic cords are equivalent to cables, equating data flow with voltage flow. Nontheless, the trigger, or the `bang`, is a feature, not a bug, unless it is used as an extension of the stimulus/response paradigm of interactivity. In other words, subordinating music events to triggers by a human operator brings out a certain military metaphor, which Lewis calls "hear and obay" (ibid., p. 11). This metaphor can easily be extended to that of weaponry itself, and to the unfortunate naming of 'bang' method of objects, a method which (generally) triggers the object's core routine.[17] In order to address this shortcoming of interactivity, Lewis relates it to rudimentary mental processes, or as he puts it, to "amoeba- or roach-like automata" (ibid., p. 11). In this sense, not only interactivity itself

[17]However unfortunate this 'bang' name is, it makes one think back to the 1946 setting of the UNIVAC computer, in the military context of the Manhattan Project, for which the computer was used to get closer to the 'H' bomb. That is to say, even if 'bang' was named differently, the computer itself would be inevitably linked to this particularly *big* bang.

is at stake by the presence of a simple model of interaction. For Lewis, the crucial aspect of this model is the empowering of the image of the composer. This intentionally (very) simple automaton promotes two fundamentally hiearchic notions that Lewis attempts to deconstruct. On the one hand, the composer as controller who would never relinquish control of the music work, that is, the modern (eurological) image of the composer, and the old ghost train that comes with it: “The social, cultural, and gender isolation of the computer music fraternity (for that is what it is)” (Rowe et al. 1993, p. 11). This image leaves improvisation, together with non-eurological thinking out of the scope of contemporary music research. On the other hand, the human operator, as the higher, architectural mind that would not allow for the nonhuman, the computer object, the computer software, to become an operational agent beyond the instructions for which it was designed. In this sense, the simple-level automaton is a symbolic restrain representing the classical concept of the human itself, which allows a non-threatening relation between man and machine that can be considered functional, productive, and operative.

One is tempted to claim that the first of these images —the reified composer— is determined by the second —the reified human—, and that their relation is a matter of depth, or inheritance. Thus, in order to redefine the composer one would have to redefine the human first. In turn, this depth would be measured against that which is nonhuman, and by extension, that which is non-composer. In Lewis’ narrative, this entails the redefinition of composition itself by making the non-composer (e.g., what was eurologically considered the ‘improvisor’ or the ‘performer’) resound back into composition, regrouping the concept ‘composer’ itself, but not as a whole, since now the extent of its terms have found places within a networked system. This is precisely what he does in *Voyager*. The composer, like the human, became regrouped in hybridity —a hybridity that cannot be considered ‘on its own’, since it escapes any idea of ownness (or one-ness), and therefore, a hybridity that is expanded in networked resonance. It is in this sense that Lewis’ proposal is geared towards an interactive (computer) music *not entierly* driven by input.

> The composer therewith relinquishes some degree of low-level control over every single bloop and bleep in order to obtain more complex macrostructural behavior from the total musical system. The output of such entities might be influenced by input, but *not entirely* driven by it (Rowe et al. 1993, p. 11) (Italics mine).

It is precisely this 'not entirely' —i.e., a phrase that I would choose to better understand as the negation of wholeness— what begins to question the basis upon which our general concept of the human is built, and by extension, the agency of everything that falls outside of its definition. It is the beginning of a breakage, a crack on the foundation of Xenakis' (old) armchair, from which the state of suspension of the concept of the music work can be understood:

> With this in mind, it becomes easier to see that Voyager is *not really a 'work'* in the modernist sense —heroic, visionary, unique (Foster 1983). Rather, I choose to explore allegory and metatextuality, the programmatic, the depictive— and through embedded indeterminacy, the contingent. Ultimately, the subject of Voyager is not technology or computers at all, but musicality itself (G. E. Lewis 1999, p. 110) (Italics mine).

Furthermore, what this fracture reveals is hybrid nature of the notion of what is real and what is virtual. Understood traditionally, or better, understood under the stipulations of the first wave cyberneticians, the composer, being the real factor in the constitution of the (modern) image of the composer, is faced with the virtuality of the computer. Upon this encounter, the virtual comes as a form of threat to replace that which is real. In this sense, this is how I would like to approach Lewis' consideration of *Voyager* as "not really a work."[18] On the one hand, as Lewis claims, the goal of the interactivity between the composer and the computer is to allow the real and the virtual, "virtuality and physicality," to engage in the production of a hybrid that "strengthens on a human scale. Seen in this light, virtuality should enhance, not interfere, with communication between us" (ibid., p. 110). Therefore, considering the role of virtuality after new media integrated theories of embodiment (See 1.1.3), the computer reveals to the human —composer, improvisor, perofmormer— the very condition of its own virtuality, that is, virtuality itself within the human.

[18] I hope the reader would forgive me for having borrowed these adjectives out of context —'entirely' and 'really'— so as to allow my argument to echo with Lewis' for a while.

In the case of *Voyager*, this virtuality is sonic, it comes as the "emotional transduction" that Lewis aims for with this computer system. Therefore, it can't be 'really' a work, because it is virtuality itself resounding back. Another way to approach this is the fact that the computer can be said to be 'listening' to the performer, given that its real-time analysis is content-based, using techniques that have been applied to music information retrieval over the years (See 1.3.1). Understood as a listener, *Voyager* engages not only with signal processing at the lower level, it engages with the resonating process of the relation to self. Furthermore, the computer is not only listening, it is *databasing*, because it is keeping record of the listened features, and in so doing, it becomes empowered with the database itself. This database of actions, however, is the sonic trace of the performance itself, which is what is most surprising of its agency, and what resounds most in time. Therefore, far from being 'really a work', but also far from Lewis' notions of narrative in the sense of "allegory and metatextuality, the programmatic, the depictive" (G. E. Lewis 1999, p. 110), I consider *Voyager* an unwork of music, one that puts into question —though, to a certain extent— the operativity of the music work itself. To a certain extent, because the notion of productivity and cohesion are still present within Lewis' music and texts; but also, to the (paradoxical) extent that it is still a 'work,' a destiny that somehow manages to persist within the practice of composition. Nontheless, and without a doubt, Lewis' claim for the "non-eurocentric computer music" (ibid., p. 107) can be a starting point to the conceptualization of the unwork.

3.2 Inoperativity

How does the concepts of inoperativity and anarchy, in their relation to database community, resonate politically in the works of database music? …

3.2.1 The Severed Object of Music: Composing Composer

> [The] Heideggerian 'work of art' is able to present a unified picture that may be used for political purposes [it] is only what it is in the world that it open…Nancy is seeking a 'workless' or 'unworking' work, *a work that refuses to create itself as a total work*. Hence, Nancy proposes an artwork that would offer itself as a permanently open whole, the concept of art remaining undecided and lacking anything that might unify it (Gratton and M.-E. Morin 2015)

I would like to refer once again to Jean-Luc Nancy's concept of inoperativity (See 2.1.4), this time in relation to the music object. I argue that, given that the inoperativity of the listening experience reveals itself as the interaction between resonance —as the *différance* within sense and sensuality— and the unworking of the network, its resulting object, instead of being a complete whole —a finished, integral 'thing', or even, a 'piece'[19]—, it becomes a severed music object. This object is different from Pierre Schaeffer's music or sound object, which comes to represent material with which to work. Neither it is related to Vaggione's concept of object, which comes from object-oriented programming, meaning every composable primitive, from the micro to the macro. In both of the above, the object is used to provide, though not without their author's intervention, a notion of *coherence* to the work.

The object I am refering to resides in memory, as the remains of the event of an exposure. It is inherently linked to the fractured way in which our own memory works, and it is impossible to define, since it has no beginning and no end. Its dimensionality includes both beginning and ending simultaneously. This object is the spectral evidence of a musical event, or better, of the happening that takes place in listening. In being evidence, it becomes subject of analysis, it is forensic. In being fractured, it is the evidence of a destruction. In being severed, and this is the central aspect that I would like to focus on, risking simultaneously the severing of the object itself, it becomes the evidence of a sacrifice. If it can be said that the music object is a severed object, then the question of its severing necessarily relates to the question of listening. Therefore, by listening —and, by

[19]Since, the notion of a 'piece' presuposes that of the whole to which it belongs.

this, I mean entering in resonance with resonance itself, exposing the self to that which returns to itself— I participate in this severing, because, just as the many ways of listening that Schaeffer, Adorno, Szendy, Idhe, Attali, have refered to over the years, in listening I choose what to listen in spite of being already deprived from that choice.

The sounds onstage are always before and after the staging. The severed object of music is what, as listeners, we grab from the stage, what we choose to rip from the sounding waves, and also what we cannot help but feeling so much a part of us before noticing it is happening. Severing is yet another way of thinking the aesthetic experience of listening, but it is not as passive as it seems. Severing empowers the listener, it is the tool of listening, the reversed stilus, the inverted mouse, the part of the human that necessarily is nonhuman. With it, we can make the world appear, but only as a fraction of it, because 'it' can never be completely. The severed object of music is always severed, but never in the same way, since there are as many severings as there are listeners, and as many listenings as there are birds. In this difference, what is resonating is the object of music, which is never one and the same because it is a singularity that appears simultaneously to many. Composers have traditionally been considered a 'source' of this object, or better, the one at the door, the key keeper that has access to the door that opens up the flow of inspiration. The composer, but also the programmer with access to the source code, which unless it is open, is hidden to the rest; and, unless you know the language, it is complete pseudo-linguistic nonsense with weird punctuation marks, closer to poetry than it is to extreme formalism.

```
# Little words that do stuff

for there in is everything there is to know
do
        echo
```

In this access to the source, the programmer and the composer are traditionally kept at a distance, as if their listening were of some other sort, engaging with the very essence of the source, drinking the water from the originary fountain, satisfying an originary thirst. Therefore, if this is the role of the composer and the programer, if this is their relation to the source, then, they are the first to perform the severing. In the hierarchy of the consequent severings, they are at the top. Further, if they are the first severers, they are the first who perform the first listening. They are the listeners at the top of the mountain, next to the source of all fountains. On the way in and out of the world, the sourcerers of condensation.

I would like to point out now, that it is not my intention here to sever the head of the sourcerer, because it is an illusion that does not allow me to do so. It is not my illusion, although I have described how I interpret it, and it comes as a product of reification of the composer, but also of the human itself as the one and only owner of the world —that is, owner of the mountain itself, and of the water, and every particle of the one and only universe. Given that, in being in resonance, listeners become the resonating world, that is, the self begins to resonate as space, then this world is what is listened to, and it is a world that has no apparent origin. However, the composition —the written score, like the written code— propose their own origin —the composer, the programmer—, thus giving an origin to the world itself, providing an 'answer' to the question of creation: Who created this music? *this* composer. The answer, therefore, has a 'this' that comes in the form of the name of the composer, which is attached to the flowing of the source. Therefore, the name of the composer is like a timbre stamp that is applied to the listening experience itself, and further, it is the severing style itself that can be named. The name of the composer becomes a synechdoche of the source itself, directly naming part of the source. This applies, quite literally in some cases, to the name of the program and the name of the programmer.[20]

Furthermore, the activity of the sourcerer lends itself to its signature. In other words, the manner in which the composer defines the music, from beginning to end, becomes the shape of the music, understanding 'shape' or 'form' as something that is at once behind and in front

[20]'Max' is named after the 'father' of computer music Max Mathews, and MAX/MSP contains Miller Puckettes's initials. Friendly gestures, most probably, but also pointers to originary sources, sources of inspiration, historical references that contextualize computer music software within broader social and environmental structures.

of the singularity of the listened music. It is behind, because it is the activity of sound sources —speakers, musical instruments, or simply media in general—, the movement of air pressure. It is in front, because it filters the memory of the activity of sound sources. However, this composed shape and the singularity act together in the moment of listening. The question is, then, regarding the dynamics of this activity. Given that this activity happens during listening, what I addressing now is precisely how the shape of the music interacts with the listening itself. That is to say, the interaction between shape —but also the form, the idea— and the singularity of the listened. Interaction, here, refers to the shared activity that occurs 'inside' listening itself, and it happens 'inside' because of the severing that needed to occur prior —or immediately at— the resonating oscillation of air pressure. This is what I consider the moment of listening that is none other than listening to music. However, once this severing has occurred, and within its momentum, it is the internal dynamics that enter into play, and it is the shape of the music what begins to delineate the shape of the listened. Understood in this way, that is, the shape of the music as a force that produces a certain listening experience, therefore, the internal dynamics is already written. The singularity of the listened becomes (almost) one and the same with the shape of the music. 'Almost,' because it is not that the listened brings no resistance to this ideal force. The singularity of the listened is resistance itself, like I have mentioned before in relation to the trace (See **??**). It acts as resistance itself, and its force is not enough to resist the command of the excellent work. This is the very presence of the masterwork, at work, the work of a master that requires the slave —a slave that is not the rest of the works but the outshunned singularities that have been muted by its very own presence. 'Almost,' in the hope that its work can be relativized, disarticulated, disentangled from the source of sources, brought down the stream to the place where singularities can resonate in endless forms of matter. However, the problem is now of a different sort. Even if resisting forces match those of the masterwork, then, like Derrida's concept of a paralyisis of memory, we can encounter a paralysis of listening itself. This paralysis. This might (also) be what Szendy means, as well, by the cutting loose of the unnatentive listener in modernity, but in a different way. It is not a paralysis caused by distraction, it is a paralysis caused by the very force that is needed to match the force of the master work. It is a paralysis that is directly called for from outside —from the shape of the music itself—, one which prevents any further listening. This is what is called for by the work of the masterwork: pure —and utterly ideal— silence.

Therefore, within these dynamics of work, what results is a function of the predicates, it is the architecture of obedience that is written in the form of a music work, with the one an only aim which is for it to 'work.' Thus, the composer engaging with this dynamics of working out the work, of creating the structures, becomes the architect of the listened, the creator of a listening that of which he himself is the only chief. The sourcerer in charge of quenching a thirst that is only there because it is always already there, beforehand, instantiated with its own creation. The question now is how can this dynamics be approached once that I have recognized that it is there. How can composition continue, a composition that does not participate in this dynamics? A composition that is not a force? A composition that is not 'really' or 'entirely' a composition? A composition that does not impose its shape? A music work that is not a work but that still resonates within listening?

What characterizes the aesthetic dimension in the severed music object is, therefore, its inoperative quality. The practice of music composition can be understood, as I already outlined, in terms of Nancy's positive, active force of unworking. For him, the condition of unworking in relation to works of art is exposed by a certain resistance present in the work of art. This resistance is a force of interruption and suspension that prevents the notion of a whole to reach completion. Thus, the concept of a total work of art is consciously and inevitably lost. This differs from the notion of an open work, for example, Umberto Eco's famous formulation.[21] The case is quite different, since in Nancy's interpretation, the work never reaches completion, but encounters interruption and fragmentation, that is, it becomes suspended upon the limit of its exposure (See

[21] "The work of art is a complete and closed form in its uniqueness as a balanced organic whole, while at the same time constituting an open product on account of its susceptibility to countless different interpretations…" (Eco 2004)

2.1.4).

This unworking, for our musical purposes here, can only be carried out in the spatiotemporal dimension of perceivable waves. It is in this dimension only where the the listening performance can be comprehended. However, Where does that leave the performativity of composition? What is the role of the composer? By putting these mechanical waves together [*com-ponere*], the composer enters into an inoperative ritual. This constitutes a music ritual in which the composed 'unwork' enters a space that is the space of the listening subject, the space of the other. In this space there is no completion since the unworking of the work presents as such in an interrupted manner. The question is, then, How can the recognition of certain patterns in the spatiotemporal dimensionality of perceivable waves be accounted for as rules of a certain kind, or as style of a certain composer, or music period? How does identity emerge from this precondition of difference and interruption? This is how the *database* enters into the framing of the practice of composition that I attempt to draw attention to.

The severing of the music object comes out of repeated conscious and unconscious processes: attention —delay, deferral, filtering, limiting, blocking— of the overwhelming world of images coming into us in the form of waves, and memory —referral, recollecting, erasure, remembering. Both of these processes can be understood as the performativity of the listening experience. Therefore, in order to understand what a music object consists of, the concepts of *repetition* and of *listening* need to be explained. Repetition is fundamental to the listening experience: it is the essence of its performance. Butler considers repetition to be the always already of performance, because gender is a rehearsed activity, it is dramatic: "actors are always already on the stage within the terms of the performance" (Butler 1988)

There is, fundamentally, no gender outside performance, just as there is no self inside the gendered body. Gender questions essentialist notions of reality by precisely residing in time: it is fictional, its status is performative, it is an 'act,' and it is an act that fabricates its own identity for itself. It is important to note here the directionality of this act, which is crucial to determine the difference Butler emphasizes between 'expression' and 'performativeness.' While the former is based on the notion that there is a preexisting identity stemming from the inner self towards the outer, social layer, the latter proposes quite the opposite. The notion of gender understood as a pre-existing identity or 'roles' is rendered void and null. Instead, Butler writes, "as performance which is performative, gender is an 'act,' broadly construed, which constructs the social fiction of its own psychological interiority" (Butler 1988). This constitution of gender exposes the temporality of performance. For example, Butler writes about the temporality of gender: "[Gender] is an identity tenuously constituted in time —an identity instituted through a *stylized repetition of acts*." This temporality is what allows the notions of gender transformation, because the repetition that gender identity calls for is different and subversive. The style that is the result is singularity itself exposed as a social and historical event.

3.2.2 Anarchy and the Unwork

In this section, I analyze the anarchic element in database practice and bring it to music composition practice.

Defining anarchy as a paradoxically productive force —a form of destruction which "produces the very thing it reduces" (Derrida and Prenowitz 1995)—, Derrida locates it at the core of the concept of the archive (See 2.2.3). As I have outlined before, databasing brings together with its relation to the archive, the archontic principle that is bound to the origin and the rule. That is to say, since the database has the potential of becoming a source, databasing becomes an activity of this source, and thus embeds the databaser with a specter of authority. Therefore, given the circumstances of this authority of databasing, claiming that composition can be identified with databasing means translating the 'archic' not only to the performativity of composition, also to the product of composing, to the composer and the composed. I have mentioned above the presence of the skin of the database, now I shall refer to the skin of the music object.

I argue that the link between the archive, the database, and the music object is this capacity to prescribe its own origin —the commencement— and rules —the command. Finally, I analyze the extent to which this anarchic element is present in the inoperative object of music, and how this presence affects the unwork of art.

My goal in this reflection on the consequences of the anarchic and the inoperative in database and composition practices is to understand the dynamics of community within both database and composition fields.

My argument is that in order to understand what is in common between database and composition, from the points of view of art, aesthetics, and politics, we need to define the transversality of the underlying structures of anarchy and inoperativity.

...

3.3 Database Subject

How can the notion of database subject enter into de political dynamics of music composition? How are the traditional models of composition affected by thinking database music as resonance and power? What is the database subject? Where is it, and how does it sound?

3.3.1 [WIP] Work In Progress

What is all this fuss about the ‘unwork’? What is the ‘work’ with music work?
[3]

```
#include <stdio.h>
#include "m_pd.h"

t_class *working_class;

typedef struct working {
	t_symbol *a_work,
```

t_symbol ∗somethingDone,
t_float ∗physicalLabor, ∗skill
char ["a_product", "A_music_piece", "a_music_work"],
t_symbol ∗an_opera,
t_symbol ∗the_work_of_an_author,
t_symbol ∗the_oeuvre,
t_symbol ∗_the_operativity_of_the_composer,
t_symbol ∗matrix_operations,
t_symbol ∗operetta,
t_symbol ∗opera_prima,
t_symbol ∗obra,
t_symbol ∗open_work,
t_symbol ∗a_work_of_art,
t_symbol ∗artistic_creation,
t_symbol ∗techne,
t_float ∗fullTime, ∗partTime,
t_symbol ∗clockwork
t_symbol ∗officiate,
t_symbol ∗office,
t_symbol ∗act,
t_symbol ∗produce,
t_symbol ∗make_it_work,
t_symbol ∗magic_work,
t_symbol ∗work_of_angels,
t_symbol ∗blueCollar,
t_symbol ∗whiteCollar,
t_symbol ∗slavework,
t_symbol ∗masterwork,
t_symbol ∗Work_as_in_the_application_of_forces // "But applied to whom?"
t_symbol ∗working_a_field,
t_symbol ∗the_internal_workings_of_structures,
t_symbol ∗work_in_an_app,
t_symbol ∗worked_out,
t_symbol ∗work_your_hat_off,
t_symbol ∗workflow,
t_symbol ∗workspace
t_symbol ∗working_for_food,
t_symbol ∗hardworking,
t_symbol ∗labour,
t_symbol ∗giving_birth,
t_symbol ∗all_that_is_remunerated_after_efforts_have_been_given,
t_symbol ∗achieve_a_goal,
t_symbol ∗a_task,
t_symbol ∗to_work_to_live,
t_symbol ∗to_have_a_working_body,

```
        t_symbol ∗functioning,
        t_symbol ∗operative
        t_symbol ∗working_like_a_bee,
        t_symbol ∗like_a["bee","ant","member_of_the_hive","worker","co−worker"]
        t_symbol ∗working["for","to","after","by"]
        t_symbol ∗working_as_an_extension_of_truth_as_well_as_lies,
        t_symbol ∗out_of_work,
        t_symbol ∗at_work,
        t_symbol ∗work_in_progress,
        t_symbol ∗working_for_the_man,
        t_symbol ∗freelancing,
        t_symbol ∗working_under_the_table,
        t_symbol ∗working_past_a_deadline
        t_symbol ∗working_in_pairs,
        t_symbol ∗teamwork,
        t_symbol ∗collaborate,
        t_symbol ∗co−operate,
        t_atom  ∗organized_labour,
        t_symbol ∗paperwork,
        t_symbol ∗networking,
        t_symbol ∗prototyping
        t_symbol ∗worked−up,
        t_symbol ∗work_the_crowd,
        t_symbol ∗work_the_system,
        t_symbol ∗work_a_miracle,
        t_symbol ∗work_your_workers,
        t_symbol ∗social_worker,
        t_symbol ∗a_ship_works_in_a_heavy_sea,
        t_symbol ∗work_the_levers,
        t_symbol ∗work_for_Facebook,
        t_symbol ∗future_work,
        t_symbol ∗framework,
} t_working;
```

3.3.2 Redifining Collaboration: Trans-Inoperativity

By describing ‘community’ in such a way, Nancy’s goal is to define the subject, which for him is the ‘singularity’ which is thus suspended, interrupted, in order to make the space of community. In this section, therefore, I present my view on collaboration, and propose a redefinition of the term, in tune with other definitions with consider it a form of dialog between the human and the nonhuman (Dan07; Oli12). In its core, the word includes 'labor,' as in the 'working' in conjunction with others. Therefore, I engage with Nancy's critique on operativity and propose a different collaboration by removing its 'labor' connotation. A definition not based in terms of an active engagement of forces forming a whole, or of singular beings forming a whole which is greater than its parts. On the contrary, I claim the new form of collaboration can be termed better as a form of collective, or *trans-inoperation*, consisting in the mutual exposure of the limits of singular, performing beings into a state of suspension and withdrawal, which provides space for the 'unworking' to appear, and for the new configuration of the listening subject to reveal itself.

3.3.3 A Database Politics of Authorship

The author has already been a topic of great discussion in the twentieth-century (Benjamin), in relation to its function (Foucault, Barthes), to music composition (Bor95, Lew00) and, more recently, in relation to art and curating (Lov04; Dan07; Ves07; Gra10). The vastness of the literature relating to authorship would extend the limits of this dissertation. However, stemming from my conception of *trans-inoperativity*, in this section I claim that a new politics of authorship needs to be analyzed in terms of its spectrality. The figure of the author, as in the case of the subject of the gendered database, remains an illusion, a ghost, or a specter. As such, the spectrality of the author (Der95) is, however, what is empowering of its figure. Therefore, I question the effectiveness of the archontic principle that is present in *trans-inoperant* works of art. In my reconceptualization of collaboration, listening subjects are exposed together, causing the *archic* —of the archontic— to never archive itself, resulting in a constant state of suspended *an*-archic resonance. The specter of the author, thus, loses the sensuality and the sense of the listening subjects in state of trans-inoperance, and thus the power of the author ceases to act.

> The other important notion that animates Voyageris that of the locally intelligent orchestra, where local decisions taken by individual players percolate up to the global level where the overall form is maintained. (G. E. Lewis 1999, p. 110).

From the ILLIAC 1 to Voyager, I consider that the above delineation accounts for an adaptation on the part of the composer. The computer as such has changed, indeed, but its technical evolution may not be as significant to computing as the effect the computer has had in the field of music composition. I have suggested that composition is a single author practice, in which the composer's technique or aesthetic intuition is traditionally admired. This is no longer the case since the computer age: now the composer and the computer —and here I bring back the notion of the database— both share the weight of composition. Put differently, composition is imagined differently when computers are involved. At this point, I will begin articulating this imagination in the form of listening, and thus introduce sonification, synchresis, resonance and force as the last elements in this discussion of what I will understand as the imaginary composition.

Although without a direct mention of the Von Neumann model as such, L. A. Hiller and L. M. Isaacson (1955-7) composed the Illiac Suite with the ILLIAC 1 computer. In their account, they describe the computer's architecture, its capability for logical operations, and immediately proceed to its use. In particular, Hiller and Isaacson used the Markov Chain Monte Carlo method to obtain an "ordering process in which specified musical elements are selected and arranged from an infinite variety of possibilities, i.e., from chaos" (Hiller; Isaacson 1959:16). Musical material was generated by statistical methods and probability matrices: far from considering sound as such, they had to simplify. By way of indexing pitch and time they developed several "experiments" which led to a score for string quartet. Despite the underlying discussions of the system (tonality, twelve-tonality, etc.), or even the possible impositions by the heads (or the heads of the heads) of their institution (i.e., the exploration of the potential of the Monte Carlo Method), they had to punch data for a computer. What I am trying to suggest here, is that Hiller and Isaacson's compositional actions were shaped by the computer itself, and hence their music cannot be analyzed without the ILLIAC 1 itself, or without the Williams tubes in which their data shifted in and out from, or without the cards used for inputting the data in the first place. This relationship between the composer and the computer has gone under several changes and developments over the years,

> Many people familiar with the sound of wind chimes have noticed that they can tell the difference between the wind's performance and that of a person who is shaking it. Or a dog might be jingling a chain, making a sound very different from that a person might make with the same chain (G. E. Lewis 1999, p. 107).

Arbitrariness is what best describes so far the role of the composer. In a similar key, Curtis Roads, while acknowledging the attractiveness of so-called bottom-up systems —which will resurface farther down this text— and opposing them to preplanned forms (i.e., top-down forms, like the sonata), argues for the freedom for the "creative composer":

"The bottom-up strategy can be fascinating, partly because its results cannot always be predicted in advance. On the other hand, why limit the scope of compositional decisions to a single time scale?"

Roads, in resonance with Vaggione's multi-scalar approach to time, holds the figure of the composer in opposition to the global, statistical processes upon which Granular Synthesis (or Xenakis, or Hiller and Isaacson's work) is built. He continues:

"To navigate the widest possible zones of creativity, the creative composer wants to float freely across time scale boundaries" (italics mine, Roads 2004:330).

Conclusion

. . . placeholder for conclusion abstract . . .

And They Are Sounding Back

I am sounding in a database, and the database is sounding back . . .

Appendices

abstract of appendices

DIANA: Database for Image and Audio Navigation

I use William Brent's `timbreID` —timbre description algorithms— and Antoine Villeret's `pix_opencv` —image descriptors using Computer Vision algorithms—, to develop a new software library for Pure Data. My model consists of a joint Database structure for Image and Audio descriptors suitable for realtime navigation. At its core, the Database is generated by calculating derivatives between both data sets, and it is performed by applying random probabilities, markov chains, or chaotic generators to this navigation. This allows for multiple paths to be traced on each navigation.

A Database Model

A detailed description of the image and audio navigation system...

> Just as a fractal has the same structure on different scales, a new media object has the same modular structure throughout. Media elements... are represented as collections of discrete samples (Manovich 2001, p. 30).

> First, data is sampled, most often at regular intervals, such as the grid of pixels used to represent a digital image. The frequency of sampling is referred to as resolution.

> Sampling turns continuous data into discrete data. . . Second, each sample is quantified, that is, it is assigned a numerical value drawn from a defined range (such as 0-255 in the case of an 8-bit greyscale image) (Manovich 2002, p. 28)

I define the points in common between Database Practice and Music Composition. I describe the main technical concepts behind Database Navigation and provide use cases from both appendices A and B, the former relating to joint image and audio databases, and the latter to text databases. I then reflect on the quality of this navigation in relation to the type of navigation and results that they obtain.

I use computer vision literature to briefly introduce and describe the most common visual descriptors. I focus on certain descriptors (TBD) which are suitable for live multimedia use, and which I will implement in Appendix A.

I use Timbre Analysis literature to briefly introduce and describe the most useful audio descriptors. I take William Brent's TimbreID library, complementing it with Tae Hong Park's dissertation on timbre recognition, and I focus on the most useful descriptors for live multimedia use (TBD), which I will implement in Appendix A in relation to the image descriptors introduced above.

ABBY: An Online Environment for Annotated Bibliographies

In order to write this dissertation, I have developed "Abby" an online Text Database tool namely to build an annotated bibliography. The program is mostly written in Javascript, with the data navigation and programming hosted in Github, and the datasets stored in the Google account that New York University has provided me. The annotated bibliography is available at `https://fdch.github.io/abby`, and the code can be accessed or cloned from `https://github.com/fdch/litrev`.

A Text Database

A detailed description of the text database model…

Glossary

"I" Operating System a mobile operating system created and developed by Apple Inc. exclusively for its hardware. See also: `https://en.wikipedia.org/wiki/IOS`. 174

Adaptive Server Enterprise SAP ASE, originally known as Sybase SQL Server, and also commonly known as Sybase DB or Sybase ASE, is a relational model database server developed by Sybase Corporation, which later became part of SAP AG. ASE is predominantly used on the Unix platform, but is also available for Microsoft Windows. See also: `https://en.wikipedia.org/wiki/Adaptive_Server_Enterprise`. 174

Apache Lauded among the most successful influencers in Open Source, The Apache Software Foundation's commitment to collaborative development has long served as a model for producing consistently high quality software that advances the future of open development. See also: `https://www.apache.org/`. 174

Apache Software Foundation an American non-profit corporation to support Apache software projects, including the Apache HTTP Server. The ASF was formed from the Apache Group and incorporated on March 25, 1999. See also: `https://en.wikipedia.org/wiki/The_Apache_Software_Foundation`. 174

Application Programming Interface In computer programming, an application programming interface is a set of subroutine definitions, communication protocols, and tools for building software. See also: `https://en.wikipedia.org/wiki/Application_programming_interface`. 174

ArangoDB ArangoDB is a native multi-model database system developed by ArangoDB Inc. The database system supports three data models with one database core and a unified query language AQL. The query language is declarative and allows the combination of different data access patterns in a single query. ArangoDB is a NoSQL database system but AQL is similar in many ways to SQL. See also: `https://en.wikipedia.org/wiki/ArangoDB`. 174

Automatic Computer Assisted Design AutoCAD is a commercial computer-aided design and drafting software application. Developed and marketed by Autodesk, AutoCAD was first released in December 1982 as a desktop app running on microcomputers with internal graphics controllers. See also: `https://en.wikipedia.org/wiki/AutoCAD`. 174

Automatic Flow for Materials Discovery A globally available database of 2,118,033 material compounds with over 281,698,389 calculated properties See also: `http://aflowlib.org/`. 174

Bark Frequency Cepstrum Coefficient See also: . 174

Boost a set of libraries for the C++ programming language that provide support for tasks and structures such as linear algebra, pseudorandom number generation, multithreading, image processing, regular expressions, and unit testing. See also: `https://www.boost.org/`. 174

Cassandra The Apache Cassandra database is the right choice when you need scalability and high availability without compromising performance. Linear scalability and proven fault-tolerance on commodity hardware or cloud infrastructure make it the perfect platform for mission-critical data. Cassandra's support for replicating across multiple datacenters is best-in-class, providing lower latency for your users and the peace of mind of knowing that you can survive regional outages. See also: `http://cassandra.apache.org/`. 174

Center for Computer Research in Music and Acoustics a multi-disciplinary facility where composers and researchers work together using computer-based technology both as an artistic medium and as a research tool. See also: `https://ccrma.stanford.edu/`. 174

Center for New Music and Audio Technologies a multidisciplinary research center within University of California, Berkeley Department of Music. The Center's goal is to provide a common ground where music, cognitive science, computer science, and other disciplines meet to investigate, invent, and implement creative tools for composition, performance, and research. It was founded in the 1980s by composer Richard Felciano. See also: `https://en.wikipedia.org/wiki/Center_for_New_Music_and_Audio_Technologies`. 174

central processing unit the electronic circuitry within a computer that carries out the instructions of a computer program by performing the basic arithmetic, logic, controlling, and input/output (I/O) operations specified by the instructions. See also: `https://en.wikipedia.org/wiki/Central_processing_unit`. 174

Cluster Weighted Modeling an algorithm-based approach to non-linear prediction of outputs (dependent variables) from inputs (independent variables) based on density estimation using a set of models (clusters) that are each notionally appropriate in a sub-region of the input space. See also: `https://en.wikipedia.org/wiki/Cluster-weighted_modeling`. 174

Coastal Data Information Program an extensive network for monitoring waves and beaches along the coastlines of the United States. Since its inception in 1975, the program has produced a vast database of publicly-accessible environmental data for use by coastal engineers and planners, scientists, mariners, and marine enthusiasts. See also: https://cdip.ucsd.edu/. 174

Comma Separated Values a delimited text file that uses a comma to separate values. See also: https://en.wikipedia.org/wiki/Comma-separated_values. 174

Common Business Oriented Language A compiled English-like computer programming language designed for business use. See also: https://en.wikipedia.org/wiki/COBOL. 174

Composition Path A music making method with interactive map interface See also: https://sihwapark.com/COMPath. 174

Computer Aided Composition See also: . 174

Computer Aided Data Driven Composition See also: Nardelli 2015. 174

Computer Aided Design the use of computers (or workstations) to aid in the creation, modification, analysis, or optimization of a design. See also: https://en.wikipedia.org/wiki/Computer-aided_design. 174

Computer Aided Software Engineering the domain of software tools used to design and implement applications. CASE tools are similar to and were partly inspired by computer-aided design (CAD) tools used for designing hardware products. See also: https://en.wikipedia.org/wiki/Computer-aided_software_engineering. 174

Computer Assisted Music Project A general purpose composition and performance software environment originally based upon William Buxton's SSSP. See also: Free and Vytas 1988. 174

Computer Based Education Research Laboratory A research center based in the University of Illinois See also: https://en.wikipedia.org/wiki/PLATO_(computer_system). 174

Conference/Committee on Data Systems Languages a consortium formed in 1959 to guide the development of a standard programming language that could be used on many computers. This effort led to the development of the programming language COBOL and other technical standards. See also: https://en.wikipedia.org/wiki/CODASYL. 174

Content Based Unified Interfaces and Descriptors for Audio/music Databases available Online a new chain of applications through the use of audio/music content descriptors, in the spirit of the MPEG-7 standard See also: Vinet, Herrera, and Pachet 2002a; Vinet, Herrera, and Pachet 2002b. 174

Couch Data Base Apache CouchDB is open-source database software that focuses on ease of use and having a scalable architecture. See also: http://guide.couchdb.org. 174

Creative Commons an American non-profit organization devoted to expanding the range of creative works available for others to build upon legally and to share. See also: https://en.wikipedia.org/wiki/Creative_Commons. 174

Data Definition Language A data definition or data description language (DDL) is a syntax similar to a computer programming language for defining data structures, especially database schemas. See also: https://en.wikipedia.org/wiki/Data_definition_language. 174

data manipulation language A data manipulation language is a computer programming language used for adding, deleting, and modifying data in a database. A DML is often a sublanguage of a broader database language such as SQL, with the DML comprising some of the operators in the language See also: https://en.wikipedia.org/wiki/Data_manipulation_language. 174

Database Managemet System a computer program (or more typically, a suite of them) designed to manage a database, a large set of structured data, and run operations on the data requested by numerous users. Typical examples of DBMS use include accounting, human resources and customer support systems. See also: https://en.wikipedia.org/wiki/Category:Database_management_systems. 174

Digital Alternate Representation of Musical Scores The DARMS project started in 1963 by Stefan Bauer-Mengelberg and it is one of the first programming languages for music engraving See also: Brinkman 1983; Erickson 1975. 174

Disk Jockey a person who plays existing recorded music for a live audience. Most common types of DJs include radio DJ, club DJ who performs at a nightclub or music festival and turntablist who uses record players, usually turntables, to manipulate sounds on phonograph records. See also: https://en.wikipedia.org/wiki/Disc_jockey. 174

Document Object Model a cross-platform and language-independent application programming interface that treats an HTML, XHTML, or XML document as a tree structure wherein each node is an object representing a part of the document See also: https://en.wikipedia.org/wiki/Document_Object_Model. 174

Domain Specific Language a computer language specialized to a particular application domain. This is in contrast to a general-purpose language, which is broadly applicable across domains. See also: https://en.wikipedia.org/wiki/Domain-specific_language. 174

Electronic dance music a broad range of percussive electronic music genres made largely for nightclubs, raves and festivals See also: https://en.wikipedia.org/wiki/Electronic_dance_music. 174

Entity Relationship A database model that describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types and specifies relationships that can exist between entities. See also: `https://en.wikipedia.org/wiki/Entity-relationship_model`. 174

Exasol AG Exasol is an analytic database management software company. Its product is called Exasol, an in-memory, column-oriented, relational database management system. See also: `https://en.wikipedia.org/wiki/Exasol`. 174

Extensible Markup Language a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. The W3C's XML 1.0 Specification and several other related specifications-all of them free open standards-define XML See also: `https://en.wikipedia.org/wiki/XML`. 174

Formula Translation a general-purpose, compiled imperative programming language that is especially suited to numeric computation and scientific computing. See also: `https://en.wikipedia.org/wiki/Fortran`. 174

Frequency Modulation In telecommunications and signal processing, frequency modulation is the encoding of information in a carrier wave by varying the instantaneous frequency of the wave. See also: `https://en.wikipedia.org/wiki/Frequency_modulation`. 174

Galvanic Skin Response GSR is another name for Electrodermal activity (EDA) is the property of the human body that causes continuous variation in the electrical characteristics of the skin. See also: `https://en.wikipedia.org/wiki/Electrodermal_activity`. 174

generative theory of tonal music a theory of music conceived by American composer and music theorist Fred Lerdahl and American linguist Ray Jackendoff and presented in the 1983 book of the same title. See also: `https://en.wikipedia.org/wiki/Generative_theory_of_tonal_music` Hamanaka, Hirata, and Tojo 2014. 174

Gesture Description Interchange Format a format for storing, retrieving and sharing information about music-related gestures. See also: **Jensenius2006a**. 174

gigabyte a unit of information equal to one thousand million (109) or, strictly, 230 bytes. See also: `https://en.wikipedia.org/wiki/Gigabyte`. 174

Graphics Processing Unit a specialized electronic circuit designed to rapidly manipulate and alter memory to accelerate the creation of images in a frame buffer intended for output to a display device. See also: `https://en.wikipedia.org/wiki/Graphics_processing_unit`. 174

Hewlett Packard an American multinational information technology company headquartered in Palo Alto, California. See also: `https://en.wikipedia.org/wiki/Hewlett-Packard`. 174

High Performance Analytics Appliance an in-memory, column-oriented, relational database management system developed and marketed by SAP SE See also: `https://en.wikipedia.org/wiki/SAP_HANA`. 174

Human Computer Interaction A field that researches the design and use of computer technology, focused on the interfaces between people (users) and computers. See also: `https://en.wikipedia.org/wiki/Human-computer_interaction`. 174

Human Metabolome Database a comprehensive, high-quality, freely accessible, online database of small molecule metabolites found in the human body. Created by the Human Metabolome Project funded by Genome Canada. See also: `https://en.wikipedia.org/wiki/Human_Metabolome_Database`. 174

Hypertext Markup Language the standard markup language for creating web pages and web applications. With Cascading Style Sheets and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web See also: `https://en.wikipedia.org/wiki/HTML`. 174

Hypertext Transfer Protocol an application protocol for distributed, collaborative, hypermedia information systems. See also: `https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol`. 174

In Memory Data Bases a database management system that primarily relies on main memory for computer data storage. It is contrasted with database management systems that employ a disk storage mechanism. See also: `https://en.wikipedia.org/wiki/In-memory_database`. 174

Incorporated Research Institutions for Seismology IRIS is a consortium of over 120 US universities dedicated to the operation of science facilities for the acquisition, management, and distribution of seismological data. See also: `https://www.iris.edu`. 174

Information Management System the first database model ever created. It was created by IBM during the early 1960s, in conjunction with two other American manufacturing conglomerates (Rockwell and Caterpillar) for NASA's Project Apollo See also: `https://en.wikipedia.org/wiki/IBM_Information_Management_System`. 174

Information Retrieval the activity of obtaining information system resources relevant to an information need from a collection of information resources. Searches can be based on full-text or other content-based indexing. See also: `https://en.wikipedia.org/wiki/Information_retrieval` https://nlp.stanford.edu/IR-book/pdf/irbookonlinereading.pdf. 174

Institut de Recherche et Coordination Acoustique/Musique a French institute for science about music and sound and avant garde electro-acoustical art music. See also: `https://www.ircam.fr/`. 174

Institute of Electronic Music and Acoustics a multidisciplinary research center within the University of Music and Performing Arts, Graz, (Austria). See also: `https://en.wikipedia.org/wiki/Institute_of_Electronic_Music_and_Acoustics`. 174

Integrated Data Store an early network database management system largely used by industry, known for its high performance. IDS became the basis for the CODASYL Data Base Task Group standards. See also: `https://en.wikipedia.org/wiki/Integrated_Data_Store`. 174

Integrated Database Management System a network model database management system for mainframes See also: `https://en.wikipedia.org/wiki/IDMS`. 174

International Business Machines Corporation an American multinational information technology company headquartered in Armonk, New York, with operations in over 170 countries. See also: `https://en.wikipedia.org/wiki/IBM`. 174

International Computer Music Conference a yearly international conference for computer music researchers and composers. It is the annual conference of the International Computer Music Association (ICMA). See also: `https://en.wikipedia.org/wiki/International_Computer_Music_Conference`. 174

International Society for Music Information Retrieval an international forum for research on the organization of music-related data. See also: `http://ismir.net`. 174

Internet Movie Database See also: `https://www.imdb.com/`. 174

Internet Protocol the principal communications protocol in the Internet protocol suite for relaying datagrams across network boundaries. Its routing function enables internetworking, and essentially establishes the Internet. See also: `https://en.wikipedia.org/wiki/Internet_Protocol`. 174

Inverse Fourier Transform See also: `https://en.wikipedia.org/wiki/Fast_Fourier_transform`. 174

JavaScript Object Notation an open-standard file format that uses human-readable text to transmit data objects consisting of attribute-value pairs and array data types (or any other serializable value) See also: `https://www.json.org/`. 174

Linear Predictive Coding a tool used mostly in audio signal processing and speech processing for representing the spectral envelope of a digital signal of speech in compressed form, using the information of a linear predictive model See also: `https://en.wikipedia.org/wiki/Linear_predictive_coding`. 174

LISt Processor a family of computer programming languages with a long history and a distinctive, fully parenthesized prefix notation. Originally specified in 1958, Lisp is the second-oldest high-level programming language in widespread use today. Linked lists are one of Lisp's major data structures, and Lisp source code is made of lists. See also: `https://en.wikipedia.org/wiki/Lisp_(programming_language)`. 174

Local Boundaries Detection Model The Local Boundary Detection Model (LBDM) calculates boundary strength values for each interval of a melodic surface according to the strength of local discontinuities; peaks in the resulting sequence of boundary strengths are taken to be potential local boundaries. See also: **DBLP:conf/icmc/Cambouropoulos01**. 174

Log Frequency Cepstral Coefficients See also: `https://en.wikipedia.org/wiki/Mel-frequenc cepstrum`. 174

Massachusetts Institute of Technology private research university in Cambridge, Massachusetts. See also: `http://www.mit.edu/`. 174

Max (Mathews) + ("Max Signal Processing", or the initials Miller Smith Puckette) also known as Max/MSP/Jitter, is a visual programming language for music and multimedia developed and maintained by San Francisco-based software company Cycling '74 See also: `https://en.wikipedia.org/wiki/Max_(software)`. 174

Max programming language See also: . 174

MetriX in Extensible Markup Language A computer music synthesis programming language based in the Music-N type. See also: `https://xamat.github.io/Thesis/html-thesis/` Amatriain 2004. 174

Microsoft Access Microsoft Access is a database management system from Microsoft that combines the relational Microsoft Jet Database Engine with a graphical user interface and software-development tools. See also: `https://products.office.com/en/access`. 174

Music information retrieval the interdisciplinary science of retrieving information from music. MIR is a small but growing field of research with many real-world applications. Those involved in MIR may have a background in musicology, psychoacoustics, psychology, academic music study, signal processing, informatics, machine learning, optical music recognition, computational intelligence or some combination of these. See also: `https://en.wikipedia.org/wiki/Music_information_retrieval`. 174

Musical Instrument Digital Interface a technical standard that describes a communications protocol, digital interface, and electrical connectors that connect a wide variety of electronic musical instruments, computers, and related audio devices. See also: `https://en.wikipedia.org/wiki/MIDI`. 174

National Aeronautics and Space Administration an independent agency of the United States Federal Government responsible for the civilian space program, as well as aeronautics and aerospace research.NASA was established in 1958, succeeding the National Advisory Committee for Aeronautics (NACA). See also: `https://www.nasa.gov/`. 174

New Interfaces for Musical Expression an international conference dedicated to scientific research on the development of new technologies and their role in musical expression and artistic performance. Researchers and musicians from all over the world gather to share their knowledge and late-breaking work on new musical interface design. See also: `https://en.wikipedia.org/wiki/New_Interfaces_for_Musical_Expression`. 174

New York University a private research university spread throughout the world. See also: . 174

Non or Not only SQL a mechanism for storage and retrieval of data that is modeled in means other than the tabular relations used in relational databases. See also: `https://en.wikipedia.org/wiki/NoSQL`. 174

Nuclear magnetic resonance a physical phenomenon in which nuclei in a strong static magnetic field are perturbed by a weak oscillating magnetic field (in the near field and therefore not involving electromagnetic waves) and respond by producing an electromagnetic signal with a frequency characteristic of the magnetic field at the nucleus. See also: `https://en.wikipedia.org/wiki/Nuclear_magnetic_resonance`. 174

Open Sound Control a protocol for networking sound synthesizers, computers, and other multimedia devices for purposes such as musical performance or show control. OSC's advantages include interoperability, accuracy, flexibility and enhanced organization and documentation. See also: `https://en.wikipedia.org/wiki/Open_Sound_Control`. 174

Oracle Corporation an American multinational computer technology corporation headquartered in Redwood Shores, California. See also: `https://en.wikipedia.org/wiki/Oracle_Corporation`. 174

Phase Vocoder a collection of phase vocoder signal processing routines and accompanying shell scripts for use in the transformation and manipulation of sounds. It is written in C and designed to be used in a UNIX environment. See also: `http://www.cs.princeton.edu/courses/archive/spr99/cs325/koonce.html`. 174

Quantuum Electrodynamics In particle physics, quantum electrodynamics is the relativistic quantum field theory of electrodynamics See also: `https://en.wikipedia.org/wiki/Quantum_electrodynamics`. 174

Raima Database Manager an ACID-compliant embedded database management system designed for use in embedded systems applications. RDM has been designed to utilize multi-core computers, networking (local or wide area), and on-disk or in-memory storage management. See also: `https://en.wikipedia.org/wiki/Raima_Database_Manager`. 174

Random Access Memory a form of computer data storage that stores data and machine code currently being used. A random-access memory device allows data items to be read or written in almost the same amount of time irrespective of the physical location of data inside the memory. See also: `https://en.wikipedia.org/wiki/Random-access_memory`. 174

Real World Computing Music Database a copyright-cleared music database (DB) that is available to researchers as a common foundation for research. It was built by the RWC Music Database Sub-Working Group of the Real World Computing Partnership (RWCP) of Japan See also: `https://staff.aist.go.jp/m.goto/RWC-MDB/`. 174

Realtime Cmix a real-time software "language" for doing digital sound synthesis and signal-processing. It is written in C/C++, and is distributed open-source, free of charge. See also: `http://rtcmix.org/`. 174

Redis Redis is an open source (BSD licensed), in-memory data structure store, used as a database, cache and message broker See also: `https://redis.io`. 174

regions of interest samples within a data set identified for a particular purpose. For example, in medical imaging, the boundaries of a tumor may be defined on an image or in a volume, for the purpose of measuring its size. See also: `https://en.wikipedia.org/wiki/Region_of_interest`. 174

Relational Model/Tasmania Extensions to the Relational Model See also: `https://en.wikipedia.org/wiki/Relational_Model/Tasmania`. 174

Sedna Sedna is a free native XML database which provides a full range of core database services - persistent storage, ACID transactions, security, indices, hot backup. Flexible XML processing facilities include W3C XQuery implementation, tight integration of XQuery with full-text search facilities and a node-level update language. See also: `https://www.sedna.org/`. 174

Short Time Fourier Transform a Fourier-related transform used to determine the sinusoidal frequency and phase content of local sections of a signal as it changes over time. See also: `https://en.wikipedia.org/wiki/Short-time_Fourier_transform`. 174

Sinusoidal Partial Editing Analysis and Resynthesis an application for audio analysis, editing and synthesis. The analysis procedure (which is based on the traditional McAulay-Quatieri technique) attempts to represent a sound with many individual sinusoidal tracks (partials), each corresponding to a single sinusoidal wave with time varying frequency and amplitude. See also: `http://www.klingbeil.com/spear/`. 174

Sound Description Interchange Format a standard for the well-defined and extensible interchange of a variety of sound descriptions. See also: `https://en.wikipedia.org/wiki/SDIF`. 174

Sparcity High-performance human solutions for Extreme Data See also: http://www.sparsity-technol com/. 174

Sparksee Sparksee is a high-performance and scalable graph database management system written in C++. Its development started in 2006 and its first version was available on Q3 - 2008. The fourth version is available since Q3-2010. See also: http://www.sparsity-technologies.com/#sparksee. 174

Speech Transformation and Representation using Adaptive Interpolation of weiGHTed spectrogram a high-quality vocoder designed for speech analysis, modification, and synthesis See also: Katayose and Kawahara 1999. 174

Structured Query Language a domain-specific language used in programming and designed for managing data held in a relational database management system, or for stream processing in a relational data stream management system. See also: https://en.wikipedia.org/wiki/SQL. 174

Structured Query Language Lite a relational database management system contained in a C programming library. In contrast to many other database management systems, SQLite is not a client-server database engine. Rather, it is embedded into the end program. See also: https://en.wikipedia.org/wiki/SQLite. 174

Structured Sound Synthesis Project an interdisciplinary project whose aim is to conduct research into problems and benefits arising from the use of computers in musical composition See also: https://www.billbuxton.com/SSSP.html. 174

Synchronized Multimedia Integration Language a World Wide Web Consortium recommended Extensible Markup Language markup language to describe multimedia presentations. It defines markup for timing, layout, animations, visual transitions, and media embedding, among other things. See also: https://en.wikipedia.org/wiki/Synchronized_Multimedia_Integration_Language. 174

Timbre Identification a collection of audio feature analysis externals for [Pd]. The classification extern (timbreID) accepts arbitrary lists of features and attempts to find the best match between an input feature and previously stored instances of training data. Besides doing identification, timbreID is also designed to facilitate real time concatenative synthesis and timbre-based orderings of sound sets. Its usage is fully explained in the accompanying helpfile. See also: http://williambrent.conflations.com/pages/research.html Brent 2010a. 174

Transmission Control Protocol / Internet Protocol The Internet protocol suite is the conceptual model and set of communications protocols used in the Internet and similar computer networks. It is commonly known as TCP/IP because the foundational protocols in the suite are the Transmission Control Protocol (TCP) and the Internet Protocol (IP). See also: https://en.wikipedia.org/wiki/Internet_protocol_suite. 174

Uniform Resource Identifier a string of characters that unambiguously identifies a particular resource. To guarantee uniformity, all URIs follow a predefined set of syntax rules, but also maintain extensibility through a separately defined hierarchical naming scheme. See also: `https://en.wikipedia.org/wiki/Uniform_Resource_Identifier`. 174

Uniform Resource Locator colloquially termed a web address, is a reference to a web resource that specifies its location on a computer network and a mechanism for retrieving it. A URL is a specific type of although many people use the two terms interchangeably. See also: `https://en.wikipedia.org/wiki/URL`. 174

Unisys OS 2200 databases The OS 2200 database managers are all part of the Universal Data System (UDS). UDS provides a common control structure for multiple different data models. Flat files (sequential, multi-keyed indexed sequential - MSAM, and fixed-block), network (DMS), and relational (RDMS) data models all share a common locking, recovery, and clustering mechanism. OS 2200 applications can use any mixtures of these data models along with the high-volume transaction file system within the same program while retaining a single common recovery mechanism. See also: `https://en.wikipedia.org/wiki/Unisys_OS_2200_databases`. 174

United Information Systems Unisys Corporation is an American global information technology company based in Blue Bell, Pennsylvania, that provides a portfolio of IT services, software, and technology. It is the legacy proprietor of the Burroughs and UNIVAC line of computers, formed when the former bought the latter. See also: `https://www.unisys.com`. 174

Videoalive Indexer Extracted Closed Captions and Metadata This is an index file in text form that shows closed captions and file offset information from a MPEG or WM video file. The .IXD file is used by VBrick Systems, Videoalive, Discovervideo.com, and other vendors. See also: `https://filext.com/file-extension/IXD`. 174

YAML Ain't Markup Language a human-readable data serialization language. It is commonly used for configuration files, but could be used in many applications where data is being stored or transmitted. See also: `https://en.wikipedia.org/wiki/YAML`. 174

Acronyms

Access Microsoft Access. 174

AFLOWLIB Automatic Flow for Materials Discovery. 174

Apache Apache. 174

API Application Programming Interface. 174

ArangoDB ArangoDB. 174

ASE Adaptive Server Enterprise. 174

ASF Apache Software Foundation. 174

AutoCAD Automatic Computer Assisted Design. 174

BFCC Bark Frequency Cepstrum Coefficient. 174

Boost Boost. 174

CAC Computer Aided Composition. 174

CAD Computer Aided Design. 174

CADDC Computer Aided Data Driven Composition. 174

CAMP Computer Assisted Music Project. 174

CASE Computer Aided Software Engineering. 174

Cassandra Cassandra. 174

CC Creative Commons. 174

CCRMA Center for Computer Research in Music and Acoustics. 174

CDIP Coastal Data Information Program. 174

CERL Computer Based Education Research Laboratory. 174

Cmix . 174

CNMAT Center for New Music and Audio Technologies. 174

COBOL Common Business Oriented Language. 174

CODASYL Conference/Committee on Data Systems Languages. 174

COMPath Composition Path. 174

CouchDB Couch Data Base. 174

CPU central processing unit. 174

CSV Comma Separated Values. 174

CUIDADO Content Based Unified Interfaces and Descriptors for Audio/music Databases available Online. 174

CWM Cluster Weighted Modeling. 174

DARMS Digital Alternate Representation of Musical Scores. 174

DBMS Database Managemet System. 174

DDL Data Definition Language. 174

DJ Disk Jockey. 174

DML data manipulation language. 174

DOM Document Object Model. 174

DSL Domain Specific Language. 174

EDM Electronic dance music. 174

ER Entity Relationship. 174

Exasol AG Exasol AG. 174

FM Frequency Modulation. 174

FORMES . 174

FORTRAN Formula Translation. 174

GB gigabyte. 174

GDIF Gesture Description Interchange Format. 174

GPU Graphics Processing Unit. 174

GSR Galvanic Skin Response. 174

GTTM generative theory of tonal music. 174

HCI Human Computer Interaction. 174

HMDB Human Metabolome Database. 174

HP Hewlett Packard. 174

HTML Hypertext Markup Language. 174

HTTP Hypertext Transfer Protocol. 174

IBM International Business Machines Corporation. 174

ICloud . 174

ICMC International Computer Music Conference. 174

IDMS Integrated Database Management System. 174

IDS Integrated Data Store. 174

IEM Institute of Electronic Music and Acoustics. 174

IFT Inverse Fourier Transform. 174

IMDB Internet Movie Database. 174

IMDBs In Memory Data Bases. 174

IMS Information Management System. 174

iOS "I" Operating System. 174

IP Internet Protocol. 174

IR Information Retrieval. 174

IRCAM Institut de Recherche et Coordination Acoustique/Musique. 174

IRIS Incorporated Research Institutions for Seismology. 174

ISMIR International Society for Music Information Retrieval. 174

IXD Videoalive Indexer Extracted Closed Captions and Metadata. 174

JSON JavaScript Object Notation. 174

LBDM Local Boundaries Detection Model. 174

LFCC Log Frequency Cepstral Coefficients. 174

LISP LISt Processor. 174

LPC Linear Predictive Coding. 174

madBPM . 174

Max Max programming language. 174

MAX/MSP Max (Mathews) + ("Max Signal Processing", or the initials Miller Smith Puckette). 174

MetriXML MetriX in Extensible Markup Language. 174

MIDI Musical Instrument Digital Interface. 174

MIR Music information retrieval. 174

MIT Massachusetts Institute of Technology. 174

MongoDB . 174

MUSIC-11 . 174

MUSIC-IV . 174

MUSIC-N . 174

MySQL . 174

NASA National Aeronautics and Space Administration. 174

Neo4j . 174

net.loadbang-SQL . 174

NetworkX . 174

NeXT . 174

NIME New Interfaces for Musical Expression. 174

NMR Nuclear magnetic resonance. 174

NoSQL Non or Not only SQL. 174

NYU New York University. 174

Oracle Oracle Corporation. 174

OS 2200 Unisys OS 2200 databases. 174

OSC Open Sound Control. 174

PostgreSQL . 174

PVC Phase Vocoder. 174

QCD-Audio . 174

QED Quantuum Electrodynamics. 174

RAM Random Access Memory. 174

RDM Raima Database Manager. 174

Redis Redis. 174

RM/T Relational Model/Tasmania. 174

ROI regions of interest. 174

RTcmix Realtime Cmix. 174

RWC Real World Computing Music Database. 174

SAP HANA High Performance Analytics Appliance. 174

SCORE . 174

SCRIVA . 174

SDIF Sound Description Interchange Format. 174

Sedna Sedna. 174

SMIL Synchronized Multimedia Integration Language. 174

SonART . 174

Sparcity Sparcity. 174

Sparksee Sparksee. 174

SPEAR Sinusoidal Partial Editing Analysis and Resynthesis. 174

SQL Structured Query Language. 174

SQLite Structured Query Language Lite. 174

SSSP Structured Sound Synthesis Project. 174

STFT Short Time Fourier Transform. 174

STRAIGHT Speech Transformation and Representation using Adaptive Interpolation of weiGHTed spectrogram. 174

Sybase . 174

TCP/IP Transmission Control Protocol / Internet Protocol. 174

timbreID Timbre Identification. 174

TimesTen . 174

TurboIMAGE . 174

Unisys United Information Systems. 174

URI Uniform Resource Identifier. 174

URL Uniform Resource Locator. 174

VMWare . 174

WebDNA . 174

XML Extensible Markup Language. 174

XPlain . 174

YAML YAML Ain't Markup Language. 174

Bibliography

Abiteboul, S. (1996). *Querying Semi-Structured Data*. Technical Report 1996-19. Stanford InfoLab. URL: http://ilpubs.stanford.edu:8090/144/.

Abiteboul, Serge, Richard Hull, and Victor Vianu (1995). *Foundations of Databases*. Addison-Wesley.

Amatriain, Xavier (2004). "An Object-Oriented Metamodel for Digital Signal Processing with a focus on Audio and Music". PhD thesis. Universitat Pompeu Fabra. URL: https://xamat.github.io//Thesis/html-thesis/node51.html.

Ames, Charles (1985). "Applications of Linked Data Structures to Automated Composition". In: *Proceedings of the International Computer Music Conference, ICMC 1985*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.1985.040/1.

Angles, Renzo and Claudio Gutierrez (Feb. 2008). "Survey of Graph Database Models". In: *ACM Computing Surveys* 40.1.

Antila, Christopher and Julie Cumming (2014). "The VIS Framework: Analyzing Counterpoint in Large Datasets". In: *Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR 2014, Taipei, Taiwan, October 27-31, 2014*. Ed. by Hsin-Min Wang, Yi-Hsuan Yang, and Jin Ha Lee, pp. 71–76. URL: http://www.terasoft.com.tw/conf/ismir2014/proceedings/T014_162_Paper.pdf.

Ariza, Christopher (2003). "Ornament as Data Structure: An Algorithmic Model based on Micro-Rhythms of Csng Laments and Funeral Music". In: *Proceedings of the International Computer Music Conference, ICMC 2003*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2003.030/1.

Ariza, Christopher (2005). "The Xenakis Sieve as Object: A New Model and a Complete Implementation". In: *Computer Music Journal* 29.2, pp. 40–60. URL: https://www.jstor.org/stable/3681712.

Assayag, Gérard et al. (1997). "An Object Oriented Visual Environment For Musical Composition". In: *Proceedings of the 1997 International Computer Music Conference, ICMC 1997, Thessaloniki, Greece, September 25-30, 1997*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1997.097.

Attali, Jacques (2009). *Noise: The Political Economy of Music*. University ofMinnesota Press.

Bachman, Charles W. (Nov. 1973). “The Programmer As Navigator”. In: *Commun. ACM* 16.11, pp. 653–658. ISSN: 0001-0782. DOI: 10.1145/355611.362534. URL: http://doi.acm.org/10.1145/355611.362534.

Ballora, Mark (2000). “Data Analysis through Auditory Display: Applications in Heart Rate Variability”. PhD thesis. McGill University. URL: http://www.markballora.com/publications/diss.pdf.

Ballora, Mark et al. (2010). “Sonification Of Web Log Data”. In: *Proceedings of the International Computer Music Conference, ICMC 2010*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2010.117/1.

Barrett, Natasha (2000). “A compositional methodology based on data extracted from natural phenomena”. In: *Proceedings of the International Computer Music Conference, ICMC 2000*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2000.123/1.

Bekke, Johannes Hendrikus ter (June 4, 1991). “Semantic data modeling in relational environments”. PhD thesis. Technische Universiteit Delft. ISBN: 90-900-4132-X. URL: http://resolver.tudelft.nl/uuid:1bacdf29-389a-422c-9b62-95ad93de1158.

Bello, Juan Pablo, Elaine Chew, and Douglas Turnbull, eds. (2008). *ISMIR 2008, 9th International Conference on Music Information Retrieval, Drexel University, Philadelphia, PA, USA, September 14-18, 2008*. ISBN: 978-0-615-24849-3.

Ben-Tal, O. et al. (2002). “Sonart: The sonification application research toolbox”. In: *Presented at the 8th International Conference on Auditory Display (ICAD), Kyoto, Japan, July 2-5, 2002*. Georgia Institute of Technology. URL: http://hdl.handle.net/1853/51376.

Bertin-Mahieux, Thierry et al. (2011). “The Million Song Dataset”. In: *Proceedings of the 12th International Society for Music Information Retrieval Conference, ISMIR 2011, Miami, Florida, USA, October 24-28, 2011*. Ed. by Anssi Klapuri and Colby Leider. University of Miami, pp. 591–596. ISBN: 978-0-615-54865-4. URL: http://ismir2011.ismir.net/papers/OS6-1.pdf.

Bittner, Rachel M. et al. (2014). “MedleyDB: A Multitrack Dataset for Annotation-Intensive MIR Research”. In: *Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR 2014, Taipei, Taiwan, October 27-31, 2014*. Ed. by Hsin-Min Wang, Yi-Hsuan Yang, and Jin Ha Lee, pp. 155–160. URL: http://www.terasoft.com.tw/conf/ismir2014/proceedings/T028_322_Paper.pdf.

Borges, Jorge Luis (1942). “Funes el memorioso”. In: *Ficciones*.

Born, Georgina (1995). *Rationalizing Culture*. University of California Press.

Bortz, Brennon, Javier Jaimovich, and R. Benjamin Knapp (May 2015). “Emotion in Motion: A Reimagined Framework for Biomusical/Emotional Interaction”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by Edgar Berdahl and Jesse Allison. Baton Rouge, Louisiana, USA: Louisiana State University, pp. 44–49. URL: http://www.nime.org/proceedings/2015/nime2015_291.pdf.

Boynton, Lee et al. (1986). “Adding a Graphical User Interface to FORMES”. In: *Proceedings of the 1986 International Computer Music Conference, ICMC 1986, Den Haag, The Netherlands,* *October 20-24, 1986*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1986.025.

Brent, William (2010a). "A Timbre Analysis And Classification Toolkit For Pure Data". In: *Proceedings of the International Computer Music Conference, ICMC 2010*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2010.044/1.

Brent, William (2010b). "Physical and perceptual aspects of percussive timbre". PhD thesis. UC San Diego. URL: https://escholarship.org/uc/item/5bx4j1fj.

Bresson, Jean and Carlos Agon (2004). "SDIF sound description data representation and manipulation in computer assisted composition". In: *Proceedings of the International Computer Music Conference, ICMC 2004*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2004.004/1.

Bresson, Jean and Carlos Agon (2010). "Processing Sound And Music Description Data Using Openmusic". In: *Proceedings of the International Computer Music Conference, ICMC 2010*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2010.129/1.

Brinkman, Alexander R. (1981). "Data Structures for a Music-11 Preprocessor". In: *Proceedings of the International Computer Music Conference, ICMC 1981*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.1981.018/1.

Brinkman, Alexander R. (1982). "Original version of the Score11 Manual". In: *Score11 Manual*. Copyright <c> 1982 by Alexander R. Brinkman. URL: http://ecmc.rochester.edu/ecmc/docs/score11/index.html#Introduction.

Brinkman, Alexander R. (1983). "A Design for a Single Pass Scanner for the DARMS Music Coding Language". In: *Proceedings of the International Computer Music Conference, ICMC 1980*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.1983.002/1.

Brinkman, Alexander R. (1984). "A Data Structure for Computer Analysis of Musical Scores". In: *Proceedings of the International Computer Music Conference, ICMC 1984*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.1984.033/1.

Bullock, Jamie, Daniel Beattie, and Jerome Turner (2011). "Integra Live : a New Graphical User Interface for Live Electronic Music". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Oslo, Norway, pp. 387–392. URL: http://www.nime.org/proceedings/2011/nime2011_387.pdf.

Bullock, Jamie and Lamberto Coccioli (2009). "Towards a Humane Graphical User Interface for Live Electronic Music". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Pittsburgh, PA, United States, pp. 266–267. URL: http://www.nime.org/proceedings/2009/nime2009_266.pdf.

Bullock, Jamie and Henrik Frisk (2009). "An Object Oriented Model for the Representation of Temporal Data in the Integra Framework". In: *Proceedings of the International Computer Music Conference, ICMC 2009*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2009.012/1.

Buneman, Peter (1997). "Semistructured Data". In: *Proceedings of the Sixteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*. PODS '97. Tucson, Arizona, USA: ACM, pp. 117–121. ISBN: 0-89791-910-6. DOI: 10.1145/263661.263675. URL: http://doi.acm.org/10.1145/263661.263675.

Butler, Judith (1988). “Performative Acts and Gender Constitution: An Essay in Phenomenology and Feminist Theory”. In: *Theatre Journal* 40.4.

Buxton, William (2016a). “Objed: The SSSP Sound Editing Tool”. In: *Youtube*. URL: https://www.youtube.com/watch?v=pUoHc_2wUjY.

Buxton, William (2016b). “Socializing Technology for the Mobile Human. Keynote, The Next Web Conference, Amsterdam/Europe.” In: *Youtube*. URL: https://youtu.be/rEeEofRShAQ.

Buxton, William, Guy Fedorkow, et al. (1978). “An Overview of the Structured Sound Synthesis Project”. In: *Proceedings of the 1978 International Computer Music Conference, ICMC 1978, Evanston, Illinois, USA, 1978*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1978.031.

Buxton, William, Sanand Patel, et al. (1980). “"OBJED" and the Design of Timbral Resources”. In: *Proceedings of the 1980 International Computer Music Conference, ICMC 1980, New York City, USA, 1980*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1980.006.

Buxton, William, William Reeves, et al. (1978). “The Use of Hierarchy and Instance in a Data Structure for Computer Music”. In: *Proceedings of the International Computer Music Conference, ICMC 1978*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.1978.012/1.

Cádiz, Rodrigo F. et al. (2015). “Sonification of Medical Images Based on Statistical Descriptors”. In: *ICMC*. Michigan Publishing.

Camara Halac, Federico (2018a). “‘This is for young ears:’ A Response to Elsa Justel’s Marelle...” In: *Open Space* 21, pp. 339–350.

Camara Halac, Federico (2018b). “A Spectral Experience: Self Convolution And Face Tracking”. Accepted at ICMC 2018 and SEAMUS 2019 conferences.

Camara Halac, Federico, Matias Delgadino, and Lucia Simonelli (Oct. 2017). *Hearing The Self: A Spectral Experience*. For 2 PS3 Eyecams, multichannel audio, screens and participants. 1440 minutes. Xuhui Art Museum, Shanghai, China: International Computer Music Conference.

Caramiaux, Baptiste, Frédéric Bevilacqua, and Norbert Schnell (2011). “Sound Selection by Gestures”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Oslo, Norway, pp. 329–330. URL: http://www.nime.org/proceedings/2011/nime2011_329.pdf.

Caraty, M. J., J. C. Richard, and Xavier Rodet (1989). “"Vowel recognition in a data base of continuous speech: experiments with local and global identification principles"”. In: *EUROSPEECH*, p. 2272.

Carey, Benjamin (2012). “Designing for Cumulative Interactivity: The $derivationsSystem$”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ann Arbor, Michigan: University of Michigan. URL: http://www.nime.org/proceedings/2012/nime2012_292.pdf.

Carlile, Simon (2011). “Psychoacoustics”. In: *The Sonification Handbook*. Ed. by Thomas Hermann, Andy Hunt, and John G. Neuhoff. Berlin, Germany: Logos Publishing House. Chap. 3, pp. 41–61. ISBN: 978-3-8325-2819-5. URL: http://sonification.de/handbook/chapters/chapter3/.

Cartwright, Mark and Bryan Pardo (June 2014). "SynthAssist: Querying an Audio Synthesizer by Vocal Imitation". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. London, United Kingdom: Goldsmiths, University of London, pp. 363–366. URL: http://www.nime.org/proceedings/2014/nime2014_446.pdf.

Casey, Michael A. and Mick Grierson (2007). "Soundspotter / remix-TV: Fast Approximate Matching for audio and video Performance". In: *Proceedings of the 2007 International Computer Music Conference, ICMC 2007, Copenhagen, Denmark, August 27-31, 2007*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.2007.205.

Casey, Michael A. and Malcolm Slaney (2006). "Song Intersection by Approximate Nearest Neighbor Search". In: *ISMIR 2006, 7th International Conference on Music Information Retrieval, Victoria, Canada, 8-12 October 2006, Proceedings*, pp. 144–149.

Charnass, Hlne (1980). "Towards a Data Base in Musicology: The Computer Processing of the Bridgman File". In: *Proceedings of the International Computer Music Conference, ICMC 1980*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.1980.051/1.

Choi, Insook, Geoffrey Zheng, and Ken Chen (2000). "Embedding a sensory data retrieval system in a movement-sensitive space and a surround sound system". In: *Proceedings of the International Computer Music Conference, ICMC 2000*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2000.146/1.

Ciardi, Fabio Cifariello (2004). "Real Time Sonification of Stock Market Data with sMax". In: *Proceedings of the International Computer Music Conference, ICMC 2004*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2004.124/1.

Clements, Peter J. (1980). "Musical Data Structures in a Multi-Use Environment". In: *Proceedings of the International Computer Music Conference, ICMC 1980*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.1980.020/1.

Codd, E. F. (June 1970). "A Relational Model of Data for Large Shared Data Banks". In: *Commun. ACM* 13.6, pp. 377–387. ISSN: 0001-0782. DOI: 10.1145/362384.362685. URL: http://doi.acm.org/10.1145/362384.362685.

Codd, E. F. (1972). "Relational completeness of data base sublanguages". In: *Database Systems*. Prentice-Hall, pp. 65–98.

Cope, David (1987). "An Expert System for Computer-Assisted Composition". In: *Computer Music Journal* 11.4, pp. 30–46. ISSN: 01489267, 15315169. URL: http://www.jstor.org/stable/3680238.

Crestel, Léopold et al. (2017). "A Database Linking Piano and Orchestral MIDI Scores with Application to Automatic Projective Orchestration". In: *Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR 2017, Suzhou, China, October 23-27, 2017*. Ed. by Sally Jo Cunningham et al., pp. 592–598. ISBN: 978-981-11-5179-8. URL: https://ismir2017.smcnus.org/wp-content/uploads/2017/10/235_Paper.pdf.

Crowley, Charles (1998). “Data Structures for Text Sequences”. In: . URL: `https://www.cs.unm.edu/~crowley/papers/sds.pdf`.

Cunningham, Sally Jo et al., eds. (2017). *Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR 2017, Suzhou, China, October 23-27, 2017*. ISBN: 978-981-11-5179-8.

Daniel, Sharon (2007). “The Database: An Aesthetics of Dignity”. In: *Database Aesthetics: Art in the Age of Information Overflow*.

Defferrard, Michaël et al. (2017). “FMA: A Dataset for Music Analysis”. In: *Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR 2017, Suzhou, China, October 23-27, 2017*. Ed. by Sally Jo Cunningham et al., pp. 316–323. ISBN: 978-981-11-5179-8. URL: `https://ismir2017.smcnus.org/wp-content/uploads/2017/10/75_Paper.pdf`.

Depalle, Philippe et al. (1993). “Generalized Diphone Control”. In: *Opening a New Horizon: Proceedings of the 1993 International Computer Music Conference, ICMC 1993, Tokio, Japan, September 10-15, 1993*. Michigan Publishing. URL: `http://hdl.handle.net/2027/spo.bbp2372.1993.038`.

Derrida, Jacques (1978). *Writing and Difference*. The University of Chicago.

Derrida, Jacques (1982). *Margins of Philosophy*. The Harvester Press.

Derrida, Jacques and Eric Prenowitz (1995). “Archive Fever: A Freudian Impression”. In: *Diacritics* 25.2.

Devaney, Johanna et al. (2015). “Theme And Variation Encodings with Roman Numerals (TAVERN): A New Data Set for Symbolic Music Analysis”. In: *Proceedings of the 16th International Society for Music Information Retrieval Conference, ISMIR 2015, Málaga, Spain, October 26-30, 2015*. Ed. by Meinard Müller and Frans Wiering, pp. 728–734. ISBN: 978-84-606-8853-2. URL: `http://ismir2015.uma.es/articles/261_Paper.pdf`.

Diener, Glendon (1985). “Formal Languages in Music Theory”. MA thesis. McGill University, Faculty of Music.

Diener, Glendon (1988). “TTrees: An Active Data Structure for Computer Music”. In: *Proceedings of the International Computer Music Conference, ICMC 1988*. Michigan Publishing. URL: `https://quod.lib.umich.edu/i/icmc/bbp2372.1988.020/1`.

Diener, Glendon (1989). “TTrees: A Tool for the Compositional Environment”. In: *Computer Music Journal* 13.2, pp. 77–85. ISSN: 01489267, 15315169. URL: `http://www.jstor.org/stable/3680043`.

Diener, Glendon R. (1992). “A Visual Programming Environment for Music Notation”. In: *Proceedings of the 1992 International Computer Music Conference, ICMC 1992, San Jose, California, USA, October 14-18, 1992*. Michigan Publishing. URL: `http://hdl.handle.net/2027/spo.bbp2372.1992.030`.

Donahue, Chris, Huanru Henry Mao, and Julian McAuley (2018). “The NES Music Database: A multi-instrumental dataset with expressive performance attributes”. In: *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, Paris, France, September 23-27, 2018*. Ed. by Emilia Gómez et al., pp. 475–482. ISBN: 978-2-9540351-2-3. URL: `http://ismir2018.ircam.fr/doc/pdfs/265_Paper.pdf`.

Dunn, Jon W. (2000). "Beyond VARIATIONS: Creating a Digital Music Library". In: *ISMIR 2000, 1st International Symposium on Music Information Retrieval, Plymouth, Massachusetts, USA, October 23-25, 2000, Proceedings*. URL: http://ismir2000.ismir.net/papers/invites/dunn_invite.pdf.

Dydo, J. Stephen (1987). "Data Structures in the Note Processor". In: *Proceedings of the International Computer Music Conference, ICMC 1987*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.1987.045/1.

Eck, Cathy van (2013). "Between air and electricity: Microphones and loudspeakers as musical instruments". PhD thesis. Leiden University.

Eco, Umberto (2004). "The Poetics of The Open Work". In: *Audio Culture: Readings in Modern Music*.

Eremenko, Vsevolod et al. (2018). "Audio-Aligned Jazz Harmony Dataset for Automatic Chord Transcription and Corpus-based Research". In: *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, Paris, France, September 23-27, 2018*. Ed. by Emilia Gómez et al., pp. 483–490. ISBN: 978-2-9540351-2-3. URL: http://ismir2018.ircam.fr/doc/pdfs/206_Paper.pdf.

Erickson, Raymond F. (1975). ""The Darms Project": A Status Report". In: *Computers and the Humanities* 9.6, pp. 291–298. ISSN: 00104817. URL: http://www.jstor.org/stable/30204239.

Ernst, Wolfgang (2013). *Digital Memory and the Archive*. University of Minnesota Press.

Esposito, Angelo, Rafael Krichevsky, and Alberto Nicolis (Mar. 2019). "Gravitational Mass Carried by Sound Waves". In: *Phys. Rev. Lett.* 122 (8), p. 084501. DOI: 10.1103/PhysRevLett.122.084501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.122.084501.

Flusser, Vilm (2011). *Into the Universe of Technical Images*. University of Minnesota Press.

Fonseca, Eduardo et al. (2017). "Freesound Datasets: A Platform for the Creation of Open Audio Datasets". In: *Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR 2017, Suzhou, China, October 23-27, 2017*. Ed. by Sally Jo Cunningham et al., pp. 486–493. ISBN: 978-981-11-5179-8. URL: https://ismir2017.smcnus.org/wp-content/uploads/2017/10/161_Paper.pdf.

Fox, Michael K., Jeremy Stewart, and Rob Hamilton (2017). "madBPM: A Modular Multimodal Environment for Data-Driven Composition And Sonification". In: *Proceedings of the International Computer Music Conference, ICMC 2017*. Michigan Publishing. URL: http://hdl.handle.net/2027/bbp2372.2017.087.

Free, John (1987). "Towards an Extensible Data Structure for the Representation of Music on Computers". In: *Proceedings of the International Computer Music Conference, ICMC 1987*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.1987.046/1.

Free, John and Paul Vytas (1986). "What Ever Happened to SSSP?" In: *Proceedings of the 1986 International Computer Music Conference, ICMC 1986, Den Haag, The Netherlands, October 20-24, 1986*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1986.004.

Free, John and Paul Vytas (1988). “The CAMP Music Configuration Database”. In: *Proceedings of the 1988 International Computer Music Conference, ICMC 1988, Cologne, Germany, September 20-25, 1988*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1988.014.

Frisson, Christian (2015). “Designing interaction for browsing media collections (by similarity)”. PhD thesis. Universit de Mons. URL: http://tcts.fpms.ac.be/publications/phds/frisson/phd-frisson.pdf.

Frisson, Christian et al. (2010). “DeviceCycle : Rapid and Reusable Prototyping of Gestural Interfaces, Applied to Audio Browsing by Similarity”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Sydney, Australia, pp. 473–476. URL: http://www.nime.org/proceedings/2010/nime2010_473.pdf.

Galloway, Alexander R. (2011). “What is New Media?: Ten Years after the Language of New”. In: *Criticism* 53.3.

García, Francisco et al. (2011). “Acquisition and Study of Blowing Pressure Profiles in Recorder Playing”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Oslo, Norway, pp. 124–127. URL: http://www.nime.org/proceedings/2011/nime2011_124.pdf.

Garton, Brad and David Topper (1997). “RTcmix - Using CMIX in Real Time”. In: *Proceedings of the 1997 International Computer Music Conference, ICMC 1997, Thessaloniki, Greece, September 25-30, 1997*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1997.106.

Gómez, Emilia et al., eds. (2018). *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, Paris, France, September 23-27, 2018*. ISBN: 978-2-9540351-2-3.

Good, Michael (2000). “Representing Music Using XML”. In: *ISMIR 2000, 1st International Symposium on Music Information Retrieval, Plymouth, Massachusetts, USA, October 23-25, 2000, Proceedings*. URL: http://ismir2000.ismir.net/posters/good.pdf.

Goto, Masataka et al. (2002). “RWC Music Database: Popular, Classical and Jazz Music Databases”. In: *ISMIR 2002, 3rd International Conference on Music Information Retrieval, Paris, France, October 13-17, 2002, Proceedings*. URL: http://ismir2002.ismir.net/proceedings/03-SP04-1.pdf.

Goto, Masataka et al. (2003). “RWC Music Database: Music genre database and musical instrument sound database”. In: *ISMIR 2003, 4th International Conference on Music Information Retrieval, Baltimore, Maryland, USA, October 27-30, 2003, Proceedings*. URL: http://ismir2003.ismir.net/papers/Goto1.PDF.

Gouyon, Fabien et al., eds. (2012). *Proceedings of the 13th International Society for Music Information Retrieval Conference, ISMIR 2012, Mosteiro S.Bento Da Vitória, Porto, Portugal, October 8-12, 2012*. FEUP Edições. ISBN: 978-972-752-144-9.

Gratton, Peter and Marie-Eve Morin (2015). *The Nancy Dictionary*. Edinburgh University Press.

Hamanaka, Masatoshi, Keiji Hirata, and Satoshi Tojo (2014). “Musical Structural Analysis Database Based on GTTM”. In: *Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR 2014, Taipei, Taiwan, October 27-31, 2014*. Ed. by Hsin-Min Wang,

Yi-Hsuan Yang, and Jin Ha Lee, pp. 325–330. URL: http://www.terasoft.com.tw/conf/ismir2014/proceedings/T059_257_Paper.pdf.
Hamilton, Robert (2006). "Bioinformatic Response Data as a Compositional Driver". In: *Proceedings of the International Computer Music Conference, ICMC 2006*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2006.123/1.
Hansen, Mark B. N. (2002). "Cinema Beyond Cybernetics, or How to Frame the Digital Image". In: *Configurations* 10.1.
Hansen, Mark B. N. (2004). *New Philosophy for New Media*. The MIT Press.
Hansen, Mark B. N. (2006). "Media Theory". In: *Theory, Culture and Society* 23.
Hashida, Mitsuyo, Toshie Matsui, and Haruhiro Katayose (2008). "A New Music Database Describing Deviation Information of Performance Expressions". In: *ISMIR 2008, 9th International Conference on Music Information Retrieval, Drexel University, Philadelphia, PA, USA, September 14-18, 2008*. Ed. by Juan Pablo Bello, Elaine Chew, and Douglas Turnbull, pp. 489–494. ISBN: 978-0-615-24849-3. URL: http://ismir2008.ismir.net/papers/ISMIR2008_173.pdf.
Hauger, David et al. (2013). "The Million Musical Tweet Dataset - What We Can Learn From Microblogs". In: *Proceedings of the 14th International Society for Music Information Retrieval Conference, ISMIR 2013, Curitiba, Brazil, November 4-8, 2013*. Ed. by Alceu de Souza Britto Jr., Fabien Gouyon, and Simon Dixon, pp. 189–194. ISBN: 978-0-615-90065-0. URL: http://www.ppgia.pucpr.br/ismir2013/wp-content/uploads/2013/09/85_Paper.pdf.
Hayles, Katherine (1993). "The Materiality of Informatics". In: *Configurations* 1.1.
Hayles, N. Katherine (1999). *How We Became Posthuman: Virtual Bodies in Cybernetics, Literature, and Informatics*. The University of Chicago Press.
Hearing the Self: Proceedings of the 43rd International Computer Music Conference, ICMC 2017, Shanghai, China, October 10 - October 17, 2017 (2017). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.2017.
Hermann, Thomas, Andy Hunt, and John G. Neuhoff, eds. (2011). *The Sonification Handbook*. Berlin, Germany: Logos Publishing House, pp. 1–586. ISBN: 978-3-8325-2819-5. URL: http://sonification.de/handbook.
Hochenbaum, Jordan, Ajay Kapur, and Matthew Wright (2010). "Multimodal Musician Recognition". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Sydney, Australia, pp. 233–237. URL: http://www.nime.org/proceedings/2010/nime2010_233.pdf.
Homburg, Helge et al. (2005). "A Benchmark Dataset for Audio Classification and Clustering". In: *ISMIR 2005, 6th International Conference on Music Information Retrieval, London, UK, 11-15 September 2005, Proceedings*, pp. 528–531. URL: http://ismir2005.ismir.net/proceedings/2117.pdf.
Humphrey, Eric, Simon Durand, and Brian McFee (2018). "OpenMIC-2018: An Open Data-set for Multiple Instrument Recognition". In: *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, Paris, France, September 23-27, 2018*. Ed. by

Emilia Gómez et al., pp. 438–444. ISBN: 978-2-9540351-2-3. URL: http://ismir2018.ircam.fr/doc/pdfs/248_Paper.pdf.

ISMIR 2000, 1st International Symposium on Music Information Retrieval, Plymouth, Massachusetts, USA, October 23-25, 2000, Proceedings (2000).

ISMIR 2002, 3rd International Conference on Music Information Retrieval, Paris, France, October 13-17, 2002, Proceedings (2002).

IV, John A. Maurer (1999). *A Brief History of Algorithmic Composition*. Online. URL: https://ccrma.stanford.edu/~blackrse/algorithm.html.

Jaimovich, Javier and R. Benjamin Knapp (May 2015). "Creating Biosignal Algorithms for Musical Applications from an Extensive Physiological Database". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by Edgar Berdahl and Jesse Allison. Baton Rouge, Louisiana, USA: Louisiana State University, pp. 1–4. URL: http://www.nime.org/proceedings/2015/nime2015_163.pdf.

Jaimovich, Javier, Miguel Ortiz, et al. (2012). "The Emotion in Motion Experiment: Using an Interactive Installation as a Means for Understanding Emotional Response to Music". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ann Arbor, Michigan: University of Michigan. URL: http://www.nime.org/proceedings/2012/nime2012_254.pdf.

Jones, Randy, Mathieu Lagrange, and W. Andrew Schloss (2007). "A hand drumming Dataset for Physical Modeling". In: *Proceedings of the 2007 International Computer Music Conference, ICMC 2007, Copenhagen, Denmark, August 27-31, 2007*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.2007.083.

Jr., Carlos Nascimento Silla, Alessandro L. Koerich, and Celso A. A. Kaestner (2008). "The Latin Music Database". In: *ISMIR 2008, 9th International Conference on Music Information Retrieval, Drexel University, Philadelphia, PA, USA, September 14-18, 2008*. Ed. by Juan Pablo Bello, Elaine Chew, and Douglas Turnbull, pp. 451–456. ISBN: 978-0-615-24849-3. URL: http://ismir2008.ismir.net/papers/ISMIR2008_106.pdf.

Karaosmanoglu, Mustafa Kemal (2012). "A Turkish Makam Music Symbolic Database for Music Information Retrieval: SymbTr". In: *Proceedings of the 13th International Society for Music Information Retrieval Conference, ISMIR 2012, Mosteiro S.Bento Da Vitória, Porto, Portugal, October 8-12, 2012*. Ed. by Fabien Gouyon et al. FEUP Edições, pp. 223–228. ISBN: 978-972-752-144-9. URL: http://ismir2012.ismir.net/event/papers/223-ismir-2012.pdf.

Katayose, Haruhiro and Hideki Kawahara (1999). "Applying STRAIGHT toward Music Systems - Accurate F0 Estimation and Application for Data-driven Synthesis". In: *Proceedings of the International Computer Music Conference, ICMC 1999*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.1999.411/1.

Kawahara, Hideki, Hideki Banno, and Masanori Morise (2004). "Acappella Synthesis Demonstrations using RWC Music Database". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Hamamatsu, Japan, pp. 130–131. URL: http://www.nime.org/proceedings/2004/nime2004_130.pdf.

Kernighan, Brian W (1978). *The C programming language*. eng. Englewood Cliffs, N.J.: Prentice-Hall. ISBN: 9780131101630.

Kirlin, Phillip B. (2014). “A Data Set for Computational Studies of Schenkerian Analysis”. In: *Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR 2014, Taipei, Taiwan, October 27-31, 2014*. Ed. by Hsin-Min Wang, Yi-Hsuan Yang, and Jin Ha Lee, pp. 213–218. URL: http://www.terasoft.com.tw/conf/ismir2014/proceedings/T039_344_Paper.pdf.

Klapuri, Anssi and Colby Leider, eds. (2011). *Proceedings of the 12th International Society for Music Information Retrieval Conference, ISMIR 2011, Miami, Florida, USA, October 24-28, 2011*. University of Miami. ISBN: 978-0-615-54865-4. URL: http://ismir2011.ismir.net/.

Klein, Norman M. (2007). “Waiting for the World to Explode: How Data Convert into a Novel”. In: *Database Aesthetics: Art in the Age of Information Overflow*.

Knees, Peter et al. (2015). “Two Data Sets for Tempo Estimation and Key Detection in Electronic Dance Music Annotated from User Corrections”. In: *Proceedings of the 16th International Society for Music Information Retrieval Conference, ISMIR 2015, Málaga, Spain, October 26-30, 2015*. Ed. by Meinard Müller and Frans Wiering, pp. 364–370. ISBN: 978-84-606-8853-2. URL: http://ismir2015.uma.es/articles/246_Paper.pdf.

Kobayashi, Ryoho (2003). “Sound Clustering Synthesis Using Spectral Data”. In: *Proceedings of the International Computer Music Conference, ICMC 2003*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2003.052/1.

Latour, Bruno (1990). “On actor-network theory. A few clarifications plus more than a few complications”. In: *Philosophia* 25.3.

Latour, Bruno (1993). *We Have Never Been Modern*. Harvard University Press Cambridge, Massachusetts.

Lewis, George (2000). “Too many notes: Computers, Complexity, and Culture in Voyager”. In: *Leonardo Music Journal* 10.

Lewis, George E. (1999). “Interacting with latter-day musical automata”. In: *Contemporary Music Review* 18.3, pp. 99–112. DOI: 10.1080/07494469900640381. eprint: https://doi.org/10.1080/07494469900640381. URL: https://doi.org/10.1080/07494469900640381.

Lindborg, PerMagnus (2017). “Pacific Bell Tower, a sculptural sound installation for live sonification of earthquake data”. In: *Proceedings of the International Computer Music Conference, ICMC 2017*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2017.033/1.

Lindemann, Eric (1990). “ANIMAL-A Rapid Prototyping Environment for Computer Music Systems”. In: *Proceedings of the 1990 International Computer Music Conference, ICMC 1990, Glasgow, Scotland, September 10-15, 1990*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1990.068.

Liu, Qian et al. (May 2013). “Cloud Bridge: a Data-driven Immersive Audio-Visual Software Interface”. In: *Proceedings of the International Conference on New Interfaces for Musical Expres-*

sion. Daejeon, Republic of Korea: Graduate School of Culture Technology, KAIST, pp. 431–436. URL: http://nime.org/proceedings/2013/nime2013_250.pdf.

Long, Rick et al. (2000). *IMS Primer*. International Business Machines Corporation. URL: http://www.redbooks.ibm.com/redbooks/pdfs/sg245352.pdf.

Loviscach, Jörn (2008). "Programming a Music Synthesizer through Data Mining". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Genoa, Italy, pp. 221–224. URL: http://www.nime.org/proceedings/2008/nime2008_221.pdf.

Manovich, Lev (2001). *The Language of New Media*. MIT Press.

Manovich, Lev (2002). "Old Media as New Media: Cinema". In: *The New Media Book*.

Manovich, Lev (2015). "There Is Only Software". In: *Vision Anew - the Lens and Screen Arts*.

Mathews, M. V. (1963). "The Digital Computer as a Musical Instrument". In: *Science* 142.3592, pp. 553–557. ISSN: 00368075, 10959203. URL: http://www.jstor.org/stable/1712380.

Maxwell, James B. and Arne Eigenfeldt (2008). "A Music Database and Query System for Recombinant Composition". In: *ISMIR 2008, 9th International Conference on Music Information Retrieval, Drexel University, Philadelphia, PA, USA, September 14-18, 2008*. Ed. by Juan Pablo Bello, Elaine Chew, and Douglas Turnbull, pp. 75–80. ISBN: 978-0-615-24849-3. URL: http://ismir2008.ismir.net/papers/ISMIR2008_158.pdf.

Mazzoni, Dominic and Roger B. Dannenberg (2001). "A Fast Data Structure for Disk-Based Audio Editing". In: *Proceedings of the International Computer Music Conference, ICMC 2001*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2001.051/1.

McCartney, James (1996). "SuperCollider, a New Real Time Synthesis Language". In: *Proceedings of the 1996 International Computer Music Conference, ICMC 1996, Hong Kong, August 19-24, 1996*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1996.078.

McCartney, James (1998). "Continued Evolution of the SuperCollider Real Time Synthesis Environment". In: *Proceedings of the 1998 International Computer Music Conference, ICMC 1998, Ann Arbor, Michigan, USA, October 1-6, 1998*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1998.262.

McCurdy, Iain et al. (2015). "Methods of Writing Csound Scores". In: *FLOSS Manuals*. URL: http://write.flossmanuals.net/csound/methods-of-writing-csound-scores/.

Melucci, Massimo and Nicola Orio (1999). "The Use of Melodic Segmentation for Content-based Retrieval of Musical Data". In: *Proceedings of the International Computer Music Conference, ICMC 1999*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.1999.355/1.

Meseguer-Brocal, Gabriel, Alice Cohen-Hadria, and Geoffroy Peeters (2018). "DALI: A Large Dataset of Synchronized Audio, Lyrics and notes, Automatically Created using Teacher-student Machine Learning Paradigm." In: *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, Paris, France, September 23-27, 2018*. Ed. by

Emilia Gómez et al., pp. 431–437. ISBN: 978-2-9540351-2-3. URL: http://ismir2018.ircam.fr/doc/pdfs/35_Paper.pdf.

Mital, Parag Kumar and Mick Grierson (May 2013). "Mining Unlabeled Electronic Music Databases through 3D Interactive Visualization of Latent Component Relationships". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Daejeon, Republic of Korea: Graduate School of Culture Technology, KAIST, pp. 227–232. URL: http://nime.org/proceedings/2013/nime2013_132.pdf.

Morawitz, Falk (2016). "Molecular Sonification of Nuclear Magnetic Resonance Data as a Novel Tool for Sound Creation". In: *Proceedings of the International Computer Music Conference, ICMC 2016*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2016.002/1.

Morin, Pat (2019). *Open Data Structures*. Creative Commons. URL: http://http://opendatastructures.org/.

Morton, Timothy (2013). *Hyperobjects: Philosophy and Ecology after the End of the World*. University of Minnesota Press.

Müller, Meinard and Frans Wiering, eds. (2015). *Proceedings of the 16th International Society for Music Information Retrieval Conference, ISMIR 2015, Málaga, Spain, October 26-30, 2015*. ISBN: 978-84-606-8853-2.

Music Technology meets Philosophy - From Digital Echos to Virtual Ethos: Joint Proceedings of the 40th International Computer Music Conference, ICMC 2014, and the 11th Sound and Music Computing Conference, SMC 2014, Athens, Greece, September 14-20, 2014 (2014). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.2014.

N. Shepard, Roger (Dec. 1964). "Circularity in Judgments of Relative Pitch". In: *The Journal of the Acoustical Society of America* 36, p. 2346. DOI: 10.1121/1.1919362.

Nakamoto, Misako and Yasuo Kuhara (2007). "Circle Canon Chorus System Used To Enjoy A Musical Ensemble Singing "Frog Round"". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. New York City, NY, United States, pp. 409–410. URL: http://www.nime.org/proceedings/2007/nime2007_409.pdf.

Nancy, Jean-Luc (1991). *The Inoperative Community*. University of Minnesota Press, Minneapolis and Oxford.

Nancy, Jean-Luc (2007). *Listening*. Fordham University Place.

Nardelli, Marco Buongiorno (2015). "materialssoundmusic: a Computer-Aided Data-Driven Composition Environment for the Sonification and Dramatization of Scientific Data Streams". In: *Proceedings of the International Computer Music Conference, ICMC 2015*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2015.072/1.

Nichols, Charles et al. (2014). "Sound of Rivers: Stone Drum: a Multimedia Collaboration, with Sonified Data, Computer-Processed Narration, and Electric Violin". In: *Proceedings of the International Computer Music Conference, ICMC 2014*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2014.065/1.

Norman, Alex and Xavier Amatriain (2007). "DATA JOCKEY, A TOOL FOR META-DATA ENHANCED DIGITAL DJING AND ACTIVE LISTENING". In: *Proceedings of the Interna-*

tional Computer Music Conference, ICMC 2007. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2007.117/1.

Nort, Doug Van, Ian Jarvis, and Michael Palumbo (2016). “Towards a Mappable Database of Emergent Gestural Meaning”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Vol. 16. 2220-4806. Brisbane, Australia: Queensland Conservatorium Griffith University, pp. 46–50. ISBN: 978-1-925455-13-7. URL: http://www.nime.org/proceedings/2016/nime2016_paper0010.pdf.

Nuanàin, Càrthach Ó, Sergi Jordà, and Perfecto Herrera (2016). “An Interactive Software Instrument for Real-time Rhythmic Concatenative Synthesis”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Vol. 16. 2220-4806. Brisbane, Australia: Queensland Conservatorium Griffith University, pp. 383–387. ISBN: 978-1-925455-13-7. URL: http://www.nime.org/proceedings/2016/nime2016_paper0076.pdf.

Osaka, Naotoshi, Ken-Ichi Sakakibara, and Takafumi Hikichi (2002). “The Sound Synthesis System "Otkinshi": Its Data Structure and Graphical User Interface”. In: *Proceedings of the International Computer Music Conference, ICMC 2002*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2002.039/1.

Parada-Cabaleiro, Emilia et al. (2017). “The SEILS Dataset: Symbolically Encoded Scores in Modern-Early Notation for Computational Musicology”. In: *Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR 2017, Suzhou, China, October 23-27, 2017*. Ed. by Sally Jo Cunningham et al., pp. 575–581. ISBN: 978-981-11-5179-8. URL: https://ismir2017.smcnus.org/wp-content/uploads/2017/10/14_Paper.pdf.

Park, Sihwa et al. (2010). “Composition With Path : Musical Sonification Of Geo-referenced Data With Online Map Interface”. In: *Proceedings of the International Computer Music Conference, ICMC 2010*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2010.002/1.

Paul, Christiane (2007). “The Database as System and Cultural Form: Anatomies of Cultural Narratives”. In: *Database Aesthetics: Art in the Age of Information Overflow*.

Pauletto, Sandra and Andy Hunt (Jan. 2004). “A Toolkit for Interactive Sonification.” In: *Proceedings of ICAD 04. Tenth Meeting of the International Conference on Auditory Display, Sydney, Australia, July 6-9, 2004. Ed. Barrass, S. and Vickers, P. International Community for Auditory Display, 2004.*

Pesek, Matevz et al. (2014). “Introducing a Dataset of Emotional and Color Responses to Music”. In: *Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR 2014, Taipei, Taiwan, October 27-31, 2014*. Ed. by Hsin-Min Wang, Yi-Hsuan Yang, and Jin Ha Lee, pp. 355–360. URL: http://www.terasoft.com.tw/conf/ismir2014/proceedings/T064_307_Paper.pdf.

Poster, Mark (2011). “Introduction”. In: *Into the Universe of Technical Images*.

Price, Robin and Pedro Rebelo (2008). “Database and Mapping Design for Audiovisual Prepared Radio Set Installation”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Genoa, Italy, pp. 311–314. URL: http://www.nime.org/proceedings/2008/nime2008_311.pdf.

Proceedings of the 1978 International Computer Music Conference, ICMC 1978, Evanston, Illinois, USA, 1978 (1978). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.1978.

Proceedings of the 1980 International Computer Music Conference, ICMC 1980, New York City, USA, 1980 (1980). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.1980.

Proceedings of the 1981 International Computer Music Conference, ICMC 1981, Denton, Texas, USA, 1981 (1981). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.1981.

Proceedings of the 1986 International Computer Music Conference, ICMC 1986, Den Haag, The Netherlands, October 20-24, 1986 (1986). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.1986.

Proceedings of the 1987 International Computer Music Conference, ICMC 1987, Champaign/Urbana, Illinois, USA, August 23-26, 1987 (1987). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.1987.

Proceedings of the 1988 International Computer Music Conference, ICMC 1988, Cologne, Germany, September 20-25, 1988 (1988). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.1988.

Proceedings of the 1996 International Computer Music Conference, ICMC 1996, Hong Kong, August 19-24, 1996 (1996). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.1996.

Proceedings of the 1997 International Computer Music Conference, ICMC 1997, Thessaloniki, Greece, September 25-30, 1997 (1997). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.1997.

Proceedings of the 1998 International Computer Music Conference, ICMC 1998, Ann Arbor, Michigan, USA, October 1-6, 1998 (1998). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.1998.

Proceedings of the 1999 International Computer Music Conference, ICMC 1999, Beijing, China, October 22-27, 1999 (1999). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.1999.

Proceedings of the 2000 International Computer Music Conference, ICMC 2000, Berlin, Germany, August 27 - September 1, 2000 (2000). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.2000.

Proceedings of the 2001 International Computer Music Conference, ICMC 2001, Havana, Cuba, September 17-22, 2001 (2001). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.2001.

Proceedings of the 2002 International Computer Music Conference, ICMC 2002, Gothenburg, Sweden, September 16-21, 2002 (2002). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.2002.

Proceedings of the 2003 International Computer Music Conference, ICMC 2003, Singapore, September 29 - October 4, 2003 (2003). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.2003.

Proceedings of the 2004 International Computer Music Conference, ICMC 2004, Miami, Florida, USA, November 1-6, 2004 (2004). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.2004.

Proceedings of the 2005 International Computer Music Conference, ICMC 2005, Barcelona, Spain, September 4-10, 2005 (2005). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.2005.

Proceedings of the 2007 International Computer Music Conference, ICMC 2007, Copenhagen, Denmark, August 27-31, 2007 (2007). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.2007.

Proceedings of the 2009 International Computer Music Conference, ICMC 2009, Montreal, Quebec, Canada, August 16-21, 2009 (2009). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.2009.

Proceedings of the 2010 International Computer Music Conference, ICMC 2010, New York, USA, 2010 (2010). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.2010.

Proceedings of the 42nd International Computer Music Conference, ICMC 2016, Utrecht, Netherlands, September 25 - October 1, 2016 (2016). Michigan Publishing. URL: http://quod.lib.umich.edu/i/icmc/bbp2372.2016.

Proutskova, Polina et al. (2012). “Breathy or Resonant - A Controlled and Curated Dataset for Phonation Mode Detection in Singing”. In: *Proceedings of the 13th International Society for Music Information Retrieval Conference, ISMIR 2012, Mosteiro S.Bento Da Vitória, Porto, Portugal, October 8-12, 2012*. Ed. by Fabien Gouyon et al. FEUP Edições, pp. 589–594. ISBN: 978-972-752-144-9. URL: http://ismir2012.ismir.net/event/papers/589-ismir-2012.pdf.

Puckette, Miller (1986). “Interprocess Communication and Timing in Real-Time Computer Music Performance”. In: *Proceedings of the 1986 International Computer Music Conference, ICMC 1986, Den Haag, The Netherlands, October 20-24, 1986*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1986.008.

Puckette, Miller (2002a). “Max at Seventeen”. In: *Computer Music Journal* 26.4, pp. 31–43. DOI: 10.1162/014892602320991356. URL: https://doi.org/10.1162/014892602320991356.

Puckette, Miller (2002b). “Using Pd as a score language”. In: *Proceedings of the 2002 International Computer Music Conference, ICMC 2002, Gothenburg, Sweden, September 16-21, 2002*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.2002.038.

Puckette, Miller (2007). “On timbre stamps and other frequency-domain filters”. In: *ICMC*. Michigan Publishing.

Puckette, Miller S. (1997). “Pure Data”. In: *Proceedings of the International Computer Music Conference, ICMC 1997*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.1997.060/1.

Puckette, Miller, Barry Vercoe, and John P. Stautner (1981). “A Real-time Music 11 Emulator”. In: *Proceedings of the 1981 International Computer Music Conference, ICMC 1981, Denton,* *Texas, USA, 1981*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1981.036.

Ramakrishnan, R. and J. Ullman (1995). *A Survey of Research in Deductive Database Systems*. Technical Report 1995-14. Stanford Infolab. URL: http://ilpubs.stanford.edu:8090/80/.

Roads, C. and Max Mathews (1980). "Interview with Max Mathews". In: *Computer Music Journal* 4.4, pp. 15–22. ISSN: 01489267, 15315169. URL: http://www.jstor.org/stable/3679463.

Roads, Curtis (2001). *Microsound*. MIT Pess.

Roberts, Charlie et al. (June 2014). "Rapid Creation and Publication of Digital Musical Instruments". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. London, United Kingdom: Goldsmiths, University of London, pp. 239–242. URL: http://www.nime.org/proceedings/2014/nime2014_373.pdf.

Rodet, Xavier, Philippe Depalle, and Gilles Poirot (1988). "Diphone Sound Synthesis Based on Spectral Envelopes and Harmonic/Noise Excitation Functions". In: *Proceedings of the 1988 International Computer Music Conference, ICMC 1988, Cologne, Germany, September 20-25, 1988*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1988.033.

Rodet, Xavier and Adrien Lefèvre (1996). "Macintosh Graphical Interface and Improvements to Generalized Diphone Control and Synthesis". In: *Proceedings of the 1996 International Computer Music Conference, ICMC 1996, Hong Kong, August 19-24, 1996*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1996.102.

Rodet, Xavier and Adrien Lefèvre (1997). "The Diphone program: New features, new synthesis methods and experience of musical use". In: *Proceedings of the 1997 International Computer Music Conference, ICMC 1997, Thessaloniki, Greece, September 25-30, 1997*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1997.111.

Rossiter, David and Wai-Yin Ng (1996). "A System for the Musical Investigation and Expression of Levels of Self-Similarity in an Arbitrary Data Stream". In: *Proceedings of the International Computer Music Conference, ICMC 1996*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.1996.085/1.

Rowe, Robert et al. (1993). "Editor's Notes: Putting Max in Perspective". In: *Computer Music Journal* 17.2, pp. 3–11. ISSN: 01489267, 15315169. URL: http://www.jstor.org/stable/3680864.

Sanden, Chris, Chad R. Befus, and John Zahng (2010). "Perception Based Multi-Genre Labeling On Music Data". In: *Proceedings of the International Computer Music Conference, ICMC 2010*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2010.003/1.

Scaletti, Carla A. (1987). "Kyma: An Object-oriented Language for Music Composition". In: *Proceedings of the 1987 International Computer Music Conference, ICMC 1987, Champaign/Urbana, Illinois, USA, August 23-26, 1987*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1987.007.

Schlei, Kevin and Rebecca Yoshikane (2016). “The Things of Shapes: Waveform Generation using 3D Vertex Data”. In: *Proceedings of the International Computer Music Conference, ICMC 2016*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2016.056/1.

Schloss, W. et al. (2001). “Towards a Virtual Membrane: New Algorithms and Technology for Analyzing Gestural Data”. In: *Proceedings of the International Computer Music Conference, ICMC 2001*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2001.103/1.

Schmeder, Andrew (2009). “Efficient Gesture Storage and Retrieval for Multiple Applications Using a Relational Data Model of Open Sound Control”. In: *Proceedings of the International Computer Music Conference, ICMC 2009*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2009.005/1.

Schöner, Bernd et al. (1998). “Data-driven Modeling and Synthesis of Acoustical Instruments”. In: *Proceedings of the 1998 International Computer Music Conference, ICMC 1998, Ann Arbor, Michigan, USA, October 1-6, 1998*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1998.265.

Schwarz, Diemo (2000). “A System for Data-driven Concatenative Sound Synthesis”. In: *Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFX-00), Verona, Italy, December 7-9*.

Schwarz, Diemo (2003). “New Developments in Data-Driven Concatenative Sound Synthesis”. In: *Proceedings of the International Computer Music Conference, ICMC 2003*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2003.099/1.

Schwarz, Diemo (2012). “The Sound Space as Musical Instrument: Playing Corpus-Based Concatenative Synthesis”. In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ann Arbor, Michigan: University of Michigan. URL: http://www.nime.org/proceedings/2012/nime2012_120.pdf.

Selfridge-Field, Eleanor (1997). “The SCORE Music Publishing System”. In: *SCORE*. From Beyond MIDI, The Handbook of Musical Codes. URL: http://scoremus.com/score.html.

Serafin, Stefania et al. (2001). “Data driven identification and computer animation of bowed string model”. In: *Proceedings of the International Computer Music Conference, ICMC 2001*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2001.071/1.

Shanks, Daniel and John W.jun. Wrench (Jan. 1962). “Calculation of Pi to 100,000 Decimals”. In: *Mathematics of Computation* 16. DOI: 10.2307/2003813.

Skinner, Robert (1990a). “Music Software”. In: *Notes* 46.3, pp. 660–684. ISSN: 00274380, 1534150X. URL: http://www.jstor.org/stable/941442.

Skinner, Robert (1990b). “Music Software”. In: *Notes* 47.1, pp. 91–101. ISSN: 00274380, 1534150X. URL: http://www.jstor.org/stable/940555.

Smith, Jordan Bennett Louis et al. (2011). “Design and creation of a large-scale database of structural annotations”. In: *Proceedings of the 12th International Society for Music Information Retrieval Conference, ISMIR 2011, Miami, Florida, USA, October 24-28, 2011*. Ed. by Anssi

Klapuri and Colby Leider. University of Miami, pp. 555–560. ISBN: 978-0-615-54865-4. URL: http://ismir2011.ismir.net/papers/PS4-14.pdf.
Smith, Leeland (1972). "SCORE: A Musician's Approach to Computer Music". In: *Journal of the Audio Engineering Society* 20.1, pp. 7–14. URL: https://ccrma.stanford.edu/~aj/archives/docs/all/649.pdf.
Solomos, Makis (2005). "An Introduction to Horacio Vaggione Musical-Theoretical Thought". In: *Contemporary Music Review* 25.4, pp. 311–326. URL: https://hal.archives-ouvertes.fr/hal-00770212.
Sturm, Bob L. (2002). "Water Music: Sonification of Ocean Buoy Spectral Data". In: *Proceedings of the International Computer Music Conference, ICMC 2002*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2002.056/1.
Szendy, Peter (2008). *Listen: A History of Our Ears*. Fordham University. URL: https://www.jstor.org/stable/j.ctt13x002m.
Taylor, Benjamin et al. (June 2014). "Simplified Expressive Mobile Development with NexusUI, NexusUp, and NexusDrop". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. London, United Kingdom: Goldsmiths, University of London, pp. 257–262. URL: http://www.nime.org/proceedings/2014/nime2014_480.pdf.
Thacker, Eugene (2006). "Review: Hansen: New Philosophy". In: *Leonardo* 39.3.
Vaggione, Horacio (1993). "Determinism and the false collective about models of time in early computer-aided composition". In: *Contemporary Music Review* 7.2.
Vaggione, Horacio (2001). "Some Ontological Remarks about Music Composition Processes". In: *Computer Music Journal* 25.1, pp. 54–61.
Varese, Edgar (2004). "The Liberation of Sound". In: *Audio Culture: Readings in Modern Music*.
Vesna, Victoria (2007a). *Database Aesthetics: Art in the Age of Information Overflow*. University of Minnesotta Press.
Vesna, Victoria (2007b). "Seeing the world in a grain of sand: the database aesthetics of everything". In: *Database Aesthetics: Art in the Age of Information Overflow*.
Vigliensoni, Gabriel and Ichiro Fujinaga (2017). "The Music Listening Histories Dataset". In: *Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR 2017, Suzhou, China, October 23-27, 2017*. Ed. by Sally Jo Cunningham et al., pp. 96–102. ISBN: 978-981-11-5179-8. URL: https://ismir2017.smcnus.org/wp-content/uploads/2017/10/180_Paper.pdf.
Vinet, Hugues (2005). "The Semantic HiFi Project". In: *Proceedings of the 2005 International Computer Music Conference, ICMC 2005, Barcelona, Spain, September 4-10, 2005*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.2005.171.
Vinet, Hugues, Perfecto Herrera, and François Pachet (2002a). "The CUIDADO Project". In: *ISMIR 2002, 3rd International Conference on Music Information Retrieval, Paris, France, October 13-17, 2002, Proceedings*. URL: http://ismir2002.ismir.net/proceedings/02-FP06-3.pdf.
Vinet, Hugues, Perfecto Herrera, and François Pachet (2002b). "The CUIDADO Project: New Applications based on Audio and Music Content Description". In: *Proceedings of the 2002 In-*

ternational Computer Music Conference, ICMC 2002, Gothenburg, Sweden, September 16-21, 2002. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.2002.093.

Visi, Federico et al. (2017). "A Knowledge-based, Data-driven Method for Action-sound Mapping". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Copenhagen, Denmark: Aalborg University Copenhagen, pp. 231–236. URL: http://www.nime.org/proceedings/2017/nime2017_paper0043.pdf.

Vogl, Richard and Peter Knees (2017). "An Intelligent Drum Machine for Electronic Dance Music Production and Performance". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Copenhagen, Denmark: Aalborg University Copenhagen, pp. 251–256. URL: http://www.nime.org/proceedings/2017/nime2017_paper0047.pdf.

Vogt, Katharina et al. (2012). "SOUNDS OF SIMULATIONS: DATA LISTENING SPACE". In: *Proceedings of the International Computer Music Conference, ICMC 2012*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2012.096/1.

von Neumann, Jon and A.W Burks (1946). "Preliminary Discussion of the Logical Design of an Electronic Computing Instrument". In: *Engineering, College of - Technical Reports*. URL: https://library.ias.edu/files/Prelim_Disc_Logical_Design.pdf.

Walker, Bruce N. and Michael A. Nees (2011). "Theory of Sonification". In: *The Sonification Handbook*. Ed. by Thomas Hermann, Andy Hunt, and John G. Neuhoff. Berlin, Germany: Logos Publishing House. Chap. 2, pp. 9–39. ISBN: 978-3-8325-2819-5. URL: http://sonification.de/handbook/chapters/chapter2/.

Wang, Hsin-Min, Yi-Hsuan Yang, and Jin Ha Lee, eds. (2014). *Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR 2014, Taipei, Taiwan, October 27-31, 2014*. URL: http://www.terasoft.com.tw/conf/ismir2014/.

Weinbren, Grahame (2007). "Ocean, Database, Recut". In: *Database Aesthetics: Art in the Age of Information Overflow*.

Wessel, Ineke and Michelle L. Moulds (2008). "How many types of forgetting? Comments on Connerton (2008)". In: *Memory Studies* 1.3.

Whalley, Ian (2014). "Broadening Telematic Electroacoustic Music by Affective Rendering and Embodied Real-time Data Sonification". In: *Proceedings of the International Computer Music Conference, ICMC 2014*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2014.046/1.

Wilkins, Julia et al. (2018). "VocalSet: A Singing Voice Dataset". In: *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, Paris, France, September 23-27, 2018*. Ed. by Emilia Gómez et al., pp. 468–474. ISBN: 978-2-9540351-2-3. URL: http://ismir2018.ircam.fr/doc/pdfs/114_Paper.pdf.

Wüst, Otto and Òscar Celma (2004). "An MPEG-7 Database System and Application for Content-Based Management and Retrieval of Music". In: *ISMIR 2004, 5th International Conference on Music Information Retrieval, Barcelona, Spain, October 10-14, 2004, Proceedings*. URL: http://ismir2004.ismir.net/proceedings/p010-page-48-paper227.pdf.

Xamb, Anna et al. (June 2018). "Live Repurposing of Sounds: MIR Explorations with Personal and Crowdsourced Databases". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Ed. by Thomas Martin Luke Dahl Douglas Bowman. Blacksburg, Virginia, USA: Virginia Tech, pp. 364–369. ISBN: 978-1-949373-99-8. URL: http://www.nime.org/proceedings/2018/nime2018_paper0081.pdf.

Xenakis, Iannis (1992). *Formalized Music: Thought and Mathematics in Music*. Pendragon Revised Edition.

Xi, Qingyang et al. (2018). "GuitarSet: A Dataset for Guitar Transcription". In: *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, Paris, France, September 23-27, 2018*. Ed. by Emilia Gómez et al., pp. 453–460. ISBN: 978-2-9540351-2-3. URL: http://ismir2018.ircam.fr/doc/pdfs/188_Paper.pdf.

Xu, Yungpeng, Changshui Zang, and Jing Yang (2005). "Semi-Supervised Classification of Musical Genre using Multi-View Features". In: *Proceedings of the 2005 International Computer Music Conference, ICMC 2005, Barcelona, Spain, September 4-10, 2005*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.2005.020.

Yang, Cheng (2001). *Music Database Retrieval Based on Spectral Similarity*. Technical Report 2001-14. Stanford InfoLab. URL: http://ilpubs.stanford.edu:8090/489/.

Yeh, Chunghsin, Niels Bogaards, and Axel Röbel (2007). "Synthesized Polyphonic Music Database with Verifiable Ground Truth for Multiple F0 Estimation". In: *Proceedings of the 8th International Conference on Music Information Retrieval, ISMIR 2007, Vienna, Austria, September 23-27, 2007*. Ed. by Simon Dixon, David Bainbridge, and Rainer Typke. Austrian Computer Society, pp. 393–398. ISBN: 978-3-85403-218-2. URL: http://ismir2007.ismir.net/proceedings/ISMIR2007_p393_yeh.pdf.

Yeo, Woon Seung and Jonathan Berger (2005). "Application of Image Sonification Methods to Music". In: *Proceedings of the 2005 International Computer Music Conference, ICMC 2005, Barcelona, Spain, September 4-10, 2005*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.2005.036.

Yeo, Woon et al. (2004). "SonART: A framework for data sonification, visualization and networked multimedia applications". In: *Proceedings of the International Computer Music Conference, ICMC 2004*. Michigan Publishing. URL: https://quod.lib.umich.edu/i/icmc/bbp2372.2004.128/1.

Young, Diana and Anagha Deshmane (2007). "Bowstroke Database : A Web-Accessible Archive of Violin Bowing Data". In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. New York City, NY, United States, pp. 352–357. URL: http://www.nime.org/proceedings/2007/nime2007_352.pdf.

Zicarelli, David (1998). "An Extensible Real-time Signal Processing Environment for Max". In: *Proceedings of the 1998 International Computer Music Conference, ICMC 1998, Ann Arbor, Michigan, USA, October 1-6, 1998*. Michigan Publishing. URL: http://hdl.handle.net/2027/spo.bbp2372.1998.274.