Title
Interface Aesthetics: Sound, Software, and the Ecology of Digital Audio Production

Permalink
https://escholarship.org/uc/item/0mv9v64c

Author
D'Errico, Michael

Publication Date
2016

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

Interface Aesthetics:

Sound, Software, and the Ecology of Digital Audio Production

A dissertation submitted in partial satisfaction of the requirements for the degree

Doctor of Philosophy in Musicology

by

Michael Anthony D’Errico

2016

ABSTRACT OF THE DISSERTATION

Interface Aesthetics: Sound, Software, and the Ecology of Digital Audio Production

by

Michael Anthony D'Errico

Doctor of Philosophy in Musicology

University of California, Los Angeles, 2016

Professor Nina Eidsheim, Chair

This dissertation investigates the ways in which the composition and production of electronic music is influenced by software and digital instrument design. Through my research, I found that musicians, sound designers, and multimedia artists are not only creative agents, but must also balance aesthetic preferences with the affordances and constraints of audio software. My inquiry combines multiple research methods. Ethnographic data gathered from participant observation and interviews with software developers, interaction designers, and electronic musicians in Southern California has revealed the extent to which digital producers increasingly borrow the tools and techniques of artists working across media platforms. Close readings and technical analyses of specific software have allowed me trace new genealogies in the evolution of music technology. Finally—influenced by the praxis-based approach of digital humanities—the creation of original software and digital applications has made it possible for me to bridge the

gap between theoretical and technical perspectives on music and media production. By applying the techniques of media theory, computer science, design, and digital humanities to critical practices in media art in the twenty-first century, the project brings to bear shifting understandings of cultural objects as they increasingly converge with computational processes.

The dissertation of Michael Anthony D'Errico is approved.

Robert W. Fink

Peter B. Lunenfeld

Timothy D. Taylor

Nina Eidsheim, Committee Chair

University of California, Los Angeles

2016

To Bess, for your strength and patience.

In loving memory of Gary D'Errico.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

First, I would like to thank my committee chair, Nina Eidsheim. I was introduced to Nina by my former advisor, Joseph Auner, who told me that I needed to meet "this opera singer and professor at UCLA who performs with Ableton *Live*." I have always valued Nina's insights and advice because her work is fundamentally grounded in a balance between theory and practice. By watching her effortlessly navigate disciplines and discourses ranging from musicology to critical race theory and medical perspectives on the voice, I have learned that interdisciplinary, practice-based work is necessary in order to gain the fullest understanding of the role of music in contemporary culture. Nina's superb research, writing, and editing skills—as well as her endless commitment to student work—are obvious to any graduate student who has had the pleasure of serving as Teaching Assistant for her "Writing About Music" class. These skills were especially apparent to me in the final months of writing this dissertation, and I am grateful to Nina for pushing me to become a better writer all the time.

I am also grateful to my other musicology and ethnomusicology committee members, Bob Fink and Tim Taylor. Bob is a true pioneer in popular music studies, and his model research and teaching continues to inspire me to pursue critical topics in a rigorous manner, regardless of how new or different they may be. It is a testament to his passion for music that some of our most fruitful academic conversations have occurred in such venues as Hollywood dance clubs, DJ nights at the hotel bar during academic conferences, or local karaoke nights to celebrate the end of a graduate seminar. Similarly, the work of Tim Taylor continues to break new ground for scholars interested in music, media, and capitalism. His book, *Strange Sounds*, was one of the first academic books I read, and it inspired me to pursue what have become the main topics of

my work: music, technology, and digital culture. In addition to retaining an encyclopedic knowledge of Marxist theory, Tim is one of the kindest and most caring people I have ever met. Our conversations about gardening and Italian food have been just as useful to me as discussions about the differences between use value and exchange value.

I am indebted to my outside committee member and Digital Humanities program advisor, Peter Lunenfeld, for constantly pushing me to conceptualize my research and creative work as *generative* practices capable of inspiring further work from scholars and practitioners. Engaging with both academics and creative professionals has been an enduring goal of mine, and Peter continues to provide useful models for thinking through what he calls "the maker's discourse." Further, his dynamic and provocative writing is a constant reminder that, when making any argument, compositional style and *design* matters just as much as content.

Special thanks are due to faculty members in the UCLA Digital Humanities program, all of whom taught me the values of experimentalism and risk-taking in my research and teaching. Stephen Mamber organized his courses as speculative labs rather than seminars, giving me one of my first introductions to the world of do-it-yourself electronics and computer programming. Miriam Posner encouraged me to pursue collaborative work in the academy, in addition to teaching me the art of project management. I have always been inspired by Miriam's care and respect for her students. Other Digital Humanities collaborators that have influenced directly the shape of this dissertation include Chris Johanson, Johanna Drucker, and Todd Presner.

In pursuing the collaborative and interdisciplinary work necessary for this project, I could not have been better situated than the UCLA Musicology department. During my five years in the department, faculty members and graduate students alike constantly pushed me to think critically about the fundamental *methods* of musicological inquiry. These questions have guided

the writing of this dissertation: how can I reshape the practice of sketch studies in the digital age? What is the nature of the musical "work" in the age of software? What are the defining features of "musicological" research in the context of increasingly public, interdisciplinary, and digital humanities programs? In particular, I wish to thank Olivia Bloechl, for inviting me to think through the ethics of care in virtual reality and computing; Ray Knapp for reminding me about the importance of history in the study of culture; Tamara Levitz, for encouraging me to think about the possibilities for social justice at the root of all humanistic work; and Brian and Elizabeth Upton, for recognizing the nature of musical practice as a form of experimental play.

Others along the way have provided big-picture guidance, encouragement, and general advice. I thank former advisors Joseph Auner, Rob Haskins, and Stephan Pennington, for introducing me to the craft of research; former teachers Bruce Gatchell and John Herman, for introducing me to the craft of teaching; managers Annie Maxfield and John Lynch, for pointing out the relevance of my skills to disciplines and industries outside of the academy; fellow musicians Ryan Baker, Jon Briggs, Vince Diamante, Jess Dilday, David Dodson, Steve Dunleavy, Richard Flanagan, Chris Hislop, Aaron Morneau, the Murphy brothers, Matt Salinder, the Trikakis brothers, Rich Vreeland, and Josh Weatherspoon, among others.

Countless scholars, intellectuals, and colleagues have supported, challenged, questioned, and generally enhanced my research in the past five years. Although each individual deserves a formal thank you letter, for now I will list the people who have most concretely shaped my thinking. Thank you Justin Burton, Mark Butler, Alessandra Campana, Will Cheng, Luis-Manuel Garcia, Stephanie Gunst, Norm Hirschy, Wendy Hsu, Robin James, Loren Kajikawa, Mark Katz, Wayne Marshall, Kiri Miller, Roger Moseley, Ali Neff, Martin Scherzinger, Jason Stanyek, and Jonathan Sterne. I would also like to thank specific UCLA graduate students for sharing their ideas and friendship with me over the years. Thank you Monica Chieffo, Benjamin Court, Wade Dean, Andrew deWaard, Albert Diaz, Oded Erez, Gillian Gower, Allison Hegel, Wendy Kurtz, Craig Messner, Andrea Moore, Tiffany Naiman, Melinda O'Brien, Marissa Ochsner, Alex Rodriguez, Jill Rogers, Arreanna Rostosky, Schuyler Whelden, and Morgan Woolsey.

A number of fellowships and grants from UCLA made this dissertation possible, including the Edwin W. Pauley Fellowship, the Andrew W. Mellon Fellowship of Distinction, the UCLA Dissertation Year Fellowship, and the Herb Alpert Student Opportunity Fund. These funding sources provided me with the travel funds, online server space, and production software that have enhanced the multimodal nature of this project.

To my parents Becky and Scott for the endless support throughout the years. You taught me to find my own way, to be open-minded, and to place the love of friends and family above the material distractions of life. To my brother Chris, you have always been right by my side, especially through the hardest times. If it were not for you buying a drum set so long ago, I would never have pursued a career in music in the first place. To the D'Errico, Henshaw, and Brooks families, I will always be a New Englander at heart, and I thank you for instilling in me the values of hard work, loyalty, respect, and care for one another. To my late father, Gary D'Errico, who always taught me to live life with a smile.

Finally, to my wife Bess. This dissertation is a product of your personal sacrifices as much as my research, and I am eternally grateful for your patience and loyalty in the past five years. You are the strongest person I have ever met, and it is your strength and courage that inspires me continually to be the best possible version of myself. I am so fortunate to be with someone who shares my strange sense of humor and passion for the unknown, and I look forward to a lifetime of adventures with you. Thank you for reminding me of what matters most.

VITA

2009	B.M., Music Education, *summa cum laude* University of New Hampshire
2010-11	Teaching Assistant Department of Music Tufts University
2011	M.A., Music Department of Music Tufts University
2011-12	Pauley Fellowship Department of Musicology University of California, Los Angeles
2012	Friends of Musicology Best Seminar Paper Award and Stipend Department of Musicology University of California, Los Angeles
2012-13	Teaching Associate GE Freshman Honors Cluster Program & Department of Musicology University of California, Los Angeles
2013	Digital Humanities Graduate Certificate Recipient Digital Humanities Program University of California, Los Angeles
2013-14	Research & Instructional Technology Consultant Center for Digital Humanities University of California, Los Angeles
2014-15	Teaching Associate Department of Musicology University of California, Los Angeles
2015-16	Digital Strategist Career Center University of California, Los Angeles
2015-16	Dissertation Year Fellowship Graduate Division University of California, Los Angeles

SELECTED PUBLICATIONS

2016 “Worlds of Sound: Indie Games, Proceduralism, and the Aesthetics of Emergence.” *Music, Sound, and the Moving Image* 9, no. 2.

2015 “Off the Grid: Instrumental Hip-Hop and Experimentalism After the Golden Age.” In *The Cambridge Companion to Hip-Hop,* edited by Justin Williams. New York: Cambridge University Press.

2015 “Electronic Dance Music in the Dubstep Era.” In *Oxford Handbooks Online.* New York: Oxford University Press.

2015 “Bassweight: Dubstep and the Transnational Space of Hip-Hop.” In *Hopeless Youth!* Tallinn, Estonia: Estonian National Museum.

2014 “Going Hard: Bassweight, Sonic Warfare, and the ‘Brostep’ Aesthetic.” *Sounding Out!* January 23, 2014, http://soundstudiesblog.com/2014/01/23/going-hard-bassweight-sonic-warfare-the-brostep-aesthetic/.

SELECTED PRESENTATIONS

2016 “Electronic Music, Computational Aesthetics, and the Californian Ideology.” Poster and workshop presented at the annual meeting of the Society for American Music, Boston, Massachusetts, March 10-13.

2015 “‘Diggin’ in the Carts’: Technologies of Play in Hip-Hop Production and Performance.” Paper presented at the annual meeting for the Society of Ethnomusicology, Austin, Texas, December 3-6.

2015 “Programming Sound: Computational Thinking in Electronic Music.” Paper presented at the fall meeting for the Pacific-Southwest Chapter of the American Musicological Society, Riverside, California, October 10.

2014 “Off the Grid: Self-Effacing Production in Juke and Footwork.” Paper presented at the annual meeting for the American Musicological Society, Milwaukee, Wisconsin, November 6-9.

2014 “Interface.” Invited panelist and performer, “Machine Fantasies: A Workshop on Music Technologies—Past, Present, and Future,” Tufts University, Medford, Massachusetts, April 4-5.

2014 “’Less Work, More Flow’: Embodied Interactivity and the Ecology of Digital Media.” Paper presented at the annual meeting for the International Association for the Study of Popular Music (US), Chapel Hill, North Carolina, March 13-16.

2014 “The Limits of Digitalism: Hypermasculinity and the New Cultures of Control.” Paper presented at the Thinking Gender conference, UCLA Center for the Study of Women, February 7.

2013 “Interface Design and Technical Practice in Digital Audio Production Software.” Paper presented at the BruinTech, “Tech-a-Thon,” UCLA, November 1.

Introduction

Interface Aesthetics

In the 2009 report, "90% of Waking Hours Spent Staring at Glowing Rectangles," satirical news outlet *The Onion* provides an account of the ubiquity of hardware and software devices in the everyday lives of Americans. Most noticeably, the author highlights the pervasiveness of technological objects throughout a variety of daily activities, describing "handheld rectangles, music-playing rectangles, mobile communication rectangles, personal work rectangles, and bright alarm cubes…"[1] While the piece may serve simply as comic relief for a wired generation, all too familiar with navigating their daily grind through the mobility of laptops and smartphones, for others the article identifies anxieties surrounding an emerging digital maximalist lifestyle. Whether we decide to embrace or resist the contemporary pervasiveness of digital media, the piece exposes a fundamental tension in our interactions with the digital world: as material technologies increasingly move to the screen space of software, what creative strategies can be employed to maneuver inherently interconnected and complex modes of practice? In short, how does one interface with media after software?

For many digital artists, technological design and practice in the twenty-first century is defined by the convergence and cross-pollination of media, resulting in the simultaneous centralization of tools and techniques on laptops and other digital workstations, as well as the proliferation of interfaces and peripherals for physically engaging that software. In response to this shift from hardware to software, artists, musicians, and technologists of various sorts have

[1] "Report: 90% Of Waking Hours Spent Staring At Glowing Rectangles," *The Onion,* June 15, 2009, accessed April 29, 2016, http://www.theonion.com/article/report-90-of-waking-hours-spent-staring-at-glowing-2747.

developed new interfaces for interactively engaging and embodying digital materials, from touch-screen devices to programming languages. As sound continues to insert itself within software applications, how have musicians and software designers adapted existing modes of engagement to musical interfaces? Moreover, how has the emergence of multimodal interfaces inculcated new conceptions of what constitutes "sonic" and "musical" material in the digital age? At a fundamental level, we might ask: what is sound after software?

It was this question that brought me to Los Angeles in 2011. At the time, the city experienced a renaissance in two seemingly unrelated cultures, both as a result of developments in media production software: first, hip-hop and electronic dance music, and second, game design. With the passing of beatmaking pioneer J Dilla in 2008, the East Los Angeles experimental club night *Low End Theory* became a hotbed for musicians, video artists, and hip-hop producers. Located far outside of the glitz and glamor of the Hollywood "EDM" scene, *Low End Theory* has always been on the cultural fringe. Most significantly, the club focuses on instrumental hip-hop and dance music performance, showcasing the technological prowess of beatmakers rather than the lyrical poetics of rappers. Indeed, the venue originated as an artists' collective dedicated to the *process* of creating art, and has since mainly attracted "beatheads" interested more in learning techniques from the artists on stage rather than dancing to their music.

While hip-hop practitioners had always admired the perceived physicality of "analog" hardware devices such as drum machines and samplers, the experimental nature of *Low End Theory* led many DJs to embrace software in their stage setups for the first time. As a result, the laptop screen created a literal and metaphorical barrier between performer and audience, encouraging musicians to seek new ways of highlighting the process of music-making for their audiences. Countless debates ensued concerning the “proper” ways in which to perform hip-hop and dance music, with professional musicians and fans alike constructing rhetorical dichotomies between the perceived “authenticity” of “analog” hardware versus the ephemerality of “digital” software. Like the introduction of the turntable to hip-hop thirty years earlier, the emergence of software in DJ culture marked yet another moment in which a new technology negotiated its changing status as a musical instrument.

Meanwhile, a group of students at the University of Southern California were radically reinventing gaming aesthetics. Small-scale releases from the independent game development team *thatgamecompany* consciously moved away from the adrenaline-fueled, hypermasculine ethos of first-person shooters, towards a non-linear experience in which players explore the virtual world on their own terms. Games such as *Flower* (2009) and *Journey* (2012) specifically foregrounded unique gameplay mechanics over narrative, highlighting the process-oriented nature of “indie” video games in the new millennium. Players learned the rules of these games through indirect exploration rather than following strictly predetermined cinematic storylines. Sound design for the games followed suit, aurally guiding the player through the virtual world in the absence of direct visual cues. As the design of the games mimicked the algorithmic nature of software, sound became more than just background noise; it became a gameplay mechanic in itself—a sonic manifestation of the rules embodied in the “hardware” of both game console and player.

For the last five years, I studied these two cultures through interviews with beatmakers, game developers, and media artists, as well as through participant observation as a sound artist, DJ, and game scholar myself. I found that I was surprisingly well-positioned to understand the aesthetics of these communities. In the decade before moving to Los Angeles, I produced beats for rappers, performed experimental multimedia sets, and composed video game music using vintage game consoles. But I only learned so much by focusing on technologies of production from the perspective of a producer. By examining the Los Angeles music and gaming cultures from the perspective of a scholar-practitioner, I realized a few things about the nature of human-computer interaction: first, software is more than just a tool. It's a platform for the expression of individual and collective identities, values, and politics. Second, the relationship between old and new media is not simply based on *remediation*—in which the new refashions the old—but also *accumulation,* as tools and techniques continue to converge in software. Most significantly, I began to understand the interfaces that surrounded myself and my work as more than just control surfaces; they were pivotal components of a saturated media environment that was increasingly altering the practices of artists and media professionals. These insights led to this dissertation.

Overview

Interface Aesthetics examines the aesthetics and technical practices of digital audio producers from multiple, sometimes conflicting perspectives, including musicians, music scholars, sound artists, software developers, and game designers, as well as my own experience working with sound across media. I use the term "interface aesthetics" to describe the ways in which the seemingly disparate processes of technological design and cultural use comingle in the creation of, and in response to, broader social values. In short, I am interested in the ways in which software acts as a metaphorical interface to practices of social and cultural formation more broadly. By studying developments in the design and use of *sound* software, in particular, I detail three shifts that have occurred in music and interactive media production during the early twenty-first century. First, I trace a shift in conceptions of sound from *text* to *experience.*

Through much of the twentieth century, sound recording technologies were designed for the purpose of audio inscription. From the phonograph to magnetic tape and the vinyl record, sound was conceived to be an archival text of sorts, faithfully preserved as it was etched into the media.[2] Thinking of sound in this way has led many practitioners down a seemingly never ending quest for fidelity—constantly seeking out *media* that most perfectly reproduces sound in its impossible, *unmediated* state. From early Victrola advertisements which claimed "the human voice *is* human" on the new phonograph,[3] to the "crystal clear fidelity" of Memorex cassette tapes, the history of sound media in the twentieth century was primarily concerned with capturing sound as an *object*.[4]

By contrast, the emergence of digital tools for sound reproduction after the new millennium has encouraged conceptions of sound as an *experience* rather than an object. The increasing rhetoric of sound "immersion" has altered previous object-oriented fidelity discourses, as in the "4K" television ads which promote "absolute audio clarity" so perfect that you do not simply hear it, but immerse yourself in it.[5] Conceptions of sound in the digital age have also become more abstract, dismissing entirely claims about sonic fidelity. Ambient musical soundscapes, for example, define many of the media experiences discussed throughout this dissertation. In contrast to the "foley" model of sound design, which strives for the faithful

[2] Shane Butler, *The Ancient Phonograph* (Cambridge, MA: The MIT Press, 2015).

[3] "The Human Voice Is Human on the New Orthophonic Victrola," *David Sarnoff Library digital archive*, March 27, 1927, accessed April 29, 2016, http://digital.hagley.org/cdm/ref/collection/p16038coll11/id/12569.

[4] "Is it live or is it Memorex," ukclassictelly, uploaded May 14, 2012, accessed April 4, 2016, https://www.youtube.com/watch?v=lhfugTnXJV4.

[5] While *Interface Aesthetics* is not a historical study, it is a history of the present that has been shaped unavoidably by the work of "media archeologists" such as Erkki Huhtamo. Huhtamo (2013) specifically examines rhetoric of "immediacy" and "immersion" in the nineteenth-century as "topoi" that have reappeared throughout the history of interactive media and technology. *Illusions in Motion: Media Archeology of the Moving Panorama and Related Spectacles* (Cambridge, MA: MIT Press, 2013).

reproduction of sounds found in the "real" world, the most popular "indie" video games and mobile music production apps provide unimposing backdrops that allow users to explore virtual environments removed from the values of sound fidelity.[6]

In addition to the shift in conceptions of sound from *text* to *experience*, I also outline a shift in the structure of media design from linear *narrative* to non-linear, process-oriented *exploration*. This second transformation has to do with how users of a given media platform experience the medium itself, rather than the content or message being conveyed by the platform. In this context, "the medium is the message," following Marshall McLuhan.[7] For example, the advent of Internet webpages and the "World Wide Web" in the mid-1990s introduced non-linear mechanisms for reading text, such as browsers, hyperlinks, and markup languages. In contrast to the linear, goal-oriented structure of paper books, these "hypertextual" forms of web-based media were meant to be explored in a fragmentary manner, the reader allowing him or herself to be distracted by the inherently linked and networked nature of online content.[8]

The shift from *narrative* to *exploration* is equally apparent in the changing trends in music software design. As software companies compete by offering the biggest and best features available to the user, musicians develop new methods for navigating these affordances. While the vast amount of add-on "plugins," constant software updates, and capacities for connecting software with external applications can be overwhelming to some musicians, others embrace the

[6] For more on "foley" audio, see Jim Stinson, "Real-Time Sound Effects: The Foley Way," *Videomaker,* July 1, 1999, accessed April 18, 2016, http://www.videomaker.com/article/7220-real-time-sound-effects-the-foley-way.

[7] Marshall McLuhan, *Understanding Media: The Extensions of Man* (New York: McGraw-Hill, 1964).

[8] Jerome McGann, *Radiant Textuality: Literature After the World Wide Web* (New York: Palgrave Macmillan, 2001).

maximalist nature of music production software as a platform for creative experimentation. In this context, musicians *explore* the interface rather than *use* it for specific purposes or intentions.

Finally, this project contributes to noting a shift in the fundamental understanding of media from *object* to *environment*.[9] Similar to representational models of sound media, software design historically has privileged a fidelity between "real" and "virtual" objects.[10] The history of personal computing has encouraged a conception of the "interface" as the visual and physical surface of a device, simply meant as a direct control mechanism for the user. While software interfaces are often designed to give the impression of "direct manipulation" between human and computer, it is important to recognize that the digital buttons, knobs, and icons with which we interact on a regular basis are just virtual *metaphors* for physical objects.[11]

This "skeuomorphic" design philosophy, in which interface elements are metaphors for "real" objects, has become less prevalent with the advent of mobile media such as smartphones.[12] Instead, abstraction has become a guiding principle in GUI design, thus focusing the computer user's attention on the environment of the operating system as a whole. In direct opposition to

---

[9] Amongst designers and developers, the software "environment" comprises the collection of resources used to support the software application, including the computer operating system, database, and development tools used to build the program. Media theorists have expanded on this integrated, networked understanding of software through the concept of "media ecology"—the study of complex communication systems as environments. See Matthew Fuller, *Media Ecologies: Materialist Energies in Art and Technoculture* (Cambridge, MA: The MIT Press, 2005); Christine Nystrom, *Towards a Science of Media Ecology: The Formulation of Integrated Conceptual Paradigms for the Study of Human Communication Systems,* doctoral dissertation, New York University (1973); Neil Postman, "The Reformed English Curriculum," in *High School 1980: The Shape of the Future in American Secondary Education,* edited by Alvin C. Eurich (New York: Pitman, 1970).

[10] Marianne Van Den Boomen, *Transcoding the Digital: How Metaphors Matter in New Media* (Amsterdam: Institute of Network Cultures, 2014).

[11] The term "direct manipulation" was coined by Ben Shneiderman in "Direct Manipulation: A Step Beyond Programming Languages," *Computer* 16, no. 8 (1983), and has since served as a fundamental concept amongst software designers.

[12] Claire L. Evans, "A Eulogy for Skeuomorphism," *Vice,* June 11, 2013, accessed April 25, 2016, http://motherboard.vice.com/read/a-eulogy-for-skeuomorphism.

the material, object-oriented metaphors that littered Mac and PC "desktops" throughout the history of personal computing in the twentieth century, the launch of operating systems such as Windows 8 and iOS 7 in the early 2000s ushered in an era of "flat" design.[13] Here, abstract shapes and colors replaced desktop metaphors such as "trash" icons and "folders" as overarching design trends to guide the user experience. Consequently, as screens got smaller and interfaces more abstract, the ways in which users interacted with them became more immediate, tangible, and visceral. Mechanical gestures such as "clicking" a mouse and "typing" on a keyboard were gradually replaced by touch-based gestures in which the user directly "tapped," "swiped," and "pinched" the interface itself. The embodied techniques afforded by mobile media thus highlight the ways in which "digital" software, despite its supposed ephemerality and immateriality, still relies on the broader hardware environment surrounding its usage.

As a result of the increasingly material design of software, in which virtual interface elements express the visceral affordances of touch-screen devices, conceptions of both sound and interface have become more fluid, moving away from representational, object-oriented models that seek fidelity.[14] Instead, sound and interface are increasingly understood as comprising an entire media environment that includes hardware and software technologies, as well as the cultural context within which these technologies circulate. The case studies presented throughout this dissertation provide both a critical survey of the ways in which software continues to shape cultural practices in the first two decades of the twenty-first century, and a practical model for the hermeneutic and semiotic analysis of software design as a cultural process rather than a fixed

[13] Amber Leigh Turner, "The history of flat design: How efficiency and minimalism turned the digital world flat," *The Next Web,* March 13, 2014, accessed April 25, 2016, http://thenextweb.com/dd/2014/03/19/history-flat-design-efficiency-minimalism-made-digital-world-flat/.

[14] Google, "Material Design: Introduction," *Google,* n.d., accessed April 25, 2016, https://www.google.com/design/spec/material-design/introduction.html.

"text." Conceptualizing the dynamics of human-computer interaction as a relational process of cultural and technical negotiation becomes especially important as software continues to ingrain itself into the global machine of neoliberal capitalism.

The rise of the "Military-Entertainment Complex" is just one example of the problematic convergence between media platforms and industries as a result of the increasing ubiquity of software. As a result of this convergence, we have seen developments such as video games for military training, the use of video game controllers in piloting military drones, and the integration of global surveillance systems with everyday technologies such as the smartphone. Throughout this dissertation, I encourage designers to cultivate an increased responsibility in conceptualizing ethical forms of human-computer interaction. By balancing critical and practical approaches to software design, *Interface Aesthetics* attempts to engage simultaneously with the concrete creative practices of musicians and software designers, as well as the ideological questions surrounding human-computer interaction more broadly.

Background

This dissertation expands on insights generated by scholars and practitioners in two subfields of music and media studies: sound studies and software studies. With the emergence of the interdisciplinary field of sound studies, scholars have pushed discussions of sonic engagement outside of strictly musical practices, asking questions related to the history and nature of sonic experience. Jonathan Sterne, for example, provides a synchronous account of developments in media such as telephony, phonography, and the radio, as well as the modern practices of medicine, physics, and industrial capitalism, thus detailing the technological preconditions for the emergence of a more integrated sound *culture*.[15] Although Sterne is partly concerned with sound's *object-oriented* ability to be recorded, inscribed, and archived in various ways, he conceives of listening to and producing sound primarily as experiential *processes* that help in shaping ongoing relationships between humans and technology.

Indeed, sound has come to play a major role in defining relational processes between society, culture, and the natural world. In effect, sound scholars have conceptualized sound itself as an interface of sorts. Stefan Helmreich uses the notion of "transduction" to describe the ways in which sound metaphorically transmutes and converts as it moves across media, similar to the ways in which sound vibrations are transduced into energy by the inner ear.[16] Transduction describes more than just the physical conversion of energy forms. Rather, the concept characterizes the entire relational network surrounding the sonic interaction, from the technical design of the system (sound, image, and mechanics of the media interface) to the aesthetic and ethical understandings of sound at stake in the listener's experience.[17] Alexandra Supper and others have examined the social and cultural consequences of sonification technologies that render scientific data into sound, as well as the ways in which devices such as the stethoscope "audify" bodily phenomena.[18] Similarly, professional sound designers working in the field of Sonic Interaction Design have developed strategies for using sound as a cognitive and embodied

[15] Jonathan Sterne, *The Audible Past: Cultural Origins of Sound Reproduction* (Durham and London: Duke University Press, 2003).

[16] Stefan Helmreich, "An anthropologist underwater: Immersive soundscapes, submarine cyborgs, and transductive ethnography," *American Ethnologist* 34 no. 4 (2007): 621-41.

[17] Adrian Mackenzie defines transduction as "a way of theorizing and figuring things primarily in terms of relationality, as processes of recontextualization and in terms of generativity." "Transduction: Invention, Innovation and Collective Life," accessed April 19, 2013, http://www.lancs.ac.uk/staff/mackenza/papers/ transduction.pdf.

[18] Trevor Pinch and Karin Bijsterveld, Introduction to *The Oxford Handbook of Sound Studies,* edited by Trevor Pinch and Karin Bijsterveld (New York: Oxford University Press, 2011), 5.

cue in the experience of non-"musical" media.[19] In each case, sound is a dynamic process of relationality *between* social, cultural, and technological agents, rather than a fixed consumer product that is controlled by a "user."

If the goal of sound studies is to expose the sociotechnical and historical infrastructures that shape the production and reception of sound, media theorists working under the rubric of "software studies" are interested in pinpointing the technical structures that shape the design and use of software. In *10 PRINT CHR$(205.5+RND(1));:GOTO 10,* Nick Montfort uses a single line of code from the Commodore 64 computer as an analytical lens in order to consider the cultural impact of software.[20] Lev Manovich's *Software Takes Command* examines media production software such as the Adobe *Creative Suite* to highlight the ways in which software techniques become established creative practices across a range of cultural communities.[21] Following in the footsteps of classic media formalists such as Friedrich Kittler and Marshall McLuhan, software studies scholars engage in hermeneutic and semiotic analyses of software programs as a means of answering more expansive questions related to the social, political, and economic effects of software, as well as the ways in which changes in software design affect cultural practice.[22]

Foundational texts in software studies deal broadly with the relationship between software programs and cultural practice—that is, the interface between technical *codes* and their

---

[19] Karmen Franinović and Stefania Serafin, eds., *Sonic Interaction Design* (Cambridge, MA: The MIT Press, 2013).

[20] Nick Montfort, *10 PRINT CHR$(205.5+RND(1));:GOTO 10* (Cambridge, MA: The MIT Press, 2013).

[21] Lev Manovich, *Software Takes Command* (New York and London: Bloomsbury, 2013).

[22] Marshall McLuhan, *The Medium is the Massage: An Inventory of Effects* (Corte Madera, CA: Gingko Press, 2001); Friedrich Kittler, *Gramophone, Film, Typewriter,* trans. Geoffrey Winthrop-Young and Michael Wutz (Stanford, CA: Stanford University Press, 1999).

realizations as social *forms.*[23] Recent scholarship in "interface criticism" extends the focus from the inner workings of software to the more user-oriented elements of HCI—the interface itself.[24] Summarizing what he terms "the interface effect," Alexander Galloway states, "the computer is not an object, or a creator of objects, it is a process or active threshold mediating between two states."[25] This relational ontology has become the dominant theme in "interface studies." Media theorists Branden Hookway and André Nusselder frame HCI through the positionality of "in-betweenness," in which the foundational experience of technological mediation comes from the active process of mediating between the two states of reality and virtuality.[26] While interface studies serves as a crucial body of research in highlighting the ontological and phenomenological experience of mediation, it has little to say about the formal and aesthetic aspects of technological experience in practice, as well as the multimodal experiences encouraged by "interactive" music and sound.

In a similarly critical gesture, design research professionals have increasingly shifted their language from the object-oriented nature of *interface* to the process-oriented nature of *interaction.* Designer Brenda Laurel claims that, in focusing on what the *computer* is doing, "interface designers are engaged in the wrong activity." Instead, she suggests that we throw out

[23] David Berry, *The Philosophy of Software: Code and Mediation in the Digital Age* (New York: Palgrave Macmillan, 2011); Wendy Hui Kyong Chun, *Programmed Visions: Software and Memory* (Cambridge and London: The MIT Press, 2011); Matthew Fuller, ed., *Software Studies: A Lexicon* (Cambridge, MA: The MIT Press, 2008); Casey Reas and Chandler McWilliams, *Form+Code in Design, Art, and Architecture* (New York: Princeton Architectural Press, 2010); Noah Wardrip-Fruin, *Expressive Processing: Digital Fictions, Computer Games, and Software Studies* (Cambridge and London: The MIT Press, 2009).

[24] Christian Ulrik Andersen and Søren Pold (eds.), *Interface Criticism: Aesthetics Beyond Buttons* (Aarhus, DK: Aarhus University Press, 2011).

[25] Alexander Galloway, *The Interface Effect* (Malden, MA: Polity Press, 2012), 23.

[26] Branden Hookway, *Interface* (Cambridge, MA: The MIT Press, 2014); André Nusselder, *Interface Fantasy: A Lacanian Cyborg Ontology* (Cambridge, MA: The MIT Press, 2009).

the term "interface" altogether, since the main focus of media design should be "what the *person* is doing with the computer—the action."[27] Laurel's work, along with that of pioneering design guru Donald Norman, has had a direct influence on the emergence of User-Experience (UX) and Interaction Design as disciplines that understand media interfaces as process-oriented "experiences" rather than material products.[28] Janet Murray expands on this work by acknowledging the significance of the social and cultural context in any technological interaction. For Murray, "interactors" (rather than "users") are engaged "with one another through the mediation of the machine, and with the larger social and cultural systems of which the automated tasks may only be a part," thus reminding designers that the mere "usefulness" of an interface is less important than its context.[29] Together, these developments in scholarship and professional practice encourage changing conceptions of the term "interface" itself.

In contrast to *object*-oriented definitions, I suggest thinking about interface as a *process* that encourages ludic forms of creativity, the convergence of tools and techniques across media, and a relational ethic of technological interaction. I argue for conceiving of interface as process because it allows us to get beyond current understandings of technological interaction as being rooted in either the phenomenological experience of "user" control over technological materials, or the technical underpinnings of the material technology itself.[30]

---

[27] Brenda Laurel, *Computers as Theatre* (Boston, MA: Addison-Wesley, 1993), 116.

[28] See also Donald Norman, *The Design of Everyday Things* (New York: Basic Books, 1988); Jacob Gube, "What is User Experience Design? Overview, Tools and Resources," *Smashing Magazine,* October 5, 2010, accessed April 1, 2016, http://uxdesign.smashingmagazine.com/2010/10/05/what-is-user-experience-design-overview-tools-and-resources/.

[29] Janet Murray, *Inventing the Medium: Principles of Interaction Design as a Cultural Practice* (Cambridge, MA: The MIT Press, 2012), 11.

[30] With the emergence of the "user-experience design (UX)" philosophy among interface designers, there has been a general trend toward understanding technological interaction as an "experience" rather than a material product. The roots of this idea can be found in Norman (1988).

By conceptualizing interface as a two-way relationship between human and technology, the term "aesthetics," as used in the title of the dissertation, similarly takes on a new meaning. Originally coined by German philosopher Alexander Baumgarten in 1750, the term aesthetics aligned with the original Greek word for "sensory perception," dealing with not only art *objects* but the *process* of human perception in its entirety.[31] While philosopher Immanuel Kant's notion of the term as a "disinterested" withdrawal from social, political, and bodily dictates of value judgment remains the most influential conception of aesthetics, interfaces complicate this objective understanding of the term.[32] On one hand, the use of human-computer interfaces is an intimately visceral and sensual practice that foregrounds the inner structures and mechanics of the technological system. Arguably, the main reason an interface such as the computer keyboard has been successful lies in the intuitive ways in which it externalizes the text-based nature of digital code, making tangible the ephemeral nature of software. On the other hand, many interfaces are designed to hide these very structures through flashy screen layouts and novel interaction patterns. Mobile media interfaces such as the iPhone touch screen, for example, are designed to foreground the physical gestures of the device's user, rather than the "rules" of the software code. As such, interfaces are far from "disinterested" agents in the shaping of aesthetic judgment and sensory perception in the digital age.

Since the interface plays such a pivotal role in the shaping of contemporary aesthetic values, the pervasive nature of interfaces in the early twenty-first century calls for a similarly integrated definition of aesthetics. Throughout this dissertation, I follow Terry Eagleton's conception of the term, recognizing that to judge aesthetically is to compare values, and that

[31] Marcia Muelder Eaton, *Basic Issues in Aesthetics* (Belmont, CA: Wadsworth Publishing, 1988).

[32] Immanuel Kant, *Critique of Judgment,* trans. Werner S. Pluhar (Indianapolis, IN: Hackett, 1987).

those values emanate from the totality of the judge and his or her context.[33] In this way, aesthetic judgment is at once a political and ethical act capable of radically transforming existing conceptions of art, culture, and society. The title of this dissertation, *Interface Aesthetics,* thus combines a process-oriented definition of "interface" with a more holistic, contextualized definition of "aesthetics," referring to the processes through which software users negotiate new forms of technological interaction as part of their social and cultural identities more broadly. Further, "interface aesthetics" refers to a scholarly method employed throughout this dissertation in which the technical structures of software are analyzed hermeneutically and semiotically in order to reveal social, cultural, and ideological structures embedded within aesthetic design.

Specifically, my research asks this question: as sound continues to become enmeshed in software, how do sound artists adapt existing embodied practices to constantly changing interfaces? Moreover, how has the emergence of alternative interfaces ushered in new conceptions of sound in the digital age? Thinking of twenty-first century media production as an ecology has significant implications for conceptions of sound. Building on Nina Sun Eidsheim's work on sound as an intermaterial and relational process, I understand sound *software* as an interface for "vibrational" transmissions that connect computer users to their social, material, and cultural contexts.[34] The idea of sound as a relational *process*—that is, as interface—introduces the more drastic notion that music is no longer "Music" as it has been defined traditionally, but a node within a broader media ecology. Indeed, as sonic ontologies increasingly become digitized, shifting from objects to processes, listening itself becomes rewired. As the examples from *Low*

[33] Terry Eagleton, *The Ideology of the Aesthetic*, 2nd Ed. (Cambridge, MA: Basil Blackwell, 1981). An overview of Eagleton's concept can be found in Carol Gigliotti, "Aesthetics of a Virtual World," *Leonardo* 28, no. 4 (1995), 290.

[34] Nina Sun Eidsheim, *Sensing Sound: Singing and Listening as Vibrational Practice* (Durham, NC: Duke University Press, 2015).

End Theory and "indie" gaming make clear, these ontological shifts occur in parallel with new cultural formations. In this context, consumers of digital music and media are not just passive listeners, but approach artistic consumption from the perspective of makers themselves. Throughout the dissertation, I refer to this new mode of HCI as *procedural listening.*

Methodology

In order to examine the interrelated and mutually dependent relationships between software design and the creative practices surrounding digital audio production across media, *Interface Aesthetics* engages multiple methodologies, including print and online discourse analysis, analyses of software and hardware devices, participant observation, and insights from practice-based research, through my work as a hip-hop producer, electronic dance music DJ, and digital media artist working with sound. These sets of inquiries have been defined by the contacts and collisions between bodies and material technologies; connections and disconnections between technological devices; and the fluid web through which tools, techniques, and practices spread when sound is encoded in software.

I chose these methods based on my fundamental position that new insights can be gained by thinking through the coterminous development between forms of analysis and forms of creative practice. Ultimately, the convergence of technical analyses, ethnographic insights, and practical experience comprises the major contribution of this project. In contrast to ethnomusicological monographs which use ethnographic methods to describe the specific practices of individual people or cultural communities, this dissertation employs interview and participant observation methods for the purpose of understanding broad, collective knowledge surrounding digital media production between the years 2000 and 2016. As a result, a majority of the "data" presented in the text comes from formal, hermeneutic, and aesthetic analyses of software, rather than direct quotes from ethnographic collaboration. The insights generated from these analyses are supported by the collective insights gathered through over ten years of creating, performing, and working with the cultural communities under discussion.[35] The pervasive emphasis I place on *design* as an intermediary between production and consumption is meant to highlight further the process-oriented nature of the term "interface aesthetics"—a general framework for thinking through relationships between theory and practice, concept and technique, aesthetics and poietics. I will now discuss the benefits and limitations of each individual method.

Hermeneutic analyses of software applications provide the bulk of my research data. The technical practice of designing possibilities for human interaction with technological systems has been broadly defined, with subfields including user interface design (UI), human-computer interaction (HCI), and user experience (UX), among others.[36] I develop a methodology for analyzing interface design that combines multiple practices within these fields. First, I examine the visual design, or "information architecture," of software and hardware interfaces, focusing on how the organization of buttons, knobs, sliders, and other control mechanisms creates a system

---

[35] The sound ethnographies of Steven Feld and other scholars working under the rubric of "sensory ethnography" have been particularly useful in demonstrating alternative ethnographic methods more appropriate to my stance as a scholar-practitioner. See, for example, Alex Rhys-Taylor, "The Essences of Multiculture: A Sensory Exploration of an Inner-City Street Market," *Identities: Global Studies in Culture and Power* 20, no. 4 (2013): 393-406; Sarah Pink, *Doing Sensory Ethnography* (London: Sage Publications, 2009); Tim Ingold, *The Perception of the Environment* (London: Routledge, 2000).

[36] Steve Krug, *Don't Make Me Think: A Common Sense Approach to Web Usability* (Upper Saddle River, NJ: Pearson Education, 2000); Jenifer Tidwell, *Designing Interfaces: Patterns for Effective Interaction Design* (Sebastopol, CA: O'Reilly Media, 2005); Jef Raskin, *The Humane Interface: New Directions for Designing Interactive Systems* (Boston, MA: Addison-Wesley Professional, 2000); Russ Unger and Carolyn Chandler, *A Project Guide to UX Design: For User Experience Designers in the Field or in the Making* (Upper Saddle River, NJ: Pearson Education, 2009); Julie A. Jacko, *The Human-Computer Interaction Handbook,* 3rd edition (Boca Raton, FL: CRC Press, 2012).

of affordances and constraints related to the usability of a given device. Then, I detail the ways in which the visual form of the interface—combined with the "back end" logic of the software code—suggests certain models for sonic and compositional form. Finally, I consider the ways in which the design of specific interface models encourages novel forms of performance. Together, these analyses broadly define the "information architecture" of contemporary interfaces. For designers, this approach may enhance the critical power of the work with which they are engaged. For scholars, this approach offers a framework for understanding the relationship between the creation of a work and its broader cultural context—a type of "sketch studies" for the digital age.[37]

Discourse analysis is equally useful in ascertaining the connections between *how* producers technically engage digital audio tools (poietics), and *why* particular practices become standardized over others (aesthetics). I examine the marketing materials for various music software; debates taking place in web forums, Facebook groups, Twitter lists, and other online communities of audio producers; and pedagogical materials, including YouTube video tutorials and trade publications. With the rise of what Kiri Miller calls "amateur-to-amateur" learning, marked by an increasing proliferation of both tools and techniques through virtual networks, discourse analyses allow me to detail emerging trends, rhetoric, and aesthetics of audio production across communities of practice.[38] Furthermore, the combined approaches to discourse

[37] Friedemann Sallis and others have posed the question of what sketch studies can offer to the study of recorded and digital music. Friedemann Sallis, ed., *Music Sketches* (Cambridge, UK: Cambridge University Press, 2015). The work of Laura Zattra, for example, attempts to reconstruct early computer music that was created using what is now obsolete software. Laura Zattra, "The identity of the work: agents and processes of electroacoustic music," *Organised Sound* 11, no. 2 (2006): 115-6. See also, Oliver Baudouin, "A reconstruction of *Stria,*" *Computer Music Journal* 31, no. 3 (2007): 75-81.

[38] Kiri Miller, *Playing Along: Digital Games, YouTube, and Virtual Performance* (New York: Oxford University Press, 2012) is exemplary in the model it creates for conducting "virtual" discourse analyses.

analysis from Michel Foucault and Friedrich Kittler offer useful paradigms for exposing the ideological conditions that give rise to various media practices.[39]

An ethnographic approach further grounds my technical and discursive methods, and Southern California—what many hail as "the next Silicon Valley"—has been a particularly fertile ground for this project.[40] Participant observation within various electronic music and digital media scenes—focused in Los Angeles, and extended virtually through my online network of DJs and producers—influenced the bulk of my research insights, as I have attended and participated in countless music performances, game festivals, technology conferences, and media workshops since coming to Los Angeles in 2011. While these close observations offer unique insights into the public performances and communal aesthetics of various subcultures, interviews with electronic musicians, game designers, and software developers provide more detailed microscopic views on the connections between the technical aspects of audio production in the studio, and the aesthetic aspects of social and cultural production within digital culture more broadly. This combination of macro and micro approaches builds on Sherry Ortner's practice of "interface ethnography," in that it not only deals with events and practices in which a subculture *interfaces* with a given public, but also with the ways in which the technological design *interfaces* with both technical practices of audio production and aesthetic aspects of cultural formation in the digital age.[41]

---

[39] Friedrich Kittler, *Discourse Networks 1800/1900* (Stanford, CA: Stanford University Press, 1990); Michel Foucault, *The Archeology of Knowledge*, trans. R. Sheridan (New York and London: Routledge, 1972).

[40] Bruce Upbin, "Why Los Angeles Is Emerging As The Next Silicon Valley," *Forbes,* August 28, 2012, accessed April 29, 2016, http://www.forbes.com/sites/ciocentral/2012/08/28/why-los-angeles-is-emerging-as-the-next-silicon-valley/.

[41] Sherry Ortner, *Not Hollywood: Independent Film at the Twilight of the American Dream* (Durham and London: Duke University Press, 2013), 26.

Finally, this study is unavoidably shaped by my perspectives as a music producer navigating constantly shifting trends in sound, software, and digital audio. Since 2005, I have produced music and sound art as a hip-hop beatmaker; a "chipmusic" artist who circuit bends, recombines, and performs with vintage computer technologies; a composer for films and interactive media; and a DJ. The idea of thinking *through* practice—or, what UCLA Design Media Arts professor Peter Lunenfeld describes as "the maker's discourse"—has gained traction across disciplines, from new media theory and the digital humanities to musicology.[42]

In her foundational work, *Boccherini's Body*, Elisabeth Le Guin describes how the kinesthetic, proprioceptive, and tactile aspects of musical performance encouraged a revelation in her methods of musicological analysis, as she writes: "…even as the centrality of a visual listening was becoming evident to me, I was increasingly convinced that certain qualities in Boccherini's music were best explained, or even solely explicable, through the invisible embodied experiences of playing it."[43] In coining the term "cello-and-bow thinking" to describe a mode of analysis which attempts to draw affective meaning from nuances in the physical gestures and muscular distribution of musical performance, Le Guin provides a useful phenomenological method for scholars attempting to combine theoretical and technical aspects of musical practice.

Digital Humanists often frame this discussion around the tension between "hacking" (making/ practicing) and "yacking" (theorizing/ conceptualizing), arguing that contemporary

---

[42] Peter Lunenfeld, "Mia Laboro: Maker's Envy and the Generative Humanities," Book Presence in a Digital Age Conference, University of Utrecht, 2012; "The Maker's Discourse," Critical Mass: the Legacy of Hollis Frampton, Invited Conference, University of Chicago, 2010.

[43] Elisabeth Le Guin, *Boccherini's Body: An Essay in Carnal Musicology* (Berkeley and Los Angeles: University of California Press, 2006), 5.

modes of humanistic research must find a productive balance between the two.[44] As Anne Burdick, Johanna Drucker, Peter Lunenfeld, Todd Presner, and Jeffrey Schnapp argue in their collaborative manifesto, *Digital_Humanities,* "Digital Humanities is a production-based endeavor in which theoretical issues get tested in the design of implementations, and implementations are loci of theoretical reflection and elaboration."[45] With this in mind, I see *Interface Aesthetics* as an experimental lab in which I can develop new tools and techniques for the "close reading" of sound in the contemporary multimedia context. To that end, I have created a companion website to this dissertation, interfaceaesthetics.com, in which I will continue to curate resources and tutorials on digital audio production, as well as host podcasts and interviews with digital designers and musicians.[46] In combining critical research and practice, I address the ways in which the tools and techniques of music analysis are intimately bound to the tools and techniques for music production.

Chapter Organization

Rather than providing a cultural and technical overview of the historical development of software, *Interface Aesthetics* instead presents a specific transitional moment in the early twenty-first century in which software became the primary platform through which artists engage with sound and media. The dissertation is organized into three sections. The first section, "Sonic

[44] "Digital Humanities is an applied field as well as a theoretical one, and the task of applying these metaconsiderations puts humanists' assumptions to a different set of tests. It also raises the stakes with regard to outcomes. Theoretical insight is constituted in this field in large part through encounters with application." Johanna Drucker, *SpecLab: Digital Aesthetics and Projects in Speculative Computing* (Chicago: The University of Chicago Press, 2009), 6.

[45] Burdick, et al., *Digital_Humanities* (Cambridge, MA: The MIT Press, 2012), 13.

[46] Mike D'Errico, "Interface Aesthetics," accessed April 25, 2016, http://www.interfaceaesthetics.com.

Architectures," discusses the ways in which the design of music production software reflects broader cultural aesthetics. Section two, "When Software Becomes Hardware," examines how the affordances of music software are materialized in both live performances with technology and everyday embodied interactions with hardware such as the iPhone. The final section, "Listening to Software," discusses the ways in which the emergent aesthetics of sound design in "indie" video games suggest a more dynamic relationship between the technical aspects of game design and the aesthetics of the player experience. Ultimately, the dissertation traces the conceptual shift from a tool-based understanding of HCI to a process-oriented and relational awareness of digital interfaces as active agents in the shaping of ethical forms of interaction with technology.

The first section of the project, "Sonic Architectures," deals with the ways in which the graphical user interfaces (GUI) of music software programs shape conceptions of foundational musical concepts such as composition and instrumentality. Like most software in the history of personal computing, the design of music production software has been guided by a dichotomy between "usability" and "hackability." On one hand, there are popular commercial software programs that simulate existing tools and techniques in order to make the application more "user-friendly" to a broader demographic of consumers. On the other hand, there are open-source, and often free, software programs whose source code is public and open to custom modifications from a community of "do-it-yourself" programmers. These disparate approaches to software development entail broader socioeconomic values, political ideologies, and cultural aesthetics. By analyzing the contrasting approaches to GUI design in software such as Ableton *Live* and *Max,* chapters one and two reveal the values, aesthetics, and ideologies fixed within music production software.

Chapter one focuses specifically on what is arguably the most popular music production software introduced to date: Ableton *Live*. In the mid-2000s, Live became the dominant software for electronic music producers working across genres. Designed for "real-time" musical performance on stage and in the studio, *Live* introduced unique affordances for engaging digital sound, such as a non-linear "Session View" which allows for the modular juxtaposition of musical ideas, as well as increased interoperability between *Live* and other media. Through design analyses of the *Live* software interface, as well as discourse analysis among producers and designers, chapter one details the ways in which the design of music software is moving away from the representation of pre-existing instruments, toward more abstract interfaces for engaging digital sound. I contextualize the aesthetic desire among producers to integrate a vast array of media and technologies into their creative workflow within the broader technological milieu of digital maximalism—a philosophy which says that the more connected you are, the better. In doing so, I answer fundamental questions regarding the multimodal nature of musical composition with the emergence of software.

Chapter two aligns the practices of musical composition in *Max* and *Pure Data* with the technical skills and design aesthetics of computer programmers, in an attempt to outline a theoretical model for computational thinking in electronic music. While *Live* has appealed to popular music and dance music producers because of the influx and accessibility of creative options presented by the interface, other music software embodies minimalist aesthetics, fostering creativity through limitations in design. *Max*, for example, is rarely described as software at all, but rather a musical "environment." As the visual and algorithmic nature of the program suggests, *Max* requires composers to embrace an alternative digital literacy when working with the software: one more akin to computer coding than traditional music composition. Outlining perspectives on the practice of coding from software studies, computer science, and the individual aesthetics of software developers—including those of original *Max* developer Miller Puckette—I present significant insights on the nature of composition and instrumentality in the digital age. Most noticeably, computational thinking encourages artists to think increasingly through the lens of designers: crafting entire systems rather than individual "works," and developing proficiencies in media formats and techniques that continue to converge in the screen space of software.

The second section of the project, "When Software Becomes Hardware," examines the ways in which hardware controllers have allowed electronic musicians to physically embody the gestural affordances of software, thus establishing a feedback loop between the body of the performer and the computational system. In recent years, touch-screen technologies have become the dominant modes of engagement with digital media, particularly in the realm of music production. The rise of mobile "apps" for creating music, kinesthetic control schemes in music-based video games, and an increasing abundance of hardware peripherals for controlling sound in digital audio workstations has coincided with the rise of "accessibility" and "tangibility" as rhetorical metaphors of control in interactive media. While digital audio production tools are often marketed for their so-called "democratizing" capabilities, it is exactly through this rhetoric of accessibility that these "controllers" make sense in the era of digital convergence. "Touch" becomes a tangible metaphor for the desire of non-mediation and connectivity that mobile social media, video games, and digital audio production thrive for, but continuously fail to achieve.

Chapter three is a case in point. Throughout the history of hip-hop and electronic dance music, the process of mixing and manipulating vinyl records between two turntable decks has become standard practice for DJs, imbuing the performance with a sense of improvisational spontaneity, and enhancing the perceived “live” presence of the DJ. However, the increasing presence of digital software in the DJ booth has thrown into question the nature of performance within electronic dance music communities. In an attempt to heighten the sense of physicality and direct manipulability when working with seemingly intangible software, producers and DJs have increasingly integrated button-based hardware “controllers” into their creative workflows. Combining design analyses of hardware devices such as the Monome “grid” controller and Ableton *Push* with analyses of performance techniques from popular electronic musicians such as Daedelus, chapter three posits the shift from “turntablism” to “controllerism” as exemplary of trends towards the emergence of a “controller culture” more broadly.

Chapter four examines the Apple iPhone as a device whose “user-friendly” design and seemingly intuitive touch-screen mechanics equalize the skill levels needed for both everyday productivity tasks and creative music production. In contrast to controllerists who use hardware controllers to distinguish themselves from commonplace users of technology, iPhone users celebrate the disintegrating distinction between expert “producers” and non-expert “users” facilitated by mobile media devices. The line between creative production and media consumption is further blurred with the integration of the *App Store*, an online shop in which users can purchase both new apps and add-on content for existing apps on the iPhone. In analyzing the ways in which mobile media software and app design “democratizes” music production practices, it is possible to understand music more broadly as comprising not simply *material* technologies (instruments, controllers) and traditional performance spaces (dance clubs, concert halls), but also a process-oriented *experience* that aligns with consumption practices inherent to capitalism in the early twenty-first century.

The final section of the project, "Listening to Software," examines the emergent and procedural aesthetics in video game audio, as captured by the following quote:

> Things which grow shape themselves from within outwards—they are not assemblages of originally distinct parts; they partition themselves, elaborating their own structure from the whole to the parts, from the simple to the complex.[47]

I read this description of organic emergence, by Alan Watts, not while perusing treatises on Zen Buddhism, or scanning self-help manuals for spiritual guidance, but rather in the opening sentences of Matt Pearson's practical guide to using the *Processing* programming language for creating digital art.[48] The conflation of computer code and algorithmic processes with organic and holistic metaphors is commonplace in many contemporary digital art scenes, and has a long history going at least as far back as the multimedia experiments of the 1960s avant-garde. How have the aesthetics of what has been broadly labeled "generative media" affected forms of digital audio production more generally? The final chapter of *Interface Aesthetics* hints at the ethical implications of examining software as a process-oriented "experience" rather than a fixed commodity.

Chapter five examines the application of generative music to emergent media experiences, thus opening up new forms of human-computer interaction—what I define as "procedural interfaces"—through an examination of popular "indie" games such as *Proteus* (2013) and *Fract OSC* (2014). In a 1996 talk, "Generative Music," Brian Eno describes the principle that formed the basis of his philosophy of ambient music: "the idea that it's possible to think of a system or a set of rules which once set in motion will create music for you."[49] With the

[47] Alan Watts, *Nature, Man, and Woman* (New York: Vintage Books, 1991).

[48] Matt Pearson, *Generative Art: A Practical Guide Using Processing* (Shelter Island, NY: Manning Publications, 2011).

[49] Brian Eno, "Generative Music," *In Motion Magazine,* July 7, 1996, accessed December 11, 2014, http://www.inmotionmagazine.com/eno1.html.

rise of procedural generation in video game design, and other forms of computer-generated media, this desire for self-generating "environments" would seemingly materialize in the multisensory space of video games. Extending a historical lineage of "generative aesthetics" throughout the twentieth century, I define the concept of emergence both technically and discursively, analyzing practices of sound design in the *Unity3D* game design software in conjunction with the theoretical and aesthetic motivations of game and sound designers themselves. Expanding on Eno's idea, procedural interfaces encourage dynamic, relational modes of technological engagement in which sonic interaction design guides the players through the virtual world, rather than a set of rules imposed from the designers themselves.

In the end, each chapter provides a case study in what I call "procedural listening"—a form of technological interaction in which the user focuses on the process-oriented mechanics of the technological system, rather than the content created by the user. In facilitating a relational, two-way dynamic between process-based systems and human interactors, the environment provided by software encourages users to be more than just *users*: it inspires them to be *makers.* Yet, while many have rightly celebrated the participatory, "democratized" nature of digital media, the situation also calls for an increased attention to the ethics of human-computer interaction and an awareness of social responsibility on the part of both software designers and users. In this way, the fundamental goals of *Interface Aesthetics* are both critical and practical. For scholars and researchers, the project aims to provide both a framework for analyzing software design as a cultural practice, and a critical understanding of the increasing role software continues to play in the social, economic, and cultural lives of Western consumers. For designers and digital media artists, the project offers suggestions for moving beyond traditional creative techniques and practices, and experimenting with new software interfaces as critical tools in themselves. By the time you finish reading this dissertation, a whole new set of software programs, digital interfaces, and media platforms will have been released, simultaneously extending and subverting traditional forms of creativity. In the midst of today's perennial upgrade culture, learning how to critically navigate emerging media and technologies becomes relevant now more than ever.

Section I

Sonic Architectures

The first section of this dissertation considers the ways in which the graphical user interfaces (GUI) of music software programs shape conceptions of foundational musical concepts such as composition and instrumentality. Like most software in the history of personal computing, the design of music production software has been guided by a dichotomy between "usability" and "hackability." On one hand, there are popular commercial software programs that simulate existing tools and techniques in order to make the application more "user-friendly" to a broader demographic of consumers. On the other hand, there are open-source, and often free, software programs whose source code is public and open to custom modifications from a community of "do-it-yourself" (DIY) programmers. These disparate approaches to software development entail broader socioeconomic values, political ideologies, and cultural aesthetics. By analyzing the contrasting approaches to GUI design in software such as Ableton *Live* and *Max,* chapters one and two reveal the values, aesthetics, and ideologies fixed within music production software.

Chapter one focuses specifically on what is arguably the most popular music production software introduced to date: Ableton *Live*. In the mid-2000s, *Live* became the dominant software for electronic music producers working across genres. Designed for "real-time" musical performance on stage and in the studio, Live introduced unique affordances for engaging digital sound, such as a non-linear "Session View" which allows for the modular juxtaposition of musical ideas, as well as increased interoperability between *Live* and other media. Through design analyses of the *Live* software interface, as well as discourse analysis among producers and designers, chapter one details the ways in which the design of music software is moving away from the representation of pre-existing instruments, toward more abstract interfaces for engaging digital sound. I contextualize the aesthetic desire among producers to integrate a vast array of media and technologies into their creative workflow within the broader technological milieu of digital maximalism—a philosophy which says that the more connected you are, the better.

Chapter two aligns the practices of musical composition in *Max* and *Pure Data* with the technical skills and design aesthetics of computer programmers, in an attempt to outline a theoretical model for computational thinking in electronic music. While *Live* has appealed to popular music and dance music producers because of the influx and accessibility of creative options presented by the interface, other music software embodies minimalist aesthetics, fostering creativity through limitations in design. *Max*, for example, is rarely described as software at all, but rather a musical "environment." As the visual and algorithmic nature of the program suggests, *Max* requires composers to embrace an alternative digital literacy when working with the software: one more akin to computer coding than traditional music composition. Outlining perspectives on the practice of coding from software studies, computer science, and the individual aesthetics of software developers—including those of original *Max* developer Miller Puckette—I present significant insights on the nature of composition and instrumentality in the digital age. Most noticeably, computational thinking encourages artists to think increasingly through the lens of designers: crafting entire systems rather than individual "works," and developing proficiencies in media formats and techniques that continue to converge in the screen space of software.

Chapter 1

Plugin Cultures: The Digital Audio Workstation as Maximal Interface

In 1990, Palo Alto-based software company Opcode released *Studio Vision,* the first software to integrate audio editing with digital MIDI notation, therefore merging the previously disparate tools of musical instruments and personal computers. By employing a simple graphical user interface (GUI) that arranged both audio waveforms and digital music notation along a linear timeline, *Studio Vision* responded to a popular desire among musicians seeking to compose recorded music using nothing but their personal computers and a $995 software program (fig. 1.1).[50] While *Studio Vision* established an enduring design layout for contemporary digital audio workstations (DAWs)—software applications for music recording and audio editing—the simplicity and purposeful functionality of the software is a far cry from the overwhelming complexity that would emerge in the design of DAWs just two decades later.

[50] Taking inflation into account, $995 in 1990 had the same buying power as about $1800 in 2016. In contrast, the average price of a full-fledged DAW in 2016 falls between $200 and $500.

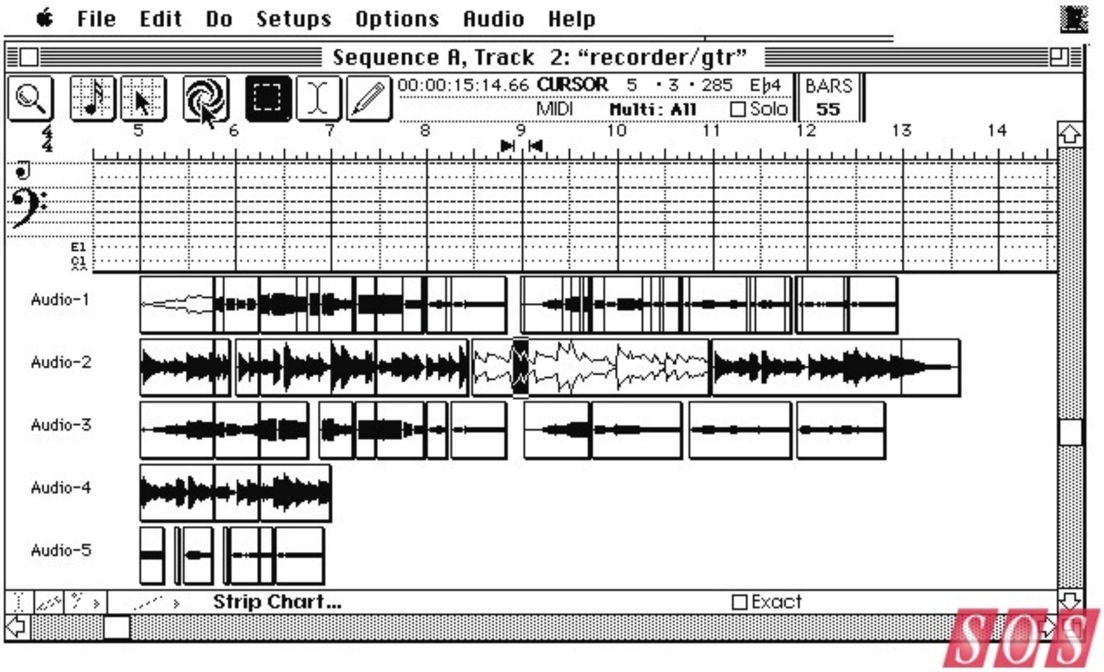


Figure 1.1: Opcode's *Studio Vision* (1990), the original digital audio workstation (DAW) (Source: "25 Products That Changed Recording," *Sound on Sound,* November 2010, accessed April 25, 2016, http://www.soundonsound.com/sos/nov10/articles/25-milestone-products.htm)

In the twenty years following the release of *Studio Vision*, countless DAWs emerged, each software expanding on the functionality of previous programs. Some DAWs, such as Propellerhead's *Reason,* modeled themselves on the design of "analog" hardware devices prevalent in physical recording studios, allowing the user to incorporate hundreds of synthesizers, effects boxes, and sampling machines into their production setup. Other DAWs, such as Ableton *Live* and *Renoise,* arranged their GUIs as vertical grids in which musicians could endlessly stack musical patterns on top of one another. As a result of the ever increasing creative options available to musicians in the age of software, musicians have learned to adapt constantly existing compositional techniques to new interfaces. In turn, DAWs have become more than just functional tools for the purpose of audio recording and composition—they have become experimental playgrounds and media environments that extend beyond the physical confines of the "analog" music studio.

How might such changing trends in musical instrument and interface design specifically shape techniques of electronic music composition and performance? Through an analysis of the technical design of the music production software *Live,* combined with analysis of discourse carried out by producers and musicians in online periodicals and through interviews, this chapter addresses this question. I specifically highlight the ways in which software enables producers to simulate existing musical practices, as well as transform and migrate these creative techniques across various digital media platforms. After examining the specific affordances and constraints embedded within the design of *Live*, I tease out the ways in which these production practices exemplify a maximalist approach to media production and consumption in twenty-first century digital cultures—an approach which promotes the position that the more technologically connected you are, the better. To this end, Alan Kay's notion of the computer as a "metamedium," a device that can "be all other media," provides a useful concept for understanding the ways in which techniques and practices from disparate media converge in software.[51]

The seemingly limitless affordances of *Live* have inspired a new generation of creatives to embrace digital music production, redefining what constitutes the recording "studio" for contemporary musicians. In addition to the modular and interoperable nature of the software, the relative accessibility and usability of *Live* has allowed it to become a primary tool in the rise of "do-it-yourself" (DIY), "bedroom" studios. As a result, this chapter raises broader questions

[51] Alan Kay and Adele Goldberg, "Personal Dynamic Media," in *The New Media Reader*, edited by Noah Wardrip-Fruin and Nick Montfort, 393-404 (Cambridge: The MIT Press, 2003).

about the increasingly blurred lines between consumption and production at the heart of contemporary musical practices. In the context of increasingly user-friendly and maximalist production practices, how do musicians deal with the technological overload that often accompanies composition with software? The techniques and aesthetics reported on suggest that software design is intimately connected both to cultural aesthetics and ethical ideologies of consumerism in technological use since the new millennium.

Instrumental Design in Ableton *Live*

In 2001, the Berlin and Los Angeles-based software company Ableton released *Live*, the now standard digital audio workstation software for music recording and performance. Until the time of *Live*'s release, music production software was catered mostly to a demographic of amateur and professional musicians interested in the ways in which software could make their *recording* and playback setup more efficient. In contrast, *Live* was designed for both audio recording and, as its name suggests, "live" *performance*. In light of this goal, the GUI of the software was designed in such a way as to facilitate rapid audio editing, the ability to loop digital samples with ease, and the seamless integration between tools for audio recording and performance (fig. 1.2). The buttons, knobs, and sliders of "analog" mixing boards are simulated in *Live*, alongside features unique to "digital" software: a vertical grid for arranging digital samples in a non-linear fashion, and an array of windows for editing audio at a micro level. Significantly, *Live* also included a drag-and-drop interface that allows for the real-time remixing of musical content.

Figure 1.2: Ableton *Live* version 1.5 (2002) (Source: "Ableton Live Audio & Loop Sequencing Software For Mac and Windows," *Beat Mode,* n.d., accessed April 25, 2016, [http://www.beatmode.com/historical/ableton-live/)](http://www.beatmode.com/historical/ableton-live/)

By integrating existing compositional techniques with non-traditional forms of musical interaction, *software* such as Live represents a two-fold challenge for the designer, as the digital musician is asked both to navigate a foreign set of performance techniques, and to embody a non-"musical" physical interface to the computer (the mouse and keyboard do not control a fixed set of musical parameters, but can also be used to check e-mails, draft manuscripts, and play video games). Historically, the design and development of music production software has strived to strike a balance between the simulation of existing musical interfaces and techniques on the one hand, and the introduction of new creative possibilities on the other. Generally, two models have dominated music software design, and both are reflected in the GUI of *Live*: first, an *instrumental* lineage that bases the design of new interfaces on pre-existing musical instruments and software. Second, a *computational* lineage that introduces process-oriented mechanics and design into music software. Through analyses of the graphical user interface elements in Live, we can see that both music composition and software design are rooted in iterative, modular, and abstract *processes*, rather than fixed *objects*. Taking this into consideration, I align digital musical composition and performance with the procedural nature of software code: as a set of process-oriented endeavors which require constant "updating" through new forms of musical training.

In what I call the *instrumental* model of software design, the relationship between "new" and "old" technologies is isomorphic, as the emerging tools seek to mimic the look and feel of existing tools. This tool-based notion of instrumentality aligns with philosopher Martin Heidegger's first definition of technology as a means to an end.[52] Considering this definition, the primary function of software is to make more efficient the human tasks previously completed with hardware. For example, it could be argued that the reason for the success of iPads and tablets is that they resemble notepads; or, the use of metaphors such as "desktops" and "folders" has allowed the personal computer to thrive as an office tool. Media theorists Jay Bolter and Richard Grusin define this relationship as "remediation," in which a new media platform "appropriates the techniques, forms, and social significance of other media and attempts to rival

---

[52] Martin Heidegger, "The Question Concerning Technology," *Basic Writings,* edited by David Farrell Krell (Harper & Row, 1977), 287.

or refashion them in the name of the real."[53] From this perspective, the success of new interfaces is dependent on the extent to which the fundamental logic and cultural understandings of previous tools and techniques are retained in their software simulations.

Historically, *musical* instrumentality has been defined in a similarly tool-based manner. As a discipline, organology arose from a functional definition of instruments that comprises both the sounds they produce and the manner in which the instruments produce those sounds.[54] For ethnomusicologists concerned with the historical preservation of "traditional" musical instruments, instruments are often perceived to be "fixed, static objects that cannot grow or adapt in themselves," and therefore can only exist as tools for the purpose of human expression and creativity.[55] Instrument *design* generally aligns with this tool-based, functional understanding of instrumentality.[56] As with any technological development—musical or otherwise—new instruments are often designed in such a way that they retain recognizable elements from previous instruments, while offering an attractive enough set of new affordances to allow the tool to be widely used.[57] In the case of electronic musical instruments, "traditional" instruments are often simulated by the computer, as virtual *tools* for the purpose of recording musical *content.*

---

[53] Jay David Bolter and Richard Grusin, *Remediation: Understanding New Media* (Cambridge, MA: The MIT Press, 2000), 65.

[54] Erich M. von Hornbostel and Curt Sachs, "Systematik der Musikinstrumente: Ein Versuch," *Zeitschrift für Ethnologie* 4 (1914). Translated by Anthony Baines and Klaus Wachsmann as "Classification of Musical Instruments," *Galpin Society Journal* 14 (1961): 3-29.

[55] Margaret Kartomi, "The Classification of Musical Instruments: Changing Trends in Research from the Late Nineteenth Century, with Special Reference to the 1990s," *Ethnomusicology* 45, no. 2 (2001): 305.

[56] Of course, I am aware of the implementation of non-"musical" technologies and devices as musical instruments (wash boards, radio static, vinyl records, and more), but I am here addressing instruments that were designed explicitly *as* instruments.

[57] For example, Paul Théberge characterizes the Theremin as an instrument that bore no resemblance to any existing musical technology, thus requiring musicians not only to adapt to unfamiliar sounds but also to learn an entirely new set of performance techniques. *Any Sound You Can Imagine,* 44.

In the history of music software, DAWs, such as Apple's *GarageBand*, remain the most common examples of these type of tools. DAWs facilitate the instrumental model of software design in three ways: first, through the use of add-on "plugins" to emulate existing musical tools; second, through the incorporation of a horizontal timeline recording layout to emulate the linear temporality afforded by previous musical media such as vinyl records and magnetic tape; and third, through the general use of interface metaphors that resemble physical objects. Together, these features retain the assumption of music software as an object-oriented tool designed for the purpose of audio inscription and recording.

Virtual studio instrument (VSTi) "plugins" are software add-ons that provide additional functionality when used alongside a DAW. Plugins exemplify the instrumental model of software design in that they are often designed as virtual metaphors for existing physical instruments. That is, in order to negotiate the transition from hardware devices to software programs, "digital" plugins imitate the look and feel of musical tools that currently only exist in "analog" form. A survey of popular plugins includes software *emulators* of instruments such as the Roland TR-808 drum machine, the Roland TB-303 bass synthesizer, and a vast array of Moog keyboards (fig. 1.3), as well as *simulators* of common techniques employed by producers while using these tools, such as grid-based rhythm and note sequencing, basic sound synthesis, and audio sampling.

Figure 1.3: (A) Various Native Instruments mixing plugins integrated within Live's session view; (B) Arturia's VST emulation of the original Moog "Minimoog" synthesizer. In both cases, the gestural affordances of the "analog" hardware (knobs, sliders, switches, meters) are emulated in the virtual software.

In addition to the incorporation of plugins, the instrumental design approach in DAWs is fostered through the use of a horizontal timeline recording layout that emulates the linear temporality afforded by previous media forms. In *Live*'s "Arrangement View," for example, the GUI is organized as a horizontal, linear recording timeline meant for the layering of multiple tracks (fig. 1.4). Producers are already familiar with the process of organizing musical ideas into complete, pre-composed structures—a fact that is evidenced by the ever present linear timeline across DAWs. By remediating the physical processes and graphical layout involved in *writing* musical ideas on sheet music, or *splicing* and rearranging magnetic tape, the timeline provides a visual analogy to earlier, "analog" forms of musical composition.

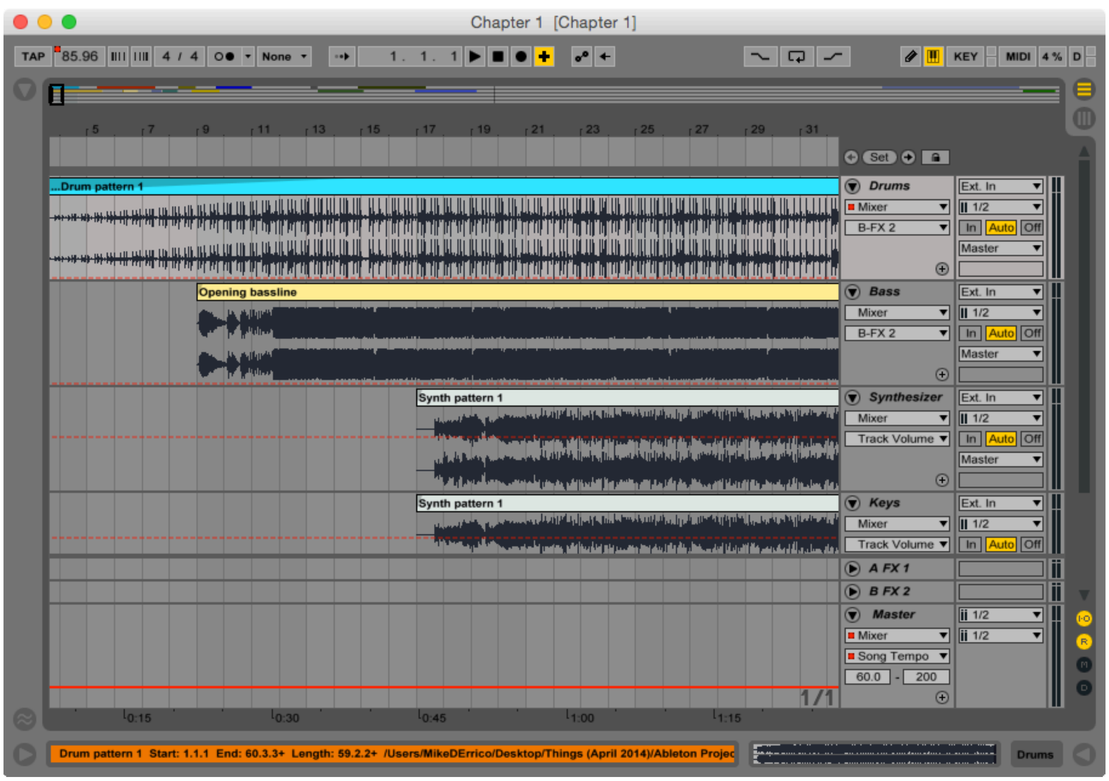


Figure 1.4: Arranging musical tracks in a linear, horizontal manner using *Live*'s "Arrangement View." Rows represent individual musical tracks (drums, bass, etc.), while the timeline runs from left to right.

Further, DAWs foster the instrumental design model through the use of design metaphors that resemble "analog" interface elements such as sliders and knobs. Since the advent of "personal computing" in the 1980s, these types of indexical icons and material metaphors have guided the design of computer operating systems.[58] For example, the "Mailbox" icon analogizes e-mail to snail mail; the "Trash" icon aligns the practices of discarding digital files and throwing away physical waste; and "Folders" on the computer "Desktop" are meant to store digital files in a similar fashion as the manila folders of an actual desktop. In each example, indexical icons are meant to relate tasks in the digital software environment to everyday tasks with which the computer user is already familiar. In this way, the suggested usage of the software is implied by the design metaphor, thus requiring little to no specialized knowledge from the user.

Material metaphors in the form of buttons, knobs, and sliders are present throughout the *Live* GUI. For example, when adding effects to a given track, the musician can manipulate specific effect parameters using minimalist interface elements that abstractly represent the rotary knobs on an analog synthesizer (fig. 1.5a). Similarly, adjusting the volume of an audio sample requires the manipulation of a vertical slider that represents a simplified version of the faders that appear on physical mixing boards (fig. 1.5b). In addition, muting, recording, and soloing individual audio tracks can be achieved by clicking one of three rectangular buttons, thus mirroring the design and functionality of a mixing board (fig. 1.5c). In these examples, the minimalist and abstract design of the interface metaphors serves to focus the musician's attention on the physical task required of each element, whether that be sliding, turning, pressing, or switching. In other words, design metaphors encourage the computer user to forget that they are

[58] Marianne Van Den Boomen, "Introduction: Metaphor, Meaning, and Code," in *Transcoding the Digital: How Metaphors Matter in New Media* (Amsterdam: Institute of Network Cultures, 2014).

interacting with software at all, instead focusing their attention on the physical hardware being simulated by the GUI.

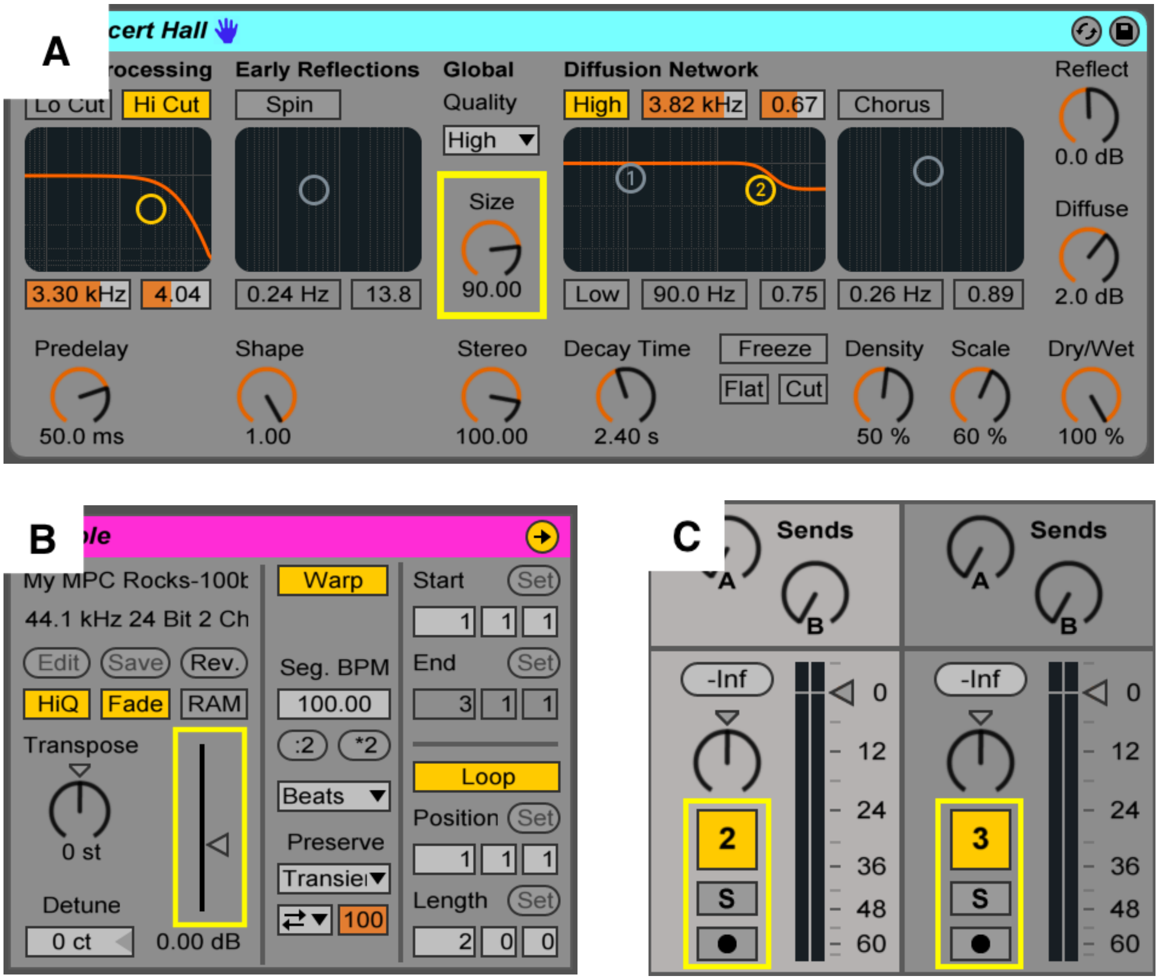


Figure 1.5: Design metaphors in *Live.* (A) rotary knobs in *Live* effects; (B) volume fader in *Live* sample editor; (C) mixing board buttons in the *Live* mixer.

As the examples of plugins, timelines, and other interface metaphors demonstrate, remediation is a common factor in digital musical training. These *tools* are designed based on the philosophy that, in order for musicians to develop technical proficiencies on new instruments, interface design should ideally resemble previous instruments in some way. However, in translating “analog” technologies to “digital” software, instrumental design models also risk reifying a limited set of practices and stunting the creative possibilities of musicians.

Musicologists have critiqued the ways in which musical media such as sheet music and recordings encourage relating to the music as a fixed text or "work." [59] Similarly, instrumental design models privilege an understanding of software as a *text-based* medium for the purpose of inscribing existing objects and practices. The concept of the musical "work" as an *object* may have made sense in the context of music printing and vinyl record distribution, but the *process-oriented* nature of software equally foregrounds the importance of algorithms, procedures, and rule-based systems. As much as the tool-based nature of *Live* appeals to computer musicians with some traditional training in audio production or musical performance, the software also incorporates computational affordances that depart from representational and metaphorical design trends.

Computational Design in *Live*

Software is built based on computational affordances that either extend, or depart from, representational and metaphorical design trends. While plugins *represent* existing musical tools and techniques, DAWs also offer the unique capability of *abstracting* what might be considered more traditional "musical" tools and techniques, thus presenting new possibilities for digital composition. In contrast to the instrumental design model, what I call the *computational* model of software design embraces the unique affordances of the computer as both an instrumental

[59] Mark Butler provides a useful survey of the ways in which "work-concepts" have been employed across a range of musical discourses as a means of asserting the compositional authority, originality, and boundedness of musical objects. *Playing With Something That Runs: Technology, Improvisation, and Composition in DJ and Laptop Performance* (New York: Oxford University Press, 2014). See also, Georgina Born, "On Musical Mediation: Ontology, Technology, and Creativity," *Twentieth-Century Music* 2 no. 1 (2005): 7-36; Albin J. Zak, *The Poetics of Rock: Cutting Tracks, Making Records* (Berkeley: University of California Press, 2001); Lydia Goehr, *The Imaginary Museum of Musical Works: An Essay in the Philosophy of Music* (Oxford and New York: Oxford University Press, 1992); Theodore Gracyk, *Rhythm and Noise: An Aesthetics of Rock* (Durham, NC: Duke University Press, 1996).

medium for the authoring of text, audio, and visual content, as well as a process-oriented medium for the authoring of processes themselves. If *tools* such as word processors and photo editors exemplify the instrumental model, the computational model is best represented by rule-based *practices* such as programming code, generating algorithmic scripts, and designing digital games. DAWs such as *Live* facilitate the computational model of software design in three ways: first, the incorporation of a non-linear temporal structure in the form of a vertical grid; second, the use of “real-time” feedback mechanisms that afford the rapid and iterative prototyping of musical material; and third, the ability to drag-and-drop interface elements in a modular manner.

In contrast to the horizontal timeline of “Arrangement View,” *Live*’s “Session View” organizes musical patterns into a vertical grid designed to facilitate the real-time re-arrangement of musical “clips.” In this *computational* design model, musical patterns are organized as literal building blocks of data that can contain anything from a short vocal sample to an entire multi-movement symphony (fig. 1.6). Session View encourages a shifting understanding of the musical “work” by fragmenting the compositional process into modular units. This non-linear aspect of the GUI remains the most radical and appealing feature of the software for many producers. Mike Huckaby claims *Live* to be “the most revolutionary sequencing and production software to emerge from the past 10 years,” largely due to the ways in which the clip grid “destroyed the concept of linear thinking while making music.”[60] By providing an alternative to the horizontal timeline, *Live*’s non-linear temporal structure fragments the notion of the holistic, self-contained musical work, and intensifies the process-oriented nature of composing with digital software.

[60] Ableton’s website continually archives blog posts and artist interviews. Huckaby’s quote was found at the following site archive: Ableton, “Artist Quotes,” n.d., accessed April 20, 2015, https://www.ableton.com/en/pages/artists/artist_quotes/.

Figure 1.6: Stacking tracks vertically in the non-linear, building-block manner of *Live* "Session View." Columns represent individual tracks and the timeline is arranged in rows.

In addition to the use of a non-linear temporal structure, *Live*'s computational design is reflected in the incorporation of "real-time" feedback mechanisms that allow for rapid and iterative prototyping of musical material. As the very name suggests, "Live" signifies mobility, emphasizing the creative flexibility that results as users transport the software using their portable laptops. From a practical perspective, the seamless workflow in which the GUI immediately displays the results of various actions by the producer (fig. 1.7) also adds to the real-time sensibility. Composer and multi-instrumentalist Angélica Negrón discusses how she uses *Live* to audition and preview her work while writing it, allowing her "to listen to both elements (electronic and acoustic) simultaneously, which is something I'm not able to do in notation software."[61] Negrón describes a composition process that exemplifies iterative design methodologies. That is, *Live* affords Negrón a process in which the creative materials are constantly reworked based on user testing and real-time evaluation of sonic materials, something she is unable to carry out in more traditional notation software.

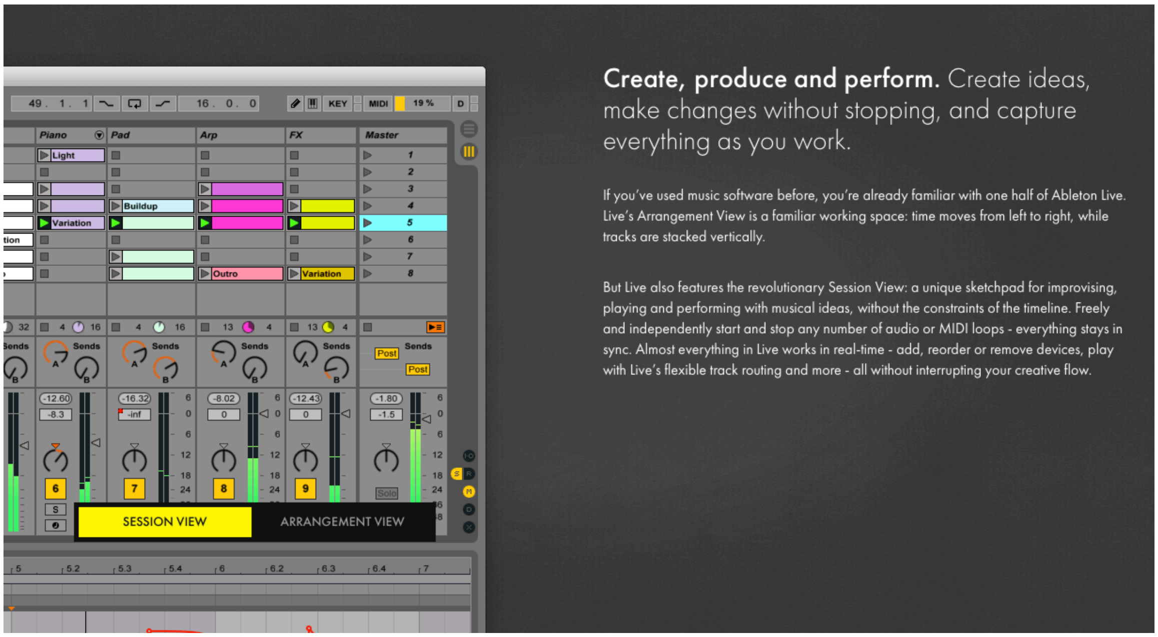


Figure 1.7: *Live 9* promotional material, emphasizing the "real-time" aspects of production and performance with the software: "Make changes without stopping, and capture everything as you work… Almost everything in Live works in real-time… all without interrupting your creative flow" (Source: Ableton, "Live," accessed April 22, 2015, https://www.ableton.com/en/live/).

In contrast to the typical recording studio model which values the *inscription* of musical material onto physical storage media, the real-time aspect of composing in *Live* instead privileges the fine-tuned *manipulation* and *re-performance* of material. Studio production blurs with stage DJing, as the producer pulls sonic material from his or her "Library," layering track after track without interrupting the workflow. As the music runs, the most minute micro parameters of the audio signal can be finely edited, with each virtual turn of a knob capable of

[61] "Angélica Negrón: Toys Noise," *Ableton,* January 13, 2012, accessed April 20, 2015, https://www.ableton.com/en/blog/angelica-negron/

being automated and recorded within each clip. For example, to create an 8-bar looping drum pattern that fades out over the course of each iteration, the producer literally draws a downward "automation" curve over the waveform of the pattern (fig. 1.8). In this way, *Live* encourages not only the re-performance of existing musical material, but also the inscription and automated playback of performance gestures such as knob-turning, slider-fading, and button-pressing. This ongoing, iterative cycle of production and reproduction prioritizes the performative aspects of sound *manipulation* rather than the inscriptional aspects of sound *recording.* In doing so, real-time feedback mechanisms foreground a shift in the conception of software from text-based code (object) to "live" instrument (process).

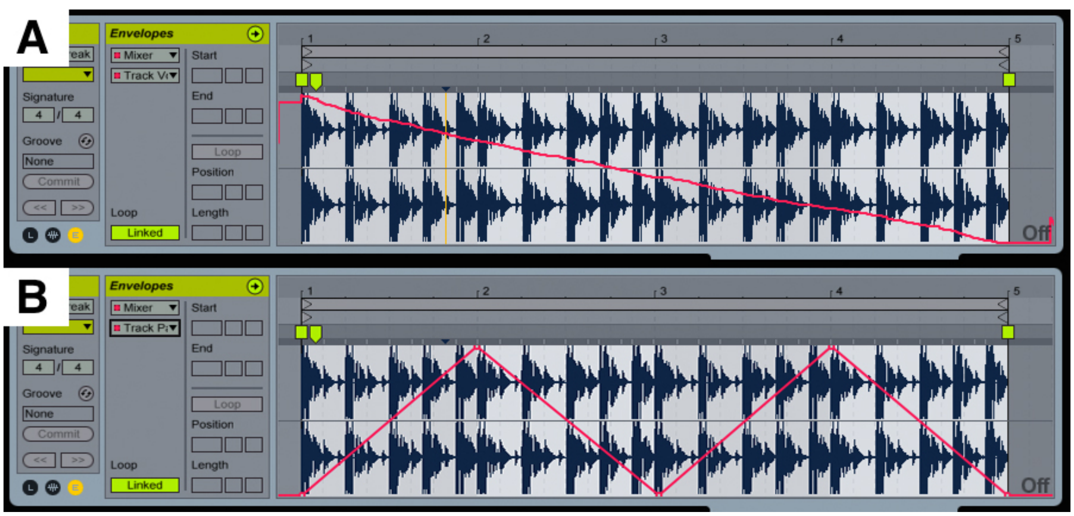


Figure 1.8: Drawing "automation curves" onto a sampled musical "clip." (A) The volume of the musical pattern starts high and fades to nothing with each iteration of the loop. (B) The volume of the pattern proceeds like a wave, with high and low crests throughout the loop.

The third defining feature of *Live*'s computational design is the modularity of interface elements. Whereas the instrumental model of design values faithful recreations of existing musical instruments, the layout of the *Live* GUI can be redesigned and reassembled by each individual musician. This malleability is most clearly present in one of the primary design functions of *Live*: the ability to drag-and-drop nearly every musical element into one another. Sound effects can be dropped onto individual musical clips, entire tracks can be dropped into the middle of an extended musical arrangement, and media from outside of the *Live* set can be dragged-and-dropped into the current workflow (fig. 1.9).

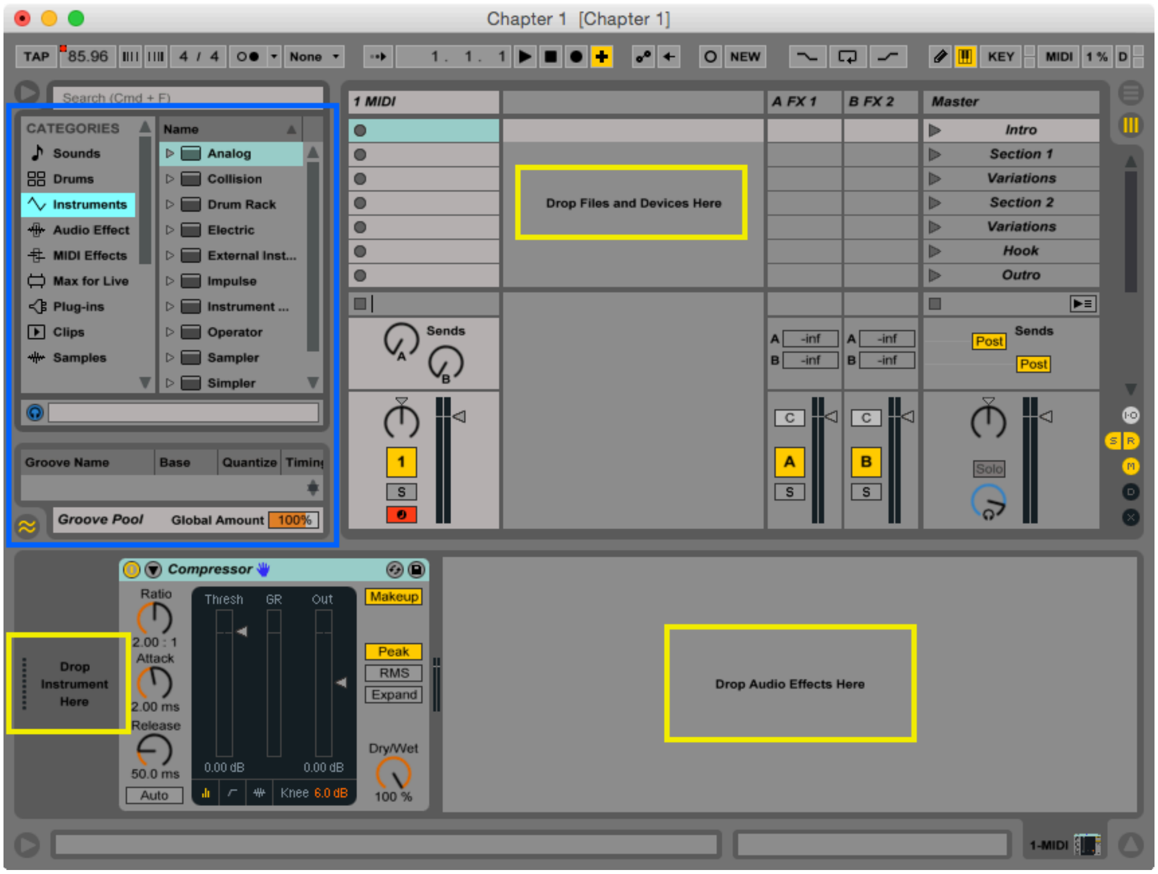


Figure 1.9: Macro perspective on *Live*'s "Session View," highlighting the possibilities for dragging-and-dropping audio files, instruments, groove templates, and effects throughout the open, seemingly transparent "windows" of the interface. The blue box highlights the "Library" from which instruments, effects, samples, and "grooves" can be dragged, while the yellow boxes highlight the spaces to which those elements can be dropped.

Significantly, the drag-and-drop function can be performed without disrupting the audio currently being played, thus allowing for the further manipulation of musical events in real-time. Experimental electronic musician Synnack describes the ways in which this functionality has allowed him to overcome the "nightmare" of exporting and importing musical material into a coherent workflow: "The fact that, in *Live*, you can just drag and drop songs into each other, and even preview them from the browser was totally revolutionary and still is… The fact that you can drag and drop effects into a song while it's playing, and even reorder them with no audio dropouts is insane."[62] For Synnack, the primary benefit of drag-and-drop in *Live* is that it does not disrupt the playback of the musical tracks, thus encouraging iterative prototyping of the composition.

Not only does drag-and-drop afford the modularity of musical material, but also of the very interface elements themselves. Like most DAWs, Live incorporates a traditional "windows, icons, menus, pointer" (WIMP) interface that makes transparent the various screens through which the producer composes. Additionally, in *Live* each segment of the production workflow—the user's sound library, the transport (play, stop, record) controls, sound effects, and track content—appears as a distinct module, clearly delineated by a thick frame that functions as a border to the corresponding window. As such, the *Live* GUI is designed to be as compact and transparent as possible, with as many interface elements visible and easily accessible at all times. Steven Johnson singles out the development of the design feature, "windows," as the most important interface innovation in personal computing, as it facilitates a "more layered and multiplicitous" onscreen space that allows users to quickly switch between various modes of thinking and practice with the click of a mouse.[63] In the context of the *Live* GUI, the WIMP interface encourages a modularity of not only the musical *content* of a given production, but also the *form* of the DAW itself.

[62] "Clint Sand: Electronic Polymath," *Ableton,* January 20, 2011, accessed April 20, 2015, https://www.ableton.com/en/pages/artists/clint_sand/.

[63] Steven Johnson, *Interface Culture* (New York: Harper Collins, 1997), 84.

Existing scholarship on the relationship between musicians and technology has focused on the ways in which musicians use technologies designed for non-musical purposes as musical instruments. Mark Katz and other musicologists describe the virtuosic performances of hip-hop DJs, specifically aligning the creative practices of turntablism with those of classical musicians.[64] David Bernstein examines the use of tape recorders and other multimedia tools as performance instruments in the 1960s avant-garde.[65] In each case, technologies not intended for music creation are valued for their capacities to become tools for the purpose of musical performance. In contrast, the *Live* GUI exemplifies a convergence of instrumental and computational design models, as previously outlined. As such, *Live* is more than just a musical instrument or tool for sound recording; it is an exploratory environment of creative affordances through which the musician navigates. Examining *Live* as a process-oriented environment rather than a fixed instrument allows us to shift the focus from the GUI itself to the broader network of tools and techniques that continue to shape the design and use of the software.

***Live* as Software Environment**

Within technology circles, the term “environment” is often used to refer to the external or internal factors that shape the design and use of a given technology. The Software Development Environment refers to the programming tools used to create the software; The scholarly discipline knows as “media ecology” specifically studies media and technology as environments that structure human perception, feeling, and value. Indeed, environmental metaphors are

---

[64] Mark Katz, *Capturing Sound:* (Berkeley and Los Angeles: University of California Press, 2004).

[65] David W. Bernstein, *The San Francisco Tape Music Center: 1960s Counterculture and the Avant-Garde* (Berkeley and Los Angeles: University of California Press, 2008).

commonplace among professional interface designers. Borrowing a term from information science, for example, we can think about the ways in which the *Live* GUI epitomizes an open and transparent "information architecture."[66] Initially defined by Richard Saul Wurman as the creation of "systemic, structural, and orderly principles to make something work—the thoughtful making of either artifact, or idea, or policy that informs because it is clear,"[67] information architecture now encapsulates various meanings and practices across information science and technology. These include the structural design of shared information environments,[68] the art and science of labeling websites and other information management programs,[69] and a community of practice focused on migrating design principles to the digital landscape.[70] *Live*'s instrumental and computational design approaches, previously discussed, can be understood as attempts to construct a broader "environment" of technical affordances that frame the digital musicians' creative workflow.

Specifically, analyzing *Live* through the structural metaphor of "information architecture" provides a model for understanding the DAW as a software environment. In physical recording studios, glass windows often serve as barriers between performance spaces in which musical content is recorded, and control rooms in which sound is managed and edited during and after

---

66 Earl Morrogh, *Information Architecture: An Emerging 21st Century Profession* (Saddle River, NJ: Prentice Hall, 2003); Louis Rosenfeld and Peter Morville, *Information Architecture for the World Wide Web* (Sebastopol, CA: O'Reilly & Associates, 1998); Richard Saul Wurman, *Information Architects* (New York: Graphis Inc., 1997).

67 Wurman, 17.

68 Andrea Resmini and Luca Rosati, *Pervasive Information Architecture: Designing Cross-Channel User Experiences* (Burlington, MA: Morgan Kauffman, 2011).

69 Christina Wodtke, *Information Architecture: Blueprints for the Web* (San Francisco, CA: New Riders, 2009); Peter Brown, *Information Architecture With XML* (Hoboken, NJ: John Wiley & Sons Ltd., 2003).

70 Wei Ding and Xia Lin, *Information Architecture: The Design and Integration of Information Spaces* (San Rafael, CA: Morgan & Claypool, 2009).

the performance. When this design principle is translated into digital software, physical space is flattened into a single screen, removing the barrier between the creative spaces of the stage (as a performance space) and studio (as a sound management and editing space). In removing the *glass* that fractures and isolates the production process into separate spaces, sound in the digital environment is allowed to filter and bleed through the various elements of the computer *screen*.

Most immediately, the architectural metaphor may evoke an instrumental conception of software as object—as "frozen music," perhaps. Following this Goethian imagery, I suggest that the process-oriented nature of composing in *Live* exemplifies software as "liquid architecture": an environment structurally defined by the movement, juxtaposition, and recombination of sound and interface elements, rather than through fixed visual structures. "All that is solid seems to melt in the cloud," media theorist Marianne Van Den Boomen puts it.[71] The modularity of musical ideas occurs first and foremost within the open framing of the GUI (for example, the WIMP interface, the ability to drag-and-drop musical tracks, and the use of interface metaphors), guiding the producer as they navigate the interconnected structures of the various interface elements.

At the same time, framing the DAW as environment also captures the ways in which the software extends *beyond* the centralized space of the *Live* interface, as an interoperable program within a broader system of digital tools and techniques. Musicologists Jason Stanyek and Benjamin Piekut describe the "leakage effects" as inherent qualities of digital recording spaces, occurring "when an activity in one area expands unexpectedly into another area, setting in

[71] Marianne Van Den Boomen, "Introduction: Metaphor, Meaning, and Code," in *Transcoding the Digital: How Metaphors Matter in New Media* (Amsterdam: Institute of Network Cultures, 2014), 12.

motion a second process, project, or concern."[72] For Stanyek and Piekut, the productive potential of software manifests in the ongoing convergence of music composition tools and techniques across media platforms. These leakage effects occur not only when musical content (plug-ins, audio files, and effects) moves between the various modules of *Live*'s sonic architecture, but also when the previously fixed space of the digital "studio" extends beyond a single dedicated software. This software convergence forms what Paul Théberge calls a "network studio," influencing the techniques and practices of other applications within the broader computational environment.[73] The metaphorical perforations inherent to the screens of software thus deconstruct the centralizing function of the recording studio glass, and the modularity inherent to the creative practices of *Live* users extends itself to the complete environment of software programs that comprise the computer hardware as a whole.

The *Live* environment's modularity is heightened by the development of standards for software interoperability, affordances which have facilitated greater interconnections between disparate media production software (fig. 1.10). In addition to standard plugin capabilities that allow for the incorporation of third-party instruments and effects into the *Live* workflow, the ReWire software protocol allows remote control and data transfer between *Live* and related software for digital audio editing (fig. 1.11). For example, the audio output of a synthesizer patch in *Max* software can be "rewired" into the mixer of *Live*, allowing the musician to control various effects and sound parameters of the *Max* synthesizer from within *Live*. Since its emergence in 1998, ReWire has become an industry standard for music production software,

[72] Jason Stanyek and Benjamin Piekut. "Deadness: Technologies of the Intermundane." *The Drama Review* 54 no. 1 (Spring 2010): 20.

[73] Paul Théberge, "The Network Studio: Historical and Technological Paths to a New Ideal in Music making," *Social Studies of Science* 34 no. 5 (2004): 776.

allowing the simultaneous transfer of up to 256 audio tracks and 4080 channels of MIDI data. In this way, software interoperability embodies what Manovich calls the "logic of 'permanent extendability'" inherent to media production in the early twenty-first century.[74]

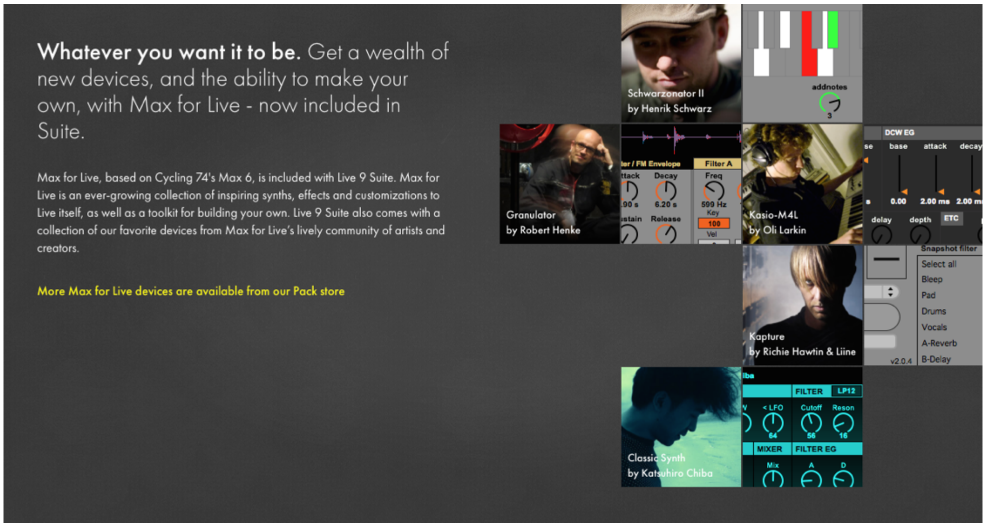


Figure 1.10: ***Live 9*** **promotional material, emphasizing the modular architecture of the interface. A specific plugin, Max for Live, is singled out as just one of the many ways in which the** ***Live*** **can be extended by networking it with other tools and software (Source: Ableton, "Live," accessed April 22, 2015, https://www.ableton.com/en/live/).**

Figure 1.11: ***Live*** **as ReWire "slave" to Avid's** ***Pro Tools.*** **The yellow box shows** ***Live*** **receiving MIDI from** ***Pro Tools*****. The blue box shows** ***Live*** **receiving audio from** ***Pro Tools.***

[74] Manovich, 156.

Software interoperability is further apparent in the phenomena of what composer David Bessell calls “feature creep,” in which plugins and functionality spread from one DAW to another as a result of commercial competition, desire for increased usability, and the need to keep up with technological advances. Examples of this “duplication of function” across music software include the ability to “fix” the timing of recorded audio (quantization), audio time-stretching and “warping,” tempo detection of audio clips, and video playback from within the DAW, among others (fig. 1.12).[75] As electronic music pioneer Miller Puckette notes, “we want software to do everything, and our notion of ‘everything’ grows broader every year.”[76] While developers such as Puckette view feature creep as a counterintuitive design method, the accumulation of tools and techniques that results from its implementation allows musicians to balance existing knowledge with new and abstract compositional practices.

[75] Bessell, 408.

[76] Miller Puckette, “Max at Seventeen,” *Computer Music Journal* 26, no. 4 (2002): 40.

Figure 1.12: "Feature creep" across DAWs. (A) Temporally stretching a sampled drum break using the "Elastic Audio" functionality of *Pro Tools 11*. (B) Accomplishing the same task using the audio "Warp" functionality of *Live 9*.

Indeed, the combined convergence of instrumental and computational design models *within* the *Live* GUI, and the proliferation of creative musical affordances *across* a larger network of software applications highlights an alternative understanding of the historical and technical relationships between media. If we approach software as an environment of affordances, we may conceptualize the ways in which software encapsulates not only temporal relationships with previous media—the *instrumental* idea that technological change occurs through either innovation or remediation—but also an *accumulation* of technologies and creative practices from the past.[77] The quest for a platform that is not just one instrumental tool, but can simulate many tools, has been at the heart of computer science since its early days. In Alan Kay's original vision of the computer, he characterizes software as a "metamedium" that can "dynamically simulate the details of any other medium, including media that cannot exist physically. It is not a tool, although it can act like many tools." As such, the computer "has degrees of freedom for representation and expression never before encountered and as yet barely investigated."[78] The notion of the computer as a metamedium is simultaneously a challenge to existing practices of human-computer interface design, as well as a metaphor for the ways in which the creative practices of one cultural arena are capable of moving between a variety of other platforms. In this mode of thinking, software environments are designed specifically to facilitate an *accumulation* of tools and techniques, rather than to only serve as a *remediation* of existing practices.

By conceptualizing the *Live* environment as a metamedium, it is possible to explain the constant migration of compositional techniques and interface designs across media production

[77] Frances Dyson, *Sounding New Media: Immersion and Embodiment in the Arts and Culture* (Berkeley and Los Angeles: University of California Press, 2009), 3.

[78] Alan Kay, "Computer Software," *Scientific American* 251 no. 3 (1984): 52-9.

software. For example, Lev Manovich points out that the "layers" function, typical of photo editing software such as *Photoshop*, has become standard technique across editing software, including vector image editors, motion graphics software, video editors, and sound editors. Despite the differences between media platforms, "*a final composition is a result of 'adding up' data*" for all these editing software.[79] Similarly, plugins and a modular GUI for sound manipulation, the core features of *Live*, serve to accumulate compositional techniques from new and existing instruments, as well as from tools and techniques that have migrated from other media platforms. The additive nature of software development thus challenges electronic music producers to learn constantly new techniques and navigate new tools, encouraging them to focus on the *event* of creative production rather than the *objects* produced, thus destabilizing fixed understandings of musical instruments, musical "works," and software itself.

As I have suggested, the convergence of instrumental and computational design models exemplifies a shift from objects to processes in approaches to music-making. By designing possibilities for extended functionality into the very core mechanics of *Live*, the creators of the program posit new conceptions of the recording studio as an inherently networked and "plugged in" information architecture composed of tools and techniques gradually accumulated from external devices and applications. Media theorist Peter Lunenfeld describes this process of technical accumulation as "stickiness," a fundamental quality of creative production in digital environments containing affordances "that allow other meaningful objects or systems to latch on to it, expand it, or burrow deep within it."[80] Software interoperability shows the extent to which a

[79] Lev Manovich, *Software Takes Command* (New York and London: Bloomsbury, 2013). Emphasis added.

[80] Peter Lunenfeld, *The Secret War Between Downloading and Uploading: Tales of the Computer as Culture Machine* (Cambridge, MA: The MIT Press, 2011), 28.

logic of stickiness is embedded within the design and development of software. In constantly seeking out perforations and "leakage effects" amongst the various spaces of the DAW, the interoperable nature of the software environment deconstructs the closed, insular walls of the physical recording studio. Ultimately, the organization of elements within the *Live* interface implies not just personal aesthetics but also—as I discuss in the final section of this chapter—formations of specific cultural communities.[81]

Maximalism as Cultural Practice

Having posited the design of *Live* as an integrated environment comprising an accumulation of creative affordances, it is possible to consider the ways in which the shift from software as fixed instrument to modular environment has shaped broader cultural practices. How might the *technical* mechanics previously outlined parallel the development of similar *aesthetic* values within electronic music communities? In the final section of this chapter, I contextualize the maximalist design of the *Live* GUI within a broader maximalist consumer aesthetic in contemporary digital culture. In doing so, I bridge the gap between the technical structure of the *software* and the *culture* that has formed around it, which includes an emerging demographic of producers, DJs, and musical genres that sonically define themselves through these tools; and the introduction of digital audio production "academies" that have worked to standardize electronic music pedagogy. In negotiating the parallel development between shifts in software design and a transpiring culture of electronic musicians, digital music pedagogues have become key mediators

[81] Many media theorists and computer scientists employ architectural analogies in describing the "environment" of software applications. Steven Johnson interrogates the cultural and political import of these metaphors, claiming that "each [software] design decision echoes and amplifies a set of values, an assumption about the larger society that frames it. All works of architecture imply a worldview, which means that all architecture is in some deeper sense political" (44).

in the standardization of increasingly changing understandings of music composition in the age of software.

As I suggested in the previous section, the design of *Live* encourages a shift in conceptions of software from a singular *tool* for the inscription of fixed content to a modular *environment* of interconnected tools. Just as the seemingly endless affordances of the software reflect a broader rhetoric of democratization at the heart of technological design and use, *Live* also affects broader cultural formations of "do-it-yourself" (DIY) musicians and composers. Indeed, the apparent usability of the software has allowed *Live* to become a ubiquitous presence in the everyday lives and personal spaces of electronic musicians. Dance music blog XLR8R's "In the Studio" series takes the reader behind the scenes of popular musicians' creative workspaces, and confirms the pervasive nature of software in the age of the "bedroom" studio (fig. 1.13). However, while software developers and musicians alike praise the seemingly limitless affordances of programs such as *Live*, musicians often struggle to deal with the overwhelming amount of creative options available to them in the instantly interconnected digital world. Scholars, journalists, and industry professionals have continuously attempted to define this sensibility of information overload in digital culture, whether through the frames of technological "ubiquity,"[82] media "ecologies,"[83] or even more broadly, epochal markers such as "the information age." None of these monikers capture the interconnections between technical design and practical use in HCI. Instead, I suggest the term "maximalism" as an overarching

[82] Ulrik Ekman, ed., *Throughout: Art and Culture Emerging with Ubiquitous Computing* (Cambridge and London: The MIT Press, 2012).

[83] Matthew Fuller, *Media Ecologies: Materialist Energies in Art and Technoculture* (Cambridge, MA: The MIT Press, 2007).

concept that encapsulates an accumulative and integrated approach to media production and consumption in the early twenty-first century.

Figure 1.13: Machinedrum in his bedroom studio, Berlin (2012)
(Source: Shawn Reynaldo, "In the Studio: Machinedrum," ***XLR8R,*** **August 27, 2012, accessed April 25, 2016, https://www.xlr8r.com/gear/2012/08/in-the-studio-machinedrum/)**

While maximalism has been used to describe an aesthetic of digital art, an ethic for technological use, and an ideology of contemporary capitalism, I use the term to describe a dominant approach to music and media production and consumption. In the context of technology, economics, and creative practice, it serves as an umbrella term for a variety of ideologies related to the mobility and accessibility of media in the digital age. Moreover, the concept functions as a point of contact for the theory and practice of contemporary music production, highlighting deep-seated anxieties about information overload while serving as a guide for computer music producers as they navigate a range of media platforms.

The maximalist attitude is most noticeable in the general technophobic anxiety amongst consumers that they will be unable to keep up with the increasing presence of technology in their everyday lives. In a paradoxical response to the fear that technology may one day render obsolete the agency of the human being, technology is embraced more fully as a means of reasserting individual control over the natural world. While some celebrate the ability to negotiate the constant influx of material technologies and information outlets, others are critical of the maximalist claim that technology will solve the world's problems.[84] The perception of media and technology as overwhelming forces in society and culture is certainly nothing new, but the widespread adoption of consumer technology such as smartphones since the first iPhone release in 2007 has made these anxieties even more personal. One iPhone user details the "murky feeling of unease" that arises after witnessing a mother steering her baby stroller with her elbows to free up her hands for smartphone usage, or upon realizing that he is the only person in a city crowd whose eyes are not glued to the screen of his phone.[85] Critical media theorist Nicholas Carr examines the ways in which web surfing via personal computing technologies has encouraged more distracted forms of human perception and cognition, thus decreasing our ability to focus for extended periods of time.

However, maximalism is more than just a social situation that inspires polarizing debate between luddites and technological celebrationists. It is also a mantra for marketing and productivity in the digital age, a cultural aesthetic, and an economic infrastructure endemic to global capitalism—what anthropologist Frenchy Lunning terms "hyperconsumerism," a gloated

---

[84] William Powers, *Hamlet's Blackberry: Building a Good Life in the Digital Age* (New York: Harper Perennial, 2011).

[85] Alex Hunley, "Hamlet's Blackberry – William Powers," *Full Stop,* August 24, 2011, accessed May 15, 2015, http://www.full-stop.net/2011/08/24/reviews/alexhunley/hamlets-blackberry-william-powers/.

economic system that feeds a media market of "monstrous proportions."[86] Similarly, theories addressing the increasingly maximal nature of communications media are premised on the critical notion that globalization has accelerated the rate at which digital products can be produced, accessed, and manipulated, resulting in a constant state of information overload.[87] For media theorist Tiziana Terranova, the consumer should primarily be concerned with how to navigate his or her way out of the vast sea of digital information and data accumulation in the twenty-first century. As Terranova writes, "network culture" is characterized by "an unprecedented *abundance* of informational output and by an *acceleration* of informational dynamics."[88] *Network Culture* does not simply outline changes in consumer dynamics as a result of capitalism at the start of the new millennium, it also diagnoses "information overload" as an ethical issue for the modern citizen and introduces questions as to "how we might start to think our way through it."[89] Information becomes the material manifestation of an ideology that privileges instant access, technological fluidity, and an omnivorous approach to media consumption.

As discussed in relation to *Live*, at the heart of the maximalist techno-economic sensibility is software—the metaphorical interface that provides users with a means for the everyday interaction and consumption of media. As musical instruments and tools increasingly

---

[86] Frenchy Lunning, ed., *Mechademia 5: Fanthropologies* (Minneapolis: University of Minnesota Press, 2010), 140.

[87] Richard H.R. Harper, *Texture: Human Expression in the Age of Communications Overload* (Cambridge, MA: The MIT Press, 2010). Harper implicitly expands on Powers' critique by outlining the general "texture" of expressive practices in contemporary culture. The terms used to describe the "feel" of communication in the twenty-first century are by and large aesthetic markers, such as "quick," "slow," "permanent," and "ephemeral." These are the types of aesthetics I am interested in teasing out in relation to the changing "texture" of contemporary music production in the age of software.

[88] Tiziana Terranova, *Network Culture: Politics for the Information Age* (London: Pluto Press, 2004), 1.

[89] Ibid.

converge in software platforms, this omnivorous approach to media consumption has broad implications for music composition and production. How might musical practice make audible the sense of information overload inherent to the maximalist ethos? Dance music critic Simon Reynolds defines maximalism in the specific context of contemporary electronic music. Describing the "rococo-florid riffs, eruptions of digitally-enhanced virtuosity, skyscraping solos, and other 'maxutiae,'" present in the music of artists such as Rustie and Flying Lotus, Reynolds' characterizations of contemporary digital music production are primarily aesthetic. His notion of both music consumption and production as driven by a "post-everything omnivorousness" reflects a shared economic sentiment with much digital maximalist discourse, highlighting the glutted mediascape and viral nature of contemporary capitalism. For Reynolds, the blurry line between maximalist production and consumption is epitomized in the technical and practical process of digital audio production, in which there are "a hell of a lot of inputs… in terms of influences and sources, and a hell of a lot of outputs, in terms of density, scale, structural convolution, and sheer majesty."[90] By conceptualizing maximalism at the level of both input (consumption) and output (production), Reynolds suggests a more isomorphic relationship between the aesthetic design of software and the shifting compositional values within digital music culture more broadly.

As I have suggested throughout this chapter, the "inputs" described by Reynolds are not simply the wide-ranging artistic influences of the producer, but also the hybrid, maximal nature of DAWs—software *environments* which foster the convergence of various musical tools into a coherent workflow, and consequently shape the "texture-saturated overload" of the music.

---

[90] Simon Reynolds, "Maximal Nation: Electronic Music's Evolution Toward the Thrilling Excess of Digital Maximalism," *Pitchfork,* December 6, 2011, accessed April 20, 2015, http://pitchfork.com/features/articles/8721-maximal-nation/.

Matthew Ingram describes the ways in which DAWs such as *Live* encourage "interminable layering," the GUI presenting music as "a giant sandwich of vertically arranged elements stacked upon one another."[91] The software's ability to tweak the parameters of any given sonic event results in what Ingram terms the "crisis of the control surface"—an overemphasis on the visual paradigm of software that "messes with a proper engagement with sound itself."[92] Echoing a common thread in maximalist discourse, Ingram perceives the inherently "wired-in" characteristics of software as deterministic properties that strictly prescribe and confine creativity, shifting the essence of musical practice from an organic "event" into an inscribed product.

As is the case with technology in everyday use, musicians have expressed two contrasting responses to the maximalist situation. On one hand, there are producers who are skeptical of the uncritical acquisition and use of new technologies. Electronic dance music producer DJ Machinedrum is cautious of the maximalist ethos surrounding the marketing and use of digital tools, echoing Ingram's "crisis of the control surface." He states, "the only problem or 'frustration' I can see coming from music technology would be the seemingly endless amount of options when it comes to software, plug-ins, synths, etc. I feel like a lot of people can become lost trying to get the best gear instead of really honing in on their craft and developing their own sound."[93] The idea of the creative process being stunted as a result of the musician getting lost in a vast sea of studio gear is a common symptom of maximalist consumption, and reflects a common developmental stage in the introduction of new technologies for music production.

---

[91] Matthew Ingram, "Switched-On," *Loops* 1 (2009): 136.

[92] Ibid.

[93] "Machinedrum: Sacred Frequencies," *Ableton,* September 25, 2012, accessed April 28, 2015, https://www.ableton.com/en/blog/machinedrum/.

On the other hand, some producers do embrace the influx of digital tools as a positive force in shaping the direction of electronic music. Los Angeles-based producer Flying Lotus sees the constant build-up of musical production tools over time as an opportunity for artists to experiment with the limits of electronic music composition and performance:

> Why not just have all these things from our past as well as all of the newest technology from today in one, and just really come up with the craziest shit we can?... With as much access as we have to all this stuff, to our musical history, our world history, we definitely can be killing shit way crazier... We have the technology![94]

Significantly, FlyLo conflates access to music technology with a general knowledge of world history, highlighting the ways in which technological accessibility is bound up with broader social and economic values regarding the unfettered acquisition of commodities in consumer capitalism. Here, the *accumulation* of technological objects, creative techniques, and forms of knowledge highlights an emerging dialogue between the *aesthetics* of software design, and the *technical practices* of composition and performance in contemporary digital culture.[95]

Machinedrum and Flying Lotus offer disparate responses to the maximalist situation in contemporary music production. As much as the influx of creative affordances through software can feel liberating and empowering, as mentioned, it can also produce a feeling of information overload. Paradoxically, the cultural response to this situation among digital music educators has

[94] Joe Muggs, "Flying Lotus: Cosmic Drama," *Resident Advisor,* March 24, 2010, accessed April 20, 2015, http://www.residentadvisor.net/feature.aspx?1175.

[95] While they played crucial roles in the formation of ethnomusicology and musicology as academic disciplines, organology and sketch studies have very much fallen by the wayside in the past twenty years. In an attempt to understand contemporary poiesis, this dissertation presents "interface aesthetics" as a new analytical model that merges recent work in "critical organology" with the ontological concerns of sketch studies. These fields are increasingly significant in the digital era, as both instruments and musical materials ("sketches") move from the museum, archive, and recording studio, to the hard drive. See, for example, Friedemann Sallis, *Music Sketches* (Cambridge: Cambridge University Press, 2015); Eliot Bates, "The Social Life of Musical Instruments," *Ethnomusicology* 56, no. 3 (Fall 2012): 363-95; Emily Dolan, "Towards a Musicology of Interfaces," *Keyboard Perspectives* 5 (2012).

been to produce even more tools and techniques for navigating the changing landscape of music software. As a result, an entire economy of pedagogical materials has arose for the purpose of instructing musicians on how to navigate new digital music production technologies. The development of this new economy has been facilitated by two parallel developments: first, software marketing schemes that promote the democratization and accessibility of *Live*; and second, the emergence of digital music production schools and other educational content.

Since its first commercial release in 2001, *Live* has come to exemplify a certain DIY aesthetic of technological accessibility that characterizes the design, distribution, and use of music software. Most noticeably, it has been at the forefront of user communities surrounding tools, techniques, and trends in digital audio production, often providing the headlines of magazines such as *Computer Music, Future Music,* and *Music Tech,* as well as online blogs such as *Create Digital Music.* These outlets simultaneously market new software and hardware to consumers, provide examples of technological use through artist interviews, and offer practical insights on how to use the tools through tutorials (fig. 1.14). In doing so, they encourage a broad accessibility and use of music software across amateur and professional demographics. The concomitant emergence of new techniques for sound manipulation and new forms of electronic music pedagogy further highlights the shift in conceptions of software from an archival medium onto which users inscribe a fixed *text* to a performance medium with which a new type of musician collaborates in the creative *process.*

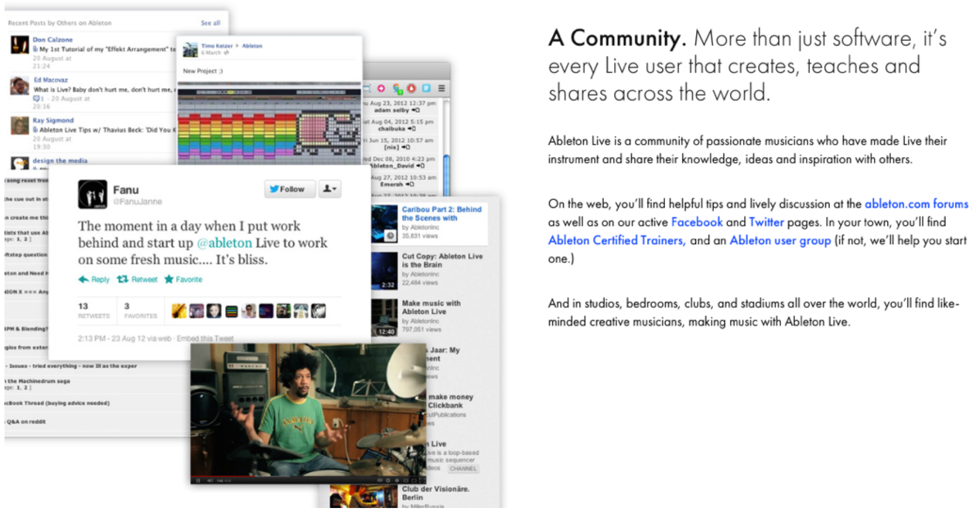


Figure 1.14: *Live 9* promotional material, emphasizing the idea of the software as not only technical components, but an entire "community." This gesture attempts to network the creative processes of "bedroom" producers with a global community of computer musicians (Source: Ableton, "Live," accessed April 22, 2015, https://www.ableton.com/en/live/).

In line with a marketing model based on the perceived democratization of new technologies for music production, the emergence of *Live* has ushered in a network of pedagogical practices to meet the needs of aspiring digital music composers.[96] In addition to the inclusion of *Live* in countless music technology courses at the university level, dedicated hybrid production schools have emerged, providing online and face-to-face instruction in a variety of digital audio topics. For example, *Dubspot* hosts courses on DJing, turntablism, sound design, mixing and mastering, music theory, and a special certificate program in *Live* (fig. 1.15). Instructors include professional DJs, producers, and sound engineers with specializations in specific software, as well as *Live* "Certified Trainers"—educators who have successfully

[96] Following Walter Benjamin's argument about photography in "The Work of Art in the Age of Mechanical Reproduction," Arild Bergh and Tia DeNora (2009) examine the ways in which recording and distribution technologies "have potentially democratised the field of aesthetic music experiences," and erased "the line between listener/fan and record producer/patron." For a comprehensive overview of the term "democratization" as it has been used in various strains of social and cultural theory, see Patryk Galuszka, "Netlabels and democratization of the recording industry," *First Monday* 17, no. 7 (July 2012).

completed Ableton's in-house certification program. Together, the discourse and pedagogy surrounding digital audio production in the 2000s has been defined by an increased desire to market the software to as broad an audience as possible, and to stabilize its use among instructors and artists across artistic disciplines, reifying the maximalist aesthetic previously outlined.

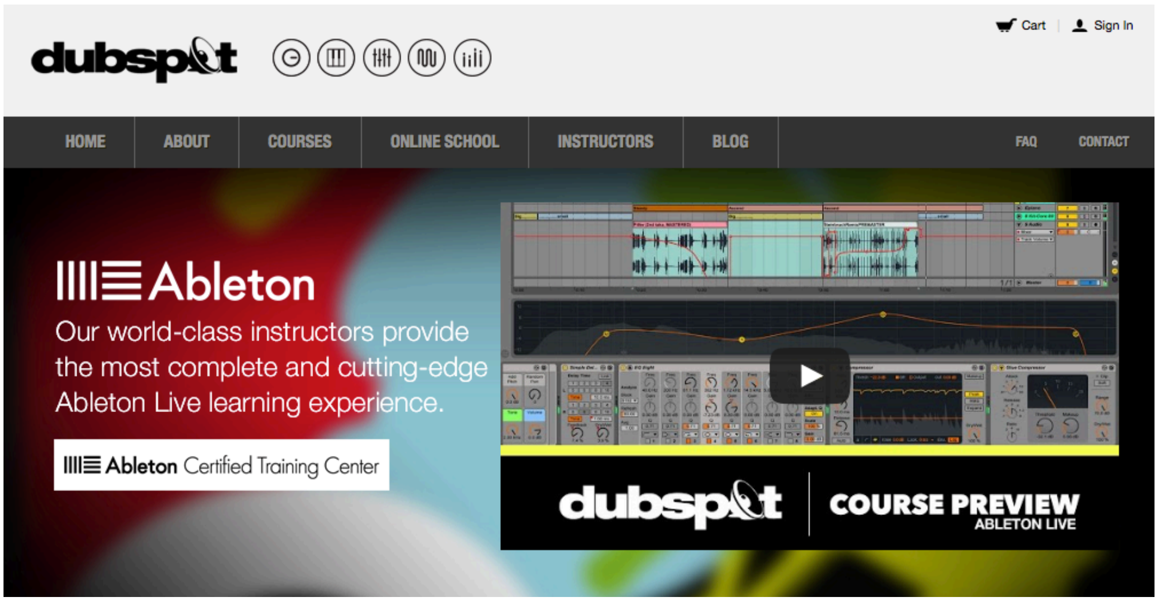


Figure 1.15: Dubspot *Live* course promotional material. The emphasis on certification reflects the standardization of electronic music pedagogy as musicians negotiate emerging technologies. (Source: Dubspot, "Ableton Live," accessed April 22, 2015, http://www.dubspot.com/ableton-live/).

Electronic musician Dennis DeSantis' book *Making Music: Creative Strategies for Electronic Music Producers* encapsulates both the struggle of maneuvering a digital software market that is growing exponentially, and the maximalist creative mindset towards composition suggested by DAWs such as Live. As Ableton's Head of Documentation, DeSantis makes a living translating changes in interface design to a dynamic consumer base. However, the book has a different aim. Rather than teaching the technical details of audio production, *Making Music* deals with the psychological hurdles that producers must overcome in order to compose music in the maximalist twenty-first century. Why is music still hard to create in the current "golden age of tools and technology," in which "a ninety-nine-cent smartphone app can give you the functionality of a million-dollar recording studio" and "tutorials for every sound design or music production technique can be found through a Google search"?[97]

DeSantis' project begins from the premise that contemporary life is full of distractions—a logic that is embedded within the multi-windowed "desktop" environment of the computer itself—and that successful electronic music production is only possible by filtering and limiting these options. In the first section of the book, "Problems of Beginning," he discusses the importance of "resisting gear lust" and "limiting acquisition of plug-ins and virtual instruments" in surpassing the feeling of inaction that often results from confronting the overwhelming nature of digital tools.[98] In line with the modular nature of the *Live* interface, DeSantis suggests "goal-less exploration… the process of simply finding a corner of your working space and letting yourself see what evolves from there" as a non-linear solution to the feeling that one must incorporate the entire DAW into their workflow.[99] Focusing on the *process* of learning discrete elements of the software interface rather than simply attempting to churn out marketable *products* prevents the musician from being too constrained by "the *mindset* of these new tools," instead allowing him or her to concentrate on compositional ideas such as track content, structure, and arrangement.[100]

DeSantis conceives of music composition through sculptural metaphors, emphasizing the process of subtraction in opposition to the additive logic and aesthetic of accumulation inherent to the design and marketing of production technologies. In terms of musical structure, DeSantis

---

[97] "What is this Book?" *Ableton,* n.d., accessed April 26, 2015, https://makingmusic.ableton.com/about.

[98] Dennis DeSantis, *Making Music: Creative Strategies for Electronic Music Producers* (Berlin: Ableton AG, 2015), 58.

[99] Ibid, 79.

[100] Ibid, 59.

suggests starting from the point of "maximal density," organizing the arrangement around that moment and subsequently chipping away extraneous musical content as needed. Whereas a dominant aesthetic of contemporary digital music production views composition in generative terms, the algorithmic nature of digital tools facilitating the seemingly "organic" growth of musical content, chapters such as "Arranging as a subtractive process" and "The power of erasing" value the *deduction* of content as a strategy in facilitating productivity.[101]

For DeSantis, navigating the maximal interface of *Live* requires a minimalist creative mindset. This paradox of consumer culture is the defining feature of maximalism. As the publication of DeSantis' book implies, in order to successfully remove the distractions inherent to technological use in the age of digital convergence, the consumer must keep consuming. In this context of perpetual consumption, maximalism is an expression of an ideology "designed to sell not only a particular commodity but consumption itself," as ethnomusicologist Timothy D. Taylor suggests.[102] Maximalism is thus defined as a process-oriented activity that values the *experience of consumption* over the consumer *product.* In this way, the broader social, cultural, and economic situation of maximalism aligns with the accumulative, process-oriented, and plugin-based design of *Live* itself, as described earlier.

In analyzing the isomorphic relationship between the technical design of *Live* and the sociocultural network through which it proliferates, this chapter has provided an account of the

---

[101] I further discuss the discourse and aesthetics of generative art and composition in chapters two and five, in relation to algorithmic composition and procedural content generation in game design, respectively. DeSantis briefly defines this aesthetic in relation to music, specifically using it as a springboard for his own counterargument of music as a subtractive process: "The idea of "process" in music refers to the development of a system or set of rules that allows some or all of the elements of the music to be generated or derived without requiring the composer to make purely intuitive creative decisions at every possible moment" (108).

[102] Timothy Taylor, *The Sounds of Capitalism: Advertising, Music, and the Conquest of Culture* (Chicago: The University of Chicago Press, 2012), 4.

ways in which media production software in the 2000s moves beyond and through the metaphorical perforations of the computer screen. Perhaps more than any other software, *Live* is the tool of choice on the *stages* of electronic dance music festivals and multimedia art installations, as well as the *studios* of film composers, video game sound designers, and "bedroom" electronic musicians and DJs. A growing network of music pedagogues has contributed to the widespread acceptance of the software by serving as mediator between software designers and emerging practices of electronic music, helping amateur and professional musicians to negotiate constantly changing design trends. The literal mobility and perceived accessibility of the software throughout social and cultural spaces has contributed to a dominant aesthetic of technological accumulation and maximalist consumption that has defined media production since the new millennium.

However, the maximalist aesthetics embedded within what we might term "plugin cultures" are not the only modes of computational practice available to electronic musicians, software designers, and instrument makers. As much as the maximal interface of the DAW evokes the promise of unlimited creative options through the additive plugin mechanics embedded within the seemingly transparent software, others perceive this influx of prescribed functionality to be creatively limiting. Describing what he calls the "deadly embrace between software and users," Miller Puckette claims that "no matter how general and powerful we believe today's software to be, it is in fact steeped in tacit assumptions about music making that restrict the field of musical possibility."[103] From this perspective, it makes perfect sense that the festival culture of electronic dance music—with all of its sonic grandiosity, digital bombast, and

[103] Miller Puckette, "The Deadly Embrace Between Music Software and Its Users," in *Electroacoustic Music Beyond Performance: Proceedings of the Electroacoustic Music Studies Network Conference, Berlin, 2014*.

excessive "drops" that overwhelm the sensorium—has become the genre par excellence for *Live* users. Whereas DeSantis and other electronic music pedagogues are concerned with introducing new compositional mindsets to the electronic musician in an attempt to help them navigate the inherent complexity of the DAW, Puckette argues that "an even more powerful strategy for managing complexity is simply to avoid it altogether."[104] In the next chapter, I test this minimalist proposition by reversing the questions posed by the design of *Live*: rather than being presented with an influx of creative options by the maximal interface, how have electronic musicians approached software that seems to provide no options at all? What is the nature of music production, instrumentality, and musicianship when the computer screen is akin to a blank canvas?

[104] Ibid.

Chapter 2

Programming Sound: Computational Thinking in Electronic Music

The Thing that Makes the Thing is More Interesting than the Thing.
– Casey Reas[105]

Software developer, music educator, and mathematician, Miller Puckette is first and foremost a pragmatic individual. When it comes to technology, he values limited tools that do seemingly unlimited things, embracing a "less is more" mentality in both his professional work and everyday life (fig. 2.1). In direct opposition to the "upgrade culture" of Apple and the rest of Silicon Valley, Puckette comprehends an almost zen-like utility in technologies that do the job right without getting in the way. Guided by what he calls the "universal principles of computer science"—portability, abstraction, reusability, and interoperability of code—Puckette's research at IRCAM in the late 1980s led to the development of *Max,* a pioneering software for process-oriented electroacoustic and real-time music performance, as well as *Pure Data,* its open source equivalent. Distinctly critical of the ways in which emerging tools for music production guide the user down a predefined path through flashy interfaces and novel mechanics, the "Max Paradigm" instead presents the user with the digital equivalent of a blank canvas: an empty screen.[106]

---

[105] "Casey Reas: The Thing that Makes the Thing is More Interesting than the Thing," *UM Stamps,* uploaded March 16, 2015, accessed May 15, 2016, https://www.youtube.com/watch?v=lu51j9V-ti0.

[106] Puckette named *Max/MSP* in honor of computer music pioneer Max Matthews. What he refers to as the "Max Paradigm" includes three computer programs—*Max/MSP, Jmax,* and *Pure Data* (Pd). For more information, see Miller Puckette, "Max at Seventeen" *Computer Music Journal* 26, no. 4 (2002): 31-43.

Figure 2.1: Miller Puckette demonstrates how to install *Pure Data* on a Raspberry Pi microcomputer (Source: "Miller Puckette – Pure Data on the Raspberry Pi," Alexander Matthews, October 26, 2012, accessed May 1, 2016, https://vimeo.com/52265243).

While the idea of building an entire working system from the ground up is a familiar challenge to computer programmers, it is certainly new to the digital musician used to working with the familiar studio interface metaphors of knobs, sliders, and linear audio editing as discussed in chapter one. How is it possible to talk about the affordances and constraints of musical interfaces in the context of software that leaves the primary design responsibilities to a user simply presented with a blank screen? In other words, what is the nature of musical instrumentality in the age of computational thinking?

Not simply a tool for recording music, *Max* is a visual programming "environment" for music production that has become increasingly popular among digital artists since its first commercial release in 1990.[107] While many commercial digital audio workstations (DAWs) present the user with a linear, timeline-based interface best suited for music *recording, Max* specializes in process-oriented digital art, presenting the user with a blank screen onto which various interactive "objects"—building blocks used to create programs in *Max*—are added. Many have lauded the open, flexible, and adaptive nature of the software, often comparing the creative experience of *Max* to the seemingly all-access control one has while programming computers. Yet, while artists have been quick to praise the effects of this software on their own work, the procedural and computational aesthetics inherent to the program are rarely contextualized outside of artistic communities.

Through an analysis of the technical design of *Max*, as well as considering the creative practices of its users, this chapter offers insights into the ways in which existing notions of musical instrumentality can be expanded as a result of the increasing integration of musical concepts and computational processes. As music composition increasingly converges with digital software, instrument designers develop interfaces that borrow from a range of media design disciplines, thus constantly introducing new sets of creative affordances for musicians and composers. Much of the research gathered have been inflected by my first-hand experiences learning and building media applications in *Max*, as well as conversations with engineers and musicians that have played key roles in the software's development and distribution.[108] The chapter begins by briefly surveying software cultures that have enabled new forms of digital

[107] *Cycling 74,* the company that sells Max, specifically describes the software as "a full kit of creative tools for sound, graphics, music and interactivity in a visual environment." "Cycling '74 Max," accessed April 18, 2016, https://cycling74.com.

[108] For example, see "Notes: An App to Think Through Your Music," a program I developed in Max for students to learn critical listening skills. Mike D'Errico, "Notes," accessed April 18, 2016, http://www.derricomusic.com/portfolio_page/notes-think-through-your-music/.

literacy, including the free software movement, maker culture, and *live* coding performances. Rather than building concrete products, these communities of "do-it-yourself" (DIY) artisans emphasize the *processes* of learning how to interact with computers in meaningful ways. Consequently, I align these new forms of digital literacy with the pedagogical framework of computational thinking—a theory that emphasizes not only the ways in which individuals learn how to use computers, but also how computers themselves shape individuals' understandings of culture and creativity more broadly.

In applying these principles to musical practices, I introduce the practice of "procedural listening" as a theoretical model that describes specifically how electronic musicians come to learn musical concepts when engaging non-linear computational processes. Procedural listening represents a new form of instrumentality in the digital age: a dominant skill developed by audio producers that allows both musicians and audiences to focus on the process-oriented mechanics of media *forms,* rather than simply audio *content.* As musicians embrace procedural listening in order to understand the inner workings of software, their conceptions of musical instrumentality shift from a *tool*-based model in which the instrument is a concrete means to a specific musical goal, to a *system*-based model in which the instrument is part of a more integrated technological network used to explore previously distinct creative possibilities. Ultimately, I extend the argument of chapter one, illustrating the ways in which broader cultural values are embedded in the design and aesthetics of musical software.

Digital Literacy and Cultural Practice

On the afternoon of Monday, December 8, 2014, President Barack Obama made history in becoming the first president of the United States to write a line of computer code. During a promotional event for Computer Science Education Week, the President sat down with a group of middle-school students in New Jersey to speak about the importance of computer science in public education, following a major announcement that the White House would be donating over $20 million in contributions to Code.org in their efforts to train 10,000 teachers in computer science education by the beginning of fall 2016 (fig. 2.2). Obama's coding session was just one of many media events to promote the "Hour of Code" campaign—an international non-profit developed to promote core values of computer science, including technological accessibility and digital literacy.[109] Since the program's launch in December of 2013, various celebrities—from Silicon Valley giants Mark Zuckerberg and Bill Gates, to actors Ashton Kutcher and Angela Bassett—have personally expressed support for the campaign, promoting the value of computer coding to grade school students, soldiers, and doctors alike.

[109] From the organization's website: "The Hour of Code is a global movement reaching tens of millions of students in 180+ countries. Anyone, anywhere can organize an Hour of Code event. One-hour tutorials are available in over 30 languages. No experience needed. Ages 4 to 104." Code.org, "Hour of Code," accessed December 12, 2014, http://hourofcode.com/us. See also, Microsoft, "Get Your Start With an Hour of Code," accessed December 12, 2014, http://www.microsoft.com/about/corporatecitizenship/en-us/youthspark/youthsparkhub/hourofcode/.

Figure 2.2: Coding session with President Barack Obama (2014) (Source: Klint Finley, "Obama Becomes First President to Write a Computer Program," December 8, 2014, accessed May 1, 2016, http://www.wired.com/2014/12/obama-becomes-first-president-write-computer-program/).

The "Hour of Code" is only one example of a broader culture that supports digital literacy across various communities of media artists.[110] Indeed, the bolstering of STEM and computer science education in public schools runs parallel with the proliferation of three cultural communities that foster similar values: the free software movement, "maker" culture, and live coding. Together, these scenes exemplify the ways in which the broader push for digital literacy has ushered in more relational forms of human-computer interaction (HCI) in the practices of computer engineers, media artists, and musicians. If the previous archetype for computer programming was the lone hacker, in full control of the software in front of him, these new

[110] See Henry Jenkins, *Confronting the Challenges of Participatory Culture: Media Education for the 21st Century* (Cambridge, MA: The MIT Press, 2009).

models of HCI conceptualize computing as a two-way, relational process of negotiation between the knowledge of the individual and the affordances of the software. This ongoing process of negotiation between a software and its "user" is fundamental to the practice of procedural listening.

Since the early 1980s, the "free software" movement has advocated for the uninhibited use, distribution, and modification of software. Computer programmer Richard Stallman's "GNU Project"—an attempt to create a computer operating system composed entirely of free software—established the values of this movement in direct opposition to the proprietary ownership model enacted by major technology corporations such as Apple and Microsoft. The following table outlines commonly perceived differences between the two models of software distribution (table 2.1):

Table 2.1

FREE SOFTWARE	PROPRIETARY SOFTWARE
Free to use	Pay to use
Online distribution	Commercial Distribution
Source code available for use	Source code unavailable for use
Emphasizes "hackability"	Emphasizes "usability"
Requires more time to learn	Requires less time to learn
Less technical support	More technical support
Anarchist-libertarian ethic of use	Democratic ethic of use
Values "freedom"	Values "innovation"

The core differences in software *design* between the two models comes down to the question of "hackability" versus "usability." In the case of free software: does the software offer greater possibilities for custom modification at the expense of being "user-friendly"? Or, in the case of proprietary software: is the software accessible to a greater range of users at the expense of being open to modifications? Digital literacy lies at the root of these concerns.[111] In order to embrace the full capabilities and extended functionality of free software, the user must overcome a gradual learning curve which often involves the development of computer coding skills. The average computer user may not be willing to dedicate the time and energy to this task. In contrast, advocates of free software see digital literacy as not only a technical challenge, but a political and ethical responsibility for reclaiming control of digital media culture from the large corporations of Silicon Valley.

With the introduction of cheap microcomputers such as the Raspberry Pi, the free software movement converged with a broader range of DIY creatives to form what has become known as "maker culture." This informal network of computer hackers, robotics enthusiasts, 3D printers, and artisans working in the traditional arts and crafts has been guided by the principles of learning-through-doing, hacktivism, and shared knowledge formation (fig. 2.3).[112] Significantly, maker culture has inspired an even greater push for computer science education in America, as microcomputers continue to be used as tools in teaching children how to code. Similar to the free software movement, "makers" see themselves in direct opposition to corporate ideologies of technological "innovation" and "progress," instead valuing knowledge formation from the ground up through amateur production practices.

[111] Doug Belshaw, *The Essential Elements of Digital Literacies,* available from http://dougbelshaw.com/ebooks/digilit/ under a CC BY license (2011).

[112] Mike Sharples et al., "Maker Culture: Learning By Making," in *Innovating Pedagogy 2013: Open University Innovation Report 2* (Milton Keynes, UK: The Open University, 2013), 33-5.

Figure 2.3: Paris Maker Faire (2015) (Source: "35,000 visitors at the Paris Maker Faire, a record," ***Makery,*** **May 4, 2015, accessed April 29, 2016, http://www.makery.info/en/2015/05/04/35000-visiteurs-a-la-maker-faire-paris-un-record/).**

Finally, digital literacy has been given a performative dimension by computer programmers who engage in "live coding," in which an on-stage programmer projects their computer screen to an audience while they code software in real-time.[113] Live coding is practiced in a range of art forms, including dance, poetry, and music, typically highlighting the technical virtuosity and improvisational skills of the programmer (fig. 2.4). In addition to the entertaining and community building mode of performance, live coding has an educational function. Online streaming hubs, such as livecoding.tv, often host instructional tutorials to help viewers learn computer programming techniques.[114] The combined performative and pedagogical functions of

[113] Ilias Bergstrom and R. Beau Lotto, "Code Bending: A New Creative Coding Practice," *Leonardo* 48, no. 1 (2015); Eduardo Ledesma, "The Poetics and Politics of Computer Code in Latin America: Codewor, Code Art, and Live Coding," *Revista de Estudios Hispánicos* 49, no. 1 (2015): 91-120; Alex McLean, Julian Rohrhuber, and Nick Collins, eds., [Special Issue on Live Coding], *Computer Music Journal* 38, no. 1 (2014); Nick Collins, "Live Coding of Consequence," *Leonardo* 44, no. 3 (2011): 207-11; Thor Magnusson, "Algorithms as Scores: Coding Live Music," *Leonardo Music Journal* 21 (2011): 19-23.

[114] *Livecoding.tv,* "Livecoding: Watch People Code Products Live," accessed April 5, 2016, https://www.livecoding.tv.

live coding foreground a community advocating for digital literacy by showing their audiences the process of coding, rather than just sitting behind a screen and "pressing play."[115]

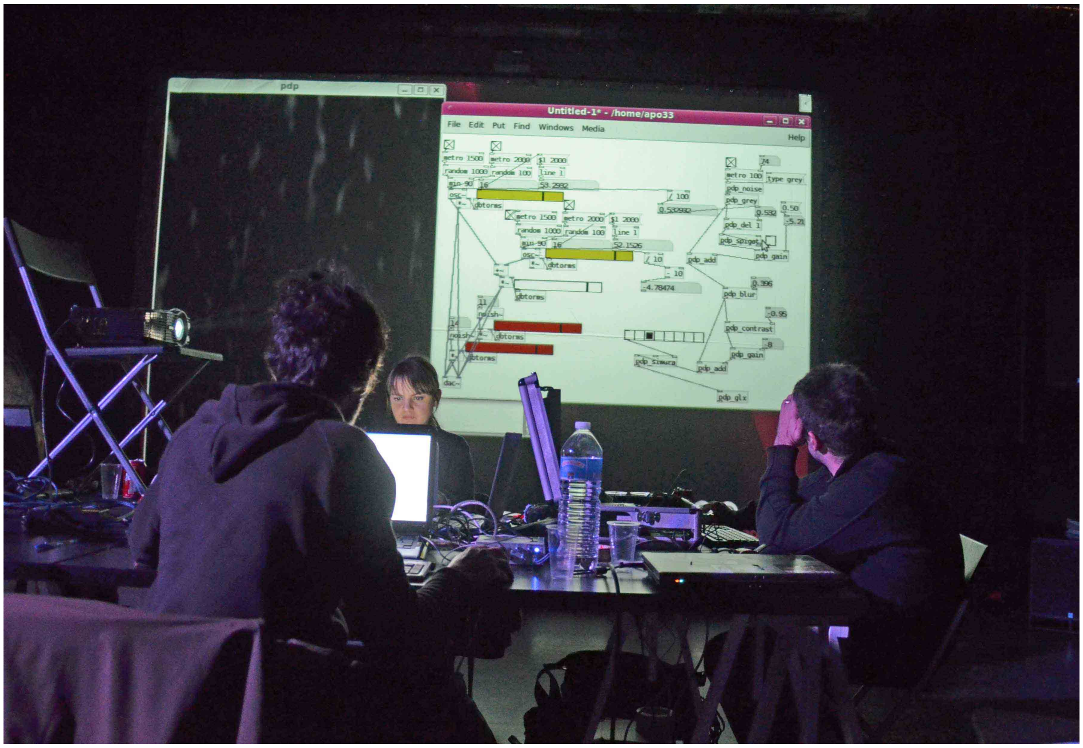


Figure 2.4: Live coding session in "La Fabrique" (Nantes, France, 2012) (Source: "Live coding session in La Fabrique (Nantes, France)," n.d., accessed April 29, 2016, http://diccan.com/Eicl.htm).

The free software movement, maker culture, and live coding represent concrete manifestations of the increased presence and influence of computer science principles on contemporary digital culture. By studying these communities on a practical level, it is possible to outline how the countercultural politics of technological alterity to Silicon Valley become fixed within the *cultural practices* of DIY computing.[116] However, despite the ways in which these communities externalize the mechanics of software to their audiences, tracing cultural aesthetics

---

[115] Chapter three discusses in more detail historical and contemporary debates surrounding live performance with computers.

[116] Chris Kelty, *Two Bits: The Cultural Significance of Free Software* (Durham, NC: Duke University Press, 2008).

as they manifest in the *technical design* of software itself is not so obvious. Focusing on the software comprising the *Max* paradigm, this chapter outlines the ways in which software design may epitomize the principles of digital literacy, principles that drive the creative practices of contemporary digital cultures. As both musical instruments and forms of musical notation, software in the *Max* paradigm encourages unique forms of HCI: composers become designers, designers become listeners, and listeners become performers. In order to establish the coterminous evolution of design and aesthetics in software development, I will briefly trace the historical emergence of the most popular programs in the *Max* paradigm: *Max* and *Pure Data*.

In many ways, the history of the *Max* paradigm traces the dichotomies between free and proprietary software previously discussed. Puckette conceptualized ideas for a real-time computer control solution for music in the 1980s at the infamous state-funded Institut de Recherche et Coordination Acoustique/Musique (IRCAM) in Paris. At the time, the center espoused a modernist ideology that valued newness and innovation above all else, which included "a hostility and contempt toward all commercial developments and especially 'low-tech' or small consumer technologies."[117] Software was a specific target, "denigrated for having no such physical embodiment, no object form, for being insubstantial and ephemeral."[118] As a result, most of the software developed at IRCAM after the advent of digital computers involved highly opaque computer programming languages that required highly specialized knowledge in general computing, acoustics, music composition, and electronic music.[119] Even in the realm of

[117] Georgina Born, *Rationalizing Culture: IRCAM, Boulez, and the Institutionalization of the Musical Avant-Garde* (Berkeley and Los Angeles: University of California Press, 1995), 184.

[118] Ibid, 233.

[119] While code appears to hold close affinity to natural language, Born notes that programming languages are "far from transparent to decode, even for the highly skilled authors themselves. However, IRCAM programmers seemed to delight in this intransigent opacity since, despite the many difficulties that it

hardware, Pierre Boulez—the director of IRCAM—reportedly told the pedagogy director that the "user-friendly" Apple Macintosh computers would come into the institute "over my dead body."[120]

Max represented a radical departure from the dominant modernist ideology of IRCAM in at least two ways. First, it is a software with a graphical user interface (GUI) known as a "patcher," rather than a text-based programming language (fig. 2.5).[121] The "patcher" design layout afforded more concrete opportunities for real-time music making, which was the primary goal of the software. Second, *Max* was designed fundamentally as a performance tool, rather than a music "compiler" whose primary purpose was to produce notated scores or audio recordings. Despite the advent of powerful *commercial* digital sampling hardware such as the Fairlight computer, which allowed for a certain level of "real-time" audio editing, the anti-commercial stance of Boulez caused IRCAM to rely instead on outdated machines. As in the early days of computer music, just to hear brief snippets of their music, composers had to wait hours—sometimes days—for the sounds to be compiled by the machine. In addition to the software's emphasis on performance and its use of a GUI, the *Max* design also departed from IRCAM ideology by integrating the "commercial" Apple Macintosh into the performance setup, eventually enabling the first version of *Max* to be used on stage.[122]

---

caused, it made programs appear artful and unstandardized expressions of collective imaginative labor." *Rationalizing Culture,* 230.

[120] Ibid, 284.

[121] Miller Puckette, "The Patcher," in *Proceedings of the International Computer Music Asssociation* (San Francisco: International Computer Music Association, 1988), 420-9.

[122] Puckette describes the Macintosh computer that was brought to IRCAM in 1987 by David Wessel, "without whose efforts I and the rest of IRCAM might have entirely missed out on the personal computer revolution." *Max at Seventeen,* 34.

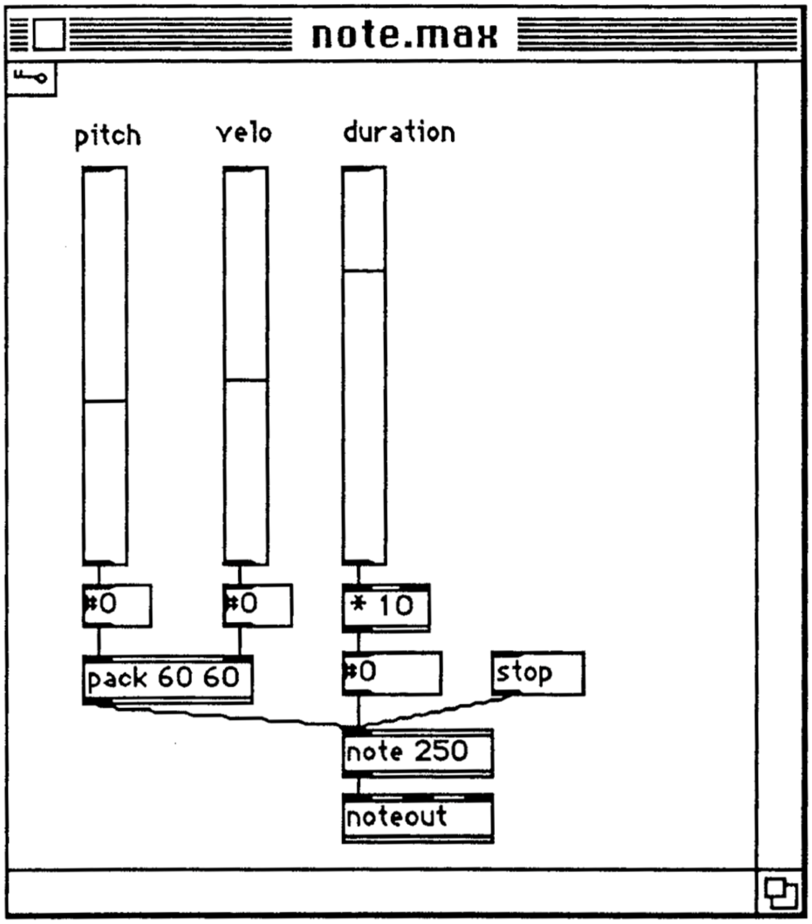


Figure 2.5: Original *Max* "patch" prototype (1988) (Source: Miller Puckette, "The Patcher," November 8, 1988, accessed May 1, 2016, http://msp.ucsd.edu/Publications/icmc88.pdf).

The discrepancy between user-friendly and programmable software design played out not only in the development of *Max* at IRCAM, but also in its subsequent commercial distribution beyond IRCAM. In 1989, IRCAM licensed the software to Opcode Systems, which sold the first commercial version in 1990. In 1996, Puckette developed *Pure Data (Pd)* as an open-source equivalent to *Max*, in an effort to make several improvements to the program. Since then, *Pd* has become central to the free software movement previously outlined. Meanwhile, *Max* has followed a more proprietary route, aligning itself with popular "commercial" software such as Ableton *Live*, and incorporating a more plugin-based interface design. Despite these differences, the *Max* paradigm continues to offer modular platforms for creating process-oriented musical experiences to digital artists across media. Additionally, the "visual programming" interface design inherent to the *Max* paradigm encourages users to develop digital literacies more akin to computer programming itself, rather than musical composition. Before considering the ways in which the paradigm encourages procedural listening and results in new forms of instrumentality amongst musicians, I will provide some practical examples of how *Max* is used.

The capabilities for interfacing among hardware devices and software applications has inspired both the creative practices of *Max* users, as well as their conceptions of music composition in digital environments. Among electroacoustic composers and others working in the Western art music tradition, *Max* has been used to supplement acoustic performance with real-time and improvisational electronics. In an interview with *Cycling 74,* the software development company that commercially distributes *Max*, computer music composer William Kleinsasser talks about the ways in which the software introduces new concepts and methods into the creative process. Describing the rise of what he calls the "integrative composer," Kleinsasser claims that "many composers now speak of a blurring difference between composition and programming."[123] The rise of the composer-programmer is a prime example of the ways in which the digital literacy afforded by *Max* has introduced a modular, systems-based approach to the more traditional compositional processes of Western art music.

Similarly, among electronic music producers working in various strains of popular music, the modular approach to composition in *Max* is embraced as a way of pushing the music into more experimental territory. In his work with Radiohead, Jonny Greenwood has used *Max* as a

[123] David Zicarelli, "An Interview with William Kleinsasser," *Cycling 74,* September 13, 2005, accessed December 11, 2014, https://cycling74.com/2005/09/13/an-interview-with-william-kleinsasser/.

respite from "all those programs [that] seemed desperate for you to write in 4/4 at 120 bpm and loop the first 4 bars."[124] Los Angeles beatmaker Daedelus uses the software in conjunction with various hardware interfaces to stretch, stutter, and juxtapose fragments of existing tracks. Here, *Max* is being used as a sort of hypersampler, in which an open interface allows multiple loops to be simultaneously micro-edited and sequenced at various tempos (fig. 2.6).[125]

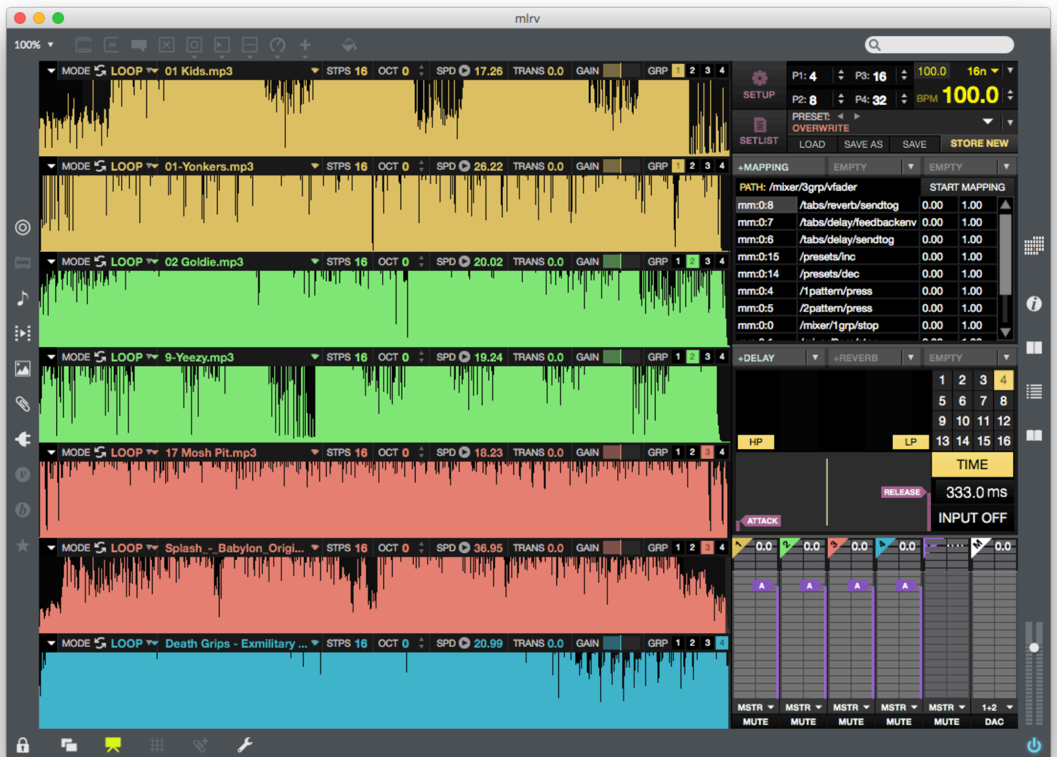


Figure 2.6: MLRv 2.2 (2011), a *Max* patch by Galapagoose, allows users to simultaneously edit and playback multiple audio loops

[124] Andrew Pask, "Mini Interview: Jonny Greenwood," *Cycling 74,* January 2, 2014, accessed March 30, 2016, https://cycling74.com/2014/01/02/mini-interview-jonny-greenwood/.

[125] Editing parameters at the "micro" level of an audio sample is possible as a result of music software that allows for high resolution digital sampling in the first place. Similar techniques are outlined in Paul Harkins, "Microsampling: From Akufen's Microhouse to Todd Edwards and the Sound of UK Garage," in *Musical Rhythm in the Age of Digital Reproduction,* edited by Anne Danielsen, 177-94 (Burlington, VT: Ashgate, 2012).

While the software has been diversely employed for the creation of music, other users have integrated *Max* into broader media applications. In the *Skube* media player project, Andrew Spitz created an interactive Spotify radio through the combination of an Arduino board, *Max* software, and the Last.fm API.[126] Skubes exist as physical, cube-shaped speakers that react to being flipped, tapped, and connected to each other by either shuffling music from a playlist, or selecting music based on songs previously accessed by the listener (fig. 2.7). In this case, *Max* is specifically used to both interface with the physical cube through the Arduino, and to control audio playback in Spotify through its integration with a custom-coded Applescript.

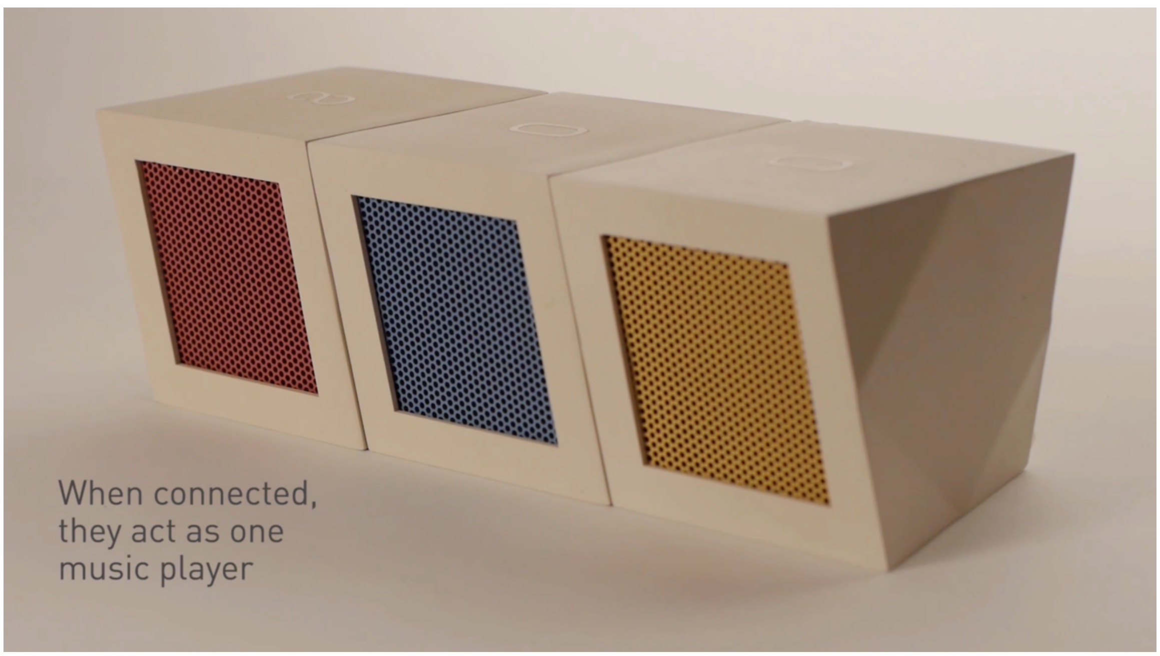


Figure 2.7: *Skube* promotional video (Source: "Skube – A Last.fm & Spotify Radio," Andrew Nip, September 12, 2012, accessed May 1, 2016, https://vimeo.com/49343337).

In contrast to the instrumental design model described in chapter one, using music software in the ways just listed requires computational forms of digital literacy which privilege

[126] Andrew Spitz, "Skube—A Last.fm & Spotify Radio," *Sound + Design,* October 16, 2012, accessed December 12, 2014, http://www.soundplusdesign.com/?p=5516.

system-oriented thinking, technological interoperability, and abstract problem-solving. In short, the *Max* paradigm encourages music *composers*—in the broad sense of the word—to think like computer *programmers*. The next section of this chapter introduces the core elements of this creative mode of computational thinking, followed by a discussion of the ways in which computational thinking—as a creative approach to digital literacy and media production—influences broader conceptions of musical instrumentality and HCI.

Computational Thinking and Proceduralism

If digital literacy focuses on the *processes* through which humans learn how to use computers, computational thinking comprises the *attitude* that individuals hold toward creative production with digital tools. In an effort to detail the ways in which working with computers might facilitate this shift in the cognitive and creative capacities of human beings, computer scientist Jeanette Wing defines the term computational thinking as a "fundamental skill for everyone, not just for computer scientists," which involves "solving problems, designing systems, and understanding human behavior by drawing on the concepts fundamental to computer science."[127] For Wing, computational thinking is more than just the ability to program a computer, as it requires a skillset to work abstractly with entire systems, rather than just its individual elements; to reframe and reformulate seemingly difficult problems into simpler languages through simulation; to think through both theoretical and practical solutions to a given problem; and, most significantly, to discover solutions in the presence of uncertainty.

[127] Jeanette Wing, "Computational Thinking," *Communications of the ACM* 49, no. 3 (2006): 33-5.

The skills captured by computational thinking prove useful not only in understanding digital interfaces of various sorts, but also the mechanics of musical instruments and compositional systems. Music educator Gena Greher and computer scientist Jesse Heines introduce the idea of "computational thinking in sound" as a pedagogical tool for music and computer science undergraduate majors, both of which are "hampered by habit, which limits their abilities to imagine alternative possibilities."[128] Developing interdisciplinary skillsets through collaboration is crucial to computational thinking in sound. For example, the ability to view musical structures as a hierarchical set of data allows students to understand melody, lyrics, and song form as chunks of connected data from which they may be able to extract meaningful patterns. Further, Greher and Heines offer practical activities to help students "crack the code" of various forms of musical notation (Western scores, MIDI piano rolls, audio waveforms, etc.), encouraging the development of abstract notation schemes that rely less on systems of representational metaphors that have become reified throughout the history of Western music, and more on the seemingly counter-intuitive mechanics and awkward physical gestures required of students when they first encounter a new instrument or interface. Greher and Heines are particularly concerned with strategies for facilitating student learning in educational contexts, and scholars have yet to examine the ways in which computational thinking has shaped the design and use of musical instruments and compositional tools. On a deeper level, we might ask: how might *computational* attitudes towards music-making influence shifting understandings of fundamental concepts such as instrumentality, composition, and performance? In attempting to

[128] Gena R. Greher and Jesse M. Heines, *Computational Thinking in Sound: Teaching the Art and Science of Music and Technology* (New York: Oxford University Press, 2014), 4.

outline a practical model for computational thinking in digital music production, I will contextualize by considering some of the working theories and principles of computer science.

With the rise of software studies, media scholars across disciplines have become increasingly interested in algorithms, code, and computational procedures. The efficiency of well-constructed code, the ability of software to evolve in relation to its hardware environment, the conceptual abstraction of code into a series of programmatic steps, and the recursion and repetition of digital processes are just a few of the values that continue to surface in the theoretical discourses and educational curricula surrounding computer science.[129] Together, these values constitute a general aesthetic of minimalist elegance and interoperability in the construction and operation of computational systems. Software developers, game designers, and electronic musicians alike apply these values in their working methods, which then get inscribed into the design of software. The mechanics of computational thinking are thus revealed in the constant negotiation between conception, design, programming, and use at the interface between software (design) and hardware (practice).

Among the many principles of computer science, *proceduralism* has become particularly influential among media theorists seeking to critically engage the technical *processes* inherent to the inner workings of software. Judson Rosebush characterizes the uniquely procedural nature of the computer as "a natural, historical evolution from conceptual process art, with the advancement that it actually scripts and enacts concepts, producing tangible personal property as

[129] For example, see Lev Manovich, *Software Takes Command* (New York and London: Bloomsbury, 2013); David Berry, *The Philosophy of Software: Code and Mediation in the Digital Age* (New York: Palgrave Macmillan, 2011); Wendy Hui Kyong Chun, *Programmed Visions: Software and Memory* (Cambridge and London: The MIT Press, 2011); Casey Reas and Chandler McWilliams, *Form+Code in Design, Art, and Architecture* (New York: Princeton Architectural Press, 2010); Noah Wardrip-Fruin, *Expressive Processing: Digital Fictions, Computer Games, and Software Studies* (Cambridge and London: The MIT Press, 2009); Matthew Fuller, ed., *Software Studies: A Lexicon* (Cambridge, MA: The MIT Press, 2008).

the result."[130] In combining conceptual design and practical structures of use, proceduralism fosters a brand of computational thinking that views simultaneously the user as programmer. In doing so, the computer itself becomes both a *tool* that simulates and models concrete objects (instrumental design model from chapter one), and a *medium* that reconfigures existing creative techniques and processes (computational design model). Expanding on Rosebush's claims, computer scientist Michael Mateas describes proceduralism as "the ability to read and write processes, to engage procedural representation and aesthetics," and to understand code as a written artifact with its own embedded aesthetics, language, and poetics.[131] Most importantly, proceduralism is meant to cultivate forms of digital literacy in artistic practices that are best suited to represent concepts of structure and process. In this way, the term provides a necessary starting point in understanding computational thinking in sound as a bridge between the abstract design of interfaces, and the concrete practices of musicians.

While Mateas and Rosebush are primarily concerned with shifts in the reception and analysis of computer code, other scholars have expanded the concept of proceduralism to include the expressive potential of human-computer interaction more broadly. Game designer and theorist Ian Bogost defines what he calls "procedural rhetoric" as a "process-intensive" engagement with media and computational systems in general, in which "expression is found primarily in the player's experience as it results from interaction with the game's mechanics and dynamics…"[132] To think procedurally is to think *in* and *through* algorithms and mechanics, as a

---

[130] Judson Rosebush, "The Proceduralist Manifesto," *Leonardo* 22, no. 5 (1989): 55.

[131] Michael Mateas, "Procedural Literacy: Educating the New Media Practitioner," in *Beyond Fun: Serious Games and Media,* edited by Drew Davidson (Pittsburgh, PA: ETC Press, 2008), accessed December 9, 2014, http://press.etc.cmu.edu/node/205.

[132] Ian Bogost, "Persuasive Games: The Proceduralist Style," *Gamasutra,* January 21, 2009, accessed December 11, 2014, http://www.gamasutra.com/view/feature/132302/persuasive_games_the_.php.

tool for critically engaging computational procedures simultaneously as objects of analysis and interactive media experiences. Together, Mateas' procedural literacy and Bogost's procedural rhetoric provide two examples of the ways in which the process-oriented nature of computing has facilitated broader forms of computational thinking across media cultures. In turn, computational thinking has shaped directly existing ideas about creativity, including temporality in formal structure, the nature of representation, and issues of authorship, concepts to which I will now turn.

First and foremost, concentrating on the underlying logic of software requires a perceptual shift away from the narrative capabilities of media, towards the non-linear relationships between discrete elements within larger systems. Bogost describes these aspects of computational systems as "unit operations… modes of meaning-making that privilege discrete, disconnected actions over deterministic, progressive systems."[133] Significantly, considering software at the unit operations level shows the ways in which computational systems are designed to be more than just metaphorical representations of existing phenomena. Examples include a computer programmer deciphering software by picking apart the individual lines of code, or a video gamer navigating an abstract world by learning its unique rules and physics (see chapter five). Systems design engineer and media theorist Wendy Hui Kyong Chun claims that the uniquely interactive elements of computational systems are precisely rooted in this ability of digital software to break down mathematical operations into a series of simple arithmetical steps, as she claims, "The programmability and accuracy of digital computers stems from the

[133] Ian Bogost, *Unit Operations: An Approach to Videogame Criticism* (Cambridge and London: The MIT Press, 2006), 3.

discretization (or disciplining) of hardware."[134] As I suggest later in my description of *Max*, the "discretization" of digital audio production into modular "objects" has allowed for new forms of computational thinking to affect electronic music composition.

In addition to altering musicians' and listeners' sense of temporality in the formal structures of creative work, the concept of proceduralism evades the representational power of digital tools, instead focusing on the power of abstraction afforded by the computer. Rather than using text description or image depiction to explain relationships between objects and events, proceduralist media forms make claims about how things work by modeling them in the process-oriented environment of the computer.[135] Whereas print writing and other textual media excel at *describing* phenomena, and visual media *depict* that same phenomena, software *synthesizes* cultural information of various sorts, including the compositional processes of music. In this way, proceduralism further acknowledges the shift posited in chapter one, in the conception of the computer from a *tool* strictly designed for the simulation of texts and images, to an *environment* for designing abstract processes.

Finally, as the experiential focus of digital media shifts from objects to environments, computational thinking reconfigures existing understandings of digital authorship. With "analog" media, it is the conventional wisdom to think of the author as the person who writes poetry, composes music, and so on. Instead, software encourages its users to author *new processes* as a fundamental element of media creation.[136] This is a particularly significant point for music production, as it allows the producer to move beyond the idea of digital audio software being

---

[134] Wendy Hui Kyong Chun, *Programmed Visions: Software and Memory* (Cambridge and London: The MIT Press, 2011), 225.

[135] Bogost, "Persuasive Games."

[136] Noah Wardrip-Fruin, *Expressive Processing: Digital Fictions, Computer Games, and Software Studies* (Cambridge and London: The MIT Press, 2009), 7.

strictly designed for the recording of audio content, to a more fluid conception of software as a real-time instrument for the performance of music that is not yet complete. This shift in conceptions of software from tool to instrument is exhibited in the rise of live coding, as well as a number of other digital music cultures that foreground acts of technological mediation as the primary objectives of performance, as I suggest in the following chapter.

Procedural Listening

A core element of computational thinking, the proceduralist approach offers unique insights into the ways in which users directly interact with the mechanics and logical structures of software. At the same time, the feedback loop between a user and a given technology also involves moments of role reversal, in which the user must take a step back and listen to the machine. What is the nature of those experiences in which a computational process has been set in motion, and the "programmer" becomes a seemingly uninvolved observer? What happens to the subjectivity and agency of the computer musician after he or she presses play? The composition of digital music simultaneously involves active moments of sound recording and editing, as well as observational moments of listening and analysis.[137] I describe this element of the digital music production process as "procedural listening," in which the creative and perceptual focus of the producer shifts from the audio *content* being created, to the *formal* mechanics of the computational system. Simultaneously a fundamental element of digital literacy for musicians

[137] Music theorist Mark Butler defines this position of being both performer and observer as "listener orientation," defined within the specific context of electronic dance music: "This term captures a widespread set of attitudes within electronic dance music. A DJ or laptop set characterized by listener orientation is simultaneously performance-based and interpretive; it encompasses both the production and consumption of sound." Mark Butler, *Playing With Something That Runs: Technology, Improvisation, and Composition in DJ and Laptop Performance* (New York: Oxford University Press, 2014), 106.

and a core consequence of computational thinking in sound, thinking about digital music production as procedural listening allow us to grasp that both digital and musical literacies are not simply about learning how to create music with computers, but also learning how to interact physically and cognitively with new instruments, technologies, and cultural practices. Both digital music composition and HCI more broadly have been typically conceptualized as top-down relationships of a human "user" providing one-way inputs or "commands" to a computer. Procedural listening complicates this dynamic by focusing on the ways in which the creative input of "users" is guided conversely by the "rules" of the computational system, thus fostering more relational forms of HCI.

Procedural listening is not limited to software-based music, but is also present in many twentieth-century art forms, including the process-oriented aesthetics of the post-1960s experimental music traditions.[138] In response to the creative affordances for music performance offered by technologies such as electro-magnetic tape, minimalist composer Steve Reich developed the idea of "music as a gradual process," referring not to "the process of composition itself, but rather pieces of music that are, literally, processes."[139] Composer Brian Eno expanded on Reich's ideas in the context of the recording studio, in a creative process he calls "generative music": "the idea that it's possible to think of a system or a set of rules which once set in motion will create music for you."[140] If Reich was concerned with the ways in which technology influenced the performance practices of human musicians, Eno focused on the influence of

[138] See chapter five for a discussion of procedural, or "generative," visual art from the mid-twentieth-century.

[139] Steve Reich, "Music as a Gradual Process," in *Writings on Music, 1965-2000* (New York: Oxford University Press, 2004), 34.

[140] Brian Eno, "Generative Music," *In Motion Magazine,* July 7, 1996, accessed December 11, 2014, http://www.inmotionmagazine.com/eno1.html.

machine processes on other machines (fig. 2.8). These examples, among many others, serve not only to detail a rich lineage of music as algorithmic process, but also to highlight the ways in which cultural responses to technological changes are historically specific. In this way, it is possible to characterize forms of musical instrumentality as co-constructors of social and cultural epochs.

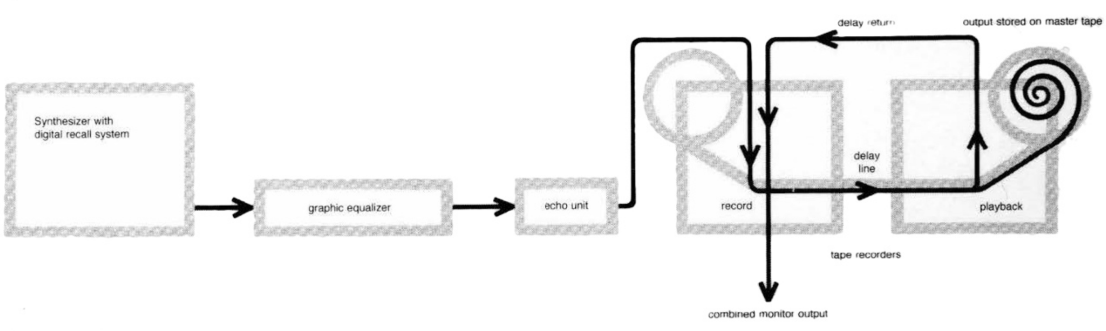


Figure 2.8: Operational diagram for Brian Eno's *Discreet Music* (1975). To the left, a delay loop is created as tape recordings playback at different speeds.
(Source: Michael Peters, "The Birth of Loop," *Para Los Pjaros,* April 25, 2015, accessed May 1, 2016, http://fripp100.tumblr.com/post/117337241586/the-birth-of-loop-by-michael-peters).

As the examples of process-oriented music attest, it is the element of perceived randomness in the perception of procedurally generated audio events that allows musicians to apply procedural listening in their creative approach. Electro-acoustic composer Kim Cascone talks about how satisfying it is to be able to "get dealt all these random events and try to make sense of it on the fly… it kind of develops a *certain way of thinking* about the material."[141] Computer artists of the 1960s used random number generators both to break the predictability that came from the human influence on computer programming, and to develop programs that could replicate the work of existing artists by simulating their "rules" and patterns. Dealing with

[141] Ben Nevile, "An Interview with Kim Cascone," *Cycling 74,* September 13, 2005, accessed December 11, 2014, https://cycling74.com/2005/09/13/an-interview-with-kim- /. Emphasis added.

randomness and complexity also remains a fundamental concern for computer scientists, whether in attempting to predict the uncertain results of algorithmic procedures or network shifts, or in theorizing the complexity of computational systems themselves.

The "certain way of thinking" about generative music described by Cascone comprises a fundamental shift in creative roles encouraged by procedural listening, as the musician learns to think like a designer rather than a composer. Most noticeably, it involves a shift away from a certain "composerly" perspective, as musicians learn to forgo compositional values such as intentionality and authorial control over the material.[142] Singer, songwriter and audio engineer Ducky views herself as a *designer* of the musical process, rather than a composer as such: "Something that I want to play with more is the randomization of things, something that I find really beautiful in sets. Maybe I can control a few things and set up whatever patches and plugins and stuff so that it's something that grows in a way where I'm not in control of everything."[143] Once again, we witness the ways in which procedural listening is focused on systematically thinking through the algorithmic, process-oriented aspects of digital performance. In turn, music composition becomes a balancing act between production and consumption, performance and observation, compositional control and computational agency.

Redirecting listening practices from "musical" content such as pitch and harmony to the organizational principles of computer code requires computer musicians constantly to develop and redevelop digital literacies. Considering core discourses in the field of interface design can

---

[142] The writings and legacy of John Cage remain the strongest advocates of an experimental musical practice in opposition to compositional intention and control. Instead, he notes that "those involved with the composition of experimental music find ways and means to remove themselves from the activities of the sounds they make." John Cage, *Silence: Lectures and Writings* (Hanover, NH: Wesleyan University Press, 1961), 10.

[143] Sam Tarakajian, "An Interview with Ducky," *Cycling 74,* November 18, 2013, accessed December 11, 2014, https://cycling74.com/2013/11/18/an-interview-with-ducky/.

help to understand how the core literacies required of procedural listening—the ability to understand the abstract nature of algorithms; competency in decoding unit operations and other discrete elements of computational systems; process-oriented thinking—are fostered on a cognitive and practical level.

Invisible Computers and the Philosophy of "Natural" Design

Music software, like all consumer products, is the result of a long and complex *design* process. As such, every software program contains within it social, cultural, technological, economic, and aesthetic values that fade to the background of the musician's attention the more he or she uses the software. In order to consider the values fixed within the software itself, it is necessary to decode the design philosophies that lead eventually to the widespread use of the program.

Dominant trends in design research primarily have been concerned with how best to achieve a "natural" interaction between humans and technologies, thus eliminating the need for users to develop new literacies for every technology with which they engage. The development of standard designs for mundane tools such as door knobs, kitchen utensils, and vehicle dashboards are just a few examples of this push towards more "intuitive" forms of human-technology interaction. Key thinker in this regard, Donald Norman, proposes not just design principles for practitioners, but an entire "psychology" of how individuals interact with everyday things.[144] For Norman, the goal is "natural design," in which the user does not experience his or her interaction with technology as an obstacle, instead focusing on the task that needs to be completed at any given moment.

[144] Donald Norman, *The Design of Everyday Things* (New York: Basic Books, 1988).

A quintessential element in natural design is maximizing the transparency between the intended use of a device and its user. Norman defines the mapping between intended actions and actual operations as interface "visibility," claiming that "just the right things have to be visible: to indicate what parts operate and how, [and] to indicate how the user is to interact with the device."[145] Consider a standard, sixteen-channel audio mixing board, for example. The device is designed to expose as many possible gestures that a user might make in manipulating and mixing sound, from turning knobs to fading sliders and plugging in cables. These interface elements are openly laid out on a flat surface, often color-coded to further highlight and distinguish actions, in an attempt to make the process of audio mixing as transparent as possible. The board is designed to make musicians feel they are not actually interacting with a mixing board at all, but rather touching and manipulating the sonic waveform itself.

By mapping the operations of a given technology in a transparent manner, visibility acts to interface designer and user. With a clear knowledge of the intended use of a given software application, the user is able to develop a mental model that simulates the inner workings of the program.[146] The more the designer exposes the fundamental properties that determine how a program could possibly be used—what Norman calls "affordances" and "constraints"—the more fully developed the conceptual model will be.[147] The thought process behind the conceptual model is this: with a knowledge of how a device *should* work, users will ideally forget that they are confronted by a technology altogether, as the interaction becomes naturalized into their everyday practice.

---

[145] Norman, *The Design of Everyday Things*, 8.

[146] Ibid, 12.

[147] Ibid, 9. Also see James J. Gibson, "Theory of Affordances," in *The Ecological Approach to Visual Perception* (Hillsdale, NJ: Lawrence Erlbaum Associates, 1986).

In addition to the development of conceptual models, the effect of natural design is often accomplished through the use of interface metaphors. In software operating systems, for example, tabs, folders, desktops, and other design metaphors are employed as navigational guides that are presumed to be understood intuitively by the broadest range of users. In the 2014 "iOS Human Interface Guidelines," Apple suggests the use of familiar interface metaphors as a strategy for increasing the feeling of transparency between users and mobile apps:

> When virtual objects and actions in an app are metaphors for familiar experiences—whether these experiences are rooted in the real world or the digital world—users quickly grasp how to use the app. It's best when an app uses a metaphor to suggest a usage or experience without letting the metaphor enforce the limitations of the object or action on which it's based.[148]

While Apple is quick to recognize the creative constraints and limitations inherent to the use of design metaphors, the marketing and use of common "everyday" technologies such as Apple products heavily relies on the user perceiving an unmediated, or "invisible," relationship to the device.

Here, we arrive at a fundamental limitation in the natural design philosophy. In emphasizing user *experience* over the technical *materials* that comprise the interface, natural design implicitly values the *mundane* forms of technological interaction that define the everyday actions so clearly delineated by Norman and others. In this way, natural design philosophy risks losing sight of a core aspect of human-computer interaction: the *creative* ways in which artists (as well as many "everyday" users) skillfully navigate and experiment with the material structures of technological devices. While many HCI experts and designers continue to believe that the ideal human-computer interface should be invisible and get out of the way, many early

---

[148] Apple, "Designing for iOS," in *iOS Human Interface Guidelines* (Cupertino, CA: Apple, 2014), accessed April 17, 2016, https://developer.apple.com/library/iOS/documentation/userexperience/conceptual/mobilehig/.

computing pioneers originally conceived of the GUI as a *medium* designed to facilitate learning, discovery, and creativity.[149] While Norman provides useful frameworks for designing objects with common usage conventions, natural design principles do not easily map onto music production software designed to generate entirely new creative experiences. In order to understand how music software can facilitate new digital literacies through computational thinking, it is necessary to look beyond philosophies of *interface* design, towards principles of musical *instrument* design.

In many ways, the everyday user interfaces and commonly used musical instruments share the natural design philosophy principles. Just as the design of a doorknob rarely departs from a handheld object that rotates in a circular fashion, so too does the design of an electric guitar rarely depart from a long wooden fret board attached to a body on which the musician is able to rest their hand. Over decades of use, the cultural techniques and physical affordances of these tools have become transparent to their users, resulting in a direct mapping between the intended design and actual use of each technology. Indeed, overcoming the very design of the instrument is a deep-seated value within many musical traditions.

In contrast, electronic musical instrument designers have recently examined the ways in which the rise of alternative interfaces complicates traditional notions of cognitive mappings between intended and actual musical operations.[150] Instrument designer Joseph Butch Rovan

---

[149] Lev Manovich, *Software Takes Command* (New York: Bloomsbury, 2013), 100.

[150] See the work of researchers affiliated with the International Conference on New Interfaces for Musical Expression (NIME). For example, Thor Magnusson, "Affordances and Constraints in Screen-Based Musical Instruments," in *Proceedings of the 4th Nordic Conference on Human-Computer Interaction: Changing Roles* (New York: ACM Press, 2006): 441-444; Atau Tanaka, et al, "Gestural Musical Affordances," in *Proceedings of the 9th Sound and Music Computing Conference* (Copenhagen, 2012): 318–325; A. Tanaka, "Mapping Out Instruments, Affordances, and Mobiles," in *Proceedings of the 2010 Conference on New Interfaces for Musical Expression* (Sydney, 2010): 15–18. In addition to NIME,

details the significance of *resistance,* rather than transparency, as a fundamental aspect of HCI in the context of music. Quoting Aden Evens, Rovan claims "the resistance of the interface 'holds within it its own creative potential… the interface must push back, make itself felt, get in the way, provoke or problematize the experience of the user'."[151] By positing the interface as an obstacle—a problematization of user experience, rather than a crutch—Rovan provides a direct counter argument to the "natural design" principles previously laid out. Similarly, programmers Chris Nash and Alan Blackwell describe the ways in which reducing the "closeness of mapping" between interface metaphor and conceptual model inculcates a shift from design principles based on *usability,* to those that encourage *virtuosity* in music systems.[152] Exposing the ways in which interfaces "get in the way," or perhaps even "create the way," rather than gradually fade away and become "invisible" to the user, instrument designers provide a productive framework for design as a tool for introducing new digital literacies to musicians and composers.

Musicological research takes this framework one step further, asking how some of our fundamental epistemological definitions of "music" are bound up with the technical design of musical interfaces. In her article, "Towards a Musicology of Interfaces," Emily Dolan defines "instrumentality" as the modes of mediation at work in the technologies that enable musical production. Historicizing the development of the keyboard interface in line with rational thought and enlightenment aesthetics, Dolan investigates the ways in which "the keyboard has

---

various articles from the archives of *Leonardo Music Journal, Journal of New Music Research,* and *Computer Music Journal* contain topics related to interface design.

[151] Joseph Butch Rovan, "Living on the Edge: Alternate Controllers and the Obstinate Interface," in *Mapping Landscapes for Performance as Research: Scholarly Acts and Creative Cartographies,* edited by Shannon Rose Riley and Lynette Hunter (New York: Palgrave Macmillan, 2009), 253.

[152] Chris Nash and Alan F. Blackwell, "Flow of Creative Interaction with Digital Music Notations," in *The Oxford Handbook of Interactive Audio,* edited by Karen Collins, Bill Kapralos, and Holly Tessler (New York: Oxford University Press, 2013).

represented a particular mode of instrumentality, namely one based on the idea of complete control."[153] If modernity presents a unique form of instrumentality shaped by a specific historical context, procedural listening presents the diminishing value of technological control as a defining feature of musical instrumentality in the age of computational thinking.

Understanding the concept of design "affordances" is crucial in explaining how interfaces might resist the control of users. For the most part, scholars have used the affordance concept as a springboard to detail the constraints of technological systems, focusing on the ways in which artists often use technology in ways unintended by the designer. But how can we talk about the affordances and constraints of musical interfaces in the context of software based on a "terminal" design; that is, software that leaves the primary design responsibilities to a user simply presented with a blank screen? Further, what types of digital literacy arise when a user is encouraged to build procedural, dynamic systems with seemingly no input or feedback from the computer?

Current understandings of both interface and instrument design are limited, in that they focus on the control of the user over the ways in which the technological system could possibly shape the creative mindset of the user. On one hand, interface designers too often focus on how design should embrace affordances that have been cultivated by users in the course of a product's history. On the other hand, instrument designers can be rightly accused of the opposite: emphasizing the ways in which users productively *subvert the constraints* of a technological system. In either case, the creative focus remains on the desires and intentions of a perceived "user" working to shape his or her technological experiences. How might forms of digital music literacy change if we reverse this model, examining the ways in which "users" act not simply as

[153] Emily Dolan, "Towards a Musicology of Interfaces" *Keyboard Perspectives* 5 (2012), 11.

agents of control over technological systems, but also *in response to* the inscribed codes, algorithms, and computational procedures of those systems?

As I have detailed here, procedural listening is both a practical tool for musicians to develop digital literacies with new software, and a form of computational thinking in sound that allows musicians to reconfigure existing conceptions of composition and performance. By combining concept (composition, design, thinking) and practice (listening, performance, literacy), procedural listening represents a new paradigm of musical instrumentality. While the term instrumentality is typically defined by the qualities of functionality and goal-oriented purpose inherent to tool-based understandings of technology, the new paradigm of instrumentality introduced by procedural listening is defined by more relational, iterative, and feedback-driven conceptions of technological use. In the specific context of musical composition, procedural listening highlights a developing mode of practice in which artists are concerned less with the direct transmission of existing musical ideas and more with the design of the human-computer performance *environment* itself.

Through analyses of *Max* and *Pure Data* in both design and practice, the next section of this chapter outlines the concept of procedural listening as a model for examining software simultaneously as a mode of musical analysis, a site of creative production, and a site for the conception of a general computational aesthetic across media. I acknowledge the connections between the programmability of computer code (as a process-oriented system with discrete steps), and the diagnostic aspects of listening (as a relational process in which the musician responds and reacts to computational processes in motion). Further, I reframe composers in the position of designers, and vice versa, encouraging the musician both to understand the logic and working mechanics of the software program, and to experiment with those mechanics in creative practice. In doing so, I offer theoretical and practical models for understanding instrumentality in the digital age from the combined perspectives of computer science, musicology, and interface design.

Music Software and the Terminal Interface

The shared modularity between computer programming and alternative interfaces for digital music composition makes the practice of procedural listening especially viable for composers working with software such as *Max* and *Pure Data*. Indeed, the software in the *Max* paradigm is appealing to the digital "makers" outlined at the opening of this chapter precisely because of the ways in which it seems to resist the limitations of the most common "proprietary" music programs. Before examining the ways in which *Max* and *Pure Data* encourage the development of alternative digital literacies for digital music producers, it is useful to contextualize that discussion by reviewing some dominant trends in the design of music software, using a product that many audio professionals agree to be the industry standard: first released in 1989, Avid's *Pro Tools* is a digital audio workstation that remains one of the most common recording programs in professional music studios.

In the realm of commercial digital audio software, interface design has been dominated by two primary affordances: the ability to record sonic material, and the opportunity to purchase "plug-ins"—third party add-ons that expand the capabilities of the software. To take stock of the constant influx of computer software, digital games, and mobile apps for music production would be a daunting task, yielding a diverse set of GUIs for manipulating sound. However, *Pro Tools*'s market dominance has solidified certain conventions over others. First, the use of linear, horizontal grids in the "Edit" window encourages the recording and inscription of musical material, regardless of source notation (fig. 2.9). This design is ubiquitous among music software, from common DAWs such as *Garage Band*, to mobile apps for music production. Second, to increase the functionality of the software, Pro Tools incorporates the use of expansion “plug-ins,” often designed using visual models and interface metaphors from “analog” tools for music production. As the *Pro Tools* interface exemplifies, the capacity of software to simulate existing instrument design standards has strongly influenced the history of musical instrument and recording studio design. Further, the software’s linear and narrative-based affordances extend twentieth-century musical aesthetics that value composer intention and mastery over sonic material.

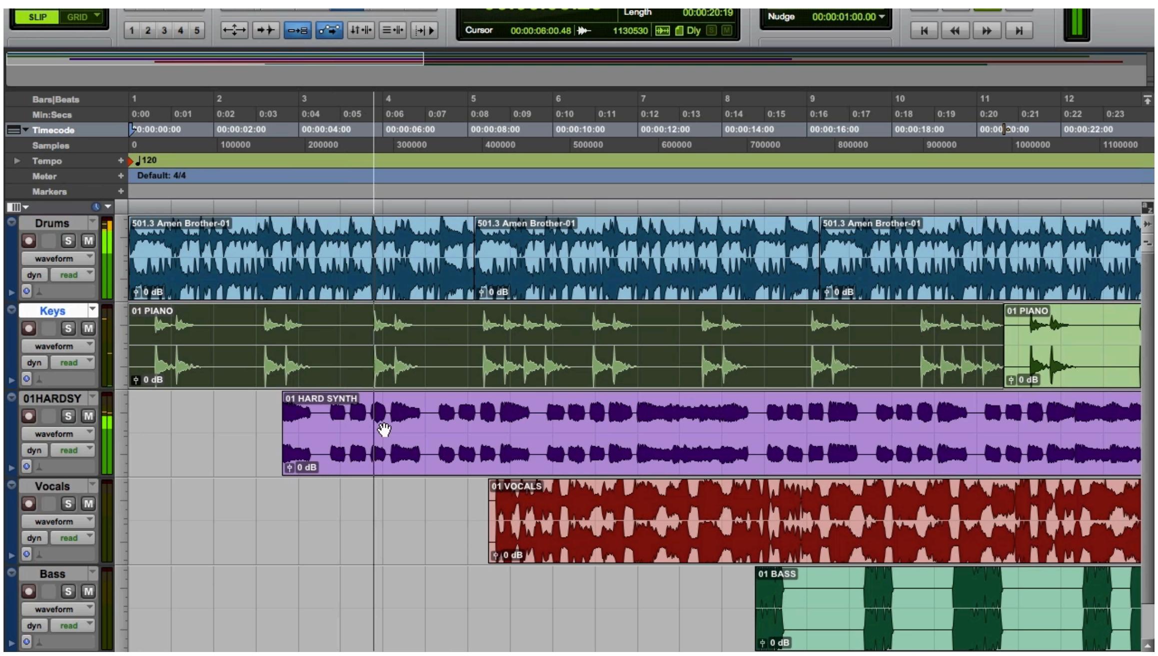
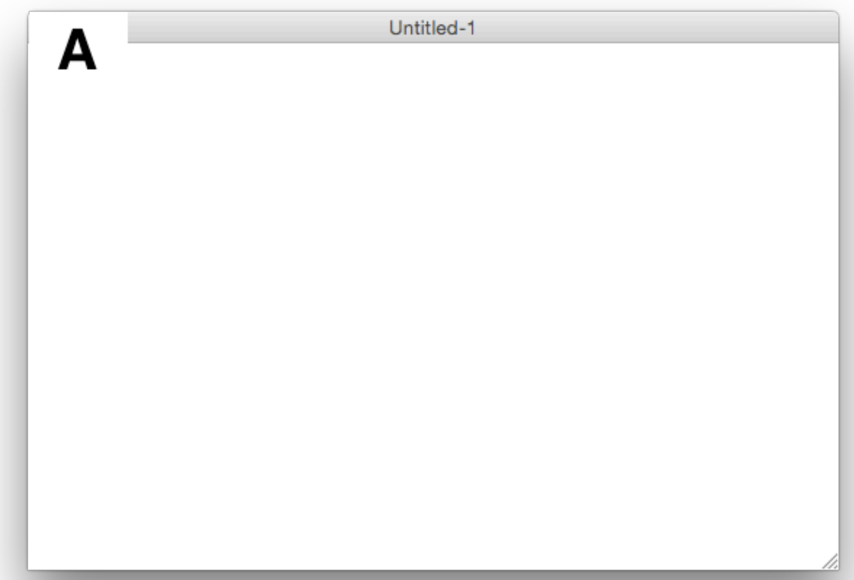


Figure 2.9: *Pro Tools 12* (2015) recording timeline. Musical tracks are organized in vertical rows, and time moves horizontally from left to right.

While software such as *Pro Tools* may be better suited for musicians looking to record through-composed tracks, often via the simulation of existing instruments and tools, software in the *Max* paradigm specializes in the creation of instruments and compositional *processes* themselves. As noted earlier, upon launching the program the user is presented with a blank screen, similar to the command line interface of the Mac operating system (fig. 2.10).



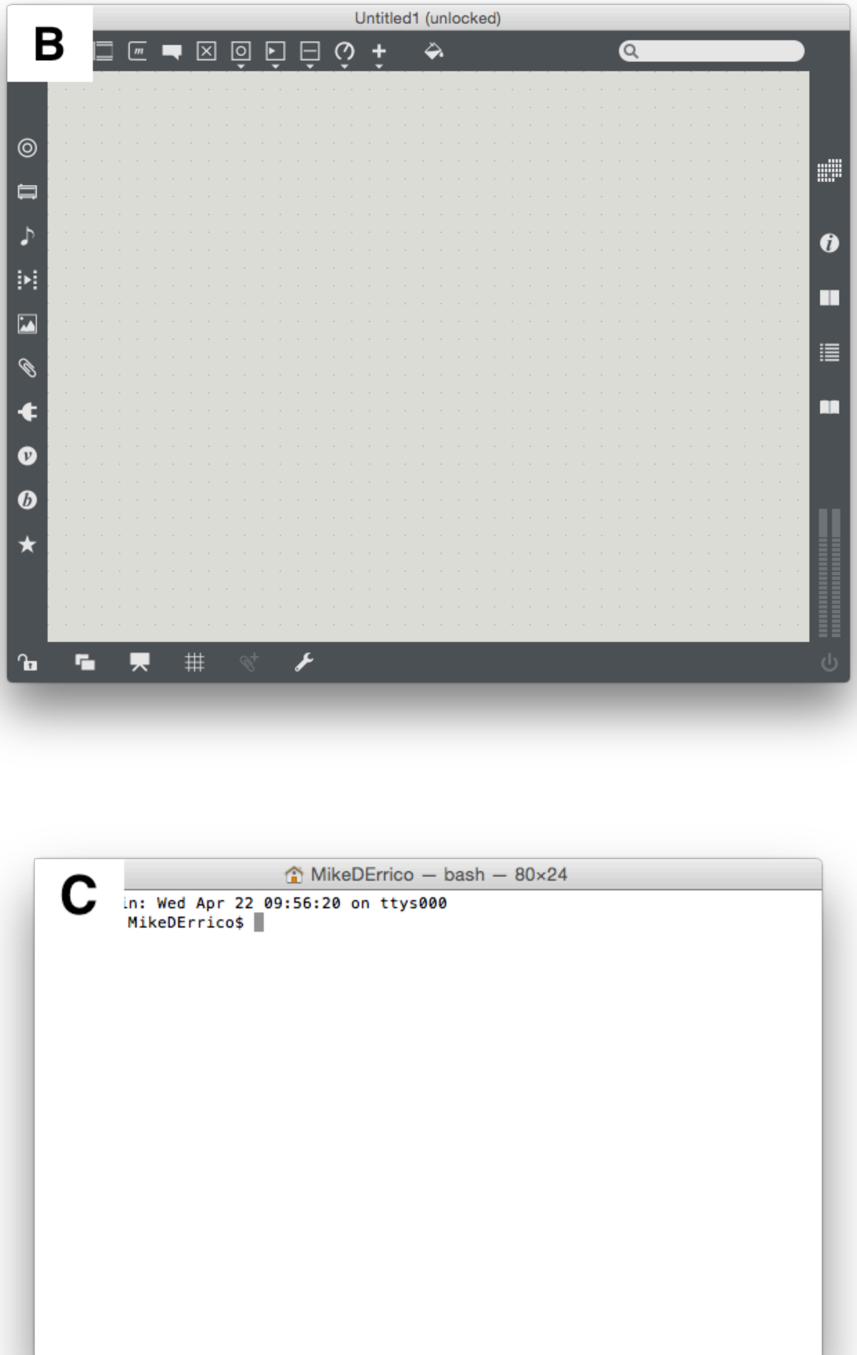


Figure 2.10: "Terminal" interfaces. (A) *Pure Data* patch; (B) *Max 7* patch, with basic editing icons surrounding the edges; (C) Mac OS X "Terminal" command line interface

As a self-proclaimed “visual programming language for media,” this blank screen design aligns the *Max* paradigm with the “terminal” interfaces of many computer programming languages. The terminal design philosophy privileges bottom-up creative environments in which the programmer-artist can create comprehensive systems or tools by working through the interactions of the smallest possible units. Composer and Radiohead experimentalist Jonny Greenwood talks about how the “open” nature of the blank *Max* screen allows him to break free from the seemingly strict conventions of “plug-in”-based music software like *Pro Tools*. Despite the supposed limitless directions being promised by the software, Greenwood describes how he was “always being led down a certain route” with *Pro Tools*. When he started using *Max*, he claims “it was like coming off the rails. Before there was all this padding between the computer and me. Now there was a blank screen as a starting point.[154] The DIY feeling of immediacy towards the *Max* interface is pervasive among users, and reflects an increasing trend towards alternative digital literacies in the development of procedural listening.[155]

In the case of *Max* and *Pure Data*, the terminal design is fostered through the inclusion of operational “commands” in the GUI known as “objects.” These modular units can be added to the blank interface through an “object browser.” Some of the more common objects include a “button,” which simply sends a “bang” (a trigger to do something) to another object; “metro,” which sends bangs at regular intervals, most often used to create a metronomic counter for a given program; and “toggle,” a switch to send on/off messages to other objects. There are approximately six hundred objects that come bundled with the commercial release of the

[154] Pask, “Mini Interview: Jonny Greenwood.”

[155] Nancy Mauro-Flude, “The Intimacy of the Command Line,” *Scan: Journal of Media Arts Culture* 10, no. 2 (2013), accessed April 18, 2016, http://scan.net.au/scn/journal/vol10number2/Nancy-Mauro-Flude.html.

software, each accomplishing a wide variety of tasks, from manipulating the GUI of the user's project, to performing arithmetic and displaying visual media (fig. 2.11).

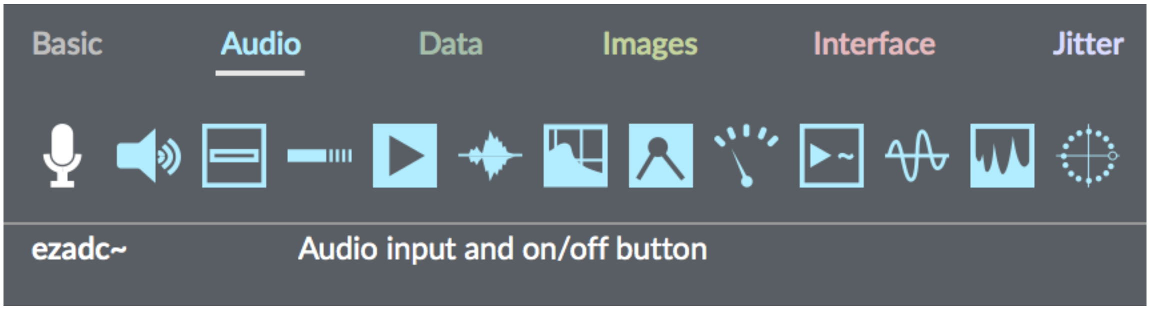


Figure 2.11: *Max 7* object browser

While the objects alone are useless, connecting various objects together through virtual cables can create complex musical applications. As a basic example, I show the *Max* "patch" for a random note generator (fig. 2.12):

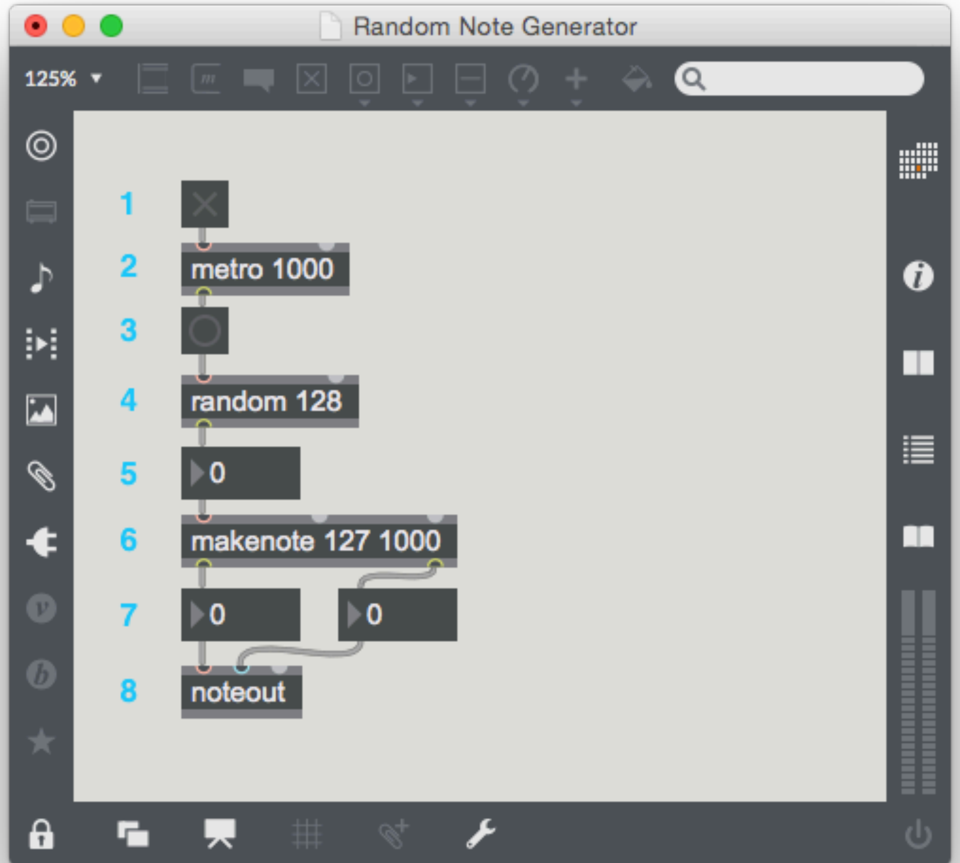


Figure 2.12: Random note generator in *Max 7*

In this patch, the square "toggle" (1) at the top triggers the following progression of actions: a "metro" (2) begins counting every second (1000 milliseconds), which triggers the circular "bang" (3) at each interval. This "bang" tells the "random" message (4) to generate a MIDI note from 0 to 127, a value which gets displayed in the "integer" box (5) just below. The MIDI number is then sent to a "makenote" object (6), which adds a velocity of 127 and a duration of one second to the MIDI value, thus creating a full set of MIDI parameters. The note and velocity (7) are finally sent to a "noteout" object (8) that transmits the numbers to a designated MIDI device (often the computer's built-in synthesizer), resulting in a sounding note every second. This sequence of actions continues as long as the "toggle" is on, designated by an "X" across the box. It would be impossible to use a custom instrument such as the random note generator in a software such as *Pro Tools,* unless a third party developer created and sold a plugin version.

Having provided a basic overview of the design and use of software in the *Max* paradigm, it is possible to discuss *Max* and similar software as tools in the development of new forms of instrumentality for digital music production. In the final section of this chapter, I detail three elements of procedural listening in working with the Max paradigm, a paradigm that has encouraged new digital literacies for multimedia artists: (1) "incomplete thinking" as a systems-based approach to composition; (2) the employment of rapid prototyping into one's creative workflow; and (3) the ability to integrate the mechanics and underlying structures of other media platforms into the composition and design of musical processes. Together, these skills highlight the increasing convergence of computation and composition, as well as provide concrete analytical models for discussing process-oriented digital art.

Procedural Listening and the *Max* Paradigm

In his 2002 essay, "How I Learned to Love a Program That Does Nothing," *Max* developer David Zicarelli defines "incomplete thinking" as a fundamental strategy for artists looking to begin working with the program. Rather than conceptualizing final goals through the realization of compositional intentions, as is often the case with other music production software, Max users must "think about the middle instead of the end, in the same sense that the programmer of a word processor is more concerned with how documents are edited than the quality of the writing being composed."[156] The idea of incomplete thinking as a focus on the *how* rather than the *what* resonates with much of the previously detailed proceduralist theories and expanded notions of process-oriented instrument design.[157] Whether in the design, marketing, or use of *Max*, the idea of the computer software as a real-time performance environment in dialogue with its user is ubiquitous, and has shaped both the ways in which new users learn the program, and the increasing convergence between artists working across media.

The incomplete aspects of the *Max* paradigm are not simply a result of the production tools afforded by the software, but also a product of the relationship between the minimal, "terminal" design of the interfaces and the structure of computer programming languages. The fact that, upon opening the software, the user is introduced to a blank screen encourages the composer to think like a programmer, in terms of non-linear and modular systems. Electronic musician Kim Cascone details the emergence of a "non-linear architecture" in the design shift from the "linear, tape deck kind of paradigm" offered by the *Pro Tools* plug-in interface, to the visual programming environment of *Max*.[158] If procedural thinking is defined by the privileging of building from the smallest to the largest units—and thus the opposite of thinking in terms of intentions, final structures, and end goals—then procedural listening in digital music production may be defined by an attention to the relationship between musical "modules" of various sorts. This systems-based approach to digital practice is at the heart of learning both general computer programming and music related programming, such as *Max*.

[156] David Zicarelli, "How I Learned to Love a Program That Does Nothing," *Computer Music Journal* 26, no. 4 (Winter, 2002): 44.

[157] Wardrip-Fruin, 7.

[158] Ben Nevile, "An Interview with Kim Cascone," *Cycling 74,* September 13, 2005, accessed March 30, 2016, https://cycling74.com/2005/09/13/an-interview-with-kim-cascone/#.VvvveMfg9go.

At the most basic level, the “incomplete” aspects of the *Max* interface can be experienced through the discrete nature of its primary building blocks: objects. Whereas “traditional” music composition often involves the juxtaposition of pre-composed musical ideas such as riffs, rhythmic patterns, melodies, and harmonies, the primary “content” of *Max* objects are not compositional ideas, but algorithmic processes that are defined in relation to one another. As Zicarelli hints at through the title of his essay, these incomplete units “do nothing” on their own, and can only be activated by connecting one to another through virtual patch cords. In order to construct a complete *Max* patch, the user must work through the various interactions between objects in a flexible, step-by-step manner, constantly aware of the multiple relational possibilities between objects at any given moment. Early computer graphics artist Manfred Mohr characterized this relationality between objects in automated systems as the major paradox of generative art, as he claims, “formwise it is minimalist and contentwise it is maximalist.”[159]

The step-by-step, algorithmic, and “incomplete” manner in which the patch is constructed fosters a diagnostic relationship between the composer and the software. As the patch “runs,” the composer views the resulting program from a macro perspective, able to pinpoint the micro processes happening every step of the way as a result of each object. In the random note generator, for example, number boxes after the “Random” and “Makenote” objects provide the composer with real-time feedback in the form of constantly changing integers, allowing him or her to perceive characteristics of the sounding note, such as volume and duration. At the same time, the “Bang” button blinks whenever it is triggered by the “Metro” object, reassuring the composer that the patch is functioning correctly. In addition to teaching users to “think outside

[159] Manfred Mohr, quoted in Frank Dietrich, “Visual Intelligence: The First Decade of Computer Art (1965-1975),” *Leonardo* 19, no. 2 (1986): 166.

the timeline" of various DAWs, this process of debugging a patch in *Max* encourages the user to focus on the micro-processes of each individual object algorithm. The composer's understanding of the mechanics and rules of the software is regularly "updated" in response to the constantly changing "state" of the *Max* patch. Computational thinking in sound thus involves a feedback loop inherent to procedural listening: the user diagnoses the status of the computational system while programming the software itself. In other words, this "integrative composer," to use a term introduced earlier, is simultaneously creator, listener, producer, and consumer.

As a result of the diagnostic aspects of procedural listening in *Max*, composers are able to employ rapid and iterative prototyping methods in the patch building process. Since the effects of any changes made to the patch are revealed immediately to the composer, he or she can substitute objects for one another in order to explore the various possibilities afforded by the patch. In the current iteration of the random note generator, for example, the "Random" object functions to generate a random number which eventually determines the frequency of the resulting note. If the composer alters the "Random" object and connects it to a "Sel" object, the patch could randomly select notes among a specific set of options. For example, in the random triad generator (fig. 2.13), the "Random" object chooses among three different notes, which are determined by the "Sel" object. The three possibilities are the three MIDI notes from the minor triad, as displayed in the "Message" boxes below the "Sel" object.

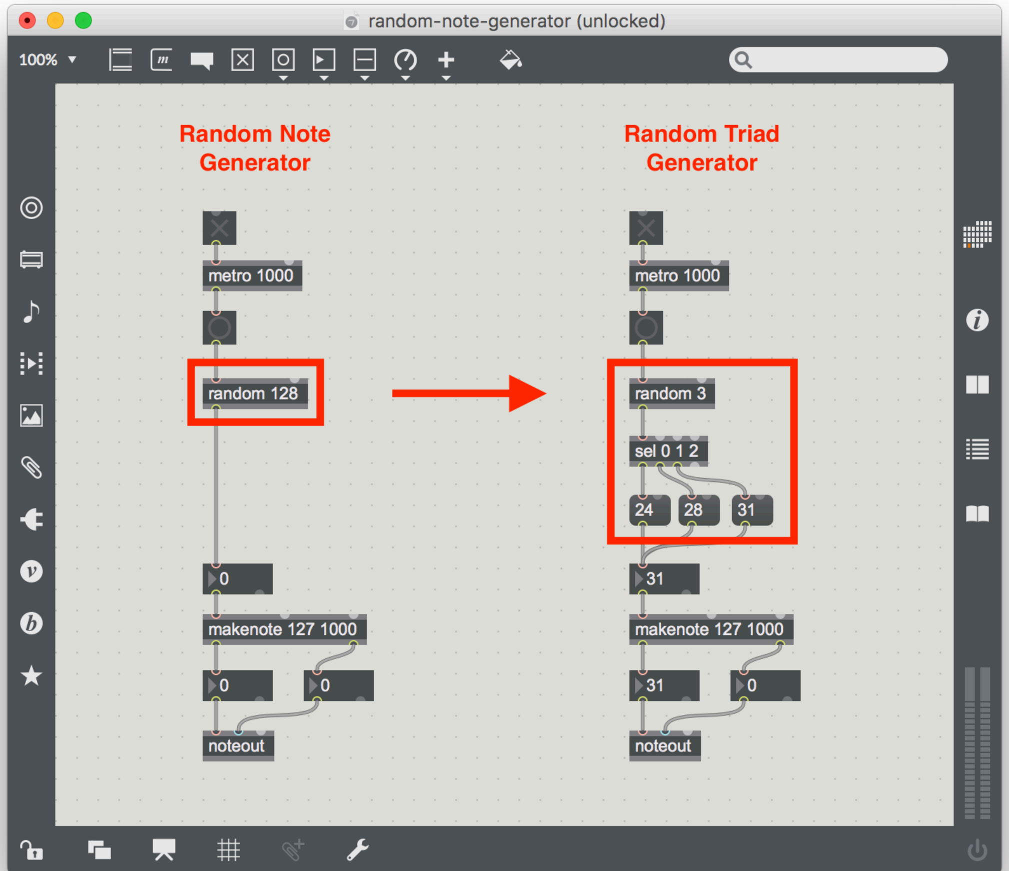


Figure 2.13: *Max 7* random triad generator patch

Among design professionals, this type of iterative prototyping—defined by the constant *process* of testing, analyzing, and refining a product—has replaced more *fixed* text documentation as a core deliverable given to project managers.[160] *Swift,* Apple's programming environment for OS X and mobile software development, incorporates iterative methods in its workflow by introducing "playgrounds"—interactive documents where *Swift* code is compiled

[160] Emily Grace Adiseshiah, "Iterative prototyping to improve the design process," *Justinmind,* January 14, 2016, accessed April 12, 2016, http://www.justinmind.com/blog/iterative-prototyping-to-improve-the-design-process/.

and run live as you type.[161] In the same way as *Max* patches allow the composer a macro perspective on the operations of the patch, *Swift* playgrounds display the program's operations in a step-by-step timeline, allowing the programmer to inspect and revise the code at any point (fig. 2.14). If code is often conceptualized as a text-based *object-oriented* representation of the *process-oriented* outcomes of software, the *Max* patch offers a tool for musical representation that is simultaneously a kind of musical score (an inscribed set of rules about sound) *and* performance (the mechanism through which those rules are sounded). The analogy to digital music composition is clear: in the context of procedural listening, people do not simply read musical notation; they also interact with it.

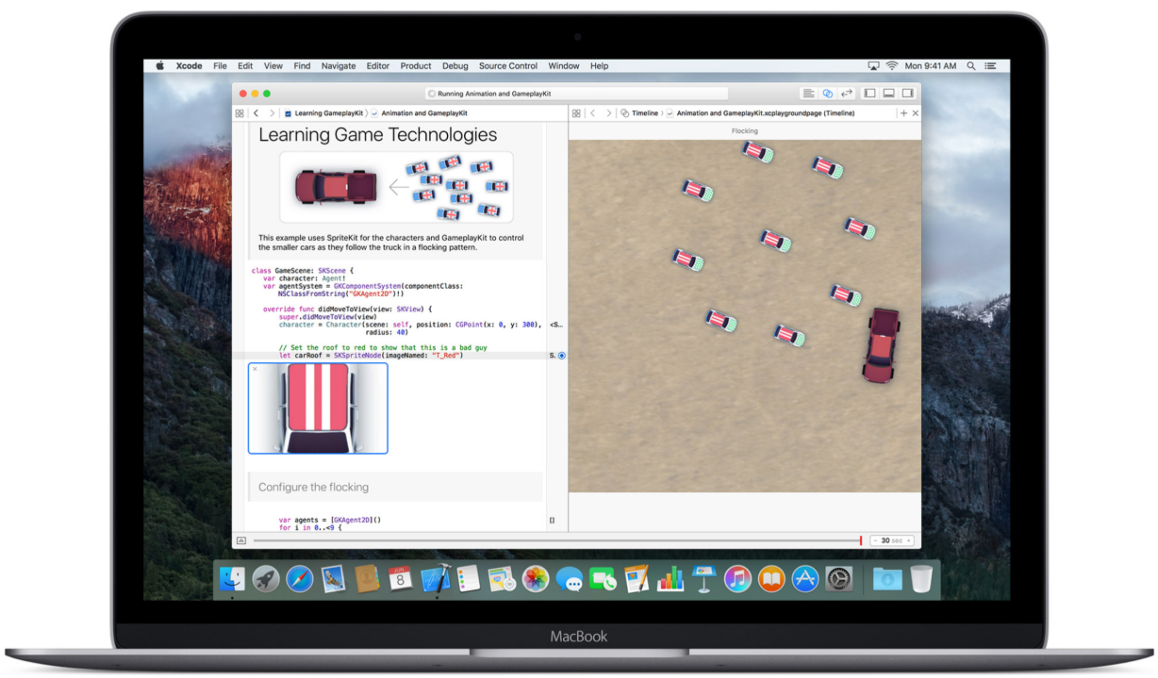
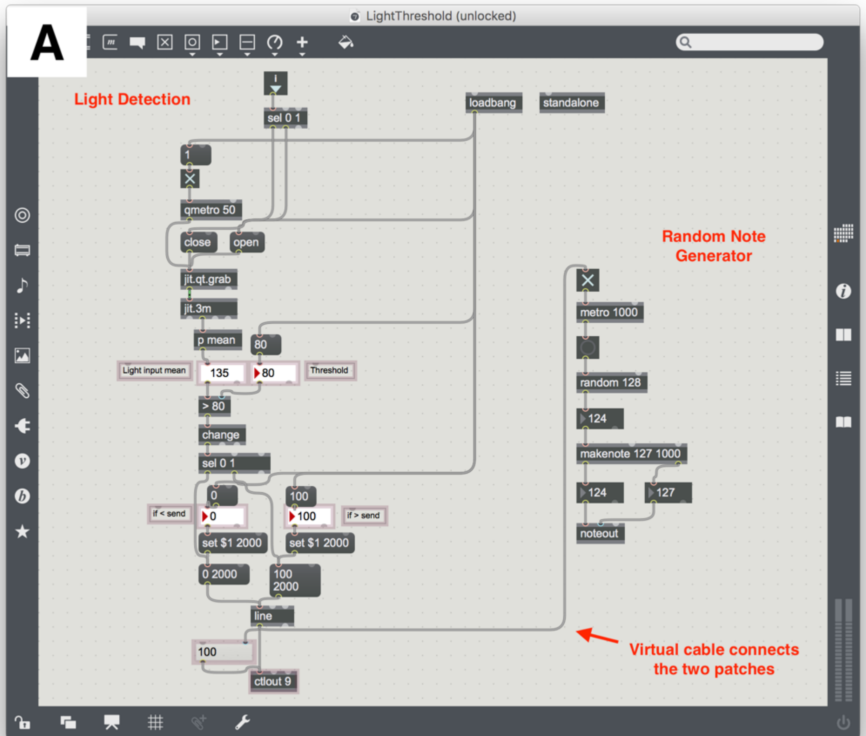


Figure 2.14: *Swift* "Playground"
(Source: Apple Developer, "Swift," accessed May 1, 2016, https://developer.apple.com/swift/).

[161] Brad Larson, "Rapid Prototyping in Swift Playgrounds," *objc.io* 16 (September 2014), accessed April 12, 2016, https://www.objc.io/issues/16-swift/rapid-prototyping-in-swift-playgrounds/.

The ability to conceptualize simultaneously micro and macro structures of a *Max* patch is facilitated by an information architecture that arranges smaller units into increasingly larger structures: objects are arranged into subpatches, which are then connected to other subpatches to form a larger patch. Picture a more complicated random note generator, triggered not by clicking manually the “Toggle” object in *Max*, but an external input such as the amount of light in a room. When the lights are on, the “Toggle” is on, thus triggering the random note generator to play, and vice versa. In order to do this, the random note generator patch must be connected to a separate light detection sensor patch.

As the figure shows (fig. 2.15a), the resulting integrated patch is much more complex visually than the random note generator by itself. In a similar manner as other common programming languages, each separate patch can be “encapsulated” into a smaller subpatch (akin to a complex dropdown menu), thus clarifying the visual layout of the overall patch (fig. 2.15b). Encapsulation provides a simplified way to hierarchize functions within a patch, allowing the composer to follow more carefully the individual processes occurring within the larger program, as well as the relationship between the micro and the macro aspect of the patch design. The act of structuring musical processes on a hierarchical level is especially important when connecting patches within the *Max* interface to software outside of it. Ultimately, the supposed end product, the “complete” patch, often serves as just another musical module within a broader performance environment that includes other musicians, instruments, or media platforms. The incomplete, modular, and iterative nature of procedural listening in *Max* encourages the application of systems-oriented design methods as a way of both handling algorithmic complexity and thinking of program functionality outside of a specific software.[162]



[162] Elisabeth Skjelten, *Complexity and Other Beasts* (Oslo, Norway: The Oslo School of Architecture and Design, 2014). See also Birger Sevaldson's website, "Systems Oriented Design," Oslo School of Architecture and Design, accessed April 12, 2016, http://www.systemsorienteddesign.net.

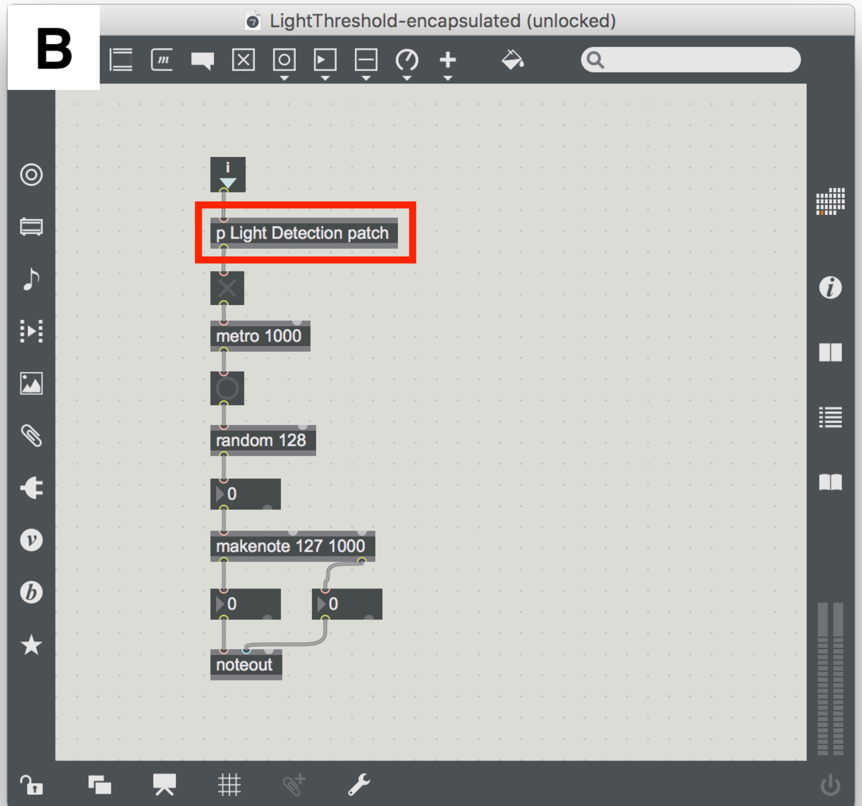


Figure 2.15: (A) *Max 7* light threshold patch; (B) light threshold patch encapsulated

Indeed, most *Max* objects are agnostic as to which media format is being controlled, allowing the integration of various media platforms and techniques into a single project. “Jitter” is a specific set of *Max* objects dedicated to video, graphics, and matrix data processing, allowing real-time cropping, stretching, and juxtaposition of moving images, as well as the handling of information through spreadsheets and databases. In turn, these text-based data handling objects are capable of interacting with audio objects dedicated to synthesis, sampling, and editing. Further, *Max* is capable of interfacing with software and hardware outside of its native environment. By working with the Software Development Kit (SDK) and Application Programming Interface (API), users can develop their own *Max* objects capable of sending control messages to web services such as YouTube, Spotify, and Last.fm, as well as hardware devices such as mobile phones, computers, and portable sensors of various sorts. When these integrated systems are setup, users can easily reroute audio to and from other music software on the same machine, using sound routing applications such as *Soundflower* and *Rewire.* In the random note generator patch, for example, double-clicking the "Noteout" object reveals a list of possible outputs for the resulting MIDI note, allowing the composer to send the output of the patch to other digital audio workstations such as Ableton *Live* or *Pro Tools* (fig. 2.16).

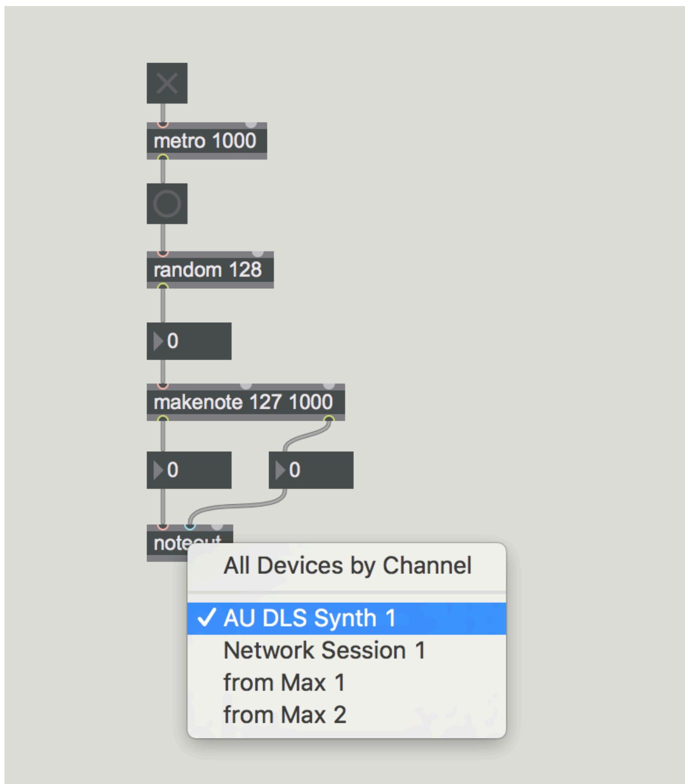


Figure 2.16: "Noteout" options in *Max 7* patch

The integration of *Max* with various external media platforms highlights yet another aspect of procedural listening: the ability to understand simultaneously the techniques of "musical" and extra-musical software. In the context of *Max*, the production environment is not simply the software alone, but countless APIs and external objects that are meant to interface

Max with other software applications and media platforms. Manovich describes this convergence of separate media techniques and tools into software as "deep remixability," a form of media composition in which "the software production environment allows designers to remix not only the content of different media types, but also their fundamental techniques, working methods, and ways of representation and expression."[163] *Max* objects serve as abstractions of broader musical gestures (triggers, pitch bends, dynamic changes) and tools [metronomes, transport controls (play, pause, stop), beat counters], functioning similarly regardless of the actual content or media platform that they control.

In the end, *Max* and *Pure Data* are programs that allow musicians to manipulate numbers in various ways. The software itself is agnostic to the output of these numbers, allowing a patch as simple as the random note generator to control external actions ranging from the manipulation of interface elements in a video game to the thermostat levels in the composers' home. Cascone talks about developing a *Max* tool that enables sound designers to generate algorithms that can be used by game programmers: "They don't have to worry about talking the same language, because all he has to say is okay, I've exposed all these controls to you. You need to send values in this range with this name to these controls."[164] In facilitating a shared technical knowledge among media producers—what Wardrip-Fruin terms the "operational logics" of technological systems—*Max* participates in the shift from the isolated architecture of the *recording* studio to the emerging *network* studio ideal posited in chapter one.[165] The increasingly integrated nature of

[163] Manovich, 46.

[164] Nevile, "An Interview with Kim Cascone."

[165] Wardrip-Fruin, 13-14.

this compositional model privileges a process-oriented relationality between media objects, transcending conventional musical parameters such as pitch, volume, and timbre.

One of the more common multimedia forms to converge with digital audio production is game design. In the practice of gaming, players learn the mechanics of the game through an embodied habituation of response developed using a physical interface. Procedural content generation in the open-world of *Proteus,* for example, is reflected in the procedurally generated musical patterns, together forming a defining aspect of the gaming experience (see chapter five).[166] In a puzzle-platformer like *Fez,* musical cues guide the player through the game's complex mazes, as well as helping him or her to discover hidden secrets only accessible by decoding rhythmic and melodic patterns.[167] Similarly, the mechanics of music software are revealed to the composer through a perceptibly transparent GUI that exposes the "guts" of the program. Puckette claims that the design of *Max* "strives for simplicity and explicitness throughout," and that abstraction in the software is meant to enhance the directness of the program's mechanics.[168] In effect, the "rules" of the *Max* paradigm are defined by the clearly delineated affordances and constraints of each individual object, combined with the ability to endlessly remix the relationships between these objects.

Bernard Suits famously defines games as "the voluntary attempt to overcome unnecessary obstacles."[169] The feedback loop inherent to procedural listening allows us to

---

[166] "Proteus' Soundscapes: The Similarities Between Procedural Generation and Improvisation," *The Phonograph,* January 6, 2012, accessed December 12, 2014, http://www.thephonograph.co.uk/2012/01/06/proteus-soundscapes-the-similarities-between-procedural-generation-and-improvisation/.

[167] Rich Vreeland, "Philosophy of Music Design in Games – Fez," Hunter Short, uploaded April 20, 2013, accessed March 30, 2016, https://www.youtube.com/watch?v=Pl86ND_c5Og.

[168] Puckette, *Max at Seventeen,* 37.

[169] Bernard Suits, *The Grasshopper: Games, Life, and Utopia* (Ontario: Broadview Press, 2005), 55.

understand the seemingly disparate processes of creating *Max* patches and embodying game mechanics through play as a series of creative challenges to which the “player” must respond. As mentioned earlier, interface design research has focused on the ideal of “natural” interaction and “invisible” computing encouraged by attempting to make the interface transparent. By exposing digital audio production as a rule-based system similar to gameplay, the forms of mediation inherent to the software are foregrounded. In turn, the mapping between intended and actual operations is made visible, and the user is able to develop a fuller understanding of the program’s creative affordances. Digital literacy is thus conceived as a dialogic interaction between human and computer, rather than a one-way relationship of command.

Procedural Listening and Cultural Praxis

As I have discussed in this chapter, the technical design of software in the *Max* paradigm integrates the creative practice of computer programmers with digital audio producers. In doing so, it introduces new forms of musical instrumentality to composers and electronic musicians working across the visual, interactive, and sonic arts. While combining analyses of technological design and creative use remains a common strategy for media theorists and computer scientists, musicologists have yet to take into account the disintegrating dichotomy between consumption and production in twenty-first century musical thought. As President Obama states, what is important for education across the disciplines of engineering and the arts is that students are familiar “with not just how to play a video game, but how to create a video game.”[170] This praxis-oriented “maker” model of digital literacy, in which software users learn computational

[170] Ezra Mechaber, “President Obama is the First President to Write a Line of Code,” *The White House Blog,* December 10, 2014, accessed December 13, 2014, http://www.whitehouse.gov/blog/2014/12/10/president-obama-first-president-write-line-code.

systems by building them, is fundamental to the experience of patching in the *Max* paradigm. Like software itself in the twenty-first century, digital music production tools are both instruments and forms of musical notation. By analyzing *Max* patches both organologically and as “scores,” this chapter foreshadows broader questions about the nature of musical pedagogy and music theory in the digital age.

The hermeneutic and semiotic study of software should constitute a fundamental skillset for musicologists and theorists working with electronic music as much as digital music literacies comprise fundamental skillsets for contemporary musicians and composers. This praxis-oriented mode of computational thinking offers a new model for the analysis of musicological works and performances—a form of sketch studies for the digital age. Procedural listening, in this mode, consists of a constant negotiation between the conceptualization and practice of digital culture. In turn, digital cultures themselves—from the free software movement to live coders and graphic artists—may be understood not simply as communities of “tinkerers” that produce technological *objects,* but also as individuals developing new *processes* of relating to existing social understandings and technical practices surrounding technology more broadly. In other words, procedural listening allows software users to hear the dynamics of cultural interaction in the technical design of human-computer interfaces themselves.

Section II

When Software Becomes Hardware

The first section of this dissertation discussed different compositional approaches to the convergent media landscape that software has introduced. Chapter one aligned the maximalist design of digital audio workstations such as Ableton *Live* with a broader maximalist approach to media and technology usage in the digital age. In contrast, chapter two considered the minimalist attitude at the root of both the "terminal" design of computer programming languages and the formation of software cultures that value a more "do-it-yourself" approach to technological use. Together, these case studies exemplify the ways in which the internal design of software applications shapes, and is shaped by, external social and cultural factors.

Shifting the analytical focus from the design of software graphical user interfaces, the second section (chapter three and four) considers how the design of hardware devices is used to externalize and materialize the affordances of software.

Chapter three examines how digital musicians distinguish themselves as "musicians" in the context of a media landscape in which music producers and performers use the same tools as office workers, people shopping or watching Netflix, and children learning how to use computers for the first time. For example, performers in the "controllerism" movement—electronic musicians who use hardware "controllers" to manipulate software in performance—integrate hardware peripherals with laptop computers in an effort to foreground the corporeal, "live" nature of performance in the digital age, in an attempt to distinguish themselves from DJs who simply "press play" on stage. In doing so, controllerists imply that the underlying cultural valence of performance with digital computer music tools is similar to existing practices of musical performance. But, what happens when there are no ways to distinguish skill levels among the various forms of technological interaction, and thus no way to distinguish between levels of skill and artistry, or even between "musician" and "non-musician"?

Chapter four examines the Apple iPhone as a device whose "user-friendly" design and seemingly intuitive touch-screen mechanics equalize the skill levels needed for both everyday productivity tasks and creative music production. In contrast to controllerists who use hardware controllers to distinguish themselves from commonplace users of technology, iPhone users celebrate the disintegrating distinction between expert "producers" and non-expert "users" facilitated by mobile media devices. The line between creative production and media consumption is further blurred with the integration of the *App Store*, an online shop in which users can purchase both new apps and add-on content for existing apps on the iPhone. In analyzing the ways in which mobile media software and app design "democratizes" music production practices, it is possible to understand "music" as comprising not simply *material* technologies (instruments, controllers) and traditional performance spaces (dance clubs, concert halls), but a process-oriented *experience* that aligns with consumption practices inherent to capitalism in the early twenty-first century.

Chapter 3

"Diggin' in the Carts":
Technologies of Play in Hip-Hop Production and Performance

In the first decades of the twenty-first century, electronic dance music and video games have emerged as dominant forms of popular cultural expression. The rise of electronic dance music as a global industry parallels the proliferation of massive multiplayer online video games, both manifesting the power of social media in mobilizing previously isolated communities of gamers and musicians. As a result of their widespread appeal, video games and electronic dance music have become the targets of American politicians and moralists who blame the cultures for everything from drug use to gun violence and general cultural decay.[171] In a similar manner as previous youth cultural movements, games and dance music simultaneously reify and subvert existing anxieties about both the increased role of media and technology in everyday life, and the resulting shifts in the nature of social relationships more broadly.

[171] These concerns echo the 1985 court case initiated by the Parents Music Resource Center (PMRC) against commercial music with violent and sexual content. United States Senate, "Record Labeling: Hearing before the Committee on Commerce, Science, and Transportation," United States Senate, Ninety-ninth Congress, First Session on Contents of Music and the Lyrics of Records, September 19, 1985, accessed April 28, 2016, http://www.joesapt.net/superlink/shrg99-529/. In the context of electronic dance music, see Rong-Gong Lin II, "ER doctors: Drug-fueled raves too dangerous and should be banned," *Los Angeles Times,* August 10, 2015, accessed April 18, 2016, http://www.latimes.com/local/lanow/la-me-ln-why-some-er-doctors-want-to-end-raves-in-los-angeles-county-20150810-story.html; Ben Sisario and James C. McKinley Jr., "Drug Deaths Threaten Rising Business of Electronic Music Fests," *The New York Times,* September 9, 2013, accessed April 18, 2016, http://www.nytimes.com/2013/09/10/arts/music/drugs-at-music-festivals-are-threat-to-investors-as-well-as-fans.html?_r=0. In the context of video games, see Heather Kelly, "Do video games cause violence?" *CNN,* August 17, 2015, accessed April 18, 2016, http://money.cnn.com/2015/08/17/technology/video-game-violence/; Brett Molina, "Obama seeks research into violent video games," *USA Today,* January 16, 2013, accessed April 18, 2016, http://www.usatoday.com/story/tech/gaming/2013/01/16/obama-gun-violence-video-games/1839879/.

More recently, the visceral experiences of music and gameplay have converged in various ways, specifically shaping the embodied practices of music and game creators themselves. The success of music video games such as *Guitar Hero* and *Rock Band* has influenced both amateur and professional musicians to think through the practical connections between musical performance and gameplay. Dubstep pioneers Skream and Benga have discussed the ways in which their use of the Sony Playstation video game console to make beats has shaped the sound of contemporary dance music.[172] In 2014, Red Bull Music Academy even launched a documentary series titled "Diggin' in the Carts", tracing the global influence of Japanese video game music from the 1980s and 90s on contemporary genres of electronic music.[173] Through interviews with game music composers and hip-hop DJs alike, the series reveals unexplored relationships between the now ubiquitous experience of gameplay in everyday life and the technical practices of electronic musicians.

Integrating theories of play from various branches of media studies with analyses of the technical design of both music and video game controllers, this chapter discusses the embodied practices of electronic music production in relation to the haptic control inherent to gameplay. Together, the coterminous rise of video games and electronic dance music charts an alternative historical narrative in the evolution of digital media. Rather than reifying the centrality of "analog" technologies such as the turntable in the birth of popular music genres, the ongoing convergence of games and music establishes forms of experimental play with emerging media as

[172] "Benga makes a Grime beat on Playstation music 2000 - BBC 2003," *GetDarker*, uploaded December 5, 2014, accessed March 31, 2016, https://www.youtube.com/watch?v=AZHnaSIZP0Y; "Lecture: Skream (Melbourne 2006)," Red Bull Music Academy, uploaded September 24, 2011, accessed March 31, 2016, https://vimeo.com/29535666.

[173] "Diggin' In The Carts - Series Trailer - Red Bull Music Academy Presents," *Red Bull Music Academy*, uploaded August 28, 2014, accessed March 30, 2016, https://www.youtube.com/watch?v=fwN_o_fi7xE.

crucial to the development of cultural production in the twenty-first century. By engaging a transitional moment in the historical evolution of hip-hop, electronic dance music, and interactive media, I provide insights into the physical and cognitive structures of sonic embodiment in gameplay and human-computer interaction more broadly.

From Turntablism to Controllerism

While digital music software has become commonplace in the studio and on the live stage, the history of hip-hop has always been rooted in the “analog” materiality and physical manipulation afforded by the vinyl record. Indeed, the turntable has played a fundamental role in shaping the music production and performance practices of hip-hop, as well as cultural aesthetics more broadly. Early DJs such as Grandmaster Flash, Grand Wizard Theodore, and Afrika Bambaataa established vinyl record mixing, cueing, beat matching, and beat juggling as primary techniques for the aspiring turntablist, showcasing both their individual virtuosity and entirely new modes of performance made possible by technological experimentation. In his seminal mix, “Adventures of Grandmaster Flash on the Wheels of Steel,” Flash popularizes the “cut”—a mixing technique in which the DJ quickly shifts the sonic output from one turntable to another—a gesture made possible by Flash’s own invention of the crossfader. Bambaataa, on the other hand, was known for his eclectic musical tastes and extensive record collection. While disco records provided the most common musical fodder for early hip-hop DJs, Bambaataa often experimented with rock, Latin, and other diverse styles from around the world, including Kraftwerk’s “Trans-Europe Express.” His 1982 track, “Planet Rock,” was an early prototype for the now pervasive concept of the remix, highlighting the turntable and vinyl record as both a musical instrument and symbol for the increasing circulation of culture through technology in the early 1980s.

In addition to functioning as a conduit for socialization through music, vinyl has helped in bridging generational gaps and aesthetic values within families. Many DJs have talked about the influence of their parents' record collection on their early childhood musical education as both a stylistic influence on their own musical aesthetics, and a sort of indoctrination into techniques of everyday music consumption. For Grand Wizard Theodore, this relationship was crucial to the history of hip-hop, as it led to the emergence of one of the music's most canonical sounds: the scratch. In one of hip-hop's most enduring creation myths, Theodore described the moment in which he was listening to his parents' records on the family turntable late at night, and his mother walked in, demanding him to turn it down. Instead of lowering the volume, he stopped the record manually by placing his hand on the vinyl and slowly moving the record back and forth, thus creating the signature scratch that has come to define the hip-hop sound.[174] Whether or not one believes the story, this moment exemplifies the importance of historical lineage in hip-hop, a cyclical process in which one expands on the possible uses of his or her inherited media through creative innovation.

As this brief historical gloss points to, vinyl secured a fundamental role in preserving hip-hop traditions through offering musicians a sense of tangible presence and real-time manipulation. Musicologist Mark Katz claims the physical immediacy of the record as the most important reason for its success, as he describes the hand resting "comfortably on the grooved, slightly tacky surface… Pushing a record underneath a turntable needle, transforming the music held within its grooves, one has a sense of touching sound."[175] The "inimitable feel" of vinyl

---

[174] Grand Wizard Theodore tells this story in *Scratch,* DVD, directed by Doug Pray (New York, NY: Palm Pictures, 2001).

[175] Mark Katz, *Groove Music: The Art and Culture of the Hip-Hop DJ* (New York: Oxford University Press, 2012), 64.

comes through not only in the performance practice of the DJ, but also in the hands of record collectors who value the dusty, aged quality of vinyl just as a book collector values the original printing of a text. In the physical manipulating the deep wax grooves on the surface of a record, the DJ may sense he or she is "touching sound" and allowed immediate access to the musical source and social context embedded within the object.

It is no coincidence, then, that an archaeological rhetoric pervades discourse surrounding record collecting within hip-hop. The process of seeking out new records for both creative inspiration and musical source material, known as "digging the crates," has become a rite of passage for aspiring DJs. According to ethnomusicologist Joseph Schloss, "one of the highest compliments that can be given to a hip-hop producer is the phrase 'You can tell he digs.'"[176] Typical dig sites for these media archivists include yard sales, online auctions, the hidden recesses of thrift stores, and the basements of old record stores. Similar to archaeologists, record collectors are equally concerned with preserving, even sanctifying, the original site of discovery while evaluating the quality of each find.

The excavation of vinyl facilitates both the construction of hip-hop's musical genealogy and the proliferation of technical innovation within the genre, Katz describing the materiality of the vinyl as "a precious substance in hip-hop" that is "authentic," "elemental," and "fundamental." Present at and largely responsible for the birth of hip-hop, Katz claims "There is more than just music inscribed in those black discs; vinyl carries with it the whole history, the DNA, of hip-hop."[177] In the late 1990s through the early 2000s, vinyl culture would confront a major practical and philosophical dilemma with the emergence of digital tools for music

---

[176] Joseph G. Schloss, *Making Beats: The Art of Sample-Based Hip-Hop* (Middletown, CT: Wesleyan University Press, 2004), 80.

[177] Katz, 218.

production. For a culture so intimately dedicated to the physicality of both the record and the performer, what happens to the structure of hip-hop's musical DNA in the context of the perceived immateriality of software? How are techniques of production and performance coping with the gradual obsolescence of vinyl?

In 2010, Technics discontinued production of the SL-1200 turntable. The iconic model was lauded for its minimalist interface and direct drive system, which afforded the DJ a particularly robust instrument with a heightened sense of tactile feedback (fig. 3.1a). The countless obituaries surrounding the device's death marked this moment as the end of an era, questioning what would become of hip-hop in the post-SL-1200 age.[178] In the same year, Apple introduced the iPad, a touch-screen portable tablet that became particularly popular among digital musicians seeking new ways of controlling the increasingly complex music production software developed for laptops (fig. 3.1b). These coterminous developments turn out to have major impact on the forms and techniques of hip-hop production and performance, marking the convergence of multiple discursive spaces within electronic dance music culture—studio producers become stage DJs, laptops converge with mobile devices, and software becomes hardware.

---

[178] Brian Barrett, "End of an Era: Panasonic Kills Off Technics Turntables," *Gizmodo,* October 28, 2010, accessed April 18, 2016, http://gizmodo.com/5675818/end-of-an-era-panasonic-kills-off-technics-turntables. The SL-1200 was rereleased in 2016, marketed to an audience of mostly audiophiles rather than DJs. Nilay Patel, "The Technics SL-1200 turntable returns in two new audiophile models," *The Verge,* January 5, 2016, accessed April 18, 2016, http://www.theverge.com/2016/1/5/10718234/technics-sl1200-sl1200g-sl1200gae-turntable-new-models-announced-release-date-ces-2016.

Figure 3.1: (A) Technics SL-1200 turntable (1972); (B) Apple iPad (2010).

While turntablism thrives on the physical dexterity of the DJ and the visibility of the vinyl record, laptop musicians often struggle with constructing convincing stage performances. Since the computer serves as the primary focal point for the stage setup, laptop DJs are often accused of playing video games or simply checking e-mail without offering the audience an entertaining performance. DJ John Devecchis disputes the notion of laptop performance as a form of DJing altogether, as he asks, "how do you know the DJ is even playing? How do you know he's not playing a pre-recorded set? How do you know he's not playing Pac-Man while he's supposed to be DJing? I want to see the DJ doing something."[179] For Devecchis, as well as many other DJs and fans of electronic dance music, it is the lack of visibility in performance techniques that delegitimizes the skill of the performer, while disrespecting the expectations of certain audience members.

Debates concerning the proper techniques of electronic music performance proliferated on the heels of such technological changes, eventually coming to a head in 2013 as a result of a controversial statement by Joel Zimmerman, also known as Deadmau5, one of the most globally renowned DJs at the time. In a blog post titled, "we all hit play," Zimmerman claimed to speak for all of the "button pushers" who were too afraid to admit that most DJs "live" performances consist of simply getting on stage and pressing play: "its no secret. when it comes to 'live' performance of EDM… that's about the most it seems you can do anyway. It's not about performance art, its not about talent either (really its not)."[180] In direct response to DJs such as John Devicchis, who prioritizes individual skill and "paying your dues" as a turntable DJ, Zimmerman celebrates the *lack* of skill and technical accessibility of DJing in the digital age,

[179] Ed Montano, "'How do you know he's not playing Pac-Man while he's supposed to be DJing?': technology, formats, and the digital future of DJ culture," *Popular Music* 29 no. 3 (2010): 410. This issue was the target of satire in Andy Samberg's *Saturday Night Live* skit, "When Will the Bass Drop," *NBC,* accessed April 18, 2016, http://www.nbc.com/saturday-night-live/video/digital-short-when-will-the-bass-drop/2783137.

[180] Deadmau5, "we all hit play," *United We Fail* blog, 2013, accessed November 9, 2015, http://deadmau5.tumblr.com/post/25690507284/we-all-hit-play. Also see DJ A-Trak's perspective: "Don't Push My Buttons," *Huffington Post,* July 23, 2012, accessed April 18, 2016, http://www.huffingtonpost.com/atrak/dont-push-my-buttons_b_1694719.html.

claiming that "given about 1 hour of instruction, anyone with minimal knowledge of ableton and music tech in general could DO what im doing at a deadmau5 concert." The post immediately went viral amongst the online community of DJs and electronic music producers, inspiring heated exchanges and countless defenses of the lineage of "live" performance in DJ culture, including Twitter rebuttals from Zimmerman's friend and fellow DJ, Sonny Moore, also known as Skrillex.

The "button pusher" debate exemplifies many of the ongoing anxieties musical cultures experience with the rise of new technologies. For some audience members, the presence of a laptop on stage seems to negate the "live" aspect of the event and thus their own physical presence at the club, leading them to think, "why not just listen to the music in the isolation of my home?" For some DJs, particularly those who have dedicated years of their lives to learning the standard techniques of turntablism, the laptop delegitimizes the creative labors of a musical tradition nearly half of a century old. These moments of discursive tension, what Rebekah Farrugia and Thomas Swiss term "moments of resistance," have always arisen during periods of technological innovation within dance music culture, reflecting the nature of technological change in society more broadly.[181]

In gauging the quality of a DJ performance solely on a particular group of skills developed by the *human* musician, the "button pusher" discussion is limited to debating the individual performer's creativity. In doing so, it ignores the broader media environment within

---

[181] Rebekah Farrugia and Thomas Swiss, "Tracking the DJs: Vinyl Records, Work, and the Debate over New Technologies," *Journal of Popular Music Studies* 17, no. 1 (2005): 33. Raymond Williams describes the historical dialectic that occurs in the negotiation of new technologies: "At first glance there are simply dire predictions based on easily aroused prejudices against new technologies. Yet there are also phases of settlement in which formerly innovating technologies have been absorbed and only the currently new forms are a threat." *Television: Technology and Cultural Form* (Hanover and London: Wesleyan University Press, 1974), 133.

which DJ performances are embedded, including the influence of stage lighting, visual displays, and constantly changing technologies of musical performance. The level of musical control afforded by the human body, although privileged by many producers and audience members alike, is just one of moving parts that comes together to form the broader media environment of the electronic music event. Indeed, one of the more significant takeaways from the situation described in "we all hit play" is Zimmerman's attempt to emphasize the importance of the seemingly extra-musical factors such as multimedia accompaniment and audience energy levels in contributing to the overall "performance."

Especially in the midst of these moments of discursive tension within dance music culture, the very combination of instruments plays a significant role in constituting the performance environment. Rather than perceiving the technologies as threats to performance standards and conventions, as in the "button pusher" debate, music theorist Mark Butler describes the increasing prevalence of hardware "controllers" in the laptop performer's arsenal as tools for externalizing the perceptibly opaque creative processes happening behind the laptop screen. According to Butler, "Rarely if ever is a 'laptop set' *only* a laptop set. Instead, the internal, digital elements of the laptop environment are externalized—made physical in the form of MIDI controllers and other hardware devices."[182] In the wake of Zimmerman's critique (or celebration?) of the state of performance in electronic dance music culture, both stage DJs and studio producers have increasingly turned to hardware controllers as a means of heightening the physicality and theatricality of their "live" presence.

---

[182] Mark Butler, *Playing With Something That Runs: Technology, Improvisation, and Composition in DJ and Laptop Performance* (New York: Oxford University Press, 2014), 96.

Controllerism and the Materiality of Software

"Controllerism" emerged in the late 2000s within the electronic music community against the heated backdrop of the button pusher debate. While the term could be used to describe a vast number of performance techniques within electronic music, musician and hardware hacker Moldover broadly defines it as being "about making music with new technology. Right now controllers are where it's at, and so that's the name for the movement. Button-pushers, finger drummers, digital DJs, live loopers, augmented instrumentalists; we're all controllerists."[183] For Moldover, controllerism represents a unique stage in the development of music technology, one that materialized at a historical moment in which the vinyl record ceased being the sole interface for performing pre-recorded musical material. Indeed, it is the vast proliferation of digital music controllers that has defined electronic dance music production amidst the perceived twilight of vinyl, helping DJs and producers to navigate emerging tools and techniques through new forms of musical practice.

The use of MIDI devices to control digital software is the most common form of controllerism, with historical roots in the broader emergence of digital technology in the 1980s. Created in 1983, the musical instrument digital interface protocol (MIDI) was a significant development in the history of music technology in that it facilitated an increasing symbiosis between "digital" computers and "analog" musical hardware such as synthesizers.[184] The initial specifications were limited in terms of what types of connections could be made between instruments—allowing one synthesizer to control another synthesizer's notes or output volume,

[183] "Moldover: Performance and Controllerism," *Ableton* blog, November 28, 2013, accessed November 12, 2015, https://www.ableton.com/en/blog/moldover-performance-and-controllerism/.

[184] For more information on the historical development of the MIDI protocol, see Jessica Feldman, "The MIDI Effect" (paper presented at Bone Flute to Auto-Tune: A Conference on Music & Technology in History, Theory and Practice, Berkeley, California, April 24-26, 2014).

for example. However, after just a few years, MIDI features were adapted to many early computer platforms, including the Apple Macintosh, Commodore 64, and Atari ST. Throughout the late 1980s and 1990s, the MIDI functionality of the ST allowed it to become a useful music sequencer and musical instrument controller for a range of musical acts, including Madonna, Fatboy Slim, and Atari Teenage Riot.

In contemporary popular music since the early 2000s, MIDI devices have not simply been used to send control messages between instruments, but also as "live" instruments to be manipulated in real-time. Grid-based interfaces with rubber pads have become commonplace in the studio and on the stage, allowing the percussive triggering and automated sequencing of digital samples. Ableton's APC, Livid's OHM, the Monome, and Novation's Launchpad, among many others, are specifically catered to the "live" triggering and micro-manipulation of both musical patterns and sonic parameters such as volume, effects, and mixer settings (fig. 3.2). Other grid controllers are fashioned as entire studio workstations in themselves. Native Instruments describes its Maschine Studio as an "ultimate studio centerpiece for modern music production," specifically emphasizing the "unprecedented physical control and visual feedback" of the interface.[185] Designed by Ableton in collaboration with Akai Professional, the company responsible for the infamous MPC series drum samplers, the *Push* controller is likewise marketed as a digital controller that blurs the line between production and performance, presenting a staggering degree of fine-tuned control while composing using Ableton *Live* software.

[185] Native Instruments, "Maschine Studio," last modified February 1, 2016, accessed March 30, 2016, http://www.native-instruments.com/en/products/maschine/production-systems/maschine-studio/.

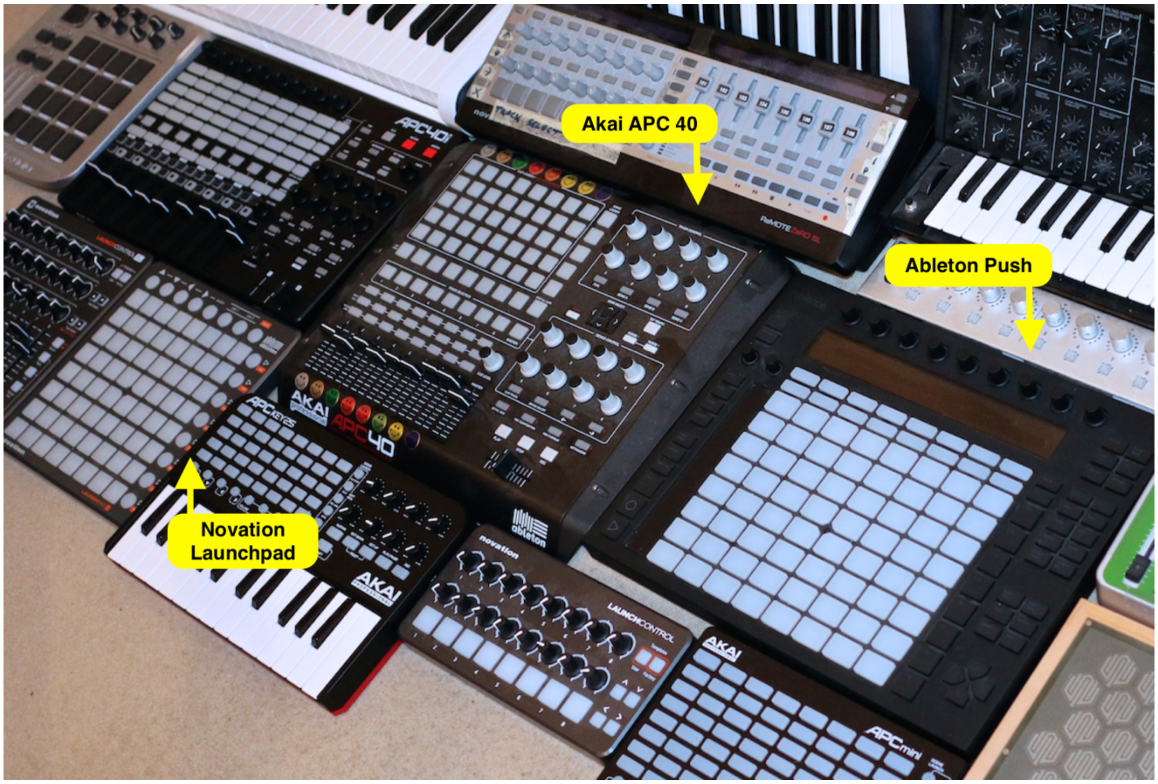


Figure 3.2: Grid-based MIDI controllers

The increasingly dialogic relationship between hardware and software in musical practice has ushered in new marketing models among software development companies. In the promotional campaign for their flagship controller, Ableton specifically emphasized the ways in which the Push hardware would allow the producer to focus less on the laptop screen in both stage performance and studio production. *Live* Certified Trainer and electronic music instructor Josh Weatherspoon describes the mindset behind this strategy from a marketing perspective:

> It's a bold and interesting thing for a company to say, "oh, just don't worry about the software." But the biggest thing that I think about for a lot of these companies is, because of piracy they have to have a way to make some money. So, if they lose a million dollars in piracy, and make 1.5 in hardware sales, it creates more of a demand and a brand tie… A lot of people are like "I got Live because of Push."[186]

[186] Josh Weatherspoon, Interview by Author, Digital Recording, Los Angeles, CA, October 23, 2014.

Just as performers use controllers to "materialize" and "externalize" the creative processes embedded within software programs, the developers of those programs design controllers as material manifestations of the software, making it more difficult for people to illegally download their products.

While grid-based controllers dominate the digital instrument industry through a carefully marketed alignment with proprietary music software, other controllerists feel limited by the creative constraints resulting from this integration. Brian Crabtree and Kelli Cain started building open source, minimalist controllers in 2006, seeking to construct "less complex, more versatile tools" than the cluttered interfaces being marketed to electronic musicians at the time. The company prides itself with operating "on a human scale," using only local suppliers and manufacturers, and embodying values of environmental and economic sustainability in their design process. This minimalist sensibility is embedded within products such as their Monome "grid" controller, in which the only control mechanism on the instrument exists in the form of small rubber buttons capable of sending simple on and off messages to open source software such as *Max* (fig. 3.3). Rather than perform with the seemingly prescribed options of proprietary software, Monome users build and freely share custom software patches that can be applied across a variety of artistic genres and creative needs.

Figure 3.3: Monome "grid" controller (2015)
(Source: Monome, "grid," last updated October 25, 2015, accessed January 5, 2016, http://market.monome.org/collections/primary/products/grid).

The alternative, do-it-yourself (DIY) rhetoric of flexibility employed by Monome designers intensifies a more fundamental concept ingrained within the aesthetics of live electronic music performance: personal "expression." With the emergence of the International Conference on New Interfaces for Musical Expression (NIME), the value of personal creative expression has become institutionalized amongst a particular demographic of electronic musicians, computer scientists, engineers, and designers. While Monome reflects this sensibility through a minimalist design aesthetic that claims to foster more adaptable forms of creativity, NIME attendees generally align themselves with the "tangible," "wearable" technology movement—focusing on interfaces that respond and react to bodily motion and gestural input, for example. The 2014 conference proceedings feature discussions of augmented pianos, modified *Guitar Hero* controllers, a wearable transducer system called "FingerSynth," tools for mediating vibrations of the vocal cord, and techniques for eye-controlled musical performance (fig. 3.4).[187] With backgrounds in music cognition, human-computer interaction, UX design, and computer science, among others, NIME participants clearly align musical expression with the embodied agency of the performer *as* controller.

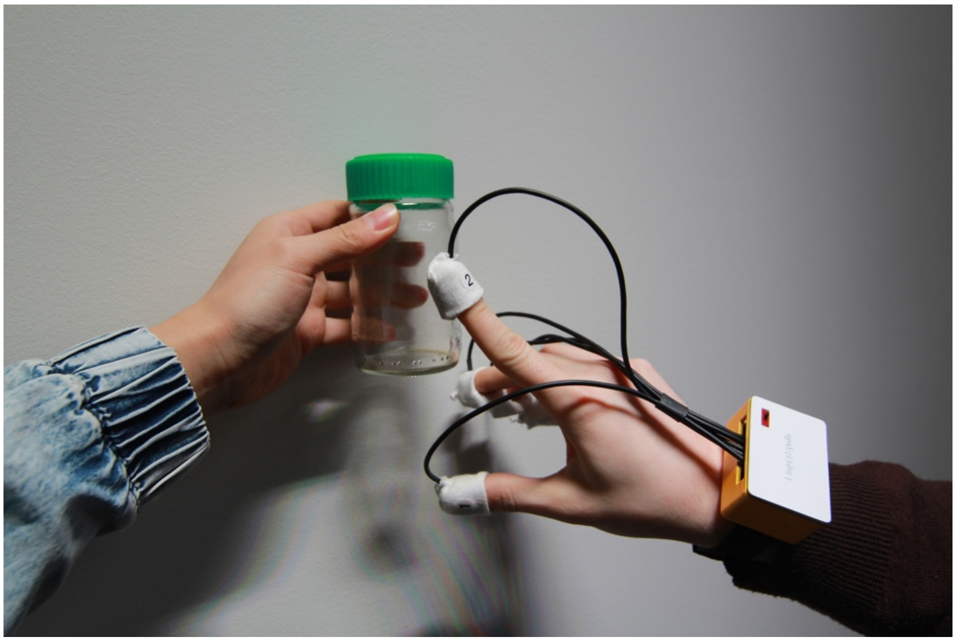


Figure 3.4: "FingerSynth" (2014), created by Gershon Dublon, creates sounds based on the density of objects it touches. [Source: Gershon Dublon and Joseph Paradiso, "FingerSynth: Wearable Transducers for Exploring the Environment with Sound," ***Proceedings of the International Conference on New Interfaces for Musical Expression,*** **Goldsmiths, University of London, UK (2014)]**

As these examples demonstrate, controllerism surfaced as an attempt by electronic musicians and designers to employ hardware as physical extensions of existing instruments, simultaneously enhancing the sense of tactile immediacy imbued by turntablism, and distinguishing themselves from the "we all hit play" paradigm detailed by Deadmau5. Indeed, Moldover defines the primary motivation for controllerism using the same critical language as

[187] NIME 2014, "Proceedings," last updated May 12, 2015, accessed January 5, 2016, http://www.nime2014.org/technical-programme/proceedings/.

vinyl purists, claiming "performers who use computer technologies as musical instruments needed a way to differentiate themselves from people who 'check their e-mail'."[188] At the same time—as the examples from NIME make clear—performing with vinyl without employing extensive sonic manipulation is also not enough for many controllerists, who emphasize "live" improvisation and the physical *display* of human-computer interaction on stage. In this way, controllerism positions itself as a progressive expansion of both laptop DJs *and* vinyl DJs who simply "hit play."

In developing more integrated human-computer interaction systems, controllerists negotiate complex and shifting conceptions of human and technological agency, embodied performance literacies, and computational forms of instrumentality. How might the stakes surrounding "live" electronic music in the 2000s simultaneously challenge and reify existing concepts of musical performance and instrumental embodiment? The next section of this chapter contextualizes controllerism by revisiting historical conceptions of technologically mediated performance from a variety of disciplinary perspectives, in an effort to outline the emergence of a play-oriented model of music performance at the interstices of process and object-oriented aesthetics.

"Liveness," Performance, and Electronic Music

Controllerism represents only one case story within a broader tradition of electronic musicians learning to navigate rapidly changing technologies, industries, and aesthetics. Indeed, the question of how to perform electronic music in a "live" setting has been contemplated since the

---

[188] Ean Golden, "Music Maneuvers: Discover the Digital Turntablism Concept, 'Controllerism,' Compliments of Moldover," *Remix,* October 2007, accessed April 17, 2016, http://www.moldover.com/press/Moldover_Remix_Oct-2007_w.jpg.

advent of tape music and electroacoustic traditions in the mid-twentieth century. When Pierre Schaeffer developed the concept of "concrete music" in the late 1940s, he was primarily concerned with the ways in which new forms of composition with phonograph discs could liberate the listener, allowing them to hear sounds acousmatically—separate from their source, no longer dependent on "preconceived sound abstractions," and removed from the "elementary definitions of music theory."[189] Foreshadowing concerns surrounding the visibility of electronic music production techniques from the audience's perspective, Schaeffer quotes Paul Valéry: "Looking at this seashell, in which I seem to see evidence of 'construction' and, as it were, the work of a hand not operating by 'chance,' I wonder: Who made it? […] *And now I strive to find out how we know that a given object is or not made by a man*."[190] This attempt to discern whether or not a human agent constructed a given sound lies at the heart of the acousmatic dilemma in various forms of electronic music, from tape and electroacoustic music to contemporary "button pushers."

In the 1950s, composers such as Vladimir Ussachevsky, his colleague Otto Luening, and experimentalist John Cage expanded on Schaeffer's ideas through the use of the tape recorder, cutting and splicing electromagnetic tape, creating loops, changing speed and transposing pitch, and reversing and delaying tape playback. Together, the various techniques associated with tape splicing offered radically new possibilities for the composer, while challenging audience members to rethink the purpose and function of "live" performance. In the context of early tape

---

189 Pierre Schaeffer, quoted in Brian Kane, *Sound Unseen: Acousmatic Sound in Theory and Practice* (New York: Oxford University Press, 2014), 16.

190 Schaeffer, quoted in Daniela Cascella, "Sound Objects: Pierre Schaeffer's 'In Search of a Concrete Music'," *Los Angeles Review of Books,* April 3, 2013, accessed April 6, 2015, http://lareviewofbooks.org/review/sound-objects-pierre-schaeffers-in-search-of-a-concrete-music, Emphasis added.

music performance, in which the human performer was entirely absent from the stage, the audience would face this question head on, forced to aurally decipher evidence of the tape's construction in the lack of extra-sonic cues.

Many composers of tape and computer music claim that this dilemma can be solved by interpreting behavioral relationships in the spectromorphology of the sounds themselves.[191] From this perspective, focusing on the gestural ways in which sound itself emerges, rises, falls, leaps, and decays allows listeners to bind aural perception to a sonic source. For example, John Young analogizes the spatial field of the electroacoustic listening experience to aspects of our "environmental reality," claiming that although sound itself may not arise from a particular environment or cultural source, "it [may] nevertheless serve to define a 'realistic' acoustic space and behave as though it were a physical entity."[192] In perceiving physical distance and behavioral relationships in acousmatic sound, Young thus assumes that the solitary tape recorder on stage may, in turn, analogize precisely those qualities of performance that designate the "live" human presence of an event. In other words, the sense of "liveness" in the work of tape and electroacoustic musicians is considered, by composers and listeners, to arise from the metaphorical gestures embedded within sound itself.

With many tape and computer music pieces, the sonic *content* of the composition is the primary focal point. When *acts* of technological mediation become foregrounded by artists, however, the overt physicality and "presence" of the performer is often valued as a force in

---

[191] John Young, "Sound Morphology and the Articulation of Structure in Electroacoustic Music," *Organised Sound* 9, no. 1 (2004): 7-14; Denis Smalley, "Spectromorphology: Explaining Sound-Shapes," *Organised Sound* 2, no. 2 (1997): 107-26; Trevor Wishart, *On Sonic Art,* edited by Simon Emmerson (Amsterdam: Harwood Academic Publishers, 1996).

[192] John Young, "The Extended Environment," *Proceedings of the 1994 International Computer Music Conference* (Aarhus: Danish Institute of Electroacoustic Music, 1994), 24.

creating "expressive" (re: "live") performances. The question of what makes a mediated performance "live" has been fundamental to the work of a range of theorists and practitioners across disciplines. Performance theorists, for example, have focused on physical *presence* as a defining feature in the ontology of performance. Peggy Phelan defines performance as "representation without reproduction," arguing that "performance's only life is in the present" and that it cannot be "saved, recorded, documented, or otherwise participate in the circulation of representations *of* representations." In other words, if a performance was to be recorded in any way, it would become "something other than performance," since performance "becomes itself through disappearance."[193] For Phelan, the essence of performance lies in the temporal uniqueness of a real-time event inherently opposed to mediated representations. Crucially, this ontology aligns itself with a politics of identity that is always in opposition to the constructed representations of subjectivity through media formats. Performance, in this context, is a channel through which subjects are able to escape the confines of hegemonic forms of mediated representation.

Musicologists have mostly extended this presence-based mode of thought, valuing the individual performer as the locus of the creative act. Expanding on philosopher Vladimir Jankélévitch's ontology of music performance as ineffable—existing in time as a material acoustic phenomenon—Carolyn Abbate argues that "real music" is most clearly expressed in "an actual live performance (and not a recording, even of a live performance)."[194] Ethnomusicologist Charles Keil developed the notion of "participatory discrepancies" (PD) to highlight the ways in which performers, in their face-to-face relationships on stage, create differential tension in the

[193] Peggy Phelan, *Unmarked: The Politics of Performance* (New York: Routledge, 1993), 146.
[194] Carolyn Abbate, "Music—Drastic or Gnostic," *Critical Inquiry* 30, no. 3 (2004): 506.

expected rhythmic, tonal, and textural aspects of pre-composed or otherwise patterned musical processes. Similar to Abbate's notion of "real music," Keil's PD model is rooted in the material, physical, and corporeal aspects of performance: sticks tapping metal, fingers plucking strings, a "dialectical materialism in action."[195] Keil specifically denigrates electronic music for being out of touch with the "natural" world, as he writes: "The expanding mix of 'mediated and live' musics seems like a limbo or purgatory to me because the organic feedback loops are not complete and co-evolving, not in touch with nature and neighbors, always limited in at least a few ways by electricity, machinery and commodity forms."[196] Employing a similar rhetoric as Phelan, Keil creates a dichotomy between the ephemerality of the live event and implied freedom of the human performer on one hand, and the seemingly rigid representational structures of electronic media and post-industrial capitalism on the other.

However, poignant counterarguments to the privileging of the corporeal amongst music and performance theorists have emerged. Philip Auslander describes the ways in which impressions of "liveness" arise from the increasing influence of technology and digital media within the performance context. In direct opposition to Phelan, Auslander argues that the historical trajectory of artistic performance in the twentieth century is defined by the rise of mediatization as a primary component of what is perceived to be "live" performance, as he writes, "[i]nitially, mediated events were modeled on live ones. The subsequent cultural dominance of mediatization has had the ironic result that live events now frequently are modeled on the very mediatized representations that once took the self-same live events as their

[195] Charles Keil, "The Theory of Participatory Discrepancies: A Progress Report," *Ethnomusicology* 39, no. 1 (1995): 3.

[196] Ibid, 13.

models."[197] Examples of the desire for recorded musical aesthetics in live performance include the use of autotune on stage, live video performances at EDM shows, and the emergence of posthumous duets (fig. 3.5).[198]

Figure 3.5: Tupac hologram performs at the Coachella music festival (2012) (Source: Jack Hamilton, "Tupac Hologram Was Inevitable," *The Atlantic,* April 17, 2012, accessed May 1, 2016, http://www.theatlantic.com/entertainment/archive/2012/04/hologram-tupac-was-inevitable/255990/).

While troubling the binary between "live" and "mediated" performance, positing the term "liveness" functions to retain the primacy of the artists' authorial *presence* and *agency* as the measure of contemporary performance. This is echoed in the language of electronic musicians themselves. Simon Emmerson details the ways in which changes in software have facilitated a shift in electronic music performance from the studio to the stage. For example, the advent of "live coding," introduced in chapter two, highlights the ways in which "what used to be the

[197] Philip Auslander, *Liveness: Performance in a Mediatized Culture,* 2nd edition (New York: Routledge, 2008), 10.

[198] For example, see Boris Kachka, Rebecca Milzoff, and Dan Reilly, "10 Tricks That Musicians and Actors Use in Live Performances," *Vulture,* April 3, 2016, accessed April 18, 2016, http://www.vulture.com/2016/03/live-performance-tricks.html; Chenda Ngak, "Tupac Coachella hologram: Behind the technology," *CBS News,* November 9, 2012, accessed April 18, 2016, http://www.cbsnews.com/news/tupac-coachella-hologram-behind-the-technology/.

studio's domain becomes available for 'live' working."[199] The emphasis on "real-time" computational processes acts to foreground the presence of the "analog" human body interacting with the "digital" computer. For Emmerson, the blurred line between creative practices in the studio and on the stage reflects what he calls a "reanimation" of technical objects previously thought to exist outside of what has traditionally been considered musical performance.[200] In contrast to the more outdated anxiety that humans may one day disappear into a sort of machine-dominated cyber-reality, Emmerson reasserts the centrality of the body in human-computer interaction, claiming the agency of human presence as the source for a "reanimation" of "dead" technologies that perceptibly lack the capabilities for dynamic change in performance.

Electronic musician Primus Luta presents a similar conflation of human agency and "real-time" performance by outlining what he calls "variability" in live electronic music performance. Luta employs this term in reference to the multifarious ways in which technical objects afford real-time manipulation by both human agents and other technical objects.[201] Theoretically, the concept of variability paves the way for thinking about an object-oriented aesthetics in digital performance—that is, focusing on "things at all scales, and pondering their nature and relations with one another as much with ourselves," as articulated by Ian Bogost.[202] In this context, assessing the "liveness" of a given electronic music performance is possible by understanding the potential variability of a particular instrument *in relation to* the ways in which

---

[199] Simon Emmerson, *Living Electronic Music* (Burlington, VT: Ashgate, 2007), 27.

[200] Ibid, 53.

[201] Primus Luta, "Toward a Practical Language for Live Electronic Performance," *Sounding Out!,* April 29, 2013, accessed January 5, 2016, http://soundstudiesblog.com/2013/04/29/toward-a-practical-language-for-live-electronic-performance/.

[202] Ian Bogost, "What is Object-Oriented Ontology? A Definition for Ordinary Folk," December 8, 2009, accessed April 18, 2016, http://bogost.com/writing/blog/what_is_objectoriented_ontolog/.

a performer is or is not exploring that variability. However, Luta explicitly states that his model is meant to describe the performance of electronic music in a way that makes it "easier to parallel with traditional western forms," and his analogies to jazz make it clear that "variability" in this context is understood as an extension of virtuosic improvisation.[203] "Liveness" is once again conflated with the physical processes of sonic manipulation afforded by the human body in resistance to technology.

While many DJs and producers see clear links between their own work and the work of jazz musicians, they downplay the unique aspects of human-computer interaction in dance music performance by privileging values such as human agency and virtuosity. In a response to Luta's series, philosopher Robin James argues that *artist* agency in performance remains the underlying concept and value driving Luta's project. Since Luta works from the widely shared perspective that "electronic music is the new jazz," he ends up transposing jazz aesthetics "into terms compatible with electronic instruments and genres," according to James.[204] In an attempt to push the discussion beyond the frame of modernist aesthetics, James poses the question of what an object-oriented aesthetics might entail, thinking specifically about expanding on the idea of "musical objects as performers." This concept of decentering the "human" from ontologies of performance has important implications in conceptualization musical production as an environment, or network, of reciprocal feedback between human and computer.

Recent musicological scholarship has introduced useful models for the integration of object-oriented ontologies with studies of music production. Simultaneously criticizing and

---

[203] Primus Luta, "Musical Objects, Variability and Live Electronic Performance," *Sounding Out!,* August 12, 2013, accessed January 5, 2016, http://soundstudiesblog.com/2013/08/12/musical-objects-variability/.

[204] Robin James, "Aesthetics and Live Electronic Music Performance," *Cyborgology,* December 13, 2013, accessed January 5, 2016, http://thesocietypages.org/cyborgology/2013/12/13/aesthetics-and-live-electronic-music-performance/.

riffing on Auslander's concept of "liveness," Jason Stanyek and Benjamin Piekut coin the term "deadness" as a response to "an unhelpful and overvalued schism between presence and absence that undergirds much literature on performance."[205] Whereas scholars of music and technology have written about the aural ruptures and disembodied *absences* produced by sound recording, performance theorists alternatively focus on the "here and now" *presence* of musical events.[206] Through analyses of posthumous duets and the historical development of recording studio technologies, Stanyek and Piekut propose a model of "intra-action" between humans and inscription devices, in which the "corpauralities" between sounds and the bodies that produce them helps "to sonify the idea of distributed personhood" throughout the recording process.[207] The case of the posthumous duet serves to emphasize the constantly changing relationships between the material histories embedded within sound media and the social network of agents involved in the music production process. Here, human and technological agency is positioned at the edge of a blurry line between presence and absence.

Ethnomusicologist Eliot Bates takes the notion of agency even further, examining the ways in which musical instruments can be regarded as constitutive of social interactions, rather than as incidental effects or reflections of those relationships. Borrowing from Bruno Latour's "Actor-Network Theory" (ANT), Bates defines instruments as social *actors,* neither subjects nor objects, but sources of action that imply "no special motivation of human individual actors, nor

---

[205] Jason Stanyek and Benjamin Piekut, "Deadness: Technologies of the Intermundane," *The Drama Review* 54, no. 1 (2010): 20.

[206] Allen S. Weiss, *Breathless: Sound Recording, Disembodiment, and the Transformation of Lyrical Nostalgia* (Middletown, CT: Wesleyan University Press, 2002); Evan Eisenberg, *The Recording Angel: Music, Records and Culture from Aristotle to Zappa* (New Haven and London: Yale University Press, 1987).

[207] Ibid, 19.

of humans in general."[208] Covering a wide variety of musical genres and cultures, he outlines the ways in which instruments seem to take on lives of their own, possessing "a propensity to teach their owners how to play them," channeling spiritual powers, and mediating interpersonal disputes in communities.[209] In the context of musical performance, this concept draws attention to the instrument as an object that both shapes and is shaped by the performer, encouraging a more relational view of the music production process.

Within electronic dance music, musical recordings and digital controllers are two of the primary "actors" involved in the broader "network" of performance. Butler, for example, argues that recordings create dialogic "networks" between *processes* of musical production and their material *products,* thus providing a potential solution to the previously outlined dichotomy between absence and presence in performance. For Butler, the technological mediation that characterizes contemporary musical performance leads to reciprocal relationships between process- and object-oriented perspectives that are dialectical rather than dichotomous. Attributing equal ontological significance to recording, performance, composition, and improvisation reveals the dynamic interaction of these modalities constantly at play within both studio production and stage performance.[210]

In the next section of this chapter, I combine the object-oriented aesthetics of Latour and others with the process-oriented aesthetics of controllerism, proposing a play-oriented model of electronic music performance. Rather than focusing on either the "live" agency of the human performer or the processes of sound manipulation afforded by the technological hardware, a

---

[208] Bruno Latour, quoted in Eliot Bates, "The Social Life of Musical Instruments," *Ethnomusicology* 56, no. 3 (2012): 372.

[209] Ibid, 364.

[210] Butler, 4.

play-oriented model of performance acknowledges the experimental negotiations that continuously emerge in every human-computer interaction. Encapsulating the core elements of procedural listening introduced in chapter two, play provides digital media producers with a necessary skillset in understanding the "rules" embedded within software. While video games emanated at a similar historical juncture as the musical genres under discussion, the rise of controllerism foregrounds the significance of embodied experimentalism inherent to what media theorists have termed the "ludic turn" in contemporary culture.[211] In addition to voicing metaphorical connections between the materiality of video games and music production, electronic music producers have increasingly rooted their creative techniques and practices in the playful logic of gaming.

Controller Design for Gaming

Having outlined the ways in which technologically mediated performance can be viewed as a dynamic relationship between objects and processes, it is possible to return to the controllerism debate with a renewed focus on the material relationships negotiated between hardware controllers and software programs. If vinyl record performance foregrounds the agency and presence of the musician, controllerist performance foregrounds the negotiation between the musician and the "rules" of the software. This dialectical relationship between hardware (human bodies, material technologies) and software (processes, logics, and mechanics of code) also finds a direct analogy in the structures of video game play.

---

[211] Joost Raessens, *Homo Ludens 2.0: The Ludic Turn in Media Theory* (Utrecht: Utrecht University, 2012); "Playful Identities, or the Ludification of Culture," in *Handbook of Computer Game Studies,* edited by Joost Raessens and Jeffrey Goldstein, 373-88 (Cambridge, MA: The MIT Press, 2006).

The status of being a "button pusher" is not simply a denigrating term for artists working with hardware controllers, but a metaphor for the convergence of a gaming logic with digital music production. Speaking of his own influences from video gaming, Flying Lotus talks about growing up as an only child who "didn't have too many friends, but I had Nintendo." Like many electronic musicians growing up in the 1980s, the dawn of the gaming age, FlyLo cites that period as formative in his creative development, proudly stating that "Those sounds are part of my youth, part of my history."[212] Images of his studio attest to this phenomenon of media convergence, writer Jeff Weiss describing his workspace as "a mess of keyboards, DVDs, video games, computers and a drum kit."[213] Glasgow's bass music pioneer Rustie talks about how his production styles emulate the way gamers play, describing his experience with the electric guitar and video games as "different means to the same end, really… there's not much difference between plucking a string and pressing a button, I think."[214] The 2000s witnessed the emergence of a new generation of electronic musicians, one that grew up on Nintendos, Game Boys, and Ataris, rather than their parents vinyl record collection, and the performance practice of pressing buttons and swiping screens reflects this.

Recently, musicologist Roger Moseley introduced "ludomusicology" as a theoretical model with which to analyze the shared experiences of play, performance, and digital embodiment in both gaming and music production. Most significantly, ludomusicology is concerned with "the extent to which music might be understood *as* a game"—as a system of

---

[212] Louis Pattison, "Video Game Loops & Aliens: Flying Lotus Interviewed," *The Quietus,* May 18, 2010, accessed November 13, 2015, http://thequietus.com/articles/04269-flying-lotus-interview-cosmogramma.

[213] Jeff Weiss, "Flying Lotus' Nocturnal Visions," *LA Weekly,* October 4, 2012, accessed June 8, 2015, http://www.laweekly.com/music/flying-lotus-nocturnal-visions-2611706.

[214] Drew Millard, "Interview: Rustie used to produce like how gamers game," *Pitchfork*, May 31, 2012, accessed November 10, 2012, http://archive.is/dzPoJ.

rule-based logics that "constitute a set of cognitive, technological, and social affordances for behaving in certain ways, for playing in and with the world through the medium of sound and its representations."[215] If, as Moseley suggests, musical scores, software code, and hardware interfaces constitute "the ludic rules according to which music is to be played," what might the technical practices of digital music producers say about the shifting nature of musical performance and instrumentality *as* play?

In order to recognize the explicit connection between gaming and music production, it is necessary to understand how the experience of play is capable of facilitating creative experiences in general. The notion of constraints as an engine for creativity and experimentation within closed, interactive systems has become an overarching framework for explaining the allure of play as a cultural force.[216] In a succinct definition that could be applied equally to music and gameplay, Bernard Suits describes gaming as "the voluntary attempt to overcome unnecessary obstacles."[217] Whereas musical play is often conceived as allowing an unfettered creative experience—the idea that technologies allows for the creation of "any sound you can imagine"—embodied interaction with games and electronic music may be more aptly characterized by the ways in which the media *resists* or *constrains* the actions of the user.[218]

---

[215] Roger Moseley, "Playing Games With Music (and Vice Versa): Ludomusicological Perspectives on *Guitar Hero* and *Rock Band*," in *Taking it to the Bridge: Music as Performance,* edited by Nicholas Cook and Richard Pettengill (Ann Arbor: University of Michigan Press, 2013), 286.

[216] Jane McGonigal, *Reality is Broken: Why Games Make Us Better and How They Can Change the World* (New York: The Penguin Press, 2011); Tom Bissell, *Extra Lives: Why Video Games Matter* (New York: Pantheon Books, 2010); Katie Salen and Eric Zimmerman, *Rules of Play: Game Design Fundamentals* (Cambridge, MA: The MIT Press, 2003).

[217] Bernard Suits, *The Grasshopper: Games, Life, and Utopia* (Ontario: Broadview Press, 2005), 55.

[218] The marketing of music technology as allowing an unfettered creative experience is discussed in-depth in Paul Théberge, *Any Sound You Can Imagine: Making Music/Consuming Technology* (Hanover, NH: Wesleyan University Press, 1997).

Design constraints are particularly useful to consider when discussing creative forms of human-computer interaction (HCI). While designers working in the fields of user experience (UX) design and HCI continue to research practical strategies for productively balancing the affordances and constraints of *everyday* technological experience,[219] game designer Brian Upton details the ways in which constraints operate across four categories of a player's *creative* experience: technical design ("the game as designed"), player experience ("the game as encountered"), socio-cultural understandings and real-world knowledge ("the game as understood"), and practical experience ("the conceptual background").[220] Upton's model usefully aligns technical and cognitive aspects of creativity with social and cultural experience, emphasizing that although constraints are *embedded* within the design of a game, it is only through play that they become *embodied* values in the player. Here, game design—and the creation of "playful" systems more broadly—is less about building one-way systems that respond to direct player input, and more about encouraging the formation of internal constraints that facilitate creative experimentation in the mind of the player. In the context of music, for example, these internal constraints constitute years of musical training on a specific instrument. Similarly, games construct these constraints through rote repetition in the players' experience of gameplay itself.

Whether embedded within the instrumentality of music or gameplay, constraints are most often perceived in the physical comportment of the player as he or she interacts with a technological apparatus, the interface shaping his or her embodied knowledge and practices. Dance scholar Harmony Bench has examined the gestural choreographies through which users

---

[219] Donald Norman, *The Design of Everyday Things* (New York: Basic Books, 1988).

[220] Brian Upton, *The Aesthetic of Play* (Cambridge, MA: The MIT Press, 2015), 33-4.

comport themselves while engaging with touch-based digital media devices, for example. Noticing the ways in which "their bodies curved into supportive architectures with which they cradled touch-screens," Bench argues that these "digital media choreographies" encourage the development of bodily techniques across media and technologies, simultaneously ushering in new understandings of physical and bodily comportment, and serving as the mechanisms for that education.[221] Bench specifically aligns musicianship with the sort of "computational literacy" of gaming, detailing the significance of rote repetition in the development of embodied knowledge within each practice, as well as the ways in which each "demand[s] a corporeal training that impacts operators' experiences of their physicality."[222] Think of the ways in which musicians, gamers, and computer operators alike must constantly update their skills based on the rapid, and often radical, changes made to common operating systems, game controllers, and digital musical interface design.[223] While scholars have previously examined the "medium-specific" modes of embodiment that reshape technological users' bodily structures, Bench's analysis is not limited to a single platform, allowing her to highlight gaming and music production as shared avenues for the embodiment of systematic design constraints that ultimately function in shaping the bodily comportment of the player.[224]

---

[221] Harmony Bench, "Gestural Choreographies: Embodied Disciplines and Digital Media," in *The Oxford Handbook of Mobile Music Studies, Volume 2,* edited by Sumanth Gopinath and Jason Stanyek (New York: Oxford University Press, 2014), 238.

[222] Ibid, 243.

[223] Ibid, 245. Software developer Miller Puckette sees the constant updating of proprietary technology as a major hindrance to good instrument design, rapidly increasing the obsolescence of musical tools and preventing the interoperability of musical tools across technological platforms.

[224] See also Ingrid Richardson, "Faces, Interfaces, Screens: Relational Ontologies of Framing, Attention and Distraction," *Transformations* 18 (2010); Ingrid Richardson, "Ludic Mobilities: The Corporealities of Mobile Gaming," *Mobilities* 5 (2010): 431-447; Ingrid Richardson and Rowan Wilken, "Haptic Vision, Footwork, Place-Making: A Peripatetic Phenomenology of the Mobile Phone Pedestrian," *Second Nature: International Journal of Creative Media* 1 (2009): 22-41; Ian MacColl and Ingrid Richardson, "A cultural somatics of mobile media and urban screens: Wiffiti and the IWALL prototype," *Journal of*

Game controllers are particularly important conduits for the transmission and negotiation of design constraints, aiding in the embodied cognition of social values, haptic metaphors for technological interaction, and expected patterns of use. In other words, controllers externalize the "rules" embedded within digital systems. According to game theorist David Myers, all video game controllers share at least two formal properties that directly shape players' embodied practices: "they employ arbitrary and simplified abstractions of the physical actions they reference, and they require some level of habituation of response."[225] Indeed, while early video game controller *hardware* was often designed to match the on-screen actions afforded by individual game *software*, contemporary controllers have adapted more uniform design conventions, so as to facilitate a steeper learning curve while allowing the device to function across a variety of games. For example, Xbox One and Playstation 4 controller schemes (the most popular handheld controllers at the time of writing) are similar in their dual-joystick layout, abstracting a complex set of buttons and triggers to letters and shapes (fig. 3.6).[226] The fact that the buttons on the controller are mapped to the representational actions on screen in an abstract way encourages the player to focus on what Myers calls "locomotor play," a form of technological engagement specifically involving "the manipulation of the interface between our

---

Urban Technology 15, no. 3 (2008): 99-116; Ingrid Richardson, "Pocket Technospaces: The Bodily Incorporation of Mobile Media," *Continuum: Journal of Media and Cultural Studies* 21 (2007): 205-15; Ingrid Richardson, "Mobile Technosoma: Some Phenomenological Reflections on Itinerant Media Devices," *The Fibreculture Journal* 6 (2005); Rowan Wilken, "From *Stabilitas Loci* to *Mobilitas Loci:* Networked Mobility and the Transformation of Place," *The Fibreculture Journal* 6 (2005).

[225] David Myers, "The Video Game Aesthetic: Play as Form," in *Video Game Theory Reader 2,* edited by Bernard Perron and Mark J.P. Wolf (New York: Routledge, 2009), 50.

[226] For more information on the technical design and use of gaming controllers, see Pippin Barr et al, "Video Game Values: Human-Computer Interaction and Games," *Interacting With Computers* 19 (2007): 180-195; Ben Heatherly and Logan Howard, "Video Game Controllers," accessed June 8, 2015, http://andrewd.ces.clemson.edu/courses/cpsc414/spring14/papers/group2.pdf.

bodies and our environment."[227] Abstraction in the hardware interface is thus used as a method for managing the complexity of the software, allowing the player to physically internalize the constraints of the controller that are required to succeed in a variety of gaming genres. How might these design constraints apply to digital music-making—a practice that asks the musician to navigate complexities in both HCI and the conflicting perspectives on "liveness" previously discussed?

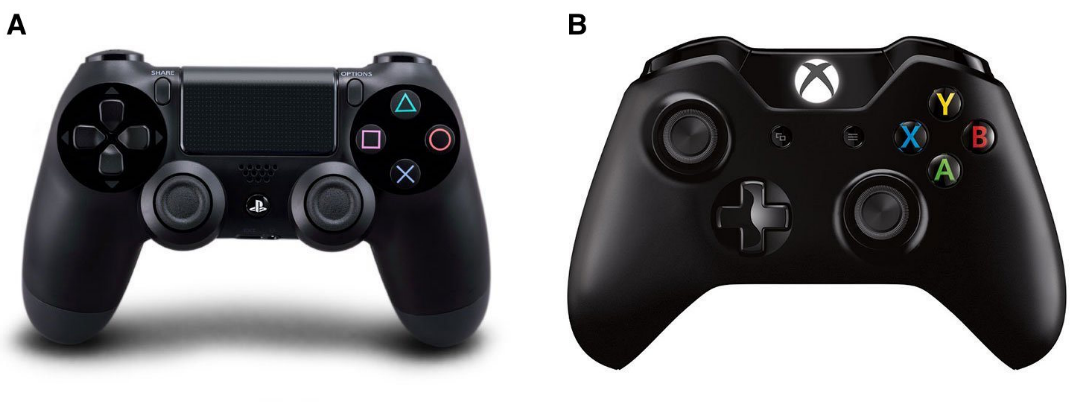


Figure 3.6: (A) Playstation 4 controller (2013); (B) Xbox One controller (2013).

Controller Design for Music-Making

As with the development of motor memory in video games, training on a musical instrument involves the internalization of the affordances and constraints of a given instrument *through* the rote repetition of bodily techniques and habituated responses. Musicologist Elisabeth Le Guin discusses the ways in which cellists physically comport themselves in relation to the cello during performance, molding themselves into a single "cellist-body" through movement and action.[228]

---

[227] Ibid, 46.

[228] Elisabeth Le Guin, "'Cello-and-Bow Thinking': Boccherini's Cello Sonata in Eb Major, 'fuori catalog'," *Echo: A Music-Centered Journal* 1, no. 1 (1999), accessed March 30, 2016, http://www.echo.ucla.edu/Volume1-Issue1/leguin/leguin-article.html.

Just as gamers embody the internal constraints embedded within the game itself, instrumentalists develop an embodied understanding of the constraints embedded within a given piece of music. Le Guin defines this skill as "anticipatory kinesthesia," in which the performer assesses the physical demands of a given piece on their body, asking such questions as "What do I need to do in order to play this? Where will I put my hands, and how will I move them?" Most instrumentalists would not be able to articulate clear answers to these questions, in the same way that most gamers would have trouble putting to words such a deeply embodied practice. Rote repetition is thus capable of facilitating the acquisition of tacit, embodied knowledge.[229]

To the extent that they also function through "arbitrary and simplified abstractions of the physical actions they reference," electronic musical instruments are even more closely aligned with the haptic structures of video game controllers. While instruments such as a violin, acoustic guitar, or snare drum reflect a 1:1 ratio between the physical gesture of instrumental attack (bowed, plucked, or strummed strings, for example) and the sonic output (string vibrates at a specific frequency), digital music controllers can be "mapped" to any number of sounds. Tapping a pad on a drum machine connected to Ableton *Live* could just as likely trigger a single snare drum sample, or an entire multi-movement symphony. In the same way that game controllers externalize the "rules" designed with the game, the constraints of electronic music production and performance are found in the limitations of the instrument's formal structures, rather than the audio content being created. In confining oneself to a simplified, abstract *hardware* interface (a sixteen-pad grid of rubber buttons, for example), the performer is able to creatively exceed what is often perceived to be over determined, complex, and formally delineated music production *software*.

[229] Michael Polanyi, *The Tacit Dimension* (Chicago: University of Chicago Press, 1966).

As is the case with embodied, tacit knowledge in game controllers, the arbitrary mappings of musical software onto hardware ask the player to internalize a constantly changing set of embodied musical techniques. This process of interface abstraction may be most clearly exemplified in the minimalist design of the Monome "grid" controller, which comprises a small rectangular box fitted with a symmetrical grid of small rubber buttons and a USB port.[230] Often, the Monome is used as a controller for the *Max* visual programming environment, which is itself a modular, open software that can be used for a variety of creative practices from electronic music synthesis to the real-time generation of 3D visuals. In this context, the button grid interface can take the form of a pitch controller alternative to the keyboard interface, an externalization of a step sequencer, a multi-track mixer or effects modulator, a visual spatialization map, and any number of other tools. Approaching the blank, terminal interface of an instrument such as the Monome, the musician must focus more on the internalization of specific software affordances, rather than the external affordances of the minimalist hardware.

This internalization of software (what we might call "music as designed," following Upton) *through* hardware ("music as encountered") has two seemingly opposing effects on electronic music production. First, as the processing power for a given musical task is increasingly delegated to the software, the physical and gestural manipulation of the hardware becomes increasingly unnecessary. This fact is highlighted by trends in game controller and interface design more broadly, which value the least amount of effort to achieve the maximum output. In the context of games, a single, slight flick of a Playstation 4 controller's right trigger may just as likely fire a gun, swing a sword, open a door, or detonate a series of explosives. In

[230] Butler describes the "adaptability" inherent to the Monome's generality of function, as "the buttons have no set correspondence with particular musical parameters or function; indeed, the interface need not control music at all." *Playing With Something That Runs,* 88.

the context of musical production and performance, the single tap of a rubber pad may just as likely trigger a single snare drum sample, a four-bar drum loop, or an entire musical album. In valuing the arbitrary design of musical gestures, digital music controllers have encouraged both musicians and audience members to develop new forms of embodied listening and production. It is this transitional moment that sparked the vehement and ongoing debates about human agency in performance detailed in the opening of this chapter.

Increased complexity in software design seems to facilitate a decreased complexity in hardware design, leading to what Bart Simon terms a "gestural minimalism" in gaming that could equally apply to musical performance. However, as the player develops an embodied knowledge of the software's "rules," he or she is able to dedicate more attention to the physical control of the hardware itself. This leads to the common experience of what Simon alternatively calls "gestural excess" in gaming, when physical movements are made in excess of what the hardware is actually capable of performing. For example, even though the joystick of a controller may be the only mechanism capable of steering a car in a racing game, the player often exceeds this limitation by gesturing with the controller itself *as* a steering wheel, dynamically contorting their entire body to the left and right as if controlling an actual car. This becomes a subconscious attempt to overcome the arbitrariness of the digital "mapping" by foregrounding the embodied metaphor on which the software is designed. Just as these gestures function to translate the "rules" of the game to the player, embodied metaphors can likewise translate a sense of musicality and performativity to an audience.

For electronic musicians, gestural excess represents a clear strategy for conveying a sense of "liveness" to their audience, while developing performance strategies for the embodied control of musical techniques embedded in software. Describing a performance from German electronic musician Stefan Betke (also known as Pole), Butler writes about what he calls the "passion of the knob," in which the producer "seems to put his whole body into the extended turning of a knob," directing an "exceptionally intense expressivity toward a small, technical component associated with sound engineering."[231] These gestural excesses are highly choreographed, as the performer "telegraphs 'expressivity'" to the pre-recorded musical material, locating him or herself as the primary agent of the sounds being heard by the audience.[232] In a way, this mode of performance is meant to foreground the "human" presence while effacing the technological apparatus. At the same time, highlighting the physical practice of interfacial mediation likewise foregrounds the mechanics and "rules" embedded within the apparatus, thus indoctrinating the audience into new modes of listening to the interface. In other words, gestural excess gives the audience a practical method for listening to the electronic music controller as a *process*-based musical instrument, rather than a tool simply to be used for the composition of sound *content.*

Daedelus, a Los Angeles-based producer and DJ, has become infamous for his use of controllers to externalize the mechanics of music software in performance. The relationship between gameplay and music is further highlighted by the type of creative work to which he dedicates himself, including interactive audio installations, sound design for video games, and controllerism in live performance. In a particularly fitting video shoot produced by the news and media website *Into the Woods*, he performs an entire "DJ" set in the middle of Portland, Oregon's *Ground Kontrol* arcade.[233] The video begins with Daedelus challenging a fellow

[231] Butler, 101.

[232] Ibid, 3.

[233] Intothewoods.tv, "Far From Home #9 – Daedelus Live Arcade Set," uploaded April 18, 2012, accessed March 30, 2016, https://vimeo.com/40608297.

beatmaker to a game of *Street Fighter 2,* followed by a montage of clicking and clacking button presses that trigger short bursts and choppy audio samples from the machine. Surrounded by the flashing lights, bleeps, and blips of vintage game consoles, the gestural excess of these two button-pushers transitions seamlessly into Daedelus' musical performance (fig. 3.7).

Figure 3.7: Daedelus performs at Ground Kontrol arcade, Portland, OR (2012) (Source: Intothewoods.tv, "Far From Home #9 – Daedelus Live Arcade Set," uploaded April 18, 2012, accessed March 30, 2016, https://vimeo.com/40608297).

As the camera shifts focus from the game consoles to the musician standing in the middle of the arcade, the visual frame immediately foregrounds a technical setup comprising a laptop and two Monome controllers. The "brain" of the operation consists of a *Max* software patch called MLRv, which allows Daedelus to control simultaneously the playback and fine-tuned editing of musical parameters in multiple audio samples. The GUI consists of eight horizontal rows, each containing a sample, with options to adjust volume, playback speed, and pitch just below each row (fig. 3.8). Using the Monomes as controllers for the MLRv software, Daedelus then physically manipulates the rows of audio in various ways. The 256-button Monome serves as the primary control mechanism, mirroring the layout of MLRv by dividing the 256-button grid into sixteen rows. The rows then spatially fragment the corresponding audio sample into sixteen parts, allowing the musician to playback specific moments in the sample by pushing the buttons within the horizontal row. The audio waveform in the software literally becomes externalized in the hardware, and the "rules" of MLRv become playable (fig. 3.9).

Figure 3.8: MLRv *Max* patch (2011).

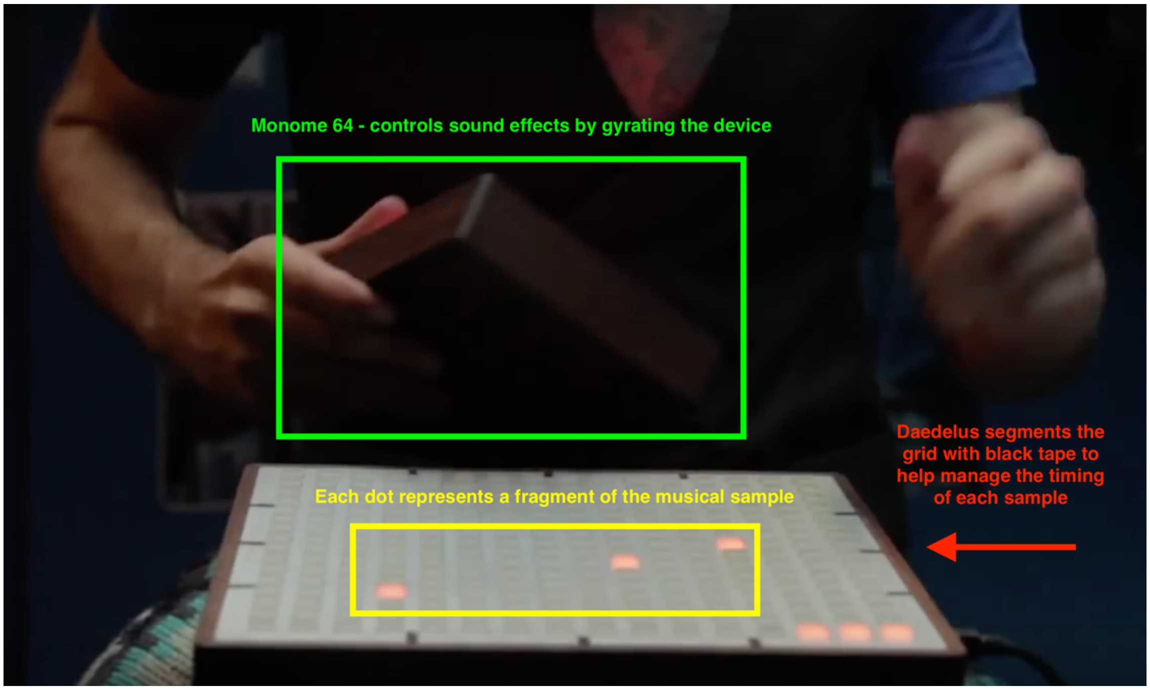


Figure 3.9: Monome 256 grid overview during Daedelus' performance (Source: Intothewoods.tv, "Far From Home #9 – Daedelus Live Arcade Set," uploaded April 18, 2012, accessed March 30, 2016, https://vimeo.com/40608297).

Daedelus' performance mannerisms further highlight the gestural excess witnessed during the gameplay depicted at the beginning of the video. The Monome is angled upward, away from the performer and towards the audience, and the laptop screen is out of sight, highlighting the physical interaction between the musician and the hardware device. Every button press by the performer is accented by a rapid withdrawal of his hand from the interface, spatially exaggerating the spectral morphologies of the sounds being controlled. While the 256-button Monome remains stationary, Daedelus twists and contorts the smaller 64-button Monome, controlling audio effects that are mapped to the device's accelerometer (the same sensor used in mobile phone technology). Rather than simply "pressing play" and letting the computer do all the work, these moments of gestural excess—combined with the abstract and minimal design of the hardware device—allow the viewer to focus visually and aurally on the musical patterns as they are chopped, stuttered, and looped by Daedelus in real-time.

The video brings to the fore key elements of the shifting nature of human-computer interaction detailed throughout this chapter. First, the virtuosic performance practices employed by Daedelus using grid-based controllers highlight the ways in which new media remediate the technical practices embedded within previous technologies (in this case, the controller remediates the turntable). Second, using *hardware* in performance to foreground the processes at play within music *software* demonstrates an increasing balance between object- and process-oriented perspectives on performance. Finally, and perhaps most obviously, the explicit alignment throughout the video of musical culture with gaming culture exemplifies the shifting media genealogies in hip-hop and electronic dance music culture outlined at the opening of this chapter. Not only is Daedelus remixing musical *content,* but he is also playing with the *forms* of relation between musical hardware (bodies, technologies) and software (algorithmic processes, rules).

As mentioned previously in the context of game controllers, the process of developing *embodied* instrumental technique with electronic music controllers consists of two steps: internalizing the affordances and constraints of the music software, and externalizing those design mechanics in the hardware. By foregrounding the hardware over the software, the complexities of the software processes can be channeled through a material device, and the feeling of non-mediation and "direct manipulation" results. As electronic musician Ander claims, the "most important thing I wanted to do was to get rid of the screen. I don't want to have a laptop on stage." Like many other producers and DJs, Ander uses a controller because it "gives you much more direct access to the music as well as to the audience," the minimal and abstract mapping capabilities offering "something which is easy to look at, where you can get a lot of information in a short time." Similar to Daedelus' use of the Monome to externalize and make visible the functions of the MLRv *Max* patch, Ander appreciates the ways in which controllers focus on "what's happening" with the perceived presence and materiality of *sound* rather than the algorithmic processes of *software*. Or, put another way, sound itself becomes a material manifestation of the algorithmic processes embedded within software. In this way, sound cannot be understood as separate from the media environment within which it is itself produced.

Similar to the ways in which the splicing of magnetic tape seemed to allow Schaeffer and others "direct" access to sound itself, the design of button-based grid controllers such as Ableton's *Push* encourages the impression of direct sonic manipulation. In covering the entire surface of the device with buttons—relegating knobs, sliders, and LCD screens to the margins of the interface—the *Push* controller defers the musician's attention to the pads themselves, thus foregrounding the *performance* (process-oriented) aspects of button-pushing and finger drumming over the *manipulation* (object-oriented) aspects of sound tweaking and fine-tuning. Redirecting the musicians' focus from the software to the hardware mirrors what can be understood as a shift from the "static" screen to the "liveness" of the broader performance environment. Decap, for example, equates the direct manipulation of the grid with the dynamism of the crowd itself: "When performing live I literally never need to look at my laptop screen. All of my attention is focused on the *Push*, and the energy of the crowd."[234]

Further, the use of sixty-four buttons (rather than the previous standard of sixteen) allows the grid itself to structure and constrain the affordances of the software, focusing the musicians'

---

[234] "Decap and Brady Watt on Push & working with Ski Beatz," *Ableton*, April 16, 2014, accessed April 18, 2016, https://www.ableton.com/en/blog/decap-and-brady-watt-push-working-ski-beatz/.

attention on the performative aspects of rhythmic and melodic sequencing. While sequencing drum patterns in real-time, for example, the Push uses color to segment the grid into two separate sections: one containing a sixteen pad grid to select and demo sounds, and another for sequencing these sounds along a rhythmic grid (fig. 3.10). This serves as another example of the ways in which abstractions of interface elements—whether in the form of colored grids in the case of music, or shapes and symbols in the case of game controllers—function in focusing the attention of the user on specific aspects of the creative experience. Despite the maximal options afforded by the *Push*, deferring the musicians' focus to a single element of the interface helps consolidate creative direction, and reduce the potentially debilitating effects of the "digital maximalist" mindset discussed in chapter one.

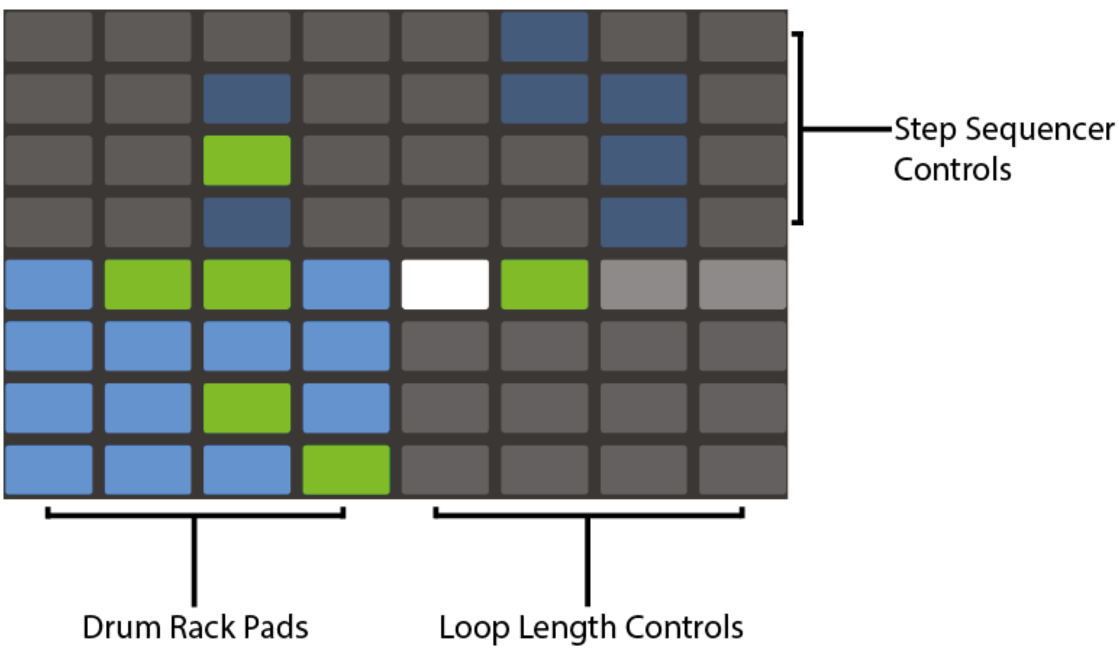


Figure 3.10: Ableton *Push* version 2 pad layout during drum sequencing. The grid is segmented into four 4x4 sections, each corresponding to specific functions. (Source: Ableton, "Using Push 2," Ableton reference manual version 9, accessed April 27, 2016, https://www.ableton.com/en/manual/using-push-2/).

Ultimately, both video game and digital music controllers make tangible the design affordances and constraints of the software being controlled. For gamers, the process of abstracting video game mechanics into the letters and shapes of controllers allows players to embody the rules of games, and therefore develop the skills required to succeed in gameplay. For musicians, the process of externalizing the mechanics of music software programs allows performers to convey "liveness" to their audiences, and therefore engage with both listeners and technology on a more dynamic level. By bringing together case studies in music and gaming, I have suggested a play-oriented model of HCI that recognizes the interconnections between hardware objects and software processes; design and use; play and performance.

Failure as Evidence of Liveness

Controllerism represents a single solution to a perennial question in digital art: how to physically interact with and manipulate creative affordances embedded in screens. The unending development of hardware for engaging with music software has rightly been criticized as an unsustainable model that runs on the desire for commercial profit—a model that is paralleled in the games industry. However, the fact that users continue to experiment with controllers, constantly challenging themselves to learn new forms of embodied interaction with their tools, highlights another important value in the experience of contemporary music and games: failure.

The necessity of failure is obvious in the case of gaming, a medium that teaches players to face death virtually over and over again. It is through the unending process of death and resurrection that the player learns from their mistakes in order to develop the skills necessary to "beat the game." Recently, the proliferation of "controllers" in media production and performance has allowed the built-in possibility of failure and imperfection to bleed into the realm of digital music. DJ Tobias Van Veen sees musical controllerism as a way of enhancing the "human" element of performance, as he claims: "Without risk of fucking-up, there is no need for the human… [but] Controllerism offers possibilities here as a way forward, by which I mean controllerism also entertains virtuosity…" Composer Kim Cascone describes failure as "a prominent aesthetic in many of the arts… reminding us that our control of technology is an illusion, and revealing digital tools to be only as perfect, precise, and efficient as the humans who build them."[235] Rather than praising the agency and virtuosity of the human over technology, "liveness" is evidenced instead in the potential for failure inherent to the process of navigating new relationships with technology.

Failure contradicts prevailing ideologies of innovation and progress inherent to design and technology industries. Each year, Apple releases swaths of computing devices, promising to make the lives of consumers better through "user-friendly" designs that are easy to navigate and seemingly fail proof. Likewise, web designers and user-experience professionals adhere to the "don't make me think" attitude, in which familiar models of interaction are borrowed from existing media to prevent the user from being cognitively or physically challenged in any way.[236] Here, "digital" tools stored in the "cloud" are marketed as catch-all solutions to the problems that exist in our material, "analog" world. Friedrich Kittler once said that "there is no software," reflecting on the ways in which automated systems are designed in such ways that they erase the mechanisms through which they work. Yet, as I have discussed throughout this chapter, both the historical relationships between media forms and the practical techniques used to navigate emerging technologies exist in a dialectical feedback loop. Through play, musical processes become materialized. It is more fitting, then, to say that there is no software without hardware.

---

[235] Kim Cascone, "The Aesthetics of Failure: 'Post-Digital' Tendencies in Contemporary Computer Music," *Computer Music Journal* 24, no. 4 (2000): 13.

[236] Steve Krug, *Don't Make Me Think, Revisited: A Common Sense Approach to Web and Mobile Usability* (San Francisco, CA: New Riders, 2014).

In exposing the potential for failure at the root of all forms of mediation, controllerism represents a single instance of a twenty-first century digital culture in the process of resisting the perennial narrative of technological process. Similar to parallel movements in interactive media—net.art, indie video games, glitch aesthetics—controllerism embraces vulnerability as a prevailing ethic of human-computer interaction. In each case, the imperfections of both the individual operator and the software itself become evidence of "liveness." Technological change, in this context, is not simply about developing new, shiny "digital" objects, but also playfully experimenting with the embodied, "analog" processes ever-present in music and media production. In an era of increased technological control, dominated by proprietary software, global surveillance systems, and the ubiquity of "smart" media, these technologies of play remind us that music, like many of the games we play, consists of rules that are designed to be broken.

Chapter 4

Ubiquitous Production: Making Music with the iPhone

Music-making is often directly linked to sound production. Conventional wisdom assumes that music is made when a sounding note leaves the instrument that produced it. Even musicians who depend heavily on technology tend to think this. Until now, this dissertation has dealt with cultural communities that distinguish themselves *as* musicians precisely by the sounds that they produce with technology. In the maximalist production setups of Flying Lotus and Rustie (chapter one), software such as *Live* is employed in order to create music in the style of electronic dance music and hip-hop. In the minimalist production setups of *Max* users (chapter two), custom musical instruments and processes are created in order to experiment more fully with the parameters of sound production. In controllerist performances (chapter three), the physical and visceral interactions of “live” sound production are valued as fundamental elements of what constitutes music. While the use of technology amongst musicians in each case study has introduced questions about whether or not digital music can have the same value as, say, music made by a classically-trained violinist, music-making is perceived, in any case, to occur at the physical or virtual point of sound production.

Contrastingly, music-making on devices such as smartphones foregrounds not the actual sounds produced by the instruments, but rather the physical actions and seemingly invisible design affordances that guide and control the mundane forms of production in the everyday use of the devices. In contrast to controllerists who use hardware controllers to distinguish themselves from commonplace users of technology, mobile media users celebrate the disintegrating distinction between expert “producers” and non-expert “users” facilitated by devices such as the iPhone. The line between creative production and media consumption is further blurred by microtransactional affordances such as the *App Store*, an online shop in which users can purchase both new apps and add-on content for existing apps on the iPhone. Since the case of the iPhone illustrates that music is not perceived to occur simply at the point of sound production, we are left with two key questions: what is the nature of music-making in the context of mobile media, and how can we understand the nature of music-making when sound production (and the device user's expertise thereof) is not perceived to be the core element of music?

In this chapter, I examine the Apple iPhone as a device whose "user-friendly" design and seemingly intuitive touch-screen mechanics equalize the skill levels needed for both everyday productivity tasks and creative music production. I suggest that music-making in the context of mobile media comprises acts of both production (through the physical gestures and design affordances of music-making apps themselves) and consumption (through the generative structure of social media sharing, as well as add-on content that can be purchased separately from the app). Specifically, I analyze the intuitive iPhone and app design affordances that allow the adaptation of the device into mundane, everyday interactions. These affordances include direct manipulation, touch-screen control gestures, elementary physics in app design, and multitasking capabilities. By analyzing the ways in which mobile media software and app design "democratizes," or facilitates non-expert music production practices, it is possible to understand digital music as comprising not only *material* technologies (instruments, controllers) and performance spaces (dance clubs, concert halls), but a process-oriented *experience* that aligns with consumption practices inherent to capitalism in the early twenty-first century.[237] Before

[237] As Timothy Taylor (2007) notes, the term "democratization" was used by player piano advertisers in the early twentieth century, and has been widely discussed in music and technology literature. Following

addressing the design affordances themselves, I will contextualize the discussion by outlining the paradigm shift from an emphasis of creative production based on specialized skill, to a form of creative production that also embraces intuitive, non-expert knowledge. This shift can be detailed in twentieth century developments in music technology, as well as the philosophy of "personal computing" in the 1980s—the design and marketing doctrine that eventually gave rise to the invention of the iPhone.

Democratization in Music Technology and Computing

With the rise of sound reproduction technologies throughout the twentieth century, music-making became increasingly available and accessible to a broader demographic of non-experts.[238] Timothy Taylor discusses the ways in which advertisers promoted the player-piano as a technology that fostered a democratization of both ability (in that it required no specialized skills to operate) and availability (in that it was more affordable than a grand piano).[239] By convincing consumers that the player piano allowed them a level of "freedom from technique," says Taylor, player-piano advertisers marked "the beginning of the transformation of the musical

---

Taylor, I use the term throughout this chapter both as an accepted vernacular concept, and an ideological tool used by software companies for the purpose of advertising. Timothy Taylor, "The Commodification of Music at the Dawn of the Era of 'Mechanical Music'," *Ethnomusicology* 51, no. 2 (2007): 281-305. See also Lisa Gitelman, "Media, Materiality, and the Measure of the Digital: Or, the Case of Sheet Music and the Problem of Piano Rolls," in *Memory Bytes: History, Technology, and Digital Culture*, edited by Lauren Rabinovitz and Abraham Geil, 199-217 (Durham: Duke University Press, 2004); Paul Théberge, *Any Sound You Can Imagine: Making Music/Consuming Technology* (Middletown, CT: Wesleyan University Press, 1997); Craig H. Roell, *The Piano in America, 1890-1940* (Chapel Hill: University of North Carolina Press, 1989).

[238] For a broad overview, see Timothy D. Taylor, Mark Katz, and Tony Grajeda (eds.), *Music, Sound, and Technology in America: A Documentary History of Early Phonograph, Cinema, and Radio* (Durham: Duke University Press, 2012).

[239] Taylor, "The Commodification of Music," 289.

experience into an object of consumption."[240] Mark Katz considers how the phonograph similarly facilitated "amateur" music-making during this time.[241] Kiri Miller, among others, have examined production practices in the context of "amateur-to-amateur" Internet platforms such as YouTube. In each case, communities of non-experts formed as a result of increased financial accessibility and usability in media and technology design.

Similarly, in the history of twentieth century computing, we can observe a gradual shift from a paradigm that we might call "specialized computing," in which technical skill is valued amongst particular communities of experts, to a newer paradigm of "personal computing." Specialized computing emerged in the midst of the technological competition occurring between nations during, and immediately following World War II.[242] The world's first electronic computers were extremely expensive, required specialized technical knowledge to operate on even a basic level, and often could only be found in specialized venues such as university research labs.[243] For example, the first commercially available electronic computer, the Ferranti

---

[240] Ibid, 291.

[241] Mark Katz, "The Amateur in the Age of Mechanical Music," in *The Oxford Handbook of Sound Studies,* edited by Trevor Pinch and Karin Bijsterveld, 459-479 (New York: Oxford University Press, 2012); *Capturing Sound: How Technology Has Changed Music* (Berkeley and Los Angeles: University of California Press, 2004).

[242] Nathan Ensmenger, *The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise* (Cambridge, MA: The MIT Press, 2012); J.C.R. Licklider, "Man-Computer Symbiosis," *IRE Transactions on Human Factors in Electronics* 1 (1960): 4-11; Norbert Wiener, "Men, Machines, and the World About," in *Medicine and Science,* edited by I. Galderston, 13-28 (New York: International Universities Press, 1954); Alan Turing, "Computing Machinery and Human Intelligence," *Mind: A Quarterly Review of Psychology and Philosophy* 59, no. 236 (1950): 433-60.

[243] Paul E. Ceruzzi, "The First Computers, 1935-1945," in *Computing: A Concise History,* 23-48 (Cambridge, MA: The MIT Press, 2012); James Gleick, *The Information: A History, A Theory, A Flood* (New York: Pantheon Books, 2011); Rich Arzoomanian, "A Complete History of Mainframe Computing," *Tom's Hardware,* June 26, 2009, accessed April 30, 2016, http://www.tomshardware.com/picturestory/508-3-mainframe-computer-history.html.

Mark 1 (1951), sold only ten units in total, for £83,000 each, or $122,000 is today's terms (fig. 4.1).

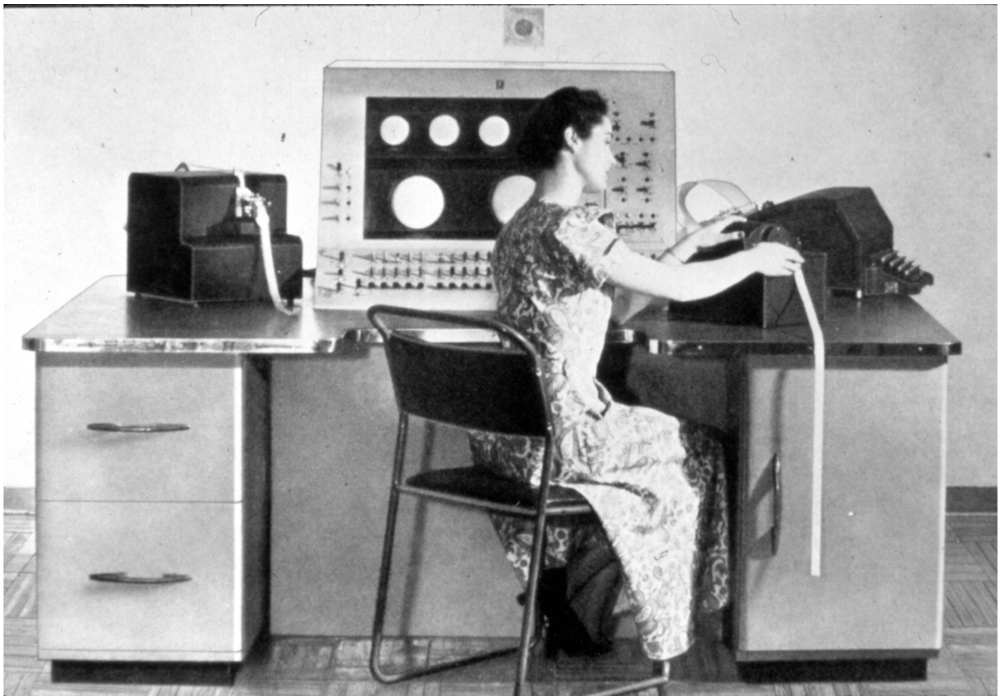


Figure 4.1: Ferranti Mark 1 (1951), photo circa 1953
(Source: Computer History Museum, September 8, 2005, accessed April 30, 2016, http://archive.computerhistory.org/projects/chess/related_materials/still-image/bak/).

Personal computing emerged following the development of the microprocessor in the early 1970s. In response to the specialized nature of electronic computers following World War II, the underlying premise of the personal computing philosophy is the belief that the affordances of technology should be democratized—that is, technologies should be commercially available at an affordable price, and usable by the broadest demographic possible.[244] In contrast to computers such as the Ferranti Mark 1, companies such as Xerox and IBM aimed to create smaller, more

[244] Roy A. Allan, *A History of the Personal Computer: The People and the Technology* (Allan Publications, 2001), accessed April 30, 2016, https://archive.org/details/A_History_of_the_Personal_Computer; Steven Levy, *Hackers: Heroes of the Computer Revolution* (Garden City, NY: Doubleday, 1984).

affordable computers that could be taken outside of the computer "lab" and into users' everyday lives (fig. 4.2). It is this line of thinking that continues in contemporary initiatives such as the "Hour of Code" campaign, discussed in chapter two.

Figure 4.2: Xerox Alto I CPU with monitor, mouse, keyboard and 5-key chording keyset (1973) (Source: Computer History Museum, n.d., accessed April 30, 2016, http://www.computerhistory.org/revolution/input-output/14/347).

Apple contributed greatly to spark the personal computer revolution on a mass commercial scale in the late 1970s. From the mid to late 1970s, Apple co-founder and designer Steve Wozniak regularly attended meetings of the "Homebrew Computer Club," an informal group of Silicon Valley computer enthusiasts. Wozniak was so inspired by the group that he began designing what would become Apple's first personal computer after just one meeting with the group.[245] Instead of designing complex interfaces only accessible to the expert computer user, Apple asked the question, what are humans *naturally* good at, and how can we embed those affordances in software and hardware products? In order to appeal to the broadest demographic of computer users, the company established a design philosophy centered on three core tenets: the marketing of "everyday" creativity, an emphasis on visual aesthetics over computational power, and a minimalist design aesthetic.

By employing a marketing strategy that focuses on the "everyday" creativity afforded by Apple products, Apple continues to represent the apex of "user-friendly" populism in personal computing. As the slogan from the original Macintosh (1984) commercials attests, Apple's design philosophy has been eternally invested in the democratizing goal of creating a "computer for the rest of us" (fig. 4.3).[246] This goal has inspired non-expert artists and musicians to embrace the Mac OS X in their creative work, as an easier to use counterpart to Microsoft's Windows operating system. With the introduction of *iMovie* video editing software in 1984, Apple products were promoted to be "the next big thing" in personal computing, turning their users into "both the director and the producer"[247]—a marketing rhetoric that promoted what Jean Burgess calls "vernacular" creativity.[248] According to Burgess, the technical design of the original Macintosh and the *iLife* suite, as well as the advertising campaign surrounding them, "constructed its users as effortlessly creative rather than extraordinarily so."[249] In contrast to

---

245 Steve Wozniak, *iWoz* (New York: W.W. Norton & Company, 2006), 150.

246 "1984 Apple's Macintosh Commercial (HD)," *Mac History,* uploaded February 1, 2012, accessed April 30, 2016, https://www.youtube.com/watch?v=VtvjbmoDx-I.

247 Jean Burgess, "Vernacular Creativity and New Media," PhD dissertation U of Queensland 2007.

248 "Steve Jobs introduces iMovie & iMac DV - Apple Special Event (1999)," EverySteveJobsVideo, uploaded December 21, 2012, accessed March 9, 2016, https://youtu.be/ngW5qCBRwxk.

249 Jean Burgess, "The iPhone Moment, the Apple Brand, and the Creative Consumer: From 'Hackability and Usability' to Cultural Generativity," in *Studying Mobile Media: Cultural Technologies, Mobile*

specialized technical skill, vernacular creativity encouraged a playful remixing of content that merged amateur creativity, technology, and everyday experience.

Figure 4.3: Advertisement – Apple Macintosh (1984) (Source: prattcomd520, "User Centered Design," accessed April 30, 2016, http://dosislas.org/pratt/spring16/comd520/user-centered-design/).

In addition to promoting the democratizing power of vernacular creativity in Apple products, the company also redirected the focus of computing from computational power and technical skill to visual aesthetics. The original iMac, introduced in 1999, embodied this shift in its focus on the color and aesthetics of the device above and beyond what Steve Jobs called the technical "mumbo jumbo" of the hardware.[250] Available in five different colors, these computers were promoted as creative platforms for the user to express themselves, closely aligning the

Communication, and the iPhone, edited by Larissa Hjorth, Jean Burgess and Ingrid Richardson (New York: Routledge, 2012), 37-8.

[250] "Steve Jobs introduces rainbow iMacs & Power Mac G3 - Macworld SF (1999)," EverySteveJobsVideo, uploaded December 21, 2013, accessed March 9, 2016, https://youtu.be/NuCYHrSig94.

company's marketing and design schemes (fig. 4.4). Since this moment, for many users, the intimate processes of interaction have become more important than the goal-oriented tasks of computing. These users respond to Apple's goal of increasingly intimate relationships between humans and their computers. This message is apparent in many of Jobs' product launch presentations, as everything from the iMac to the iPod is described as "beautiful," "gorgeous," and even "delicious."[251]

Figure 4.4: Apple's "Yum" ad campaign (1999) (Source: Dylan Tweny, "Vintage Posters Highlight a Century of Innovation," ***Wired,*** **February 3, 2011, accessed April 30, 2016, http://www.wired.com/2011/02/vintage-posters/)**

The emphasis on visual aesthetics is most noticeable in Apple's signature minimalist design aesthetic. This minimalist sentiment is summarized by the belief of Apple's Chief Design Officer, Sir Jonathan Ive, in the phrase "good design is as little design as possible."[252] The 2015

[251] Ibid. See also, "Incredible amazing awesome Apple," Kanal von justanotherguy84, uploaded September 15, 2009, accessed March 22, 2016, https://www.youtube.com/watch?v=Nx7v815bYUw.

[252] "Making the iPhone: The Design Tao of Jony Ive," *Bloomberg,* September 19, 2015, accessed October 5, 2015, http://www.bloomberg.com/news/videos/b/41d8c560-9bc6-479c-a3b2-6dfd2b038aca.

Macbook laptop, for example, includes a streamlined, rounded rectangle casing that provides the guiding design framework for everything contained within it, from the keyboard enclosure and mouse trackpad to the rounded square software icons and screen notifications. Interface elements on the hardware consist of three simple elements: a single peripheral output on the side of the machine, a flat trackpad, and a black and white rectangular keyboard (fig. 4.5). Cut from a single piece of aluminum, the unibody exterior of the device accentuates a soft, clean surface, enhancing the minimalist control surfaces present on the computer. In reducing the number of *hardware* elements on the device, the Macbook focuses the users' attention on the *software* processes taking place on screen.

Figure 4.5: Apple Macbook (2015) (Source: Apple, "Macbook," n.d., accessed April 30, 2016, http://www.apple.com/macbook/. Annotations added)

Apple's minimalist design aesthetic has prospered with the emergence of mobile media. The history of personal computing—as with many technological innovations of the twentieth and twenty-first centuries—is a story of increasing miniaturization, portability, and accessibility. As devices get smaller, the body learns to confine its physical gestures so as not to exceed the confines of the screen. Perhaps more than any other device, the iPhone foregrounds the *aesthetics* of human-computer interaction by solidifying Apple's minimalist design philosophy. This is most noticeable in the control mechanisms of the device. While most smartphones at the time contained button-based keyboards that took up half of the phone's front panel space, Apple removed the physical keyboard entirely in favor of direct touch-screen control (fig. 4.6). According to Ive, this was Apple's primary design strategy in constructing the iPhone, as it would encourage users to defer all mental and physical attention to the tactile space of the touch screen, rather than the limited amount of buttons and switches outside of it.[253] In focusing the user's attention on the visceral connection between the human finger and the smooth glass screen, Apple further promotes vernacular creativity, imbuing the sense of a direct, unmediated connection between the device and its user.

[253] *Objectified*, directed by Gary Hustwit, Plexi Productions, 2009.

Figure 4.6: Apple iPhone (2007) (Source: "Retro Phone Review: the original Apple iPhone," ***Tech Radar,*** **July 20, 2015, accessed April 30, 2016, http://www.techradar.com/news/phone-and-communications/mobile-phones/retro-phone-review-the-original-apple-iphone-1299387).**

Together, developments in music technology and personal computing marked a cultural shift in the perceived role of technology among "everyday" users. Whereas traditional forms of music-making and computing valued the development of specialized, technical skills, the new paradigms of non-expert music-making and personal computing valued the proliferation of non-specialized creative production. In doing so, these developments have continued to encourage shifting understandings of the role of music in everyday life. Mobile music production with iPhone apps serves as the most recent development in both music software, technology, and personal computing. In this context, we might ask: how might the design affordances of musical apps affect the processes of "everyday" music-making and computing? Further, what happens to conceptions of music if entire songs can be composed and distributed with nothing but a mobile phone and a $1 piece of software? Has the democratization of technology made music available to everybody, or has it diluted music's claim as a specialized cultural practice?

Apps and the Design of Everyday Interaction

The advent of mobile media represents the apex of personal computing values, initiating a democratization of both technical ability and affordability. Specifically, the iPhone embeds the tacit and elementary bodily comportments of computer users into the physical design of the device itself, as well as the software apps stored on it. In this section of the chapter, I consider four of the most common design affordances in mobile media design, including the employment of simple touch-screen control gestures, designing elementary virtual physics into apps, the effect of direct manipulation, and the multitasking capabilities inherent to iPhone usage. These design affordances exploit forms of physical and cognitive interaction that humans are naturally familiar with and, as such, require no specialized training or technical skills. In doing so, the iPhone blurs the boundary between creative production and the mundane tasks of everyday life. As the personal computer literally and metaphorically moves from the "lab" to the "home" to the "pocket," the hardware itself seems to gradually disappear, the iPhone ingraining itself into the everyday interactions of its user.

The employment of simple touch-screen control gestures is the most unique design affordance introduced by the iPhone. Most notably, the touch-screen aligns the practice of creative production with the naïve gestures of the human body, rather than the specialized technique of a musical instrument. Apple has gone so far as to copyright specific physical actions, thus standardizing touch-screen gestures such as tapping, dragging, flicking, swiping, double-tapping, pinching, touching and holding, and shaking. In order to establish these gestures as fundamental mechanics in a range of iPhone apps, Apple regularly publishes a set of "iOS Human Interface Guidelines" that instructs designers on how to employ specific actions for specific purposes.[254] For example, "Flick" is to be used for scrolling quickly through a set of content, "Shake" will undo or redo an action, and "Pinch" is used to zoom in or out. Designers are specifically discouraged from introducing new gestures into the app unless it is a video game (fig. 4.7). As Apple claims, "In games and other immersive apps, custom gestures can be a fun part of the experience. But in apps that help people do things that are important to them, it's best to use standard gestures because people don't have to make an effort to discover them or remember them." These guidelines highlight the proprietary nature of iOS as a dominant operating system reaching into not only users' everyday lives but also gestures. Apple's use of a limited set of everyday gestural control also exhibits a radical departure from traditional "musical" techniques, skills, and forms of interaction.

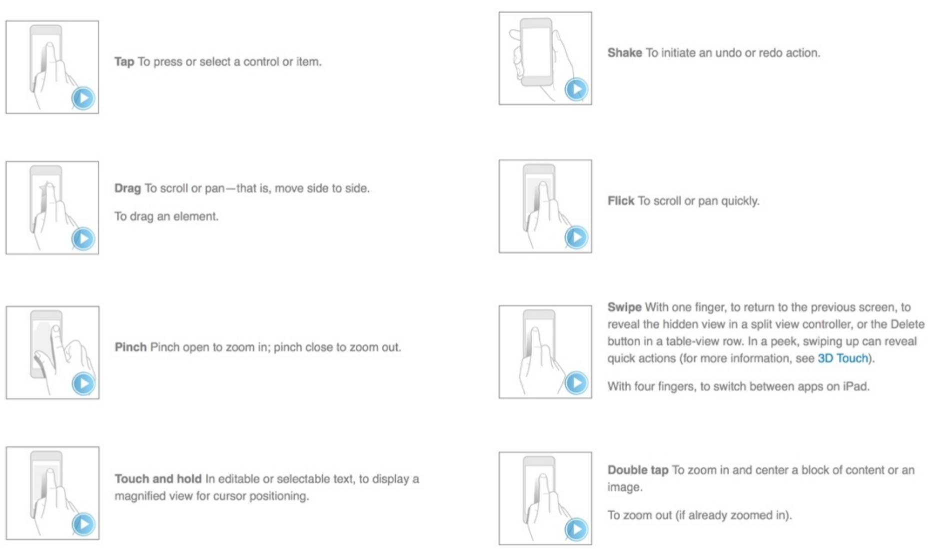


Figure 4.7: Apple iOS Standard Gestures (Source: Apple, "Interactivity and Feedback," in *iOS Human Interface Guidelines,* updated March 21, 2016, accessed April 30, 2016, https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/InteractivityInput.html).

[254] "Interactivity and Feedback," in *iOS Human Interface Guidelines,* last updated November 5, 2015, accessed December 30, 2015, https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/InteractivityInput.html.

Music production app developers have been particularly successful in embracing touch-screen gestures to facilitate more intuitive composition with mobile devices. Music production app developers such as Native Instruments' *Traktor DJ* promises its users the ability to "touch the groove" by using tapping and pinching to edit directly audio waveforms (fig. 4.8a). More abstract apps such as *Patatap* fragment the iPhone screen into twenty-four blocks that trigger audio samples when tapped (fig. 4.8b). Propellerhead's *Figure* employs a trackpad-style control mechanism that allows the musician to create rhythmic variety in their musical patterns by swiping along an X-Y grid (fig. 4.8c). Others employ touch-screen mechanics for pedagogical purposes. The *Clapping Music* app, for example, challenges the smartphone user to tap the screen in time with Steve Reich's infamous minimalist composition, "Clapping Music" (fig. 4.8d).[255] In order for users to develop the digital (and musical) literacies required of apps, they must develop a habituation of response through repeated practice. Like many forms of interaction with mobile media, music production apps thus rely on the omnipresence of the iPhone in the everyday routines of the user.

[255] Upon the release of the app in 2015, users were scored based on the tightness of their performance with a pre-recorded track, and they could even upload their high scores for a chance to perform Reich's composition on stage with the London Sinfonietta.

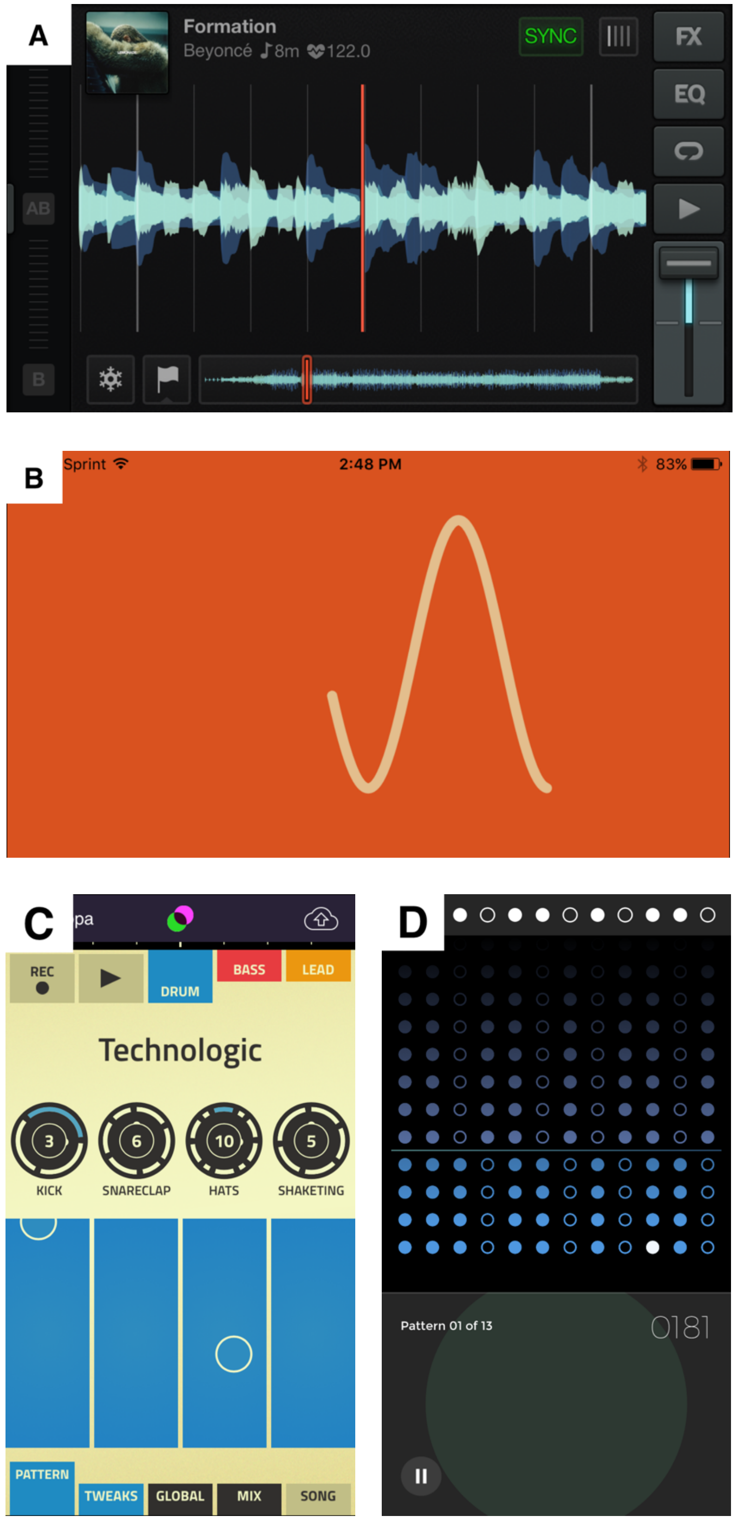


Figure 4.8: Touch gestures in iPhone music apps. (A) Native Instruments, *Traktor DJ* (2013); (B) Jonathan Brandel, *Patatap* (2014); (C) Propellerhead, *Figure* (2012); (D) Touchpress Limited, *Steve Reich's Clapping Music* (2015).

In order to teach users how to incorporate these new control gestures into their physical comportments with mobile media, apps designers incorporate elementary physics—programming scripts that cause *virtual* interface elements to simulate the behavior of *physical* objects—into app user interfaces. This design affordance gives the user the impression that they interact with physical objects, rather than a flat glass screen. In order to be effectively operated, post-WIMP interfaces—that is, interfaces that diverge from the "window, icon, menu, pointing device" paradigm, such as the iPhone—require users to viscerally internalize the affordances of the app in the absence of more traditional computer interfaces such as the mouse and keyboard.[256] This internalization is achieved through the elementary physics imbued by app design, which include sensations such as inertia and springiness amongst interface elements. For example, a user encounters elementary physics in action when the color of an app icon darkens once it is pressed. This effect provides the illusion that the virtual objects present throughout the device (buttons, in this case) have mass.

These "naïve physics" can also extend to our embodied memory of hardware such as computer keyboards that are simulated in the iPhone graphical user interface (GUI), as Ingrid Richardson suggests.[257] This means that users learn to navigate the device both by routinely practicing new sets of physical gestures, and by translating to the smartphone their embodied knowledge developed from interactions with past media forms; pressing buttons on an "analog" telephone, for example. The metaphors and analogies embedded within apps afford users the illusion that the apps are based on intuitive forms of interaction, rather than complex software

[256] Robert J.K. Jacob et al, "Reality-based interaction: a framework for post-WIMP interfaces," in *CHI 2008 Proceedings: Post-WIMP,* Florence, Italy, April 5-10 (2008): 201.

[257] Ingrid Richardson, "Touching the Screen: A Phenomenology of Mobile Gaming and the iPhone" in *Studying Mobile Media*, 144.

code or musical technique. Thus, the user is encouraged to further embody the physical gestures of the iPhone touch-screen in their everyday lives.

Abstracting *physical* gestures into *virtual* touch-screen gestures through elementary physics is particularly crucial for music production apps, which often succeed or fail based on the extent to which they emulate the tactile affordances of physical instruments. Native Instruments' *iMaschine* is a good example of this. Designed to emulate the classic drum machines and samplers used by hip-hop and electronic dance music producers, the primary focus of the interface is a sixteen-pad grid of square buttons used to control sound samples (fig. 4.9). Similar to classic samplers such as the Akai MPC, each pad is assigned an individual sample that can be edited by adjusting length and volume, or adding effects. Extra features of the app, including the ability to construct rhythmic patterns, arrange songs, and set playback parameters, are relegated to the outer margins of the screen. Similar to the design of the iPhone itself, this layout focuses the users' attention on the *process* of "pushing" the buttons on the grid rather than the *objects* simulated.

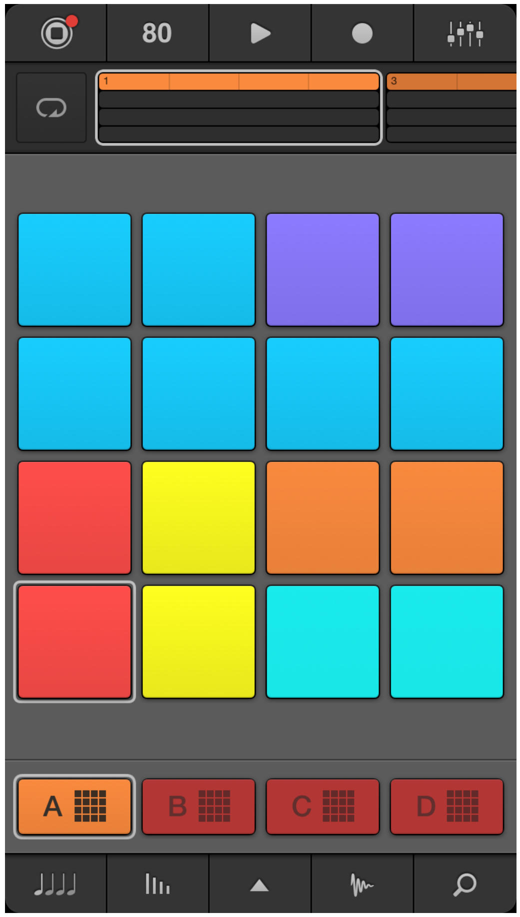


Figure 4.9: Sample pads in *iMaschine*. Similar to the Ableton *Push* controller, described in chapter three, colors are used to organize sample types on the grid.

In order to model previous techniques of beat production, for example, elementary physics in the app are directly mapped to physical gestures required by the “analog” machines being simulated. When the virtual pads on *iMaschine* are “pushed,” they illuminate and display a 3D gradient around their edges, giving the impression of visual depth and weight to GUI elements. Similarly, moving a virtual fader by sliding a finger across the glass screen of the iPhone gesturally emulates the smoothness of adjusting a fader on an “analog” mixer. These synesthetic illusions allow the user to internalize the abstract metaphors and elementary physics on which the app’s core mechanics are based.

Together, touch-screen gestures and elementary app physics encourage the development of another common iPhone design affordance: direct manipulation. In general, direct manipulation relies on the foregrounding of the *material* content over extraneous administrative interface elements.[258] The iPhone gives the user the impression of immediate, direct manipulation of content by combining the physicality of touch-screen gestures with a maximal screen real estate made up of images, audio waveforms, and video clips, rather than GUI elements. Despite the small screen hardware, increased screen resolution facilitates even greater information transmission and detail of content, allowing the user to interact with the maximum amount of interface elements. For example, this feature allows iPhone users to edit images with a similar degree of clarity and precision as using a desktop software program such as Adobe Photoshop. Further, the operating system tends to distribute all content on the same horizontal surface, cementing the idea that content creation and editing is continuous, rather than interrupted. As the typical navigational elements such as "Back" buttons, text input bars, and arrows are minimized, the content itself becomes the interface, and users feel that they are naturally exploring the device rather than being led down a goal-oriented path (fig. 4.10).

[258] Edward Tufte, "iPhone Resolution by Edward Tufte," Edward Tufte, uploaded November 25, 2008, accessed March 25, 2016, https://www.youtube.com/watch?v=YslQ2625TR4.

Figure 4.10: iOS *Photos* app. Content (image) is foregrounded over GUI elements, which are relegated to the borders of the screen.

By removing extraneous interface elements from the screen, iPhone apps also foreground the process-oriented mechanics at work behind the interface, allowing the user to play with the "rules" of the software in various ways. This form of playful interaction, similar to the "locomotor play" of video gaming and music-making described in chapter three, enhances the feeling of direct manipulation in a few ways. First, audio sampling apps often expose the waveform of musical tracks, giving the musician the ability to stretch, loop, or cut-up audio by tapping or swiping the screen. In the "Edit Sample" screen of *iMaschine,* for example, the user can set the start and end points of a sample by sliding their finger across the screen, rather than using the less intuitive WIMP interface to zoom in on a waveform and drag the pointer across the sample (fig. 4.11).

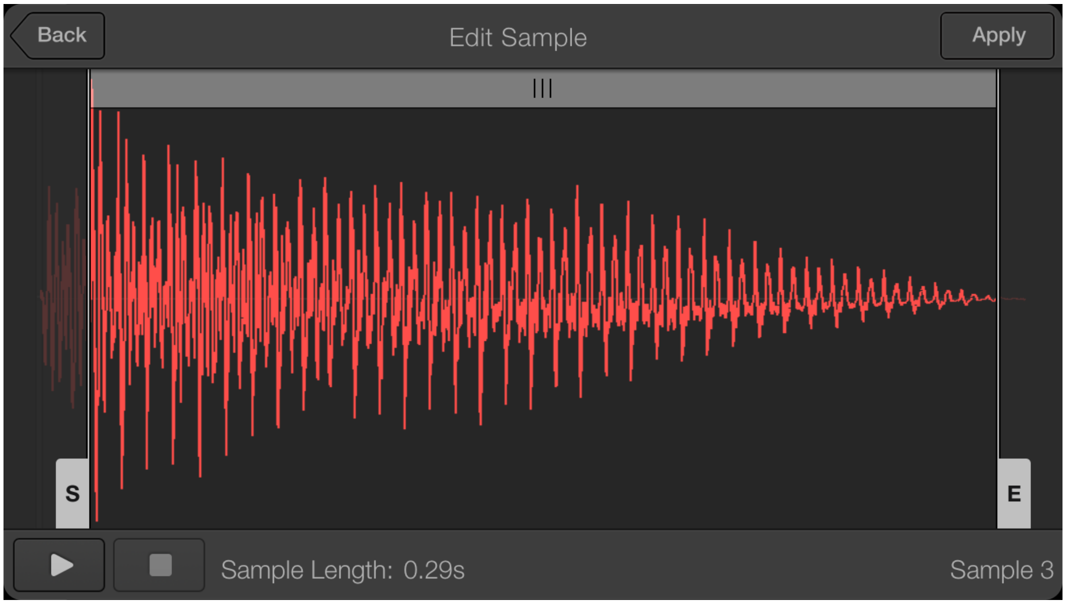


Figure 4.11: *iMaschine* "Edit Sample" screen. Sample length is controlled by sliding start points (marked "S") and end points (marked "E") with a finger.

Second, music apps often abstract more traditional musical instruments into shapes, providing more game-like experiences. Apps like *Musyc* (2013) and *Scape* (2012) apply sets of rules to simple shapes, resulting in complex rhythmic and harmonic relationships as the shapes interact with each other (fig. 4.12). In combining the sampling capabilities of digital tools with the playful systems of video games, iPhone apps provide users with a feeling that they are interacting naturally with a process-oriented *experience,* rather than the clunky *tool* in their hands.[259] Whereas the GUI originated in the realm of labor, with its focus on *function*, Søren Bro Pold and Christian Ulrik Andersen note that "the new app interface clearly has its roots in (digital) culture with an *aesthetic* interface, inspired by games, software culture and cultural

[259] The idea of music and media production as an entire experience, rather than a tool-based endeavor, is common in recent literature on "post-digital" aesthetics. Composer Kim Cascone foreshadowed many of these trends in "The Aesthetics of Failure: Post-Digital Tendencies in Contemporary Computer Music," *Computer Music Journal* 24, no. 4 (2000); See also Christopher Haworth, "Sound Synthesis Procedures as Texts: An Ontological Politics in Electroacoustic and Computer Music," *Computer Music Journal* 39, no. 1 (2015): 41-58.

interfaces in general."[260] In contrast to the critiques that usability in design "dumbs down" the user experience, direct manipulation serves as a condition of possibility for intuitive forms of non-specialist creativity in music production.

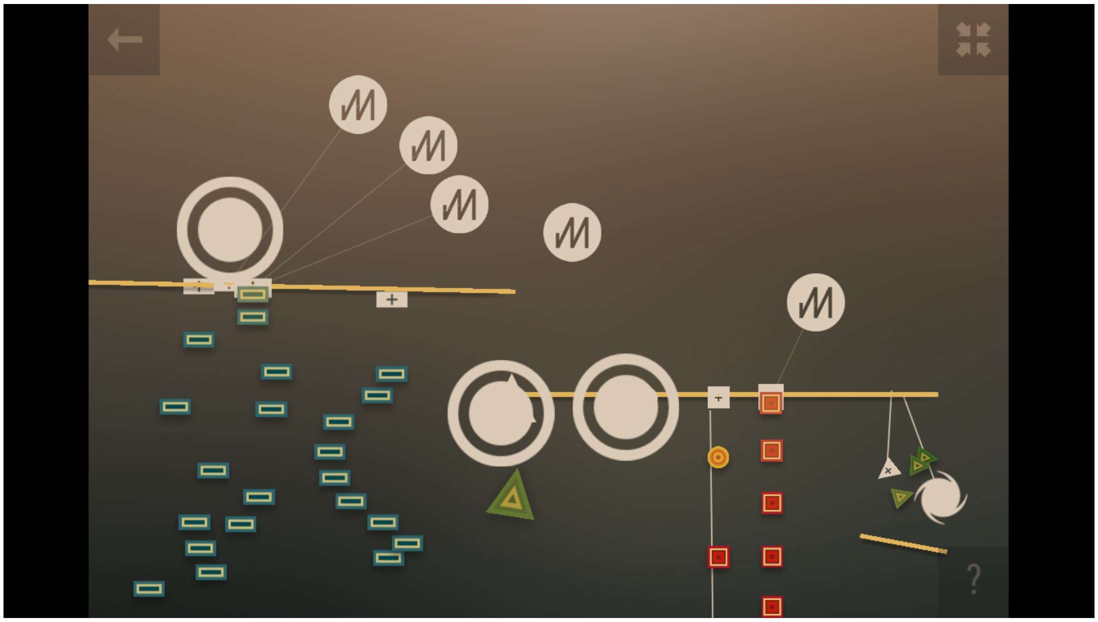


Figure 4.12: Fingerlab's *Musyc* iPhone app (2013). Shapes represent different types of sounds that play back as they collide with each other.

However, direct manipulation also presents a paradox for app designers. Just as the touch-screen presents new possibilities for *gestural* manipulation, the users' focus on the small screen serves to anchor their body by fixing attention on the software, thus preventing a more fine-tuned degree of physical control with the device. After the introduction of smartphones, across most of the world, large crowds of people are formed in which all eyes are glued to smartphone screens as fingers rapidly type away. Aden Evens describes the resulting "fealty of stillness and sameness" in mobile media interactions.[261] For controllerists, this situation

[260] Søren Bro Pold and Christian Ulrik Andersen, "Controlled Consumption Culture: When Digital Culture Becomes Software Business," in *The Imaginary App,* edited by Paul D. Miller and Svitlana Matviyenko (Cambridge, MA: The MIT Press, 2014), 17.

[261] Aden Evens, "Touch in the Abstract," *SubStance* 70, no. 3 (2011): 70.

represents the worst consequence of "usability" in personal computing, as the usability of the iPhone interface reduces mobile music producers to button pushers who are no different from the average mobile media user checking their e-mail or texting their friends. In contrast, iPhone apps embrace the disintegrating distinction between creative and mundane work on the device. This brings me to the final major design affordance of iPhone apps: the ways in which the software allows for creative multitasking.

Due to touch-screen gestures, elementary physics, and direct manipulation affordances embedded within app design are shared across the various activities with which the iPhone user is engaged, multitasking skills are crucial to everyday mobile media usage. The act of composing music on smartphones, for example, is often done in conjunction with a range of other activities, from checking e-mail to scheduling appointments and browsing news feeds. As a result, app designers must take into account what Robert Rosenberg calls the "field composition" of both the musical experience and the material context of the musicians' everyday life.[262] That is, the designer must consider how the affordances being offered to the user are capable of structuring their general field of awareness in everyday life.

On one hand, the smartphone encourages a heightened focus on the device itself, allowing the user to ignore their external environment, as mentioned earlier. It is no coincidence, then, that many of the promotional materials for music production apps portray the musician interacting with the software in public venues that are typically perceived to be difficult to navigate socially. For example, the promotional video for *Figure* depicts a man entering a crowded subway train, eyes fixed on his iPhone and over-the-ear headphones safely sheltering

---

[262] Robert Rosenberg, "The Spatial Experience of Telephone Use," *Environment, Space, Place* 2, no. 2 (2010): 61-75.

him from the noise surrounding him (fig. 4.13).[263] As the video time lapses through the duration of his commute, the man never looks away from the app. The advertisement presents *Figure* as both a distraction from the outside world and a tool for increasing one's focus on the creative process. It is the quintessential image of the always-online individual who is simultaneously completely enmeshed in the social network and absolutely separate from it—a condition that has come to define mobile media usage more broadly.[264]

Figure 4.13: *Figure* promotional video (2012)
(Source: "Figure for iPhone and iPad," *Propellerhead,* uploaded April 4, 2012, accessed April 30, 2016, https://www.youtube.com/watch?v=gLLjRH6GJec).

On the other hand, the user-experience workflows of apps are designed in such a way as to allow for multitasking in response to the demands of life outside of the smartphone. Mechanics of interruptability allow the iPhone to become embedded into users' everyday lives, precisely by encouraging usage in small doses. Initially coined by Jesper Juul to describe the

[263] "Figure for iPhone and iPad," *Propellerhead,* uploaded April 4, 2012, accessed April 30, 2016, https://www.youtube.com/watch?v=gLLjRH6GJec.

[264] Sherry Turkle, *Alone/ Together: Why We Expect More From Technology and Less From Each Other* (New York: Basic Books, 2011).

ways in which play becomes intertwined with everyday routines, interruptability is similarly employed as a mobile design strategy to maximize user productivity while they work with multiple apps simultaneously.[265] The larger screens of laptops facilitate multitasking by segmenting the screen into multiple "windows." Swiping gestures on the trackpad allow the user to switch between active software programs, reorganize windows currently visible on screen, and divide the screen into multiple "desktop" workspaces (fig. 4.14).[266]

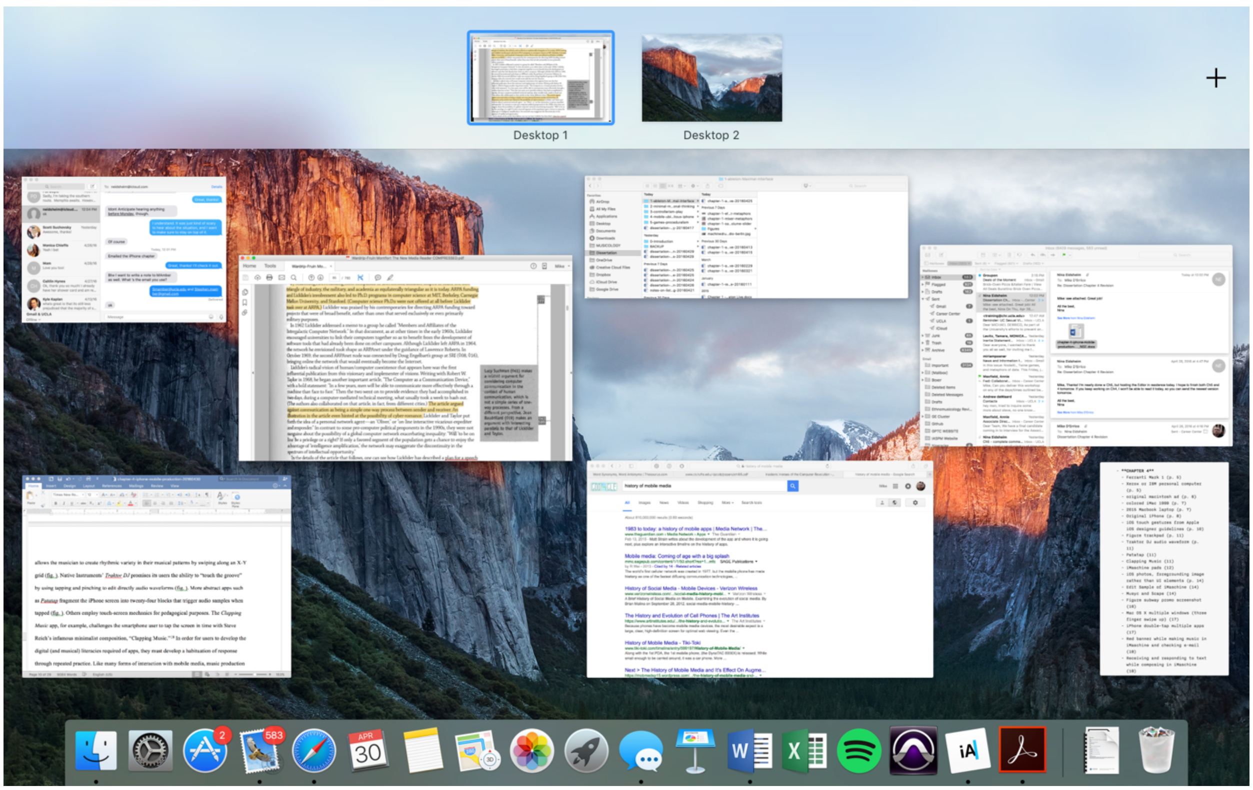


Figure 4.14: Apple Mac OS X "Show Desktop" trackpad gesture reveals active software programs

The iPhone achieves this multitasking workspace by organizing information spatially in the manner of folders, thus maximizing the real estate of the smaller screen by adding multiple

[265] Jesper Juul, *A Casual Revolution: Reinventing Video Games and Their Players* (Cambridge, MA: The MIT Press, 2010), 37.

[266] For more information on Mac OS X trackpad gestures, see Apple, "Use Multi-Touch gestures on your Mac," n.d., accessed April 30, 2016, https://support.apple.com/en-us/HT204895.

layers of depth to the interface.[267] When multiple tasks are performed, the interface collapses into a system of layers, allowing the user to switch between apps with ease and speed. At any moment in the user experience, a double-tap on the home button reveals an accordion-like menu of app "tabs" at the top of the screen that serve as navigational "breadcrumbs" (fig. 4.15). In this context, composing and producing music occurs within the device and interface, and is therefore given no more or less value than texting, checking e-mail, or playing a video game.

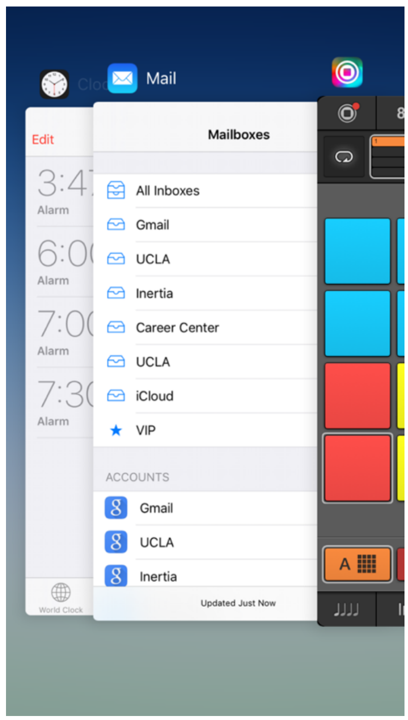


Figure 4.15: iOS "App Switcher" feature, activated by double-tapping the device's "Home" button

By allowing the user to "pause" the music-making process in this way, the iPhone affords a continuous production and *uninterrupted* creative flow amongst the various tasks with which the mobile media user is engaged. Imagine you are making a beat in *iMaschine,* for example. You receive an important e-mail, and a banner notification appears at the top of the screen. As

[267] Apple, "About Multitasking on your iPhone, iPad, and iPod touch," n.d., accessed April 30, 2016, https://support.apple.com/en-us/HT202070.

the music plays, you tap the notification and are immediately taken out of *iMaschine* and into the *Mail* app. At the same time, a red banner affixes itself at the top of the screen, letting you know that an active recording session is taking place in *iMaschine,* and that you are able to return to the app simply by tapping the red banner (fig. 4.16a). Here, despite the somewhat complicated multitasking going on, the use of banners at the periphery of the user's attention serves as a cognitive glue for the user, creating the feeling of uninterrupted, perpetual production. Text message banner notifications are even less disruptive, as the user is able to respond to the text in the notification itself, without having to leave the current app. After receiving the text message (fig. 4.16b), the user swipes down on the banner, and a text box appears. Then, iMaschine darkens and blurs, thus momentarily foregrounding the act of texting over music production (fig. 4.16c). In these examples, the mechanics of interruptability illustrate the convergent nature of production on the iPhone, heightening the musician's feeling that the creative process of music-making is as ubiquitous and mundane as checking their e-mail.

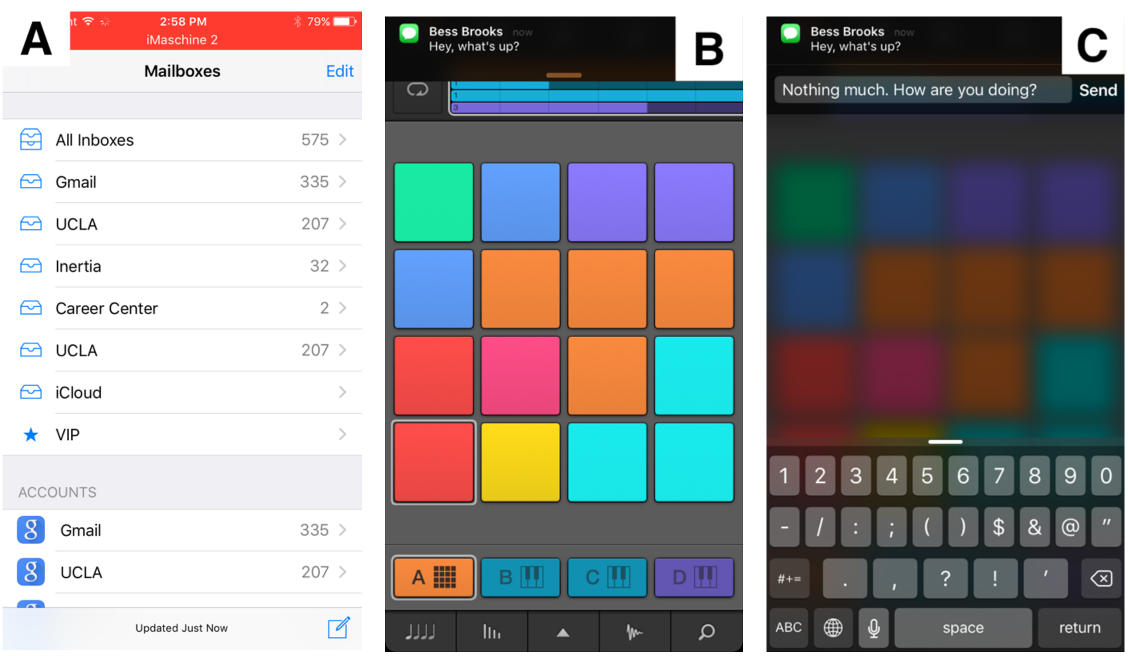


Figure 4.16: Multitasking and interruptability on the iPhone

As I have suggested, the design affordances listed—the employment of simple touch-screen control gestures, designing elementary virtual physics into apps, the effect of direct manipulation, and the multitasking capabilities inherent to iPhone usage—increase the appeal of the iPhone for a broad demographic of non-expert users by foregrounding intuitive forms of interaction over specialized technical skill. In doing so, the previously isolated practices of creative music-making and everyday production tasks are integrated within the immediately accessible and all-encompassing smartphone. Of course, even before the rise of the iPhone, the design of personal computers was already moving in the direction of increased mobility and pervasive use. In the 1991 article, "The Computer for the 21st Century," Mark Weiser foreshadows an emerging age of "ubiquitous computing" in which computers would "weave themselves into the fabric of everyday life until they are indistinguishable from it," thus fading to the background of users' attention.[268] According to Weiser, rather than being a *technological* issue, the desired "invisibility" of the computer is a *psychological* issue related to the development of tacit knowledge. As he states, "whenever people learn something sufficiently well, they cease to be aware of it."[269] The now standard physical actions of swiping and tapping the iPhone screen highlight the tacit corporeal knowledge required of media consumption and production in the age of mobile media, arguably constituting the defining characteristics of the everyday, vernacular creativity that has come to define personal computing.

Mobile app developers have also embraced marketing rhetoric that emphasizes ubiquitous computing as a democratizing force. iPhone music app developers Robert Hamilton, Jeffrey Smith, and Ge Wang consider the rise of "ubiquitous music" as a primary consequence of

---

[268] Mark Weiser, "The Computer for the 21st Century," *Scientific American* (September 1991): 94.
[269] Ibid.

mobile media usage, as apps "increasingly transport us into a world where we do not have to immerse ourselves in computers, but instead take computing into our physical world and nearly every part of our daily life."[270] Native Instruments' *iMaschine* advertisement offers musicians the ability to "make music anywhere," embedding full-fledged audio sampling and editing capabilities in a five dollar iPhone app (fig. 4.17a). In reference to their *GarageBand* iOS app, Apple expresses to consumers that "the world is your stage. This is your instrument" (fig. 4.17b).[271] Propellerhead's *Figure* is described as a "fun music-making app for instant inspiration," allowing the musician to "create music in no time with *Figure*'s dead-easy touch interface" (fig. 4.17c).[272] While previous visions for the future of personal computing, such as virtual reality, involved immersing individuals *inside* of a computer, ubiquitous computing "brings the computer outside, into our daily, lived experiences," as Adriana de Souza e Silva and Daniel Sutko note.[273] In contrast to software such as *Pro Tools,* which emphasizes the professional and specialized nature of the program, music app marketing highlights the ways in which the rhetoric of ubiquitous computing in the lineage of personal computing has influenced a general desire for the development of non-expert digital music and media production tools.

---

[270] Robert Hamilton et al, "Social Composition: Musical Data Systems for Expressive Mobile Music," *Leonardo Music Journal* 21 (2011): 64.

[271] Apple, "GarageBand," accessed March 23, 2016, http://www.apple.com/ios/garageband/.

[272] Propellerhead, "Figure," accessed March 23, 2016, https://www.propellerheads.se/figure.

[273] Quoted in Jason Farman, *Mobile Interface Theory: Embodied Space and Locative Media* (New York: Routledge, 2012), 11.

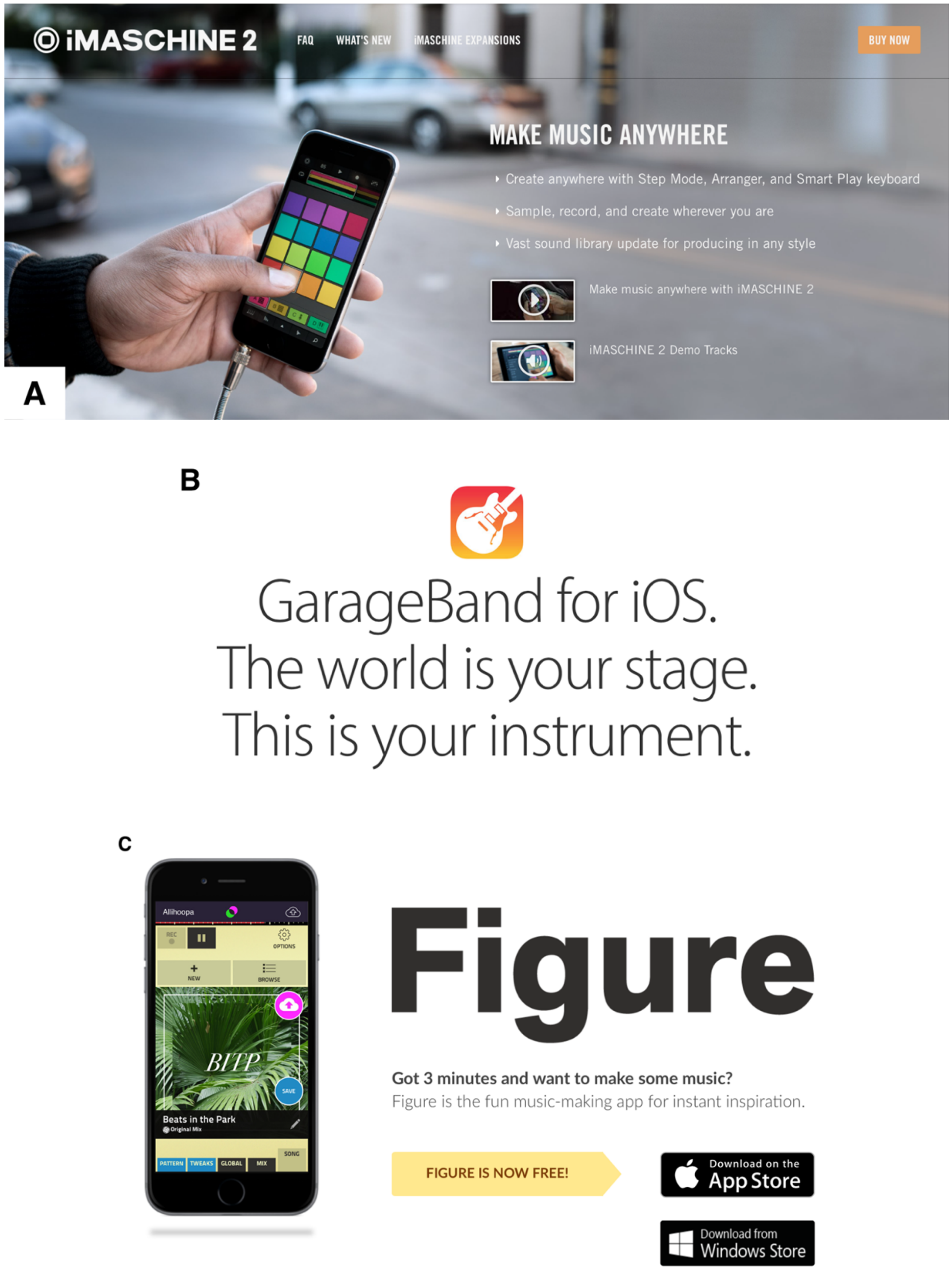


Figure 4.17: (A) *iMaschine 2* promotional material (Source: Native Instruments, "iMaschine 2," n.d., accessed April 30, 2016, http://www.native-instruments.com/en/products/maschine/maschine-for-ios/imaschine-2/); (B) *GarageBand* iOS promotional material (Source: Apple, "iOS – Garageband," n.d., accessed April 30, 2016, http://www.apple.com/ios/garageband/; (C) *Figure* promotional material (Source: Propellerhead, "Figure," n.d., accessed April 30, 2016, https://www.propellerheads.se/figure)

This situation brings up some broader questions. When mobile app developers build apps, and non-“musicians” make music with apps, is this really music-making at all? Or, are consumers simply buying into a specific brand experience? Is the question of what constitutes “real music” even useful anymore? In the final section of this chapter, I consider the implications of iPhone and app design affordances on practices of technology and media consumption. I suggest that the seemingly intuitive, natural, and ubiquitous affordances embedded within apps aligns the practices of mobile music *production* and digital media *consumption* more broadly. In doing so, it is possible to understand music as comprising not only *material* technologies (instruments, controllers) and performance spaces (dance clubs, concert halls), but a process-oriented *experience* that aligns with the maximalist consumption practices described in chapter one.

Ubiquitous Computing as Controlled Consumption

Mobile media exemplifies a dominant understanding of technology in the context of consumer capitalism—the idea that consuming technology can solve all of our problems and provide instant gratification in our everyday lives. When Apple first introduced the iPhone, the major issue Steve Jobs saw in existing smartphones was that they all included a bunch of plastic buttons that remained the same for every application. By removing the physical keyboard and increasing the screen size of the device, the iPhone GUI could literally become anything, depending on the nature of the app. The icon-grid interface layout that has come to define the iPhone reflects this tool-based conception of the device (fig. 4.18). It is a Swiss Army Knife of solutionism, offering to better your health, relationships, productivity, work, and everything else in your life with just a single tap.[274]

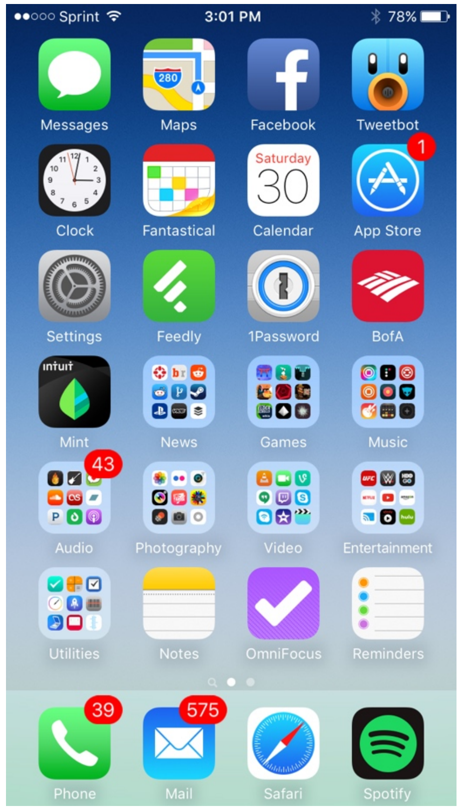


Figure 4.18: iPhone running iOS 9 (2015) home screen grid interface

Similar to the maximalist attitude towards technology described in chapter one, some view the iPhone a "tethered appliance" on which users have developed a dangerous overdependence.[275] Others view the iPhone as another useless gadget that reflects the problematic dynamic of "technological solutionism," in which software perpetuates the very

---

[274] The icon-grid interface, and mobile media more broadly, reflects what Joss Hands calls "platformativity": a gradual shift away from the Internet as a single hub for web-based cultures, towards a multiplicity of platforms including 'cloud'-based software, social media, tablets, smartphones, and 'app'-based interfaces. "Politics, Power, and 'Platformativity'," *Culture Machine* 14 (2013).

[275] Jonathan Zittrain, *The Future of the Internet—And How to Stop It* (New Haven and London: Yale University Press, 2008).

problems that it promises to alleviate.[276] Svitlana Matviyenko describes the ways in which "the 'needs' come with apps as part of the package, which means the 'solutions' are being sold to us along with the 'problems' they are mean to resolve.[277] Specifically, Matviyenko pinpoints associations between technology and happiness in advertising rhetoric, including Nokia's app store slogan, "Think appy thoughts," or Apple's popular motto, "There's an app for that."[278] In both cases, apps are marketed as catch-all solutions to problems that often arise, paradoxically, from the ubiquitous presence of technology in all aspects of our lives.

In the context of music and media production apps more specifically, solutionism manifests in the rhetoric of democratization and accessibility introduced at the beginning of this chapter—the idea that, with an iPhone, anybody can make anything anywhere. Apple claims that the *GarageBand* app, for example, allows users to "create incredible beats no matter where you are… with just a few taps." Similarly, with Apple's *iMovie* app, all it takes is "a few taps, a few swipes, and you're ready for your big premiere."[279] The marketing of the *Pages* app is perhaps the most blunt example, as they claim that the software allows you to "effortlessly create stunning documents… Writing has never been easier. Period."[280] By aligning simple touch-screen gestures with professional media production techniques, Apple both perpetuates its marketing ethos of vernacular creativity and appeals to the perennial anxiety that many users

[276] Evgeny Morozov, *To Save Everything, Click Here: The Folly of Technological Solutionism* (New York: Public Affairs, 2013); See also Jean Baudrillard on the idea of the "gadget," in William Merrin, "The Rise of the Gadget and Hyperludic Me-dia," *Cultural Politics* 10, no. 1 (2014): 8.

[277] Svitlana Matviyenko, Introduction to *The Imaginary App,* edited by Paul D. Miller and Svitlana Matviyenko (Cambridge, MA: The MIT Press, 2014), xxvii.

[278] Ibid, xvii. See also "iPhone 3g Commercial 'There's An App For That' 2009," *CommercialKid,* uploaded February 4, 2009, accessed April 30, 2016, https://www.youtube.com/watch?v=szrsfeyLzyg.

[279] Apple, "Creativity Apps: iOS," accessed March 23, 2016, http://www.apple.com/si/creativity-apps/ios/.

[280] Ibid.

hold towards creativity with the personal computer. Indeed, the seemingly immediate nature of interaction with the iPhone is posited as a catch-all solution to a long-standing unease and awkwardness that many feel when producing art and music with the standard WIMP interface. The title "smartphone" itself blurs the line between expert and non-expert production, encouraging users that it is not only the phone, but also its user, that is "smart."

The convergence of creative practices in mobile media production has resulted in the formation of what we might call an "upgrade culture," in which users simultaneously consume and produce content as part of an endlessly iterative digital economy. This framework for understanding the self-perpetuating organization of mobile media economies was foreshadowed by Henry Lefebvre as far back as 1967, when he introduced the concept of "controlled consumption." According to Lefebvre, everyday life in advanced industrialized societies has become a "voluntary programmed self-regulation" contained within a "closed circuit" of "production-consumption-production."[281] In the context of the iPhone, for example, the act of consuming apps perpetuates the very needs the software promises to solve. Lefebvre outlines four principles that characterize societies of controlled consumption: (1) A cybernetic industrial infrastructure integrating and handling production, distribution, exchange, and consumption is developed around the product; (2) The consumption is controlled through programming that closely monitors consumer behavior and the effects of marketing through tracking and surveillance; (3) Controlled obsolescence is programmed into the product, limiting its functionality and its durability; and (4) the overall effect of controlled consumption is a

[281] Henri Lefebvre, quoted in Chris Butler, *Henri Lefebvre: Spatial Politics, Everyday Life and the Right to the City* (New York: Routledge, 2012), 29.

significant reorganization and troubling of specific practices of everyday life.[282] These four principles are most clearly demonstrated by the fundamental mechanism that drives both production and consumption on the iPhone: the *App Store*.

Apple's *App Store* represents a centralized marketing app, a meta-app that structures the production and distribution of software on iOS devices (fig. 4.19). Represented by a simple pencil, paintbrush, and ruler icon, the store promises "productivity" above all else, offering one and a half million apps ranging from self-help tips to video games, word processors, digital audio workstations, and design tutorials. App prices are fairly low, for the most part, typically anywhere from completely free to five or six dollars. In addition to this initial fee, some apps contain "In-App Purchases" that either provide the user with extended functionality, or hook them into long-term use through a subscription model. Significantly, the vast majority of software developed for the iPhone are created by third-party companies. Therefore, Apple both strictly controls the content and distribution of apps in the store, and relies on outside labor to maintain its mechanics.

[282] Bro Pold and Andersen, "Controlled Consumption Culture: When Digital Culture Becomes Software Business," in *The Imaginary App,* 23-4.

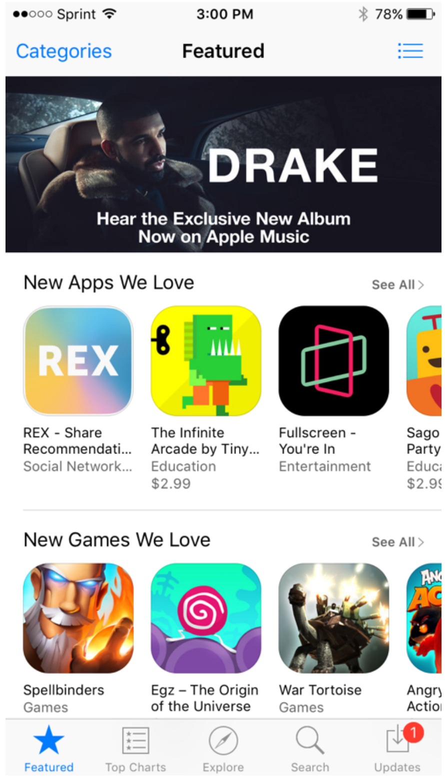


Figure 4.19: iPhone *App Store* (Screenshot taken by author, April 30, 2016)

Controlled consumption reflects the dominant ideology of consumer capitalism following the rise of software—an ideology "designed to sell not only a particular commodity but consumption itself," according to Timothy D. Taylor.[283] Controlled consumption in the app store succeeds through the deployment of "microtransactions" such as in-app purchases, software subscriptions, and downloadable content (DLC). Recently, Google claimed that the best method for hooking users into the extended lifecycle of a product is to fracture the consumer journey into "hundreds of real-time, intent-driven micro-moments," thus allowing the smartphone user "to act on any impulse at any time" to achieve "immediate results."[284] Through microtransactions in the

[283] Timothy D. Taylor, *The Sounds of Capitalism: Advertising, Music, and the Conquest of Culture* (Chicago and London: University of Chicago Press, 2012), 4.

[284] Google, "Micro-Moments: Introduction," accessed March 26, 2016, https://www.thinkwithgoogle.com/micromoments/intro.html.

app store, users' labor is thus more substantial than the apps themselves in preserving the controlled consumption model.

While the apps themselves are either given away or sold for a couple dollars, In-App Purchases come in the form of either monthly subscription payments or hefty one-time fees of over one hundred dollars. In iOS video games, for example, the rise of the "freemium" model—in which DLC can be purchased by gamers seeking extra levels or other perks—has managed to hook even the most casual players into the extended lifecycle of the product. The combat strategy game, *Clash of Clans*, grosses over one million dollars a day, largely from DLC purchases that allow the player to strengthen their village.[285] In the context of music production apps, DLC often comes in the form of sample packs or expansion plug-ins for the software. The *iMaschine* "Expansions Store," for example, regularly releases new virtual drum kits, synthesizers, and samples with which the musician can build tracks (fig. 4.20). In each case, In-App Purchases extend the act of acquiring software from a single event to a gradual process. Perpetual *consumption* becomes a primary mechanic of both gaming and music *production.*

[285] Rebecca Borison, "The 15 Highest-Grossing iPhone And iPad Games," *Business Insider,* May 20, 2014, accessed March 28, 2016, http://www.businessinsider.com/highest-grossing-iphone-and-ipad-games-2014-5?op=1.

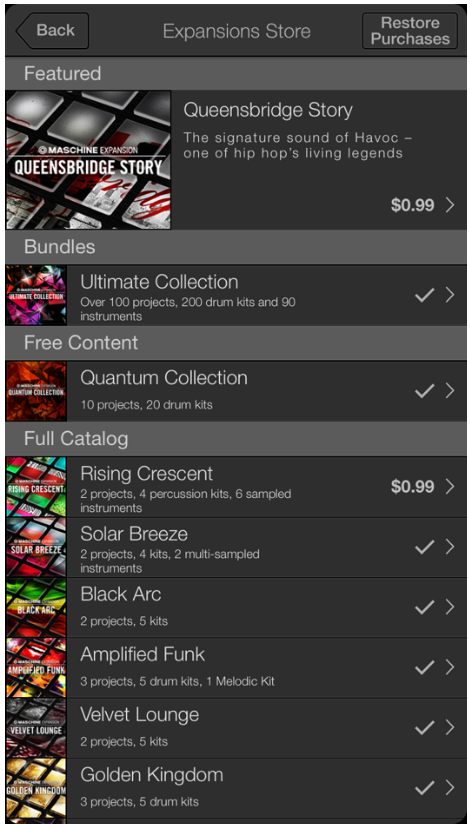


Figure 4.20: *iMaschine 2* "Expansions Store"

Of course, the iPhone is more than just a tool of controlled consumption, simply providing users with instant fixes to their personal problems. It is also a generative creative platform.[286] In addition to DLC sold by developers following an app's release, the ability of users to develop and share original content is often built into the app's design. Music production apps are especially aware of the benefits in employing "user-generated content," which typically involves musicians both sharing their creations with other users and, in turn, promoting the app itself through social media. Akai's *iMPC Pro* allows the user to upload directly tracks created in the app to social media networks such as SoundCloud (fig. 4.21a). *Figure* integrates the *Allihoopa* social music platform, allowing users to share their creations with thousands of other

[286] Joost Raessens, "Homo Ludens 2.0: The Ludic Turn in Media Theory," inaugural address at Utrecht University, November 19, 2010, accessed March 8, 2016, http://dspace.library.uu.nl/handle/1874/255181.

"campfire crooners," "table top drummers," and "shower DJs" involved in mobile music creation (fig. 4.21b). Similar to the rhetoric surrounding *Ocarina*, the *Allihoopa* platform claims to "bring all of us, everyone, together for a free and open exchange of music pieces, ideas and people, all united in 'doing music' just for the sake of it."[287] If In-App Purchases epitomize the controlled consumption model inherent to the app store, affordances for creating user-generated content highlight the generative potential of the iPhone as a creative platform.[288]

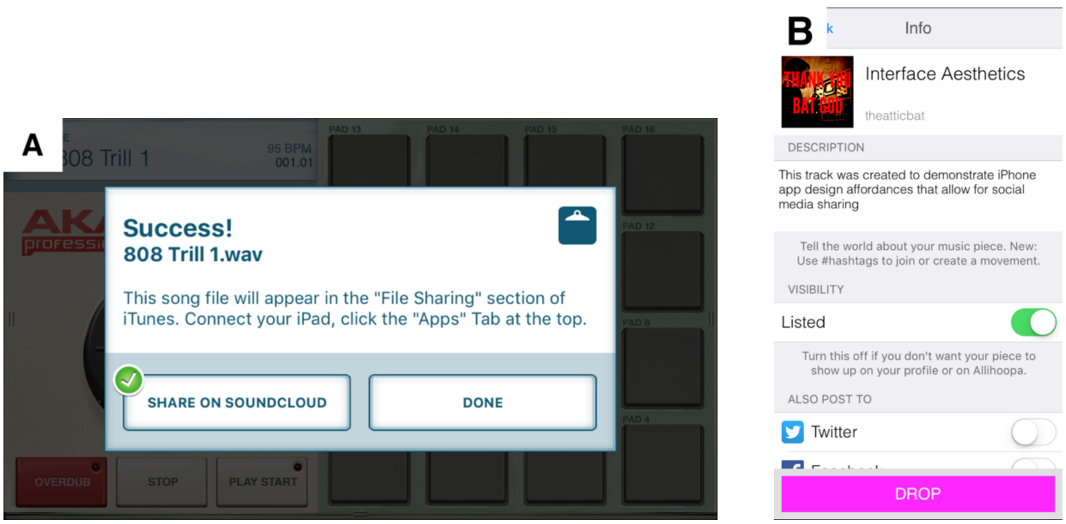


Figure 4.21: iPhone music app social media sharing. (A) Akai Professional, *iMPC for iPhone* (2013); (B) Allihoopa, *Figure* (2016)

As the logics of gaming increasingly infiltrate various media—from cinema to video games and mobile media—the aesthetics of media production cultures similarly converge. In contrast to Jonathan Zittrain's notion of the iPhone as *simply* a tethered appliance, the convergent

---

[287] "About Allihoopa," n.d., accessed March 28, 2016, https://allihoopa.com/about?&_suid=14591937644860864354913122952.

[288] Jean Burgess notes, "What the iPhone lacks in technological 'hackability,' therefore, it makes up for in social and cultural 'generativity,' thanks to its usability and the proliferation of apps that extend its functionality, and more importantly thanks to the creative, social and communicative activities of its millions of users who have integrated the iPhone, as a platform, into their everyday lives. "The iPhone Moment, the Apple Brand, and the Creative Consumer: From 'Hackability and Usability' to Cultural Generativity," in *Studying Mobile Media,* 40.

nature of apps within the smartphone also constitutes what William Merrin calls a “hyperludic” interface: a digital gadget designed for endless functioning, and with which we can never exhaust our play.[289] In this context, the intuitive physical gestures and elementary design affordances of music production apps play a crucial role in facilitating the instantly networked, informal sharing economies of the “remix culture” that has come to define creative production on digital platforms.[290] Think about the ways in which “liking,” “retweeting,” “sharing,” and “favoriting” social media posts, for example, perpetuates the viral nature of online content distribution. Here, the ludic affordances of the interface itself continuously generate even more forms of playful interaction. Music-making, in this context, is far more than the simple acquisition of goods, “but the mean mode of relating to goods, and to one another,” as Taylor suggests.[291] These examples embrace the ways in which mobile media production is capable of empowering the user and encouraging a sense of social togetherness, rather than creating a sea of mindless consumers.

The popular *Ocarina* app (2008) is a case in point. Just a year after Apple released the iPhone, mobile app development company Smule launched *Ocarina*—the first app to turn the smartphone itself into a musical instrument. The app offers the best of both worlds for mobile music production and cellular communication. Upon launching *Ocarina*, the user can choose either to view a tutorial on how to make music with the app (fig. 4.22), or proceed directly to the ocarina interface itself. The ocarina interface consists of four separate holes and an antenna icon at the bottom of the screen (fig. 4.23). In order to play musical notes, the user simply blows into the iPhone microphone while covering combinations of holes with his or her fingers. The

---

289 Merrin, 16.

290 Lawrence Lessig, *Remix: Making Art and Commerce Thrive in the Hybrid Economy* (New York: Penguin, 2008); Aram Sinnreich, *Mashed Up: Music, Technology, and the Rise of Configurable Culture* (Amherst and Boston: University of Massachusetts Press, 2010).

291 Taylor, *The Sounds of Capitalism,* 239.

sounding notes all correspond to a specific musical scale, which is chosen by the player. Tilting the phone downward while blowing into the microphone adjusts the vibrato rate and depth of the sounding note. Together, these affordances abstract the nuances of playing an actual ocarina—breath control, understanding musical scales, and producing vibrato with fingers rather than a digital technology—into an easy to use app.

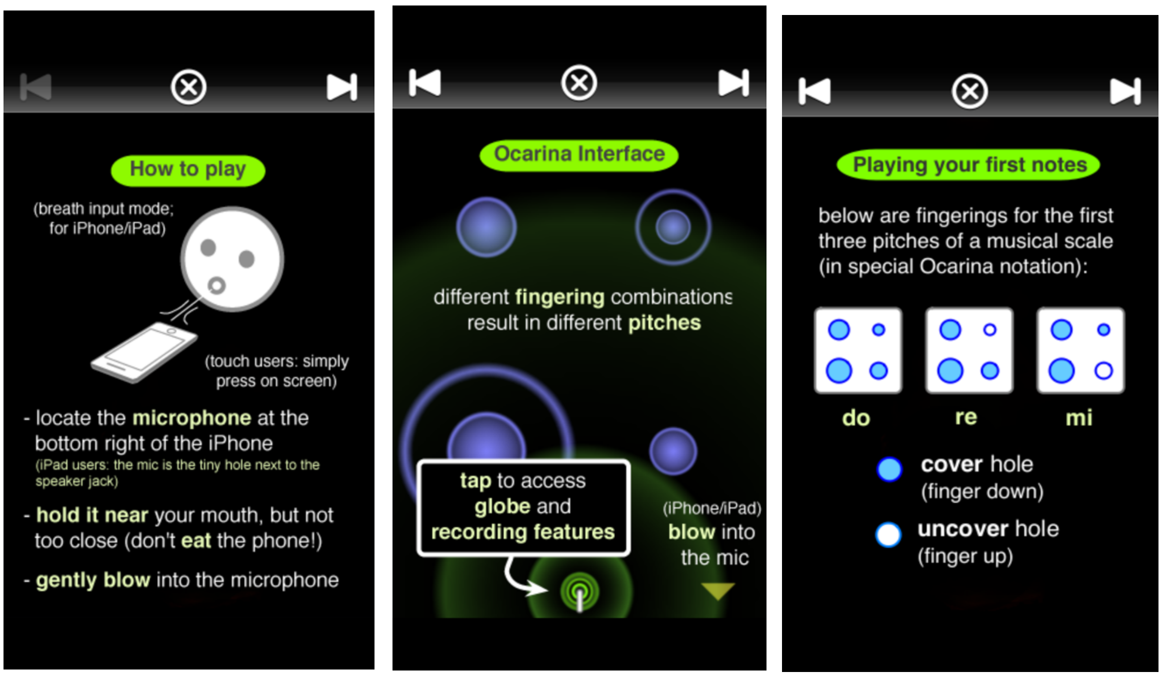


Figure 4.22: *Ocarina* tutorial

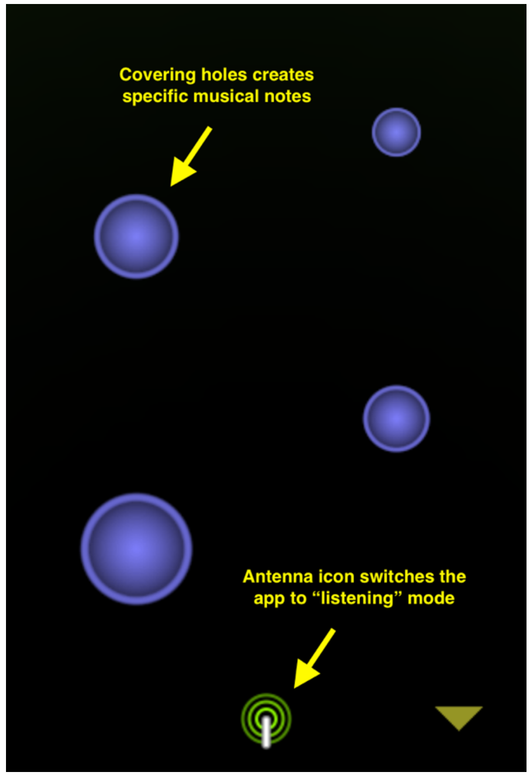


Figure 4.23: *Ocarina* main interface

Like other apps described in this chapter, *Ocarina* similarly incorporates social media sharing capabilities. By tapping the antenna icon at the bottom of the ocarina interface, the user is taken to a 3D map of the world that displays bright lights and plays sounds from locations where other *Ocarina* users are creating music with the app (fig. 4.24). Double-tapping locations on the map allows the player to "zoom in" on specific regions and listen to melodies being played in real-time by *Ocarina* users in those areas. For musicians, the usability of the *Ocarina* interface combines with the global networking capabilities of the iPhone to encourage enhanced forms of ubiquitous musical production inherent to instruments such as the harmonica. For casual smartphone users, the app transforms the iPhone from an everyday tool into a creative platform for individual and collective expression.

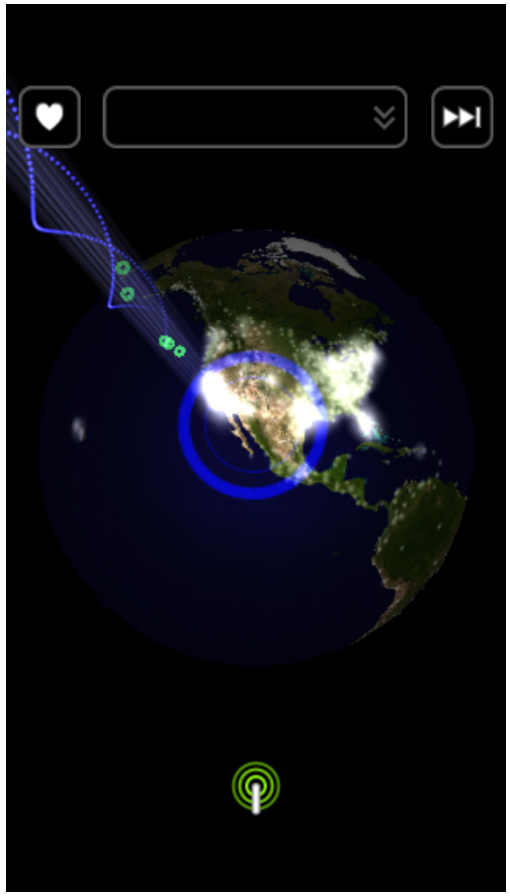


Figure 4.24: *Ocarina* social media "listening" mode

On a technical level, *Ocarina* established many of the iPhone app design affordances discussed earlier. The app employs a graphical design that blends familiar visual and haptic metaphors from existing musical instruments with abstract patterns and gestural affordances unique to the smartphone. Further, the app embraces the ubiquitous computing design philosophy established by Weiser. App designer Ge Wang discusses the ways in which the app shifts the "typical screen-based interaction to a physical interaction, in our corporeal world," where the user engages the device with "palpable dimensions of breath, touch, and tilt."[292] Additionally, social media integration and the ability to upload user-generated content to the

[292] Ge Wang, "Ocarina: Designing the iPhone's Magic Flute," *Computer Music Journal* 38, no. 2 (Summer 2014): 10

"cloud" promotes the development of user communities surrounding the app, further enhancing the perception of intuitive, direct manipulation in the corporeal use of the device. Tapping a globe icon in the app displays a 3D world map visualizing the location of other *Ocarina* users currently playing the instrument, imbuing the musician with a sense of being instantly connected to a global network of likeminded individuals.

On a broader level, the generative nature of the app epitomizes the ways in which mobile media reflects a shift in perceptions of technology and cultural value, from one that privileges technical skill to one that embraces a disintegrating dichotomy between expert and non-expert users. As a result of the democratization of gestures, design affordances, and social connectivity in mobile media production and consumption, music-making is valued for experiences it generates rather than the sounds themselves. For example, an American soldier in Iraq describes *Ocarina* as "my peace on earth. The globe feature that lets you hear everybody else in the world playing is the most calming art I have ever been introduced to." For this user, the values of global interconnectivity inherent to social and mobile media are embedded within the design of the app, which is perceived to "bring the entire world together without politics or war."[293] Here, producing musical sounds is less important than the experience of connecting with other *Ocarina* users from around the world.

At the same time, it is hard to dismiss the ways in which users become literally addicted to their smartphones, their eyes glued to the glass screen as they attempt to drown out the world around them. While developers promote user-generated content as a positive force that encourages creativity and the open sharing of ideas across a social network, others view it as a form of free labor generated for the profit of the developers. In the same way that app design

[293] Wang, 20.

hooks users in by focusing the physical and cognitive capabilities of the user, the free promotion and cloud-based content development garnered by the iPhone exemplify the type of "immaterial labor" captured by what many have termed "cognitive capitalism."[294] In this situation, control over an intellectual workforce increasingly withdraws from direct management of the production process, and instead exercises "an indirect power, based on ownership of intellectual assets, in the way a landowner might extract revenues from tenants of a piece of land."[295] Here, the aesthetics of app design align with the economics of software development and distribution, as the structures through which design and economics function are hidden from the user.

Ultimately, the iPhone is *both* a generative creative platform and a tool for the proliferation of controlled consumption in twenty-first century capitalism. Music-making, for iPhone users, occurs not in the realm of sound production itself, but rather in the perceptibly democratizing affordances of the device, as well as in the generative social and consumer platform the iPhone has become. The unique aspect of Apple's upgrade culture is not the way in which it creates polarizing debates between technological solutionists and luddites, but the ways in which it turns the acts of sociocultural negotiation at the heart of all technological change into a dialogic experience. As new apps are developed every minute and new iPhone hardware is rolled out on an annual basis, non-expert musicians and everyday users of mobile media continue experiment with music-making, often for the first time in their lives. In contrast to the controllerists described in chapter three, these individuals are not concerned with distinguishing

[294] From Carlo Vercellone, "The new articulation of wages, rent and profit in cognitive capitalism," Queen Mary University School of Business and Management, February 29, 2008; Yann Moulier-Boutang, *Cognitive Capitalism* (Boston: Polity, 2011).

[295] Nick Dyer-Witheford, "App Worker," in *The Imaginary App,* 135.

themselves as “musicians,” but rather embrace their dual status as musical producers and media consumers.

As I have suggested throughout this chapter, the major innovation of mobile media design lay in its ability to embed the creative processes of music-making into the everyday life of the consumer, therefore integrating the previously separate venues of work, leisure, and personal well-being. Just as the multitasking capabilities afforded by iPhone design have resulted in the integration of various forms of mundane labor, companies such as Apple are increasing the extent to which consumers rely on mobile technologies in order to fulfill these everyday practices. At the same time, it’s hard to dismiss the sense of joy felt when someone with no traditional musical training creates music for the first time. As much as the iPhone turns the process of music-making into a form of labor to generate profit for Apple, the experimental design of many music-making apps reveals the importance of shared play and exploration at the heart of all musical experiences, regardless of technical skill.

Section III

Listening to Software

The final section of this dissertation examines the emergent and procedural aesthetics in video game audio, as captured by the following quote:

> Things which grow shape themselves from within outwards—they are not assemblages of originally distinct parts; they partition themselves, elaborating their own structure from the whole to the parts, from the simple to the complex.[296]

I read this description of organic emergence, by Alan Watts, not while perusing treatises on Zen Buddhism, or scanning self-help manuals for spiritual guidance, but rather in the opening sentences of Matt Pearson's practical guide to using the *Processing* programming language for creating digital art.[297] The conflation of computer code and algorithmic processes with organic and holistic metaphors is commonplace in many contemporary digital art scenes, and has a long history going at least as far back as the multimedia experiments of the 1960s avant-garde. How have the aesthetics of what has been broadly labeled "generative media" affected forms of digital audio production more generally? The final chapter of *Interface Aesthetics* hints at the ethical implications of examining software as a process-oriented "experience" rather than a fixed commodity.

Chapter five examines the application of generative music to emergent media experiences, thus opening up new forms of human-computer interaction—what I define as "procedural interfaces"—through an examination of popular "indie" games such as *Proteus* (2013) and *Fract OSC* (2014). In a 1996 talk, "Generative Music," Brian Eno describes the

[296] Alan Watts, *Nature, Man, and Woman* (New York: Vintage Books, 1991).

[297] Matt Pearson, *Generative Art: A Practical Guide Using Processing* (Shelter Island, NY: Manning Publications, 2011).

principle that formed the basis of his philosophy of ambient music: "the idea that it's possible to think of a system or a set of rules which once set in motion will create music for you."[298] With the rise of procedural generation in video game design, and other forms of computer-generated media, this desire for self-generating "environments" would seemingly materialize in the multisensory space of video games. Extending a historical lineage of "generative aesthetics" throughout the twentieth century, I define the concept of emergence both technically and discursively, analyzing practices of sound design in the *Unity3D* game design software in conjunction with the theoretical and aesthetic motivations of game and sound designers themselves. Expanding on Eno's idea, procedural interfaces encourage dynamic, relational modes of technological engagement in which sonic interaction design guides the players through the virtual world, rather than a set of rules imposed from the designers themselves.

In the concluding pages of the project, I outline some broader reflections on procedural listening, suggest the benefits of considering interface aesthetics as a critical method for researchers and practitioners, and offer directions for future research.

[298] Brian Eno, "Generative Music," *In Motion Magazine,* July 7, 1996, accessed December 11, 2014, http://www.inmotionmagazine.com/eno1.html.

Chapter 5

Worlds of Sound: Indie Games, Proceduralism, and Emergent Aesthetics

Following the growth of low-cost Internet game distribution platforms such as *Steam*, “indie” became a widespread descriptor for a gaming culture comprising developers, players, and distributors interested in creating an alternative to what was rejected as a glutted, uninspired, and corporatized game market. The rise of the indie aesthetic has followed a number of parallel developments in the broader gaming world, including the proliferation of gestural controllers such as the Wiimote and Xbox Kinect,[299] open-world, “sandbox”-style games such as the popular *Minecraft* (2009), an ever-expanding market for “casual” mobile gaming,[300] and an increased community of amateur game developers and distributors. As a set of technical practices among game designers, the indie development community has foregrounded key features of contemporary video games, including dynamic ambient audio, computer generated simulations of natural ecologies, and a focus on game mechanics over visual fidelity and narrative complexity.[301] Together, these cultural, aesthetic, and technological shifts typify the ways in which experimental, independent, or alternative approaches to game design both deploy

[299] Pippin Barr, James Noble, and Robert Biddle, “Video Game Values: Human-Computer Interaction and Games,” *Interacting With Computers* 19 (2007): 180-195.

[300] Jesper Juul, *A Casual Revolution: Reinventing Video Games and Their Players* (Cambridge, MA: The MIT Press, 2009).

[301] Game mechanics refer to the “rule based systems that facilitate and encourage a user to explore and learn the properties of their possibility space through the use of feedback mechanisms.” In other words, they are the affordances that structure gameplay for the player. Daniel Cook, “What are Game Mechanics?” *Lostgarden,* October 23, 2006, accessed April 8, 2016, http://www.lostgarden.com/2006/10/what-are-game-mechanics.html.

digital aesthetics typical of large-scale studio development and yet also attempt to differentiate their small-scale offerings from those of major software companies.[302]

Through analyses of sound design in games such as *Fract OSC* (2014) and *Proteus* (2011), this chapter aligns the aesthetics and creative practices of game designers in the indie gaming community with traditions of generative art and music.[303] In doing so, I outline a broader musical turn that has occurred within gaming culture—one in which sound becomes a primary mechanic in the shaping of generative aesthetics at the heart of multiple forms of digital art. While new media or "post-media" aesthetics have often prioritized continuity or rupture in relation to disciplinary histories like those of cinema (Manovich 2001), literature (Hayles 2008), or fine art (Bourriaud 1998), the contemporary practices and historical precedents of generative aesthetics ground an understanding of the gaming experience as establishing both a historical lineage with earlier forms of process-oriented computer art, as well as an increased focus on sound as a key medium for design and creativity in interactive media.

Significantly, sound designers embed generative mechanics within a game's sound engine which, rather than simply being a "hidden" part of the software code, manifests in

---

[302] Useful surveys of the relationship between indie game aesthetics and the oppositional attitudes of indie culture can be found in Lipkin (2012), Ruffino (2012), Guevara-Villalobos (2011), and Martin and Deuze (2009).

[303] "Sound design" can be a vague term among digital media artists. Since musicians and composers working across industries and media platforms—including film, video games, installation art, mobile app development, and architectural acoustics, among others—have loosely employed the term, it has lost its specific meaning and disciplinary focus. For an exemplary case study of these debates in action, see Andy Farnell, "Perspectives on Sound Design," *Designing Sound* blog, August 27, 2013, http://designingsound.org/2013/08/perspectives-on-sound-design/. In this essay, I use the term to describe the ways in which the gameplay mechanics combine with the diegetic and non-diegetic musical elements of the overall game in constructing a holistic sonic experience informed by the dual layers of design and practice. This approach reflects conventional wisdom within the emerging discipline of "sound studies," which views sound design as a social force comprising shared knowledges, practices, and cultural imaginations. See Jonathan Sterne, "Sonic Imaginations," in *The Sound Studies Reader,* edited by Jonathan Sterne (New York: Routledge, 2012).

broader practices shared between indie game players and generative music composers, as I argue below. Rather than perceiving the technical structure or code of the game as determining form, effects, and reception, as a platform-oriented study might do (Bogost and Montfort 2009), locating the sound engine in this larger history of process-oriented practices and forms allows us to understand the ways in which sound in digital media is not simply a fixed background effect, but also a key element in the broader audio-visual interface that is generated through the very process of human-computer interaction. Further, framing game software as a manifestation of a larger generative aesthetics suggests, as well, a new model for sound-oriented gaming experiences.

This chapter thus argues for a re-framing of our historical understanding of the relationships between computational art forms, foregrounding the sound design of indie games and generative media practices more broadly. Ultimately, I provide not only an account of the ways in which sound and musical design have defined the ludic interests of the indie game movement, but also a theoretical model that seeks to address the relationship between interactive systems, musical and sonic meaning, and user experience both in games and across digital platforms.

Indie Games and Emergent Aesthetics

The video game industry in the early twenty-first century has been culturally and economically divided between "indie" game developers on one hand, and "Triple-A" games on the other. Technically, the term Triple-A refers to a grading scale used to denote production value in the games industry, with "AAA" being the highest rating. On a practical level, though, this distinction is rooted in the size of the budget and production team dedicated to the release of a given game. Indie game development often consists of small teams, sometimes even a sole individual with a meager budget. In contrast, Triple-A development is carried out by multiple creative teams housed within large entertainment companies with immense budgets. As a result of the considerable resources poured into Triple-A production, these games are sometimes categorized as "blockbusters"—a classification justified by the often massive opening weekend sales.[304]

In addition to economic disparities, the indie versus Triple-A dichotomy can be characterized by contrasting audio-visual aesthetics. Differing ideas about the role of sound in games, for example, has often grounded the broader split between the two cultures. Following the "blockbuster" analogy, Triple-A games typically align their aesthetics with those of popular American action cinema. In first-person shooter franchises such as *Call of Duty,* sound typically enhances the effects of visual cues, serving as sonic amplifiers of the (mostly) male action hero at the edge. Assault rifle barrages are echoed by quick rhythmic bass and percussion chops, while the visceral contact of pistol whips and lobbed grenades marks ruptures in time and space as slow motion frame rates mirror bass "drops" in sonic texture and rhythmic pacing. "Hardness" is the overriding affect; compressed, gated kick and snare drum samples combine with coagulated basslines made up of multiple oscillators vibrating at broad frequency ranges, colonizing the soundscape by filling every chasm of the frequency spectrum. In this example, sound is the affective catalyst for the emanation of an unabashedly assertive, physically

[304] For example, *Call of Duty: Black Ops III* sold $550 million in its opening weekend (2015).

domineering, and adrenaline-addicted masculine culture that has defined a large portion of Triple-A games.[305]

The aesthetics of indie games are subtler, employing minimalist audio-visual aesthetics reflective of the small-scale nature of independent game development. For example, in 2009, indie development company Mojang released its sandbox survival classic, *Minecraft.* In contrast to the cinematic graphic fidelity and epic action sequences of many blockbuster first-person shooter games, *Minecraft* drops the player into a randomly generated world of pixelated landscapes, with no resources and few instructions on how to proceed. As the player learns the mechanics of the game through almost completely unfettered exploration, an entire ecology of plants, animals, and weather patterns seems to emerge spontaneously, resulting in what feels like an endless world of lo-fi abstraction. Ambient sounds and peaceful piano melodies fade in and out, providing the soundtrack to everything from planting seeds in a garden to watching the sunset. By providing subtle cues that help guide the player through the game, music functions like any other building block in the virtual sandbox—as architecture, landscape, and built environment.

The refreshing feeling of non-linear exploration afforded by the open world of *Minecraft* has appealed to an entire generation of gamers seeking to escape the seemingly rigid gameplay of many Triple-A releases. Within only five years, *Minecraft* became one of the top three highest grossing games of all time, selling over fifty-four million copies and influencing an entire

[305] Mike D'Errico, "Going Hard: Bassweight, Sonic Warfare, and the 'Brostep' Aesthetic," *Sounding Out* blog, January 23, 2014, accessed April 19, 2016, https://soundstudiesblog.com/2014/01/23/going-hard-bassweight-sonic-warfare-the-brostep-aesthetic/.

generation of indie game designers and developers.[306] Moreover, it has defined many features of subsequent video games, including a focus on underlying game systems and mechanics rather than cinematic display resolution and processing power, as well as the "do-it-yourself" (DIY), entrepreneurial spirit of indie culture more broadly. Designer Craig Stern captures the oppositional attitude well, describing indie as "a way for small, independent developers without ties to publishers to effectively market and distribute their games… 'Indie' empowers the smallest and the freest of game developers. It is a declaration of creative freedom."[307]

This attitude of embracing limited resources and self-consciously attempting to assert oppositional practices in game design has encouraged the development of highly experimental games, as mentioned, often conceptualized as sandbox spaces designed for open-world exploration rather than linear narrative. Developer Jonathan Mak describes his musical puzzle-platformer *Sound Shapes* as "walking through a world of music."[308] In describing *Fez,* the vast "2D platformer set in a 3D world," designer Phil Fish states, "FEZ aims to create a non-threatening world rich with ambiance, a pleasant place to spend time in."[309] Games such as *Osmos* and *Splice* position the player within extreme macro and micro ecologies, from cosmic space to microbial DNA, highlighting the acute vulnerability and insignificance of the player.

---

[306] Megan Farokhmanesh, "Minecraft console sales surpass PC, 'almost 54M' copies sold in total," *Polygon,* June 25, 2014, accessed April 29, 2016, http://www.polygon.com/2014/6/25/5843358/minecraft-console-sales-54m-copies-sold.

[307] Craig Stern, "What makes a game indie: a universal definition," *Sinister Design,* August 22, 2012, accessed April 19, 2016, http://sinisterdesign.net/what-makes-a-game-indie-a-universal-definition/.

[308] "E3 2011 Machinima Coverage - Sound Shapes on the PS VITA Game Demo & Interview," *Machinima*, uploaded June 9, 2011, accessed April 19, 2016, https://www.youtube.com/watch?v=_rRU0bNeJ20.

[309] "Fez," *Polytron,* accessed June 1, 2014, http://fezgame.com.

While game developers never fully agree on what is or is not "independent" or "mainstream," I build my argument on the position that the audio-visual aesthetics of indie games are defined first and foremost by the presence of what artists have called "emergent aesthetics."[310] In contrast to aesthetic theories of the eighteenth and nineteenth centuries, which attempted to create rules to explain things already present in the world, emergent aesthetics attempt to create rules which, in turn, generate the world itself.[311] In regards to *Minecraft* and the other games previously mentioned, emergent aesthetics enhance the player's experiential engagement with the virtual world, as the visual environment is seen to literally materialize in the process of gameplay. In this context, sound occupies a dual position. First, audio *content* arises in a seemingly spontaneous manner, as a result of a feedback loop between the player's actions and randomizing algorithms in the game's software code. Second, the *processes* and "rules" of gameplay are often defined by emergent mechanics ingrained within the game's sound design. Emergent aesthetics in sound thus allow the game environment to become an interactive digital instrument of sorts. Rather than pursuing a set of goals towards a linear narrative with a pre-composed soundtrack, emergent aesthetics encourage the player to interact critically with the mechanics of game itself through forms of procedural listening. I will now provide overviews of

[310] Mihai Nadin, "Emergent Aesthetics: Aesthetic Issues in Computer Arts," *Leonardo* 2 (1989): 43-8. Scholars have introduced many terms to describe audio and gameplay that exists as more than simply a sonic backdrop to the overall media experience, including "procedural," "dynamic," "adaptive," "interactive," and "generative," among others (see *The Oxford Handbook of Interactive Audio* for a broad overview). Throughout this essay, I use the adjective "emergent" as an overarching theme that encompasses the features and aesthetic effects shared by these terms, including non-linearity of narrative, goal-less exploration, system components that respond and react to user input in "real-time," and abstraction of audiovisual content.

[311] Glen Carlson, "Emergent Aesthetics," glencarlson.com, accessed April 19, 2016, http://www.glencarlson.com/emergent/Carlson_EmergentAesthetics.pdf.

two games that manifest such emergent aesthetics in sound: *Fract OSC* (2014) and *Proteus* (2013).

"Explore an abstract broken-down world built on sound. Rebuild its structures and forgotten machinery. Create your own sounds and music within the world."[312] This game synopsis from the developers' website is perhaps the only set of instructions you will receive before entering the cold, dark world of Phosfiend Systems' *Fract OSC*—a music-based puzzle game released on both Mac and Windows operating systems in April of 2014. Like many indie games, *Fract* basically consists of a 3D, "open-world" environment that privileges rules and game mechanics over visual fidelity. Upon launching the game, you are placed on an unlit platform of black polygons, presented with what appears to be a white neon elevator in front of you, and a spiral ramp just beyond your purview (fig. 5.1).

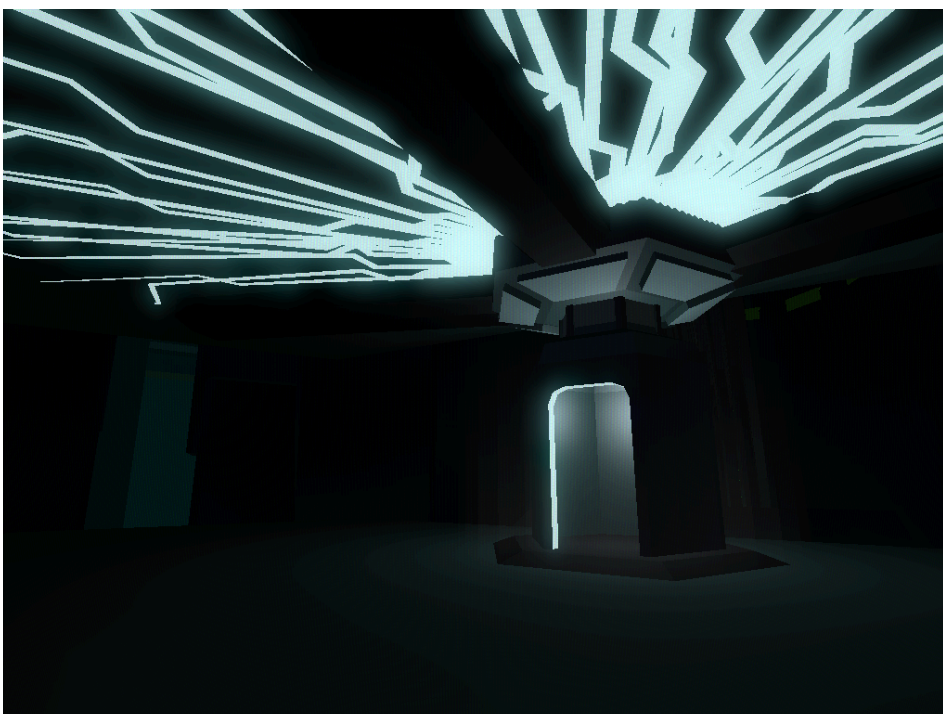


Figure 5.1: The screen upon entering the initially bleak world of *Fract OSC*

[312] Phosfiend Systems, "Fract," fractgame.com, accessed April 19, 2016, http://fractgame.com.

As you ascend the ramp, you are confronted with an enclosed structure of locked modules only later revealed to be components of a real-time subtractive synthesizer. There are few resources at your disposal, virtually no in-game hints on how to proceed, and a limited graphical user interface. As you traverse this sparse yet monumental world of sci-fi abstraction, you begin to be introduced to the simulated physics and mechanics of the visual space. You are introduced to these aspects through an exploration of sound.

The visual environment of *Fract* consists of abstract fragments of actual synthesizer components—buttons, knobs, sliders—that function as sonic and spatial markers, both allowing the player to create original sounds through the game's built-in subtractive synthesis engine, and to navigate the virtual world through warp points (fig. 5.2). Meanwhile, the pre-composed soundtrack evolves with the players' understanding of the game mechanics, starting from static bass drones and subtle synth pads to vibrant arpeggios and immense lead melodies that mirror the towering structures and radiant lighting schemes gradually generated throughout. While there are no clearly defined goals in *Fract*, the primary purpose of the game is to gradually reconstruct and revive a defunct world through an exploration of its physical and sonic possibilities. Sound is the primary building block of gameplay, and the fundamental mediator between the game and the player.

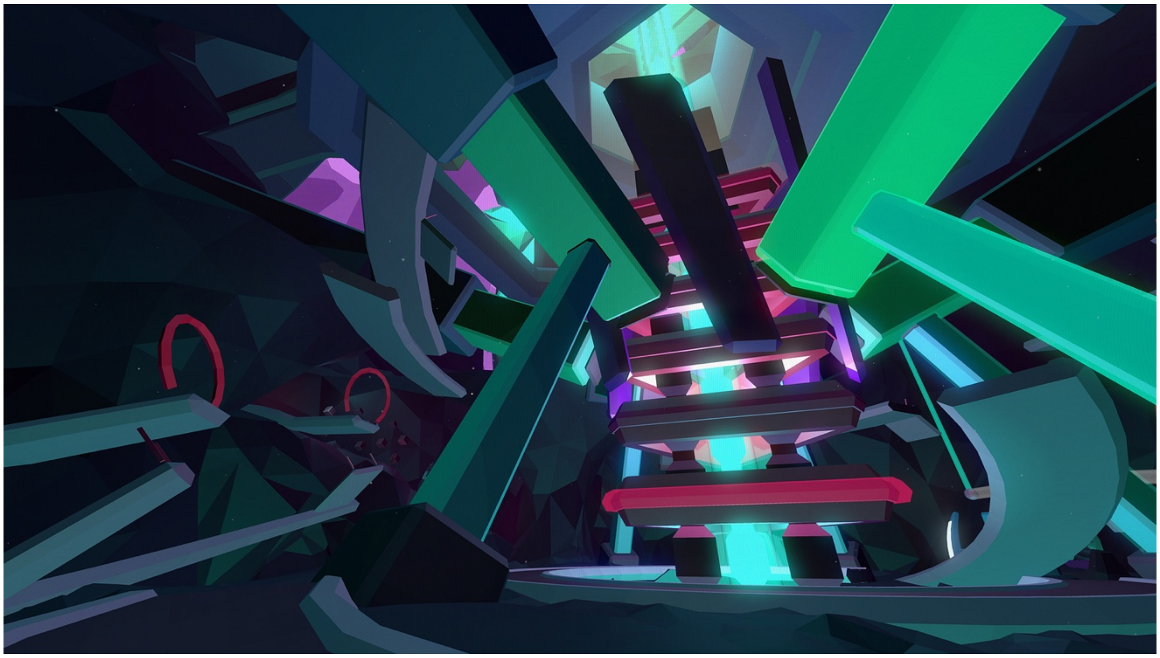


Figure 5.2: In the guts of the synthesizer. The bass synth module being constructed in-game (Source: FRACTgame, Flickr account, February 11, 2013)

Considering the historical ways in which emergent aesthetics arise and recombine in games, *Fract OSC* aligns the dynamics of early puzzle games such as *Myst* (1993) to the musical interests of rhythm games such as *Frequency* (2001). Each game results in a non-linear experience based on exploration through a world whose mechanics are defined by its sonic structure and the gestural articulation of that structure by the player. The exploratory, open-ended design of *Fract* highlights broader emergent aesthetics that have influenced the 'do-it-yourself' attitude of indie game design and development. This aesthetic embraces the funding situation of limited resources and accomplishes seemingly unlimited things, such as create and display vast 3D environments using finite computer processing power, or construct complex systems based on a single rule or game mechanic. These emergent aesthetics are noticeable in both the game design and player experience, as the audiovisual environment is seen to emerge literally in the process of gameplay.

Ed Key and David Kanaga's *Proteus*—a self-proclaimed "game of audio-visual exploration and discovery"—is another example of a game that uses emergent aesthetics to provide an alternative experience to the blockbuster, Triple-A titles.[313] Released as a ten dollar digital download for Mac and PC computers in 2013, the pixelated 3D world of *Proteus* is designed in such an open-ended way that nearly every element of the game is built using design principles intended to facilitate the perception of emergence in the player experience. Each time the game is loaded, a sparse island landscape is procedurally generated based on the global positioning system location of the user, resulting in the creation of a unique world each time it is played. It appears as there are no strict rules to follow, and the player is given no specific controls to navigate the game. One is simply dropped on an island, free to observe the world unfolding before them, emphasizing a typical focus of the indie aesthetic on the system rather than the player or designer.

The sound of *Proteus* is generated from a combination of ambient noises and synthesized pitches produced from every aspect of the visual environment; it can be triggered based on the player's proximity to specific objects in the world, and varies in frequency and timbre depending on the ways in which the player attempts to interact with the objects. For example, walking towards an animal causes it to scurry away, producing rapid, high-frequency string plucks in its wake. The longer the player explores a given interaction, the richer and more complex the sonic result. Loitering in a small forest for a long enough period of time, the player will hear a single synthesizer pad tone develop into an immense ambient soundscape in parallel with an increasing number of falling leaves from the trees. Indeed, sound acts as one of the few sensorial guides

---

[313] Twisted Tree Games, "Proteus," twistedtreegames.com, accessed April 19, 2016, http://twistedtreegames.com/proteus/.

through the open world of *Proteus,* intensifying the emergent aesthetics that define the visual landscape (fig. 5.3). In this way, the gameplay experience is inherently process-oriented, experiential, and improvisational. The mechanics of game design work in conjunction with player experience, as the seemingly random juxtaposition of sound and image creates emergent aesthetics in gameplay.

Figure 5.3: A typical afternoon in *Proteus*. Leaves fall from bright pink trees as the sun makes its way across the sky. Abstract blue dots float along the ground, beaming with high-pitched synthesizer tones as I approach.

Current Understandings of Sound and Game Design

These brief initial accounts of *Fract OSC* and *Proteus* have started to suggest how sound plays a crucial role in defining the aesthetics and mechanics of play in indie games. In spite of this, current game scholarship has yet to address the shared relationships between *sound* design and *game* design. Media theorists and musicologists alike tend to understand sound in gaming as either a matter of the subjective experience of the player or the structural or platform properties of the game as a designed system. Recent musicological scholarship in game studies, for example, has provided useful frameworks and insights in understanding how the multisensory space of video games influences the subjective and cognitive capacities of the player. Karen Collins posits the notion of "kinesonic synchresis" as the ability of game sound to carry "connotations of its haptic and visual associations even in the absence of these modalities in media."[314] In this context, player interaction with sonic material means that sound becomes a sensory guide for the player as he or she navigates the virtual world, as in *Fract*. Musicologists William Cheng and Kiri Miller examine the ways in which players construct ethical subject positions in response to virtual actions committed through forms of embodied interaction in games such as *Fallout* and *Guitar Hero.*[315] While these scholars, among others, have done excellent work in bringing the experience of sound to the forefront of game studies, their research focus generally remains on the subjectivity of the player rather than the technical work of game and sound designers.

At the same time, research from sound *designers* and game audio *engineers* has provided practical knowledge in how sound can be structured technically to create emergent experiences for the player, thus foregrounding the technical systems of the game over the aesthetic experiences of the player. The programming technique known as procedural content generation (PCG) has been a particularly strong focus among developers interested in the "dynamic" and "adaptive" capabilities of audio in gameplay.[316] Since the commercial proliferation of video

[314] Karen Collins, *Playing With Sound: A Theory of Interacting with Sound and Music in Video Games* (Cambridge: The MIT Press, 2013), 39.

[315] William Cheng, *Sound Play: Video Games and the Musical* Imagination (Oxford and New York: Oxford University Press, 2014); Kiri Miller, *Playing Along: Digital Games, YouTube and Virtual Performance* (New York and London: Oxford University Press, 2012).

[316] Andy Farnell, "Procedural Audio Theory and Practice," in *The Oxford Handbook of Interactive Audio,* edited by Karen Collins, Bill Kapralos, and Holly Tessler (New York: Oxford University Press, 2014).

games in the 1980s, PCG has facilitated emergent aesthetics in game design by generating content randomly or pseudo-randomly through algorithmic computational processes rather than through custom-modeled work by the game or sound designer. For example, Collins describes Japanese composer and artist Toshio Iwai's *Otocky* (1987), a side-scrolling shooter game in which the sound of the player's firing rhythmically quantizes in real-time, becoming the melodic accompaniment to a two-note bass line.[317] Moreover, Collins and others have highlighted the ways in which PCG gave early game developers an efficient strategy for preserving computer memory and storage, thus foreshadowing the indie aesthetic of embracing limited resources in game development. *Times of Lore* (1988), for example, used random-number generators to give the impression of variety despite a limited number of pre-programmed musical patterns.[318] Significantly, PCG is agnostic to media or content type, generating anything from levels and physical assets to user interface elements and sound design. As a result, PCG is now a common technique for creating digital content among indie game developers as well as artists across media platforms.

In contrast to the work of musicologists and game audio professionals, recent work from game scholars has attempted to align the *technical* practice of PCG with broader *aesthetic* values in the culture of indie gaming. For example, Ian Bogost characterizes the indie aesthetic through what he calls the "proceduralist style" of game design, highlighting design trends such as an orientation towards process and introspection, a foregrounding of game rules and mechanics, and

[317] Karen Collins, "An Introduction to Procedural Music in Video Games," *Contemporary Music Review* 28, no. 1 (2009): 7.

[318] Niels Böttcher, Héctor P. Martinez, and Stefania Serafin, "Procedural Audio in Computer Games Using Motion Controllers: An Evaluation on the Effect and Perception," *International Journal of Computer Games Technology* (2013).

an abstraction of visual content (rather than verisimilitude).[319] By briefly returning to the example of *Minecraft*, we can see how this approach plays out. The low visual fidelity of the game world evokes the retro aesthetic of vintage 8-bit game consoles, in which graphical elements were built by piecing pixels together one by one. The pixelated visual aesthetic of the game mirrors the primary game mechanic in *Minecraft,* which involves piecing together actual blocks of raw materials in order to build functional objects. The abstraction of visual content inherent to both visual design and gameplay encourages players to focus on their role as a builder, logically piecing together distinct combinations of blocks in order to build shelter, cook food, farm crops, and more. Player introspection is encouraged because no preset rules are established for the relationships the player chooses to foster in the game. The vast scale of the emergent worlds in *Minecraft* facilitate both completely hermetic existences and the proliferation of thriving civilizations. Just like the virtual construction of the game's terrain through PCG, learning how to play *Minecraft* is an emergent process in itself.

These procedural elements are most noticeable in the *sound* of indie games: the sound functions as an aural guide through the mechanics of the game. Yet, despite the rich history of procedural audio in gaming more broadly, questions surrounding sound and music in indie games are noticeably absent in Bogost's discussion. Indeed, the combined aesthetic and technical practices of indie game developers and sound designers remain to be theorized by music and media scholars, even as the sonic and musical aspects of such practices are understood by players and scholars as crucial elements in both procedural game design and in discussions surrounding the meanings of indie game culture. In the next section of this chapter, I align the procedural

[319] Ian Bogost, "Persuasive Games: The Proceduralist Style," *Gamasutra,* January 21, 2009, accessed December 11, 2014, http://www.gamasutra.com/view/feature/132302/persuasive_games_the_.php.

style of the indie games movement with a tradition of generative aesthetics in the late twentieth century. Charting an alternative media genealogy in the history of gaming allows me to construct a more nuanced model for understanding the interconnected sociotechnical relationships between sound design and gameplay, as well as a sound-based understanding of the historical relationship between various forms of digital art and computational media.

Situating Game Sound Within Generative Aesthetics

The previously outlined emergent aesthetics in indie games extends a long and diverse tradition of "generative" aesthetics in media art since the 1960s. Tracing this history not only allows me to suggest shared relationships between computational media platforms in the twentieth century, but also to make more explicit connections between the emergent aesthetics of musical composition and gaming. Broadly defined, generative aesthetics focus on the relinquishment of creative control in the creation of artistic content, instead privileging the ways in which aesthetic effects and values are embedded within the technical structure of computational systems themselves. In his overview of visual artists working with computers from the 1960s and 1970s, Frank Dietrich notes the ways in which many artists eschewed standard forms of human aesthetics and creativity in favor of the computer's "nonhumanness," which was seen to facilitate experiments in random image generation and algorithmic drawing.[320] Frieder Nake's *Matrix Multiplications,* for example, converts a mathematical process—a matrix successively being multiplied by itself—into an interval-based grid. Like many early experiments with algorithmically generated art, the symmetrical yet seemingly random layout of *Matrix*

[320] Frank Dietrich, "Visual Intelligence: The First Decade of Computer Art (1965-1975)," *Leonardo* 19, no. 2 (1986): 162.

Multiplications visually depicts the process of its own making. The perceived stoicism of the computer is pitted against the conventional standards of beauty imbued by the "natural" world.

Essentially, generative art is about creating processes that, in turn, create artworks. Mitchell Whitelaw writes that, in generative art, "the work is entirely shaped by the construction of its underlying system, its configuration of entities and relations."[321] In focusing on the materiality of the creative medium as an engine for the creation of art, generative artists often embed their aesthetic values within the technical structure of artworks themselves—a phenomenon shared by the use of PCG in game design, as previously discussed. For example, the work of multimedia studio and design group Universal Everything specializes in the augmentation of public space through the creation of "digital motion canvases" and dynamic "video walls" that respond and react to the physical bodies that inhabit them. In their 2014 "Infinity Room" video installation created for Microsoft, image repetition and recursion are combined with a constantly evolving visual backdrop as a way of highlighting the ubiquitous presence and fleeting nature of digital data. In the South Korean "Hyundai Vision Hall" video wall, computer generated visual noise provides abstract simulations of seemingly "natural" phenomena such as bubbling lava and crashing waves, spread across 720 microtiles on a twenty-five meter wide and four meter high wall, and surrounded by thirty-six channel surround sound. These virtual spaces exemplify the ways in which generative art is shaped by a process-oriented relationship between the design of a technical system and the feedback of its users.

The skills fostered in creating generative art and design of these sorts—logic, computer programming, algorithmic thinking—contrast strikingly with the standard "fidelity" model of

[321] Mitchell Whitelaw, "System Stories and Model Worlds: A Critical Approach to Generative Art," in *Readme 100: Temporary Software Art Factory,* edited by Olga Goriounova, 135-54 (Norderstedt: BoD, 2005), 3.

artistic design which values faithful reproduction of "real" objects and textures. Rather, concepts surrounding generative aesthetics are more closely aligned to those found in historical discussions of artificial intelligence and computer science, focusing on the construction of iterative systems that are capable of self-generating content "organically."[322] To return to the *Minecraft* example, Microsoft began testing artificial intelligence platforms in the spring of 2016 in order to "help agents sense and act" within the game.[323] The indie games movement has embraced the perceived "organicism" of generative aesthetics as an alternative to the proprietary, large-scale nature of blockbuster game development—a cultural stance shared by other forms of digital art.

In addition to its use in computational arts, generative aesthetics is found in experimental and electronic music of the mid- to late twentieth century. Deriving from compositional theories of Darmstadt composers such as Karlheinz Stockhausen and Gottfried Michael Koenig, the indeterminate music of John Cage, and the gradual processes of Steve Reich, "generative music" has described responses to compositional methods based on the intentional control of musical elements, instead focusing on self-generating compositional systems.[324] While these composers applied computational concepts and procedures to musical performance, early computer music

---

[322] Peter Cariani, "Emergence and Artificial Life," in *Artificial Life II, SFI Studies in the Sciences of Complexity,* edited by Christopher Langton et al (New York: Addison-Wesley, 1991); Christopher Langton, "Artificial Life," in *Artificial Life, SFI Studies in the Sciences of Complexity,* edited by Christopher Langton, 1-47 (New York: Addison-Wesley, 1989); Martin Gardner, "Mathematical Games: The Fantastic Combinations of John Conway's New Solitaire Game 'Life'," *Scientific American* 223 (1970): 120-23; Alan Turing, "Computing Machinery and Intelligence," *Mind: A Quarterly Review of Psychology and Philosophy* 59, no. 236 (1950): 433-60.

[323] Kelsey Atherton, "Microsoft is Training AI in Minecraft," *Popular Science,* March 14, 2016, accessed March 15, 2016, http://www.popsci.com/microsoft-is-training-ai-in-minecraft.

[324] Rene Wooller, Andrew Brown, et al, "A Framework for Comparison of Processes in Algorithmic Music Systems," in *Proceedings from Generative Arts Practice,* 109-24 (Sydney: Creativity and Cognition Studios Press, 2005).

pioneers embedded generative aesthetics within computational systems themselves. In 1957, Max Mathews wrote *MUSIC*, the first computer program capable of generating digital audio waveforms, as well as one of the first widely used programs for making music on a computer.[325]

Recall, software developer Miller Puckette wrote *Max* in the mid-1980s as an homage to Mathews, a program that by the 1990s became ubiquitous among musicians and artists seeking to create generative music in real-time. Max is based on principles rooted in the procedural nature of computer programming.[326] The work of Mathews, Puckette, and other computer musicians interested in algorithmic, generative, and "machine" processes highlights the significance of artificial intelligence (AI) and "organic" emergence in musical composition. With the rise of the "procedural style" among game designers, the core techniques of AI and emergence would also be embraced by sound designers.

Emergence as a New Model for Understanding Game Sound

Having traced seemingly disparate histories of emergent aesthetics in indie games and generative aesthetics in twentieth-century artistic movements, I propose the concept of "emergence" as an overarching framework for understanding the interconnected relationships between the design of interactive systems and the experience of human-computer interaction in sound, games, and software. The concept of emergence has often been used to talk about how the technical development and social effects of technology can be understood in non-linear ways.[327] Social

[325] Tae Hong Park, "An Interview With Max Mathews," *Computer Music Journal* 33, no. 3 (2009): 9-22.

[326] Miller Puckette, "Max at Seventeen," *Computer Music Journal* 26, no. 4 (2002): 31-43.

[327] Although he does not explicitly use the term "emergence," Raymond Williams outlines technological cause and effect as a two-way dialectic rather than a linear progression. *Television, Technology and Cultural Form* (New York and London: Routledge, 2003).

constructionist approaches like those of Bruno Latour understand the existence of sociotechnical systems as preconditions for the emergence of more complex sociotechnical systems.[328] According to Latour's "Actor-Network-Theory," both technology and society are seen as dynamic intermediaries that emerge throughout a broader history of sociotechnical evolution. For example, Sandy Stone describes the ways in which multiplicities of identities are capable of emerging at the interface between socially constructed bodies and "virtual" technologies.[329]

Among professional designers, emergence has been used as a framework for integrating the sociocultural factors of user experience with the technical affordances of specific products. Interaction designers have introduced the idea of "emergent interfaces" as a strategy in outlining a more integrated framework for feature development within a shared software environment, for example.[330] In the context of games, designer Penny Sweetser introduced the term "emergent gameplay" to describe the ways in which "interactions between objects in the game world or the players" actions result in a second order of consequence that was not planned, or perhaps even predicted, by the game developers."[331] Together, the concepts of "emergent interface" and "emergent gameplay" highlight the interconnected relationships between the *mechanics* of sound design and player *experience* in indie games. Examining the concept of emergence from the

---

[328] Bruno Latour, *Reassembling the Social: An Introduction to Actor-Network Theory* (New York: Oxford University Press, 2005); "Technology is Society Made Durable," in *A Sociology of Monsters? Essays on Power, Technology and Domination, Sociological Review Monograph*, edited by John Law, 103-31 (London: Routledge, 1991); Michel Callon and Bruno Latour, "Unscrewing the Big Leviathan: how actors macrostructure reality and how sociologists help them to do so," in *Advances in Social Theory and Methodology: Toward an Integration of Micro- and Macro-Sociologies,* edited by K.D. Knorr-Cetina and A.V. Cicourel, 277-303 (Boston, MA: Routledge and Kegan Paul, 1981).

[329] Allucquère Rosanne Stone*, The War of Desire and Technology at the Close of the Mechanical Age* (Cambridge, MA: The MIT Press, 1995).

[330] Márcio Ribeiro, Paulo Borba and Claus Brabrand, *Emergent Interfaces for Feature Modularization* (Gewerbestrasse: Springer International Publishing, 2014).

[331] Penny Sweetser, *Emergence in Games* (Boston: Course Technology, 2008), 178.

perspectives of both social theorists and professional designers foregrounds the ways in which technical systems co-emerge with social structures and values, allowing us to consider the integrated and dynamic relationship between society, culture, and technology.

By expressing generative aesthetics through the technical structure of the interactive system, then, artists, musicians, and designers conceptualize gaming as an integrated media experience. In short, generative *aesthetics* and emergent *gameplay* coalesce to form what I call a "procedural interface." In chapter two, I introduced the concept of procedural listening to describe the practice in which computer users focus their attention on the process-oriented mechanics at play in the inner workings of software itself, rather than the content or narrative being enacted by the interaction. While the term "interface" most commonly evokes the GUI of computer operating systems, or more generally the control surface of a given object, I use the term procedural interface to describe the overall experience enacted by the sociotechnical system comprising the media platform, including the values and aesthetics designed into its technical infrastructures.[332] Procedural interfaces provide bridges between the *mechanics* of rule-based systems and the *aesthetic* experience of the musician or player in response to those systems.

When considering interfaces as entire sociotechnical systems, we can conceptualize the ways in which both the technical structure and the user's experience of the software emerge in a seemingly spontaneous manner. As the user interacts with a procedural interface, the mechanics of the system are gradually made apparent to the interactor, and it is this decoding of the system's "rules" that becomes the primary function of the media experience. This concentrated

[332] This idea expands on the work of media scholars working in the subdisciplines of interface studies and postdigital aesthetics. David M. Berry and Michael Dieter, eds., *Postdigital Aesthetics: Art, Computation and Design* (New York: Palgrave Macmillan, 2015); Branden Hookway, *Interface* (Cambridge and London: The MIT Press, 2014); Christian Ulrik Andersen and Søren Bro Pold, eds., *Interface Criticism: Aesthetics Beyond Buttons* (Aarhus: Aarhus University Press, 2011).

attention to the dynamics of technological interaction is apparent in a range of media arts, including computer-generated visual art, generative music, and indie games, as previously described. In each case, the morphology of digital sound, modeled from the seemingly “organic” principles of artificial intelligence and real-time synthesis, functions as a sensory guide in the experience of emergence. The idea of procedural interfaces pushes beyond existing conceptions of generative aesthetics and emergent gameplay by focusing on the space in-between process (gameplay) and product (game), foregrounding the fundamental ways in which a user comes to know, interact with, and embody a sociotechnical system through aesthetic ludic practices.

Fract OSC **as Procedural Interface**

Fract OSC, discussed earlier, provides a paradigmatic example of such a procedural interface. The process of foregrounding the mechanics of sonic interaction is most apparent in *Fract’s* “Studio Mode.” As the player enters a small portal-like enclosure near the entrance of the main game world, the dark landscape of *Fract* is brightened by a transparent panel containing a colorful grid. Although there are no textual instructions or visual cues on how to proceed, players familiar with electronic music composition recognize the abstract knobs and virtual buttons as simulations of graphical interface elements typical of digital audio workstation software design (figs. 5.4 and 5.5). As the player tinkers with the transparent neon LCD display in front of them, sounds gradually emerge, morph, and fade away. *Fract* is not simply a *game* in which players explore the inner structures of a synthesizer, it is also a compositional *environment* for creating original music. Following musician and sound artist Norbert Herber, the game may be conceptualized as a “composition-instrument… a work that can play and be played simultaneously,” resulting in a form of emergent gameplay that represents “a kind of becoming where myriad events collide to unfold as experience in the course of their play."[333] Here, the techniques and processes of *audio* production—rather than 2D or 3D visual cues—guide the fundamental mechanics of gameplay.

Figure 5.4: Fract OSC 'Studio' sequencer. At the top of the screen, the player sequences individual notes of a synthesizer along a one bar grid divided into sixteenth notes. At the bottom, the player programs the resulting musical patterns into a song.

[333] Norbert Herber, "The Composition-Instrument: Emergence, Improvisation, and Interaction in Games and New Media," in *From Pac-Man to Pop Music: Interactive Audio in Games and New Media,* edited by Karen Collins (Burlington, VT: Ashgate, 2008), 104.

Figure 5.5: Fract OSC 'Studio' editor. After sequencing and programming synthesizer patterns, the player can edit parameters of the instrument, including volume, tone, and timbre.

"Studio Mode" is an apt component of what makes *Fract* a procedural interface, in that it is an *aesthetic* element of gameplay that reveals the *technical* structure of the game's sound engine. The *Fract* sound engine is a fundamental mediator in the creation of emergent gameplay, as it is both designed using principles of PCG and shaped by the compositional choices of the player. As players tinker with music composition and audio synthesis in "Studio Mode," they simultaneously set the sound engine in motion and they are made aware of its technical structure. Like many generative art forms, *Fract* is self-referential and celebratory in regards to the processes of its own construction.

Developed by programmer Henk Boom, the *Fract* sound engine is constructed using the generative music and synthesis capabilities of *Pure Data (Pd)*—an open-source instance of the popular *Max* visual programming 'environment', initially developed by Puckette, as mentioned above.[334] Similar to the cultural position of indie games themselves, *Pd* is the epitome of alternative software for audio production. Puckette originally designed the program to liberate musicians from what he perceived to be creatively limiting features of commercial digital audio workstations, as I discussed in chapter two. Most commonly, audio production software conceptualizes sound as recorded material that is temporally mapped onto crucial points of what is essentially a linear narrative structure.[335] In contrast, *Pd* encourages the composition of generative sound by allowing the designer to relate digital sound processing functions ("objects") to one another using virtual patch cords, resulting in process-oriented musical composition systems that respond to user input in real-time (fig. 5.6).

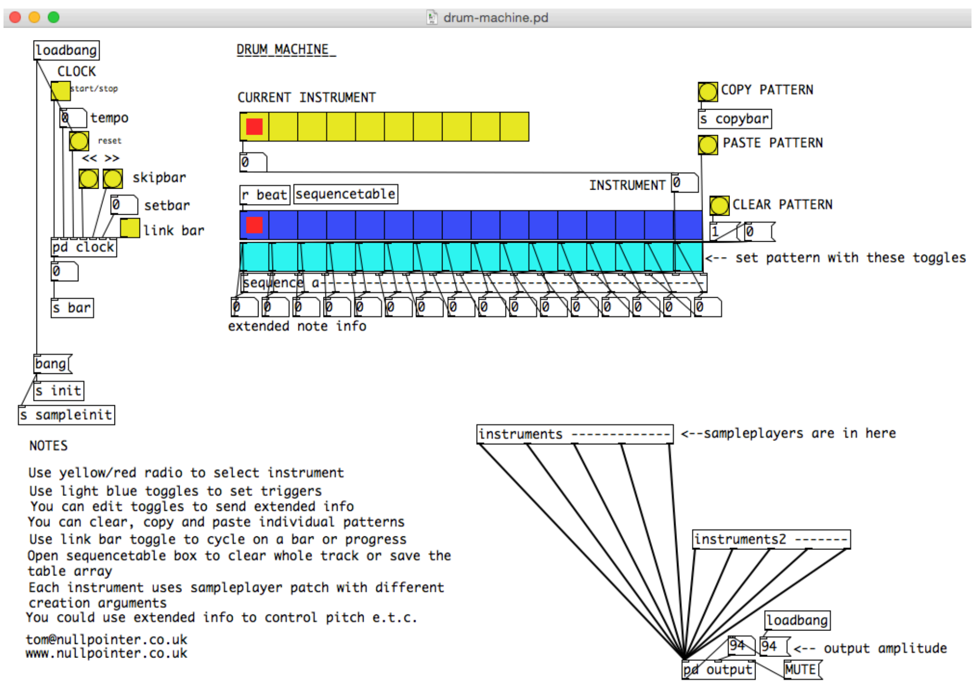


Figure 5.6: "PD Drum Machine" by Nullpointer, in the style of a Roland TR-808 (Source: Nullpointer, accessed May 1, 2016, http://www.nullpointer.co.uk/content/pure-data-patches/).

[334] Bryan W.C. Chung, *Multimedia Programming With Pure Data* (Birmingham: Packt Publishing, 2013); Eric Lyon, *Designing Audio Objects for Max/MSP and Pd* (Middleton, WI: A-R Editions, 2012).

[335] David Bessell, "Blurring Boundaries: Trends and Implications in Audio Production Software Developments," in *The Oxford Handbook of Interactive Audio,* edited by Karen Collins, Bill Kapralos, and Holly Tessler (New York: Oxford University Press, 2013).

The "object-oriented" nature of *Pd* is highlighted by the fact that it is a "visual programming language" that allows users to create programs by manipulating graphical elements of the interface, rather than using text-based code to specify commands. Since the development of the first computer languages in the 1970s, one of the main goals of visual programming has been to make the practice of coding easier to learn.[336] *Scratch,* the most popular visual programming language, is designed specifically to teach children how to code, organizing graphical elements as colorful blocks that fit together like Lego pieces. Echoing the core elements of all procedural interfaces, visual programming languages such as *Pd* and *Scratch* make apparent to the user the rules of the software through graphical and audio cues. In a similar way, indie games such as *Minecraft* and *Fract OSC* employ visual programming principles in the structure of both their technical design (using a visual programming language to build the software's sound engine) and gameplay (the player arranges graphical elements to build the virtual world as he or she experiences it). The resulting interface is procedural in that both the technical code and aesthetic audiovisual content are procedurally generated in real-time.

Boom integrates the modular sound engine of *Fract* with the 3D, generative visual aesthetic of the game. In order to combine disparate sound and game software, he uses *libpd*—a collection of resources that turns *Pd* into an embeddable audio synthesis library. While *Pd* is a stand-alone software program that runs in a desktop computer environment, *libpd* allows programmers to embed the work they create with *Pd* into other software. Most commonly, the library has been used by iOS developers looking to integrate the generative music capabilities of *Pd* with mobile app development software. For example, the developers of the 2008 iPhone app

[336] Rémi, "The Maturity of Visual Programming," *craft ai,* September 29, 2015, accessed April 11, 2016, http://www.craft.ai/blog/the-maturity-of-visual-programming/.

RJDJ used libpd to create what they referred to as "reactive music" programs that manipulated pre-existing musical content in various ways based on real-time user interaction. The default *RJDJ* music player receives ambient noise from the listener's acoustic environment, procedurally remixing and distorting the content as the iPhone user goes about their day.[337] The proliferation of generative music apps parallels the rise of indie games such as *Fract OSC* and *Proteus,* aligning the vernacular creativity imbued by mobile media (discussed in the previous chapter) with the ludic affordances of procedural interfaces.

Specifically, Boom uses libpd to integrate *Pd* with *Unity3D*, a game engine especially popular among indie developers working with 3D virtual worlds. Released in 2005, just a couple years before the launch of the first iPhone, *Unity* prides itself on its interoperability across media platforms (fig. 5.7). As the name suggests*, Unity* is a cross-platform engine used to develop games for a range of media, including personal computers, gaming consoles, mobile devices, and websites. Examples of games created using *Unity* include iOS puzzle apps *Prune* and *The Room Three,* open world console games *Firewatch* and *Rust,* as well as virtual reality "experiences" *Job Simulator* and *Tea Party Simulator 2015.* These games share core elements of the previously outlined "procedural style" in indie games, including an orientation towards process and introspection, a foregrounding of game rules and mechanics, and an abstraction of visual content (rather than verisimilitude). The generative audio capabilities of *Pd* and the adaptability of *Unity* combine to form the procedural *interface* of *Fract* as a whole—a game which audiovisual content, as well as technical infrastructure, is defined by a real-time, process-oriented dialogue between player and system.

[337] Jason Kincaid, "RJDJ Generates an Awesome, Trippy Soundtrack for Your Life," *TechCrunch,* October 13, 2008, accessed April 11, 2016, http://techcrunch.com/2008/10/13/rjdj-generates-an-awesome-trippy-soundtrack-for-your-life/.

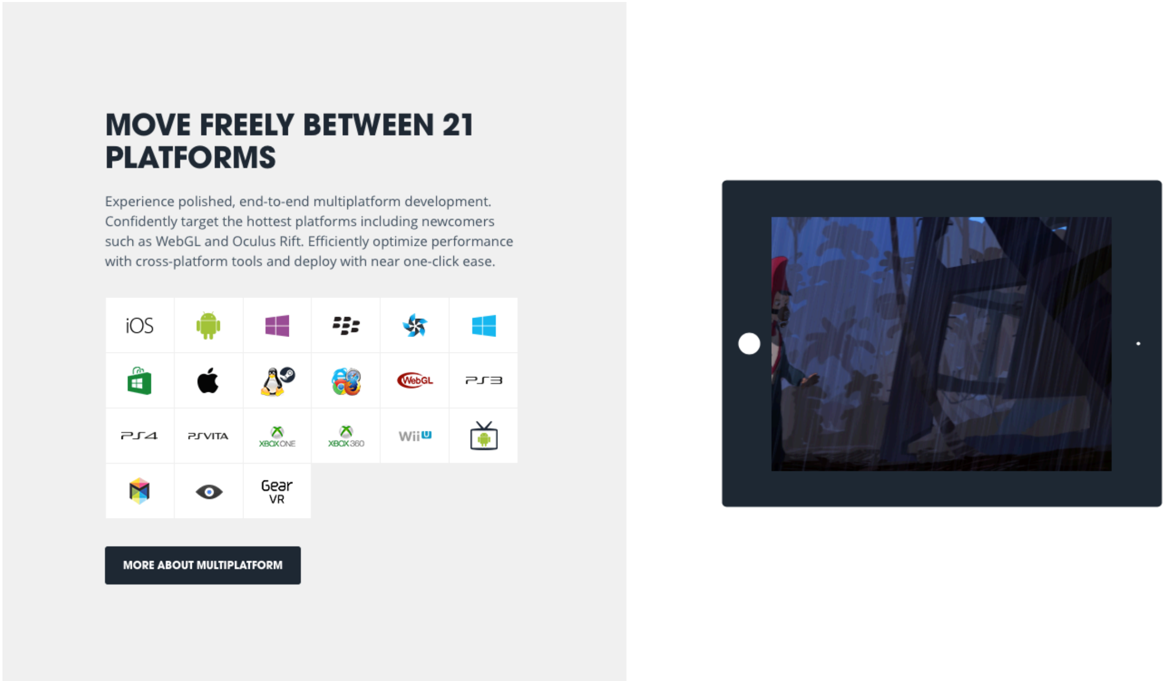


Figure 5.7: Promotional material highlighting the interoperability of *Unity3D* across twenty-one media platforms (Source: Unity3D, "Multiplatform," accessed May 2, 2016, [https://unity3d.com/unity/multiplatform)](https://unity3d.com/unity/multiplatform).

Similar to the object-oriented interface of *Pd*, *Unity* is a visual programming environment that has become popular among developers for the efficiency with which one can make simple 3D game objects perform complex procedures. Game assets of various sorts are easily dragged and dropped into a 3D space, affording the user a "plugin" style modularity of layout similar to digital audio workstations such as Ableton *Live*, as described in chapter one (fig. 5.8). For example, a virtual basketball object could be dropped into the 3D editor, which would consequently display a 3D basketball in the window. Then, a physics simulator could be dropped onto the basketball object in the editor, causing the ball to move in a specific way. Finally, a sound sample of a bouncing basketball could be dropped onto the moving ball, which would cause the object to playback the sample every time it made contact with a solid surface. The instant feedback of the drag-and-drop editor thus conveys to the designer the impression of direct manipulation—the feeling that he or she is directly interacting with the virtual world itself, rather than a programming software. In building the iOS puzzle game *Monument Valley,* lead developer Peter Pashley said he knew *Unity* would be the best software possible because the game "is all about creating beautiful places, and for that you need artists to be able to work *directly with the world*."[338] Just as the player of *Fract OSC* is placed in the position of audio designer, the software used to build the game positions the game designer in the role of a player, embracing ludic mechanics in an accessible manner to build the virtual environment.

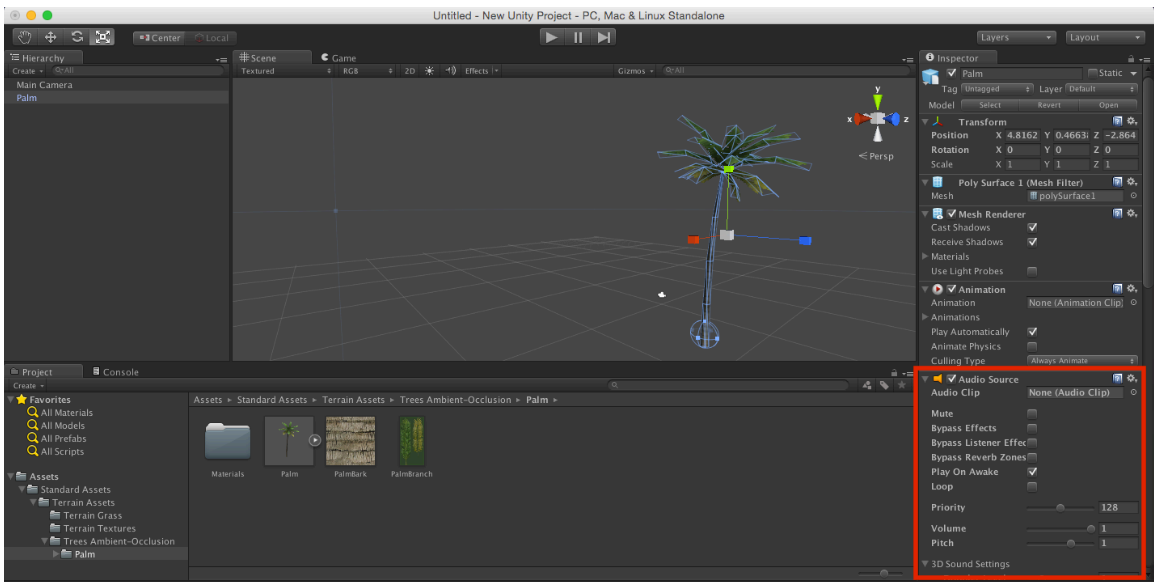


Figure 5.8: *Unity3D* "Scene" view. A 3D-modeled palm tree is dragged from the "Asset" library at the bottom-center of the screen. In the sidebar to the right, the user can manipulate parameters of the tree, including the sounds the reverberate from it as the player approaches (see highlighted box)

The feeling of direct manipulation imbued by the visual programming environment not only makes visible to the user the "rules" of the programming language itself, but also creates a more playful relationship between the abstract nature of coding and the aesthetic experience of

[338] Quoted in "Monu/Mental," *Unity3D,* January 23, 2014, accessed April 11, 2016, https://unity3d.com/showcase/case-stories/monument-valley. Emphasis added.

gameplay. Game developer John Alvarado describes *Unity* as a “component-based game object system” in which the user can attach coded functions to any game asset, including inanimate materials, avatars, and sound. According to Alvarado, “It’s real easy to add code components to any object you create in the game, whether it be a box you just made or an animated character. It’s a very modular, object-oriented way of adding functionality to an object in the game.”[339] In the same way that audio-visual characteristics and rules for interaction can be dragged-and-dropped onto game assets through *Unity’s* 3D editor, so too can computer scripts be coded into the objects.

Let us return to the basketball example. The designer is not limited to the 3D drag-and-drop editor in terms of the types of relationships being created between the basketball and other game objects. If the designer wants to create an automated process in which the basketball bounces ten times every time it is dropped on a solid surface, he or she can code original rules for interaction using *MonoDevelop*—an integrated development environment (IDE) supplied with *Unity* that functions like most text editors for computer coding. The unique aspect of *MonoDevelop* is that it allows the designer to create custom behaviors for game assets that can be accessed from within the *Unity* editor. In this way, the abstraction of text-based computer code becomes externalized and made playable in the visual programming environment of *Unity*. While the editor itself may appear to be object-oriented in its layout, the primary goal of procedural interfaces in game design is to make explicit and tangible the previously implicit processes hidden behind software algorithms.

[339] Jon Brodkin, “How Unity3D Became a Game-Development Beast,” *Dice,* June 3, 2013, accessed November 18, 2014, http://news.dice.com/2013/06/03/how-unity3d-become-a-game-development-beast/.

The process-oriented nature of the *Unity* editor is amplified by its capacity for rapid prototyping. In the same way that the editor blurs the line between visual programming and text-based coding, so too does the software blur the line between design and gameplay. As such, the creative workflow in *Unity* echoes the practices of musicians working with software such as *Live* (chapter one) and *Max* (chapter two), both of which allow the creator varying degrees of rapid prototyping and modularity of interface layout.

As the interactive objects and rules of the virtual world are being constructed, the developer can switch between "Scene" (edit) and "Game" (play) modes, effectively alternating between the act of graphical programming and the experience of the code as it would appear to the player. The designer can thus witness the effects of his or her programming choices in real-time, without having to compile and run the game as a process separate from *Unity.* Mathieu Girard, CEO of Amplitude Studios, aligns the usability of visual programming languages such as *Scratch* with that of *Unity,* claiming that the greatest part of *Unity* is its workflow, and "how easily you can create, edit, and integrate data and code. Building inspectors, editors... everything is *child's play*."[340] This ability to rapidly prototype the game world in real-time is an experience shared by both designer and player, further highlighting the integrated technical and aesthetic affordances of *Fract* as a procedural interface. In effect, the design process becomes as experimental and improvisational as the experience of the game itself.

The integration of emergent gameplay mechanics through the use of scripts in *Unity* and generative music techniques through the incorporation of *Pd* distinguishes the sound design of *Fract* from traditional examples of procedural game sound. While procedural sound generation

[340] Quoted in "Rural Legend," *Unity3D,* April 25, 2014, accessed April 11, 2016, https://unity3d.com/showcase/case-stories/endless-legend. Emphasis added.

in games is often used to create musical variety or avoid bogging down the game's processor, as I described earlier in this chapter, the sounds being produced as a result of player input in *Fract* are synthesized in real-time by the game's sound engine (fig. 5.9). In this way, the purpose of the sound engine aligns with the purpose of the game itself: to rebuild a defunct synthesizer by piecing together its core components. Indeed, while a prerecorded soundtrack provides a sonic backdrop to many crucial moments in the simplistic narrative of *Fract*, sonic interaction design in the game is defined for the most part by a generative soundscape that emerges in the process of the player's dialogue with a constantly evolving audiovisual interface.

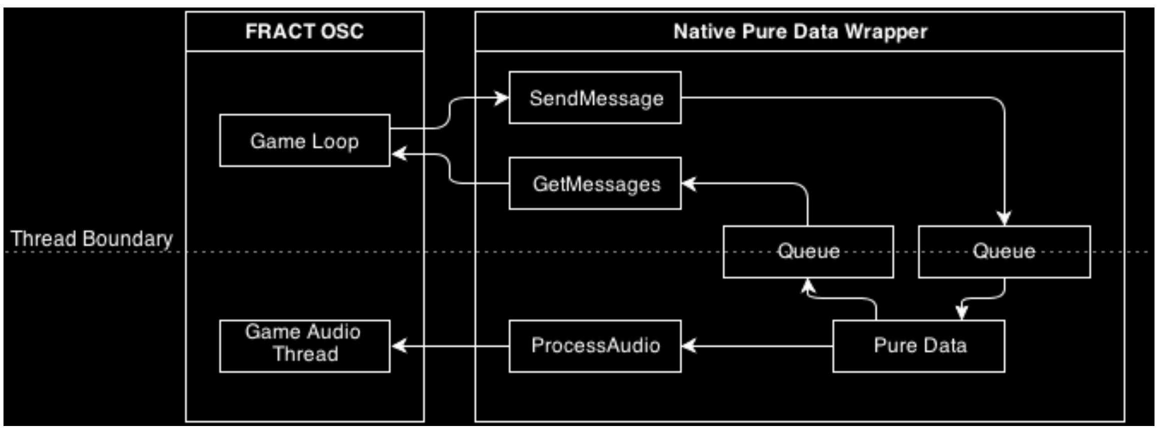


Figure 5.9: ***Fract OSC*** **to** ***Pd*** **audio workflow, created by the game developers. Sounds emerge from a feedback loop between player interactions in the game (displayed on the left) and audio processing capabilities in** ***Pd*** **(displayed on the right) (Source: FRACTgame, Flickr, June 4, 2013)**

In this chapter, I have traced the ways in which larger histories of generative aesthetics have been instrumental to contemporary practices and interests in indie gaming development. I have suggested the concept of procedural interface as a model for understanding key instances in which sound and musical design are closely integrated with game design in early twenty-first century indie games. Games like *Fract OSC,* I further argue, provide a prototype for sound designers seeking to enhance the procedural and generative aspects of gameplay. Here, the mechanics of both audio-visual production and play are revealed to the user through iterations in which the ludic material or meaning of the game itself becomes playable. In doing so, *Fract* reveals the increasing interconnections between the technical design, aesthetic experience, and contemporary discourses of play surrounding digital media not simply in indie games but, I note, across a larger range of artistic and professional contexts.

Examining the contemporary gaming experience as a procedural interface comprising both emergent gameplay mechanics and generative sound design provides a useful bridge between player- and developer-oriented perspectives in game and music scholarship. By building emergent aesthetics into both the technical structure of a game and the subjective experience of the player, it is possible to view the gaming experience as a sociotechnical system. This framework allows us to comprehend more distributed forms of agency in the experience of gameplay, dethroning the sound designer or composer as the sole factor in the creation of game sound. As examples of what Herber calls "composition-instruments," games such as *Fract OSC* position the player simultaneously in the roles of composer, designer, and experiential observer. Further, the close relationship between sound and image fostered by emergent aesthetics encourages sound designers and composers to broaden their knowledge of design and production techniques outside of strictly "musical" practices. As I have suggested throughout this dissertation, the appearance of new interfaces for human-computer interaction always occurs in parallel with new conceptions of fundamental ideas regarding art and culture in the digital age.

The procedural interfaces of indie games are just one set of examples that highlights the ways in which the artistic tools are becoming more abstract and complex, their design affordances and possible uses underdetermined and hence translatable across media formats and sites of technological production. As such, game developers and sound designers have begun to borrow software popular among computer musicians to facilitate emergent gameplay, just as

DJs, VJs, and electronic musicians have incorporated software such as *Unity* in an effort to enhance the process-oriented aspects of their performances. This network of sociotechnical practices presents game and media scholars with both useful conceptual tools in understanding the convergent nature of interactive media, as well as a concrete field of relevant sociotechnical vernacular and implicit theories at work in current practices of media production. Contemporary software for sound design and gaming continues to transform the creative possibilities of artists working within a larger historical trajectory of generative aesthetics and sociotechnical emergence. As game designers and digital musicians continue to explore the worlds of sound at the interface of culture and technology, these latent histories are manifesting in much broader areas of cultural production.

Conclusion

Interface Futures

In this dissertation, I have considered the ways in which music-making and performance have been influenced by the design of music production software. I have discussed the ways in which interface design reflects broader cultural aesthetics; I have examined how the affordances and constraints of software are materialized in live performance with technology and everyday embodied interactions with hardware; and I have explored the ways in which emergent aesthetics of sound in interactive media might suggest a more dynamic relationship between the technical aspects of design and the aesthetics of user experience. Across the case studies presented, I have defined procedural listening as a dynamic, two-way process of human-computer interaction (HCI) in which technology users focus on the structures at work in the system with which they are engaging, rather than the content created. But I have not yet addressed the social implications of procedural listening as a general method of HCI outside of creative communities of practice. In the concluding pages of this project, I will outline some broader reflections on procedural listening, suggest the benefits of considering interface aesthetics as a critical method for researchers and practitioners, and offer directions for future research.

Having established a shared relationship between the design and use of software interfaces, it is possible to examine the practice of procedural listening in a new light. Indeed, negotiating new tools into digital media production practices is not simply about reconfiguring existing technologies and cultural aesthetics. Interfaces are more than just the material metaphors and graphical user interfaces (GUIs) that define our practical interactions with computers; they are also a defining force in the emergence of a broader collective social and cultural consciousness in the first decades of the twenty-first century. Interfaces teach us how to relate to technology, culture, and society; interfaces shape, and are shaped by, cultural forms in an interactive way that both expands on and subverts previous practices; and interfaces shape the physical and gestural comportments with which we interact with material technologies and objects in the world. Procedural listening—the process of learning how to listen to and interact with music and media in an experimental, speculative, and emergent manner—is the result of these shared affordances that software has introduced. When I first started researching the electronic music and interactive media cultures of Southern California, I had one simple question: what is sound after software? Procedural listening is one of the many possible answers to this question.

Procedural listening offers three unique contributions to existing understandings of HCI: first, it encourages users to critically and actively reflect on their relationships with software; second, it contradicts dominant technological tropes of intentionality, mastery, and control; and third, it inspires an understanding of digital literacy as a lifelong process of experiential learning and discovery. The commercial technology industry of Silicon Valley is driven by an "upgrade" mentality in which constant innovation and technological change is valued above all else. As a result, users of technology anxiously embrace the maximalist and consumerist aesthetics described in chapters one and four. By focusing the user's attention on the unique design affordances of the device, as well as the interface's capacity for experimental play, procedural listening encourages a more measured approach to music and media consumption.[341] Further,

[341] In this way, procedural listening may be viewed as an expanded form of what Pauline Oliveros calls "deep listening," a practice that is intended "to heighten and expand consciousness of sound in as many dimensions of awareness and attentional dynamics as humanly possible." Pauline Oliveros, *Deep Listening: A Composer's Sound Practice* (Lincoln, NE: iUniverse, 2005).

procedural listening shifts the focus of the design process from the fixed *objects* sold on the commercial market to the *process-oriented* needs and desires of individual users and creative communities. In the end, this relational understanding of interface will provide both designers and users a stronger sense of purpose in their interactions with technology.

In the same way that procedural listening encourages critical opposition to the upgrade culture of Silicon Valley, the process also contradicts dominant technological tropes of intentionality, mastery, and control.[342] These tropes are apparent not only in cultural practices, as in the idea of compositional intent and mastery of technique described in chapters two and five, but also in the driving technological force behind global capitalism—the "Military-Entertainment Complex" (MEC). The MEC is defined by the close cooperation and sharing of resources between video game designers and the military; between movie producers and the United States government; and between military propaganda and its realization in the entertainment industry more broadly.[343] Interfaces play a primary role in the shaping of the MEC, as evidenced by the use of video game controllers in the piloting of military attack drones, the use of virtual reality technologies in soldier training, and the longer historical convergence between the Internet and wartime surveillance networks. By encouraging shared agency between computers and their users, procedural listening suggests an ethical paradigm of HCI that is sensitive to the context and implications of technological use, rather than a simple tool-based, one-way relationship in which a user imposes commands on the technology.

---

[342] Robert Romanyshyn suggests our experience of technology has become a dream "of domination, mastery, and control of nature." *Technology as Symptom and Dream* (New York: Routledge, 1989), 211.

[343] Stephen Stockwell and Adam Muir, "The Military-Entertainment Complex: A New Facet of Information Warfare," *The Fibreculture Journal* 1 (2003), accessed April 26, 2016, http://one.fibreculturejournal.org/fcj-004-the-military-entertainment-complex-a-new-facet-of-information-warfare.

In addition to providing forms of HCI that suggest alternatives to dominant technological and ideological structures of global capitalism, procedural listening inspires an understanding of digital literacy as a lifelong process of experiential learning and discovery. Ray Kurzweil, inventor and writer who has worked across the domains of music, philosophy, and computer science, once said that biology itself is a software process, claiming that humans are constantly "walking around with outdated software running in our bodies, which evolved in a very different era."[344] As computers users continually update the operating systems on their laptops, or upgrade the smartphones in their pockets, they learn anew all the physical gestures and cognitive shifts required of computational systems. Following the algorithmic flow, ephemerality, and fluidity of software itself, procedural listening encourages computer users to embrace uncertainty and foster perennially shifting understandings of the world. Despite the apparent complexity in the software program as a whole, procedural listeners understand that the moment of digital creation begins with a single line of code.

While this dissertation addresses very specific digital cultures in Southern California as they navigate new technologies, I consider these conclusions to be critical lessons for researchers and practitioners involved in sound, design, and interactive media more broadly. The hermeneutic analysis of interface design invites music scholars to understand software as a crucial platform in which conceptions of musicality, instrumentality, and performance are continually being negotiated. More fundamentally, the dissertation introduces design as a core element in the study of musical cultures. For media scholars, *Interface Aesthetics* advances an understanding of sound as a central feature in the constantly expanding ecology of digital media.

---

[344] Brian Wang, "Ray Kurzweil Predicts Radical Life Extension and blood cell sized supercomputers in 25 years," *Next Big Future*, October 17, 2013, http://nextbigfuture.com/2013/10/ray-kurzweil-predicts-radical-life.html.

Further, the practice-oriented ethnographic methods present throughout the project broaden the intellectual scope and topical reach of media theory subfields such as software studies and interface studies.

For designers, the project introduces a critical framework with which to question long-standing practices and assumptions in the technology industry. For example, despite cogent arguments against the use of design metaphors from scholar-practitioners Brenda Laurel, Janet Murray, and others, the practice of designing interface elements as virtual representations of physical objects remains ever-present amongst professional designers. By following the experimental and abstract design examples provided throughout this dissertation, designers may, in turn, challenge users to think outside the box in their interactions with technology. Further, user research methods amongst designers may be enhanced by embracing the "thick description" and context-oriented ethnographic methods of anthropologists and ethnomusicologists.[345] While some practitioners may argue that intellectual critique has no place in the design process, it is precisely the balance of theory and practice that provides this dissertation with its most critical edge.

My main goal in writing this dissertation was to provide a critical and analytical model with which to understand both my own experiences as a digital media artist working with sound, as well as the shifting landscape of media in the age of software. Having established the practice-oriented theoretical and methodological frameworks present throughout the project, it is now possible to consider briefly worthwhile paths for future research. Following my previous discussion of the MEC, the role of software in global surveillance systems would be a

[345] Clifford Geertz, "Thick Description: Toward an Interpretative Theory of Culture," in *The Interpretation of Cultures* (New York: Basic Books, 1973).

particularly significant topic to pursue. "Wearable" devices and motion capture technologies represent arguably the most dominant trends in interface design and—as a result of their relatively recent emergence—the cultural meanings and practical uses of these technologies are still up for grabs. As such, research in global surveillance systems could provide a political and ethical angle to the discussions about tacit knowledge and embodiment present in chapters three and four. In addition, the practice-oriented methods demonstrated throughout *Interface Aesthetics* could contribute to emerging literature on race and gender dynamics in digital production cultures.[346] Just as the software boom of the early 2000s has supposedly "democratized" technological design and use, it has also sparked renewed discussions about the gender gap in the technology industry.[347] By combining ethnographic work in the technology industry with the analytical study of the software that drives the industry, future research in interface aesthetics could generate critical discussions amongst designers, users, and intellectuals on the topics of labor, ethics, and social justice.

It is impossible to anticipate the path technology will follow in the coming years. In 1977, the founder of the Digital Equipment Corporation told the attendees of the World Future

---

[346] Tara McPherson, "Why Are the Digital Humanities So White? or Thinking the Histories of Race and Computation," in *Debates in the Digital Humanities,* edited by Matthew K. Gold (Minneapolis: University of Minnesota Press, 2012); Tara Rodgers, "Towards a Feminist Historiography of Electronic Music," in *The Sound Studies Reader,* edited by Jonathan Sterne, 475-89 (New York and London: Routledge, 2012); Mara Mills, "Do Signals Have Politics? Inscribing Abilities in Cochlear Implants," in *The Oxford Handbook of Sound Studies*, edited by Trevor Pinch and Karin Bijsterveld, 320-46 (New York: Oxford University Press, 2011).

[347] Susan Wojcicki, "Closing the Tech Industry Gender Gap," *Huffington Post,* January 27, 2016, accessed April 26, 2016, http://www.huffingtonpost.com/susan-wojcicki/tech-industry-gender-gap_b_9089472.html; Lauren Gilmore, "The gender gap in technology can byte me," *The Next Web,* December 1, 2015, accessed April 26, 2016, http://thenextweb.com/insider/2015/12/01/the-gender-gap-in-technology-can-byte-me/#gref; Catherine Bracy, "Closing The Tech Industry's Gender Gap Requires Better Data," *NPR,* June 25, 2013, accessed April 26, 2016, http://www.npr.org/sections/alltechconsidered/2013/06/24/195144754/closing-the-tech-industry-s-gender-gap-requires-better-data.

Society that “there is no reason for any individual to have a computer in his home.”[348] Twenty years later, the Chief Technology Officer of Microsoft claimed that “Apple is already dead.”[349] Indeed, digital media cultures are often as fleeting and short-lived as software itself, and this makes the scholarly study of these communities all the more urgent. Yet, these failed predictions about our technological future only serve to support my conception of software interaction as an experimental, process-oriented endeavor in which existing conceptions of creativity, productivity, and sociocultural identity are constantly questioned. In the process of software’s eternal becoming, the screens in which we dedicate so much of our physical and cognitive capacities act as both windows through which we view the world, and mirrors that depict the evolving interface between our personal values, identities, and aesthetics.

---

[348] Ken Olsen, quoted in Jonathan Gatlin, *Bill Gates: The Path to the Future* (New York: Harper Collins, 1999), 39.

[349] Rob Beschizza, “The 15 Dumbest Apple Predictions of All Time,” *Wired,* November 1, 2007, accessed April 26, 2016, http://www.wired.com/2007/11/analysts-dont-k/.

Bibliography

"1984 Apple's Macintosh Commercial (HD)." *Mac History,* uploaded February 1, 2012. Accessed April 30, 2016, https://www.youtube.com/watch?v=VtvjbmoDx-I.

"25 Products That Changed Recording." *Sound on Sound,* November 2010. Accessed April 25, 2016, http://www.soundonsound.com/sos/nov10/articles/25-milestone-products.htm.

"35,000 visitors at the Paris Maker Faire, a record." *Makery,* May 4, 2015. Accessed April 29, 2016, http://www.makery.info/en/2015/05/04/35000-visiteurs-a-la-maker-faire-paris-un-record/.

Abbate, Carolyn. "Music—Drastic or Gnostic." *Critical Inquiry* 30, no. 3 (2004): 505-536.

Ableton. "Ableton Artist Quotes." N.p., n.d. Accessed April 20, 2015. https://www.ableton.com/en/pages/artists/artist_quotes/.

"Ableton Live Audio & Loop Sequencing Software For Mac and Windows." *Beat Mode,* n.d. Accessed April 25, 2016, http://www.beatmode.com/historical/ableton-live/.

Adiseshiah, Emily Grace. "Iterative prototyping to improve the design process." *Justinmind,* January 14, 2016. Accessed April 12, 2016. http://www.justinmind.com/blog/iterative-prototyping-to-improve-the-design-process/.

Adorno, Theodor W. *The Culture Industry: Selected Essays on Mass Culture.* New York and London: Routledge, 1991.

Akrich, Madeleine. "The De-Scription of Technical Objects." In *Shaping Technology/ Building Society: Studies in Sociotechnical Change,* edited by Wiebe E. Bijker and John Law. Cambrige and London: The MIT Press, 1992.

Allan, Roy A. *A History of the Personal Computer: The People and the Technology.* Allan Publications, 2001. Accessed April 30, 2016, https://archive.org/details/A_History_of_the_Personal_Computer.

Andersen, Christian Ulrik and Søren Bro Pold. *Interface Criticism: Aesthetics Beyond Buttons*. Aarhus: Aarhus University Press, 2011.

"Angélica Negrón: Toys Noise." *Ableton,* January 13, 2012. Accessed April 20, 2015. https://www.ableton.com/en/blog/angelica-negron/.

Apple. "About Multitasking on your iPhone, iPad, and iPod touch." n.d. Accessed April 30, 2016, https://support.apple.com/en-us/HT202070.

________. "Creativity Apps: iOS." Accessed March 23, 2016, http://www.apple.com/si/creativity-apps/ios/.

________. "Designing for iOS." In *iOS Human Interface Guidelines.* Cupertino, CA: Apple, 2014). Accessed April 17, 2016. https://developer.apple.com/library/iOS/documentation/userexperience/conceptual/mobilehig/.

________. "Use Multi-Touch gestures on your Mac." n.d. Accessed April 30, 2016, https://support.apple.com/en-us/HT204895.

Arzoomanian, Rich. "A Complete History of Mainframe Computing." *Tom's Hardware,* June 26, 2009. Accessed April 30, 2016, http://www.tomshardware.com/picturestory/508-3-mainframe-computer-history.html.

Atherton, Kelsey. "Microsoft is Training AI in Minecraft." *Popular Science,* March 14, 2016. Accessed March 15, 2016, http://www.popsci.com/microsoft-is-training-ai-in-minecraft.

Auslander, Philip. *Liveness: Performance in a Mediatized Culture.* New York and London: Routledge, 2008.

Baines, Anthony and Klaus Wachsmann. "Classification of Musical Instruments." *Galpin Society Journal* 14 (1961): 3-29.

Barr, Pippin, James Noble, and Robert Biddle. "Video Game Values: Human-Computer Interaction and Games." *Interacting With Computers* 19 (2007): 180-195.

Barrett, Brian. "End of an Era: Panasonic Kills Off Technics Turntables." *Gizmodo,* October 28, 2010. Accessed April 18, 2016, http://gizmodo.com/5675818/end-of-an-era-panasonic-kills-off-technics-turntables.

Bates, Eliot. "The Social Life of Musical Instruments." *Ethnomusicology* 54, no. 3 (2012): 363-395.

________. "What Studios Do." *Journal on the Art of Record Production* 7 (2012). Accessed April 20, 2015. http://arpjournal.com/what-studios-do/.

Baudouin, Oliver. "A reconstruction of *Stria.*" *Computer Music Journal* 31, no. 3 (2007): 75-81.

Belshaw, Doug. *The Essential Elements of Digital Literacies.* Available from http://dougbelshaw.com/ebooks/digilit/ under a CC BY license (2011).

Bench, Harmony. "Gestural Choreographies: Embodied Disciplines and Digital Media." In *The Oxford Handbook of Mobile Music Studies, Volume 2,* edited by Sumanth Gopinath and Jason Stanyek. New York: Oxford University Press, 2014.

"Benga makes a Grime beat on Playstation music 2000 - BBC 2003." GetDarker, uploaded December 5, 2014. Accessed March 31, 2016. https://www.youtube.com/watch?v=AZHnaSIZP0Y.

Bergstrom, Ilias and R. Beau Lotto. "Code Bending: A New Creative Coding Practice." *Leonardo* 48, no. 1 (2015).

Bernstein, David W. *The San Francisco Tape Music Center: 1960s Counterculture and the Avant-Garde.* Berkeley and Los Angeles: University of California Press, 2008.

Berry, David M. and Michael Dieter, eds. *Postdigital Aesthetics: Art, Computation and Design*. New York: Palgrave Macmillan, 2015.

Berry, David. *The Philosophy of Software: Code and Mediation in the Digital Age.* New York: Palgrave Macmillan, 2011.

Beschizza, Rob. "The 15 Dumbest Apple Predictions of All Time." *Wired,* November 1, 2007. Accessed April 26, 2016, http://www.wired.com/2007/11/analysts-dont-k/.

Bessell, David. "Blurring Boundaries: Trends and Implications in Audio Production Software Developments." In *The Oxford Handbook of Interactive Audio,* edited by Karen Collins, Bill Kapralos, and Holly Tessler. New York: Oxford University Press, 2013.

Bijker, Wiebe E., Thomas P. Hughes, and Trevor Pinch. *The Social Construction of Technological Systems.* Anniversary Edition. Cambridge and London: The MIT Press, 2012.

Bissell, Tom. *Extra Lives: Why Video Games Matter.* New York: Pantheon Books, 2010.

Bogost, Ian. *Persuasive Games: The Expressive Power of Videogames.* Cambridge and London: The MIT Press, 2007.

________. "Persuasive Games: The Proceduralist Style." *Gamasutra,* January 21, 2009. Accessed December 11, 2014. http://www.gamasutra.com/view/feature/132302/persuasive_games_the_.php.

________. *Unit Operations: An Approach to Videogame Criticism.* Cambridge and London: The MIT Press, 2006.

________. "What is Object-Oriented Ontology? A Definition for Ordinary Folk." December 8, 2009. Accessed April 18, 2016, http://bogost.com/writing/blog/what_is_objectoriented_ontolog/.

Bolter, Jay David and Richard Grusin. *Remediation: Understanding New Media.* Cambridge, MA: The MIT Press, 2000.

Borison, Rebecca. “The 15 Highest-Grossing iPhone And iPad Games.” *Business Insider,* May 20, 2014. Accessed March 28, 2016, http://www.businessinsider.com/highest-grossing-iphone-and-ipad-games-2014-5?op=1.

Born, Georgina. “On Musical Mediation: Ontology, Technology, and Creativity.” *Twentieth-Century Music* 2, no. 1 (2005): 7-36.

________. *Rationalizing Culture: IRCAM, Boulez, and the Institutionalization of the Musical Avant-Garde.* Berkeley and Los Angeles: University of California Press, 1995.

________. “Recording: From Reproduction to Representation to Remediation.” In *The Cambridge Companion to Recorded Music,* edited by Nicholas Cook, Eric Clarke, Daniel Leech-Wilkinson, and John Rink, 286-304. Cambridge and New York: Cambridge University Press, 2009.

Böttcher, Niels, Héctor P. Martinez, and Stefania Serafin. “Procedural Audio in Computer Games Using Motion Controllers: An Evaluation on the Effect and Perception.” *International Journal of Computer Games Technology* (2013).

Bourdieu, Pierre. “Structures, *Habitus,* Practices.” In *The Logic of Practice,* 52-65. Translated by Richard Nice. Stanford: Stanford University Press, 1990.

Bourriaud, Nicolas. *Relational Aesthetics.* Dijon, France: Les Presses du Réel, 1998.

Bracy, Catherine. “Closing The Tech Industry's Gender Gap Requires Better Data.” *NPR,* June 25, 2013. Accessed April 26, 2016, http://www.npr.org/sections/alltechconsidered/2013/06/24/195144754/closing-the-tech-industry-s-gender-gap-requires-better-data.

Brand, Stewart. *The Media Lab: Inventing the Future at M.I.T.* New York: Penguin Books, 1988.

Brodkin, Jon. “How Unity3D Became a Game-Development Beast.” *Dice,* June 3, 2013. Accessed November 18, 2014. http://news.dice.com/2013/06/03/how-unity3d-become-a-game-development-beast/.

Brovig-Hanssen, Ragnhild. “Opaque Mediation: The Cut-and-Paste Groove in DJ Food’s ‘Break’.” In *Musical Rhythm in the Age of Digital Reproduction*, edited by Anne Danielsen, 159-76. Burlington, VT: Ashgate, 2010.

Brown, Peter. *Information Architecture With XML.* Hoboken, NJ: John Wiley & Sons Ltd., 2003.

Bull, Michael. *Sound Moves: iPod Culture and Urban Experience.* New York and London: Routledge, 2008.

Burdick, Anne, Johanna Drucker, Peter Lunenfeld, Todd Presner, and Jeffrey Schnapp. *Digital_Humanities.* Cambridge and London: The MIT Press, 2012.

Burgess, Jean. "The iPhone Moment, the Apple Brand, and the Creative Consumer: From 'Hackability and Usability' to Cultural Generativity." In *Studying Mobile Media: Cultural Technologies, Mobile Communication, and the iPhone*, edited by Larissa Hjorth, Jean Burgess and Ingrid Richardson. New York: Routledge, 2012.

Butler, Chris. *Henri Lefebvre: Spatial Politics, Everyday Life and the Right to the City*. New York: Routledge, 2012.

Butler, Mark. *Playing With Something That Runs: Technology, Improvisation, and Composition in DJ and Laptop Performance.* New York: Oxford University Press, 2014.

Butler, Shane. *The Ancient Phonograph.* Cambridge, MA: The MIT Press, 2015.

Cage, John. *Silence: Lectures and Writings.* Hanover, NH: Wesleyan University Press, 1961.

Calleja, Gordon. *In-Game: From Immersion to Incorporation.* Cambridge and London: The MIT Press, 2011.

Callon, Michel and Bruno Latour. "Unscrewing the Big Leviathan: how actors macrostructure reality and how sociologists help them to do so." In *Advances in Social Theory and Methodology: Toward an Integration of Micro- and Macro-Sociologies.* Edited by K.D. Knorr-Cetina and A.V. Cicourel, 277-303. Boston, MA: Routledge and Kegan Paul, 1981.

Cariani, Peter. "Emergence and Artificial Life." In *Artificial Life II, SFI Studies in the Sciences of Complexity,* edited by Christopher Langton et al. New York: Addison-Wesley, 1991.

Carlson, Glen. "Emergent Aesthetics." glencarlson.com. Accessed April 19, 2016, http://www.glencarlson.com/emergent/Carlson_EmergentAesthetics.pdf.

Cascella, Daniela. "Sound Objects: Pierre Schaeffer's 'In Search of a Concrete Music'." *Los Angeles Review of Books,* April 3, 2013. Accessed April 6, 2015. http://lareviewofbooks.org/review/sound-objects-pierre-schaeffers-in-search-of-a-concrete-music.

Cascone, Kim. "The Aesthetics of Failure: 'Post-Digital' Tendencies in Contemporary Computer Music." *Computer Music Journal* 24, no. 4 (2000).

Ceruzzi, Paul E. "The First Computers, 1935-1945." In *Computing: A Concise History,* 23-48. Cambridge, MA: The MIT Press, 2012.

Cheng, William. *Sound Play: Video Games and the Musical Imagination.* Oxford and New York: Oxford University Press, 2014.

Chion, Michel. *Audio-Vision: Sound on Screen.* New York: Columbia University Press, 1994.

Chun, Wendy Hui Kyong. *Programmed Visions: Software and Memory.* Cambridge and London: The MIT Press, 2011.

Chung, Bryan W.C. *Multimedia Programming With Pure Data.* Birmingham: Packt Publishing, 2013.

"Clint Sand: Electronic Polymath." *Ableton,* January 20, 2011. Accessed April 20, 2015. https://www.ableton.com/en/pages/artists/clint_sand/.

Code.org. "Hour of Code." Accessed December 12, 2014, http://hourofcode.com/us.

Collins, Karen. "An Introduction to Procedural Music in Video Games." *Contemporary Music Review* 28, no. 1 (2009): 5-15.

________, ed. *From Pac-Man to Pop Music: Interactive Audio in Games and New Media.* Burlington, VT: Ashgate, 2008.

________. *Game Sound: An Introduction to the History, Theory, and Practice of Video Game Music and Sound Design.* Cambridge, MA: The MIT Press, 2008.

________. *Playing With Sound: A Theory of Interacting with Sound and Music in Video Games.* Cambridge, MA: The MIT Press, 2013.

Collins, Nick. "Live Coding of Consequence." *Leonardo* 44, no. 3 (2011): 207-11.

Cook, Daniel. "What are Game Mechanics?" *Lostgarden,* October 23, 2006. Accessed April 8, 2016. http://www.lostgarden.com/2006/10/what-are-game-mechanics.html.

Coyne, Richard. *The Tuning of Place: Sociable Spaces and Pervasive Digital Media.* Cambridge and London: The MIT Press, 2010.

Cox, Geoff, and Alex Mclean. *Speaking Code: Coding as Aesthetic and Political Expression.* Cambridge and London: The MIT Press, 2012.

Csikszentmihalyi, Mihaly. *Flow: The Psychology of Optimal Experience.* New York: Harper Collins, 1991.

D'Errico, Mike. "Going Hard: Bassweight, Sonic Warfare, and the 'Brostep' Aesthetic." *Sounding Out* blog, January 23, 2014. Accessed April 19, 2016, https://soundstudiesblog.com/2014/01/23/going-hard-bassweight-sonic-warfare-the-brostep-aesthetic/.

Danielsen, Anne, ed. *Musical Rhythm in the Age of Digital Reproduction*. Burlington, VT: Ashgate, 2010.

De Sola Pool, Ithiel. *Technologies of Freedom: On Free Speech in an Electronic Age.* Cambridge, MA: Harvard University Press, 1983.

Deadmau5. “we all hit play.” *United We Fail* blog, 2013. Accessed November 9, 2015. http://deadmau5.tumblr.com/post/25690507284/we-all-hit-play.

“Decap and Brady Watt on Push & working with Ski Beatz.” *Ableton*, April 16, 2014. Accessed April 18, 2016, https://www.ableton.com/en/blog/decap-and-brady-watt-push-working-ski-beatz/.

DeSantis, Dennis. *Making Music: 74 Creative Strategies for Electronic Music Producers.* Berlin: Ableton AG, 2015.

Dietrich, Frank. “Visual Intelligence: The First Decade of Computer Art (1965-1975).” *Leonardo* 19, no. 2 (1986).

“Diggin' In The Carts - Series Trailer - Red Bull Music Academy Presents.” Red Bull Music Academy, uploaded August 28, 2014. Accessed March 30, 2016). https://www.youtube.com/watch?v=fwN_o_fi7xE.

Ding, Wei and Xia Lin. *Information Architecture: The Design and Integration of Information Spaces.* San Rafael, CA: Morgan & Claypool, 2009.

DJ A-Trak. “Don’t Push My Buttons.” *Huffington Post,* July 23, 2012. Accessed April 18, 2016, http://www.huffingtonpost.com/atrak/dont-push-my-buttons_b_1694719.html.

Dolan, Emily. “Toward a Musicology of Interfaces.” *Keyboard Perspectives* 5 (2014): 1-14.

Dourish, Paul. *Divining a Digital Future: Mess and Mythology in Ubiquitous Computing.* Cambridge and London: The MIT Press, 2011.

________. *Where the Action Is: The Foundations of Embodied Interaction.* Cambridge and London: The MIT Press, 2004.

Drucker, Johanna. *SpecLab: Digital Aesthetics and Projects in Speculative Computing.* Chicago and London: University of Chicago Press, 2009.

Dyson, Frances. *Sounding New Media: Immersion and Embodiment in the Arts and Culture.* Berkeley and Los Angeles: University of California Press, 2009.

“E3 2011 Machinima Coverage - Sound Shapes on the PS VITA Game Demo & Interview.” *Machinima*, uploaded June 9, 2011. Accessed April 19, 2016, https://www.youtube.com/watch?v=_rRU0bNeJ20.

Eagleton, Terry. *The Ideology of the Aesthetic*, 2nd Ed. Cambridge, MA: Basil Blackwell, 1981.

Eaton, Marcia Muelder. *Basic Issues in Aesthetics.* Belmont, CA: Wadsworth Publishing, 1988.

Eidsheim, Nina Sun. *Sensing Sound: Singing and Listening as Vibrational Practice.* Durham, NC: Duke University Press, 2015.

Eisenberg, Evan. *The Recording Angel: Music, Records and Culture from Aristotle to Zappa.* New Haven and London: Yale University Press, 1987.

Ekman, Ulrik, ed. *Throughout: Art and Culture Emerging with Ubiquitous Computing.* Cambridge and London: The MIT Press, 2012.

Ellul, Jacques. *The Technological Society.* New York: Vintage Books, 1964.

Emmerson, Simon. *Living Electronic Music.* Burlington, VT: Ashgate, 2007.

Eno, Brian. "Generative Music." *In Motion Magazine,* July 7, 1996. Accessed December 11, 2014. http://www.inmotionmagazine.com/eno1.html.

Ensmenger, Nathan L. *The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise.* Cambridge and London: The MIT Press, 2010.

Erlmann, Veit, ed. *Hearing Cultures: Essays on Sound, Listening, and Modernity*. Oxford and New York: Berg, 2005.

Evans, Claire L. "A Eulogy for Skeuomorphism." *Vice,* June 11, 2013. Accessed April 25, 2016, http://motherboard.vice.com/read/a-eulogy-for-skeumorphism.

Evens, Aden. *Sound Ideas: Music, Machines, and Experience.* Minneapolis and London: University of Minnesota Press, 2005.

________. "Touch in the Abstract." *SubStance* 70, no. 3 (2011).

Farman, Jason. *Mobile Interface Theory: Embodied Space and Locative Media.* New York and London: Routledge, 2012.

Farnell, Andy. "Perspectives on Sound Design." *Designing Sound* blog, August 27, 2013. Accessed August 19, 2015. http://designingsound.org/2013/08/perspectives-on-sound-design/.

________. "Procedural Audio Theory and Practice." In *The Oxford Handbook of Interactive Audio,* edited by Karen Collins, Bill Kapralos, and Holly Tessler. New York: Oxford University Press, 2014.

Farokhmanesh, Megan. "Minecraft console sales surpass PC, 'almost 54M' copies sold in total." *Polygon,* June 25, 2014. Accessed April 29, 2016, http://www.polygon.com/2014/6/25/5843358/minecraft-console-sales-54m-copies-sold.

Farrugia, Rebekah, and Thomas Swiss. "Tracking the DJs: Vinyl Records, Work, and the Debate over New Technologies." *Journal of Popular Music Studies* 17, no. 1 (2005).

Feld, Steven. "Doing Anthropology in Sound." *American Ethnology* 41, no. 4 (2004): 461-74.

________. "Places Sensed, Senses Placed: Toward a Sensuous Epistemology of Environments." In *Empire of the Senses: The Sensual Culture Reader,* edited by David Howes. Oxford and New York: Berg, 2005.

Feldman, Jessica. "The MIDI Effect." Paper presented at Bone Flute to Auto-Tune: A Conference on Music & Technology in History, Theory and Practice, Berkeley, California, April 24-26, 2014.

"Figure for iPhone and iPad." *Propellerhead,* uploaded April 4, 2012. Accessed April 30, 2016, https://www.youtube.com/watch?v=gLLjRH6GJec.

Foster, Hal. "Design and Crime." In *Design and Art,* edited by Alex Coles. Cambridge and London: Whitechapel and The MIT Press, 2007.

Foucault, Michel. *The Archeology of Knowledge*. Translated by R. Sheridan. New York and London: Routledge, 1972.

Franinović, Karmen, and Stefania Serafin. *Sonic Interaction Design.* Cambridge and London: The MIT Press, 2013.

Frith, Simon, and Simon Zagorski-Thomas, eds. *The Art of Record Production: An Introductory Reader for a New Academic Field.* Burlington, VT: Ashgate, 2012.

Fuller, Matthew. *Behind the Blip: Essays on the Culture of Software.* Brooklyn: Autonomedia, 2003.

________. *Media Ecologies: Materialist Energies in Art and Technoculture.* Cambridge, MA: The MIT Press, 2007.

________, ed. *Software Studies: A Lexicon.* Cambridge, MA: The MIT Press, 2008.

Galloway, Alexander R. *The Interface Effect.* Malden, MA: Polity Press, 2012.

Galuszka, Patryk. "Netlabels and Democratization of the Recording Industry." *First Monday* 17, no. 7 (July 2012) http://firstmonday.org/article/view/3770/3278.

Gardner, Martin. "Mathematical Games: The Fantastic Combinations of John Conway's New Solitaire Game 'Life'." *Scientific American* 223 (1970): 120-23.

Gatlin, Jonathan. *Bill Gates: The Path to the Future.* New York: Harper Collins, 1999.

Geertz, Clifford. "Thick Description: Toward an Interpretative Theory of Culture." In *The Interpretation of Cultures.* New York: Basic Books, 1973.

Gibson, James J. *The Ecological Approach to Visual Perception*. Hillsdale, NJ: Lawrence Erlbaum Associates, 1986.

Gigliotti, Carol. "Aesthetics of a Virtual World." *Leonardo* 28, no. 4 (1995).

Gillespie, Tarleton. "The Politics of 'Platforms'." *New Media & Society* 12, no. 3 (2010): 347-364.

Gilmore, Lauren. "The gender gap in technology can byte me." *The Next Web,* December 1, 2015. Accessed April 26, 2016, http://thenextweb.com/insider/2015/12/01/the-gender-gap-in-technology-can-byte-me/#gref.

Gitelman, Lisa. "Media, Materiality, and the Measure of the Digital: Or, the Case of Sheet Music and the Problem of Piano Rolls." In *Memory Bytes: History, Technology, and Digital Culture*. Edited by Lauren Rabinovitz and Abraham Geil, 199-217. Durham: Duke University Press, 2004.

Gleick, James. *The Information: A History, A Theory, A Flood.* New York: Pantheon, 2011.

Goehr, Lydia. *The Imaginary Museum of Musical Works: An Essay in the Philosophy of Music.* Oxford and New York: Oxford University Press, 1992.

Gold, Matthew K., ed. *Debates in the Digital Humanities.* Minneapolis: University of Minnesota Press, 2012.

Golden, Ean. "Music Maneuvers: Discover the Digital Turntablism Concept, 'Controllerism,' Compliments of Moldover." *Remix,* October 2007. Accessed April 17, 2016. http://www.moldover.com/press/Moldover_Remix_Oct-2007_w.jpg.

Google. "Material Design: Introduction." *Google,* n.d. Accessed April 25, 2016, https://www.google.com/design/spec/material-design/introduction.html.

Gopinath, Sumanth and Jason Stanyek, eds. *The Oxford Handbook of Mobile Music Studies, Volume 1.* Oxford and New York: Oxford University Press, 2014.

Gordon, Rich. "The Meanings and Implications of Convergence." In *Digital Journalism: Emerging Media and the Changing Horizons of Journalism,* edited by Kevin Kawamoto. Oxford: Rowman & Littlefield, 2003.

Gracyk, Theodore. *Rhythm and Noise: An Aesthetics of Rock.* Durham, NC: Duke University Press, 1996.

Greene, Paul D. and Thomas Porcello, eds. *Wired For Sound: Engineering and Technologies in Sonic Cultures.* Middletown, CT: Wesleyan University Press, 2005.

Greenfield, Adam. *Everyware: The Dawning Age of Ubiquitous Computing.* San Francisco, CA: New Riders Publishing, 2006.

Greher, Gena R., and Jesse M. Heines. *Computational Thinking in Sound: Teaching the Art and Science of Music and Technology.* New York: Oxford University Press, 2014.

Gube, Jacob. "What is User Experience Design? Overview, Tools and Resources." *Smashing Magazine,* October 5, 2010. Accessed April 1, 2016, http://uxdesign.smashingmagazine.com/2010/10/05/what-is-user-experience-design-overview-tools-and-resources/.

Guevara-Villalobos, Orlando. "Cultures of Independent Game Production: Examining the Relationship Between Community and Labour." In *Think Design Play: Proceedings of the 2011 Digital Games Research Association Conference.*

Hamilton, Robert, et al. "Social Composition: Musical Data Systems for Expressive Mobile Music." *Leonardo Music Journal* 21 (2011).

Hands, Joss. "Politics, Power, and 'Platformativity'." *Culture Machine* 14 (2013).

Haworth, Christopher. "Sound Synthesis Procedures as Texts: An Ontological Politics in Electroacoustic and Computer Music." Computer Music Journal 39, no. 1 (2015): 41-58.

Heidegger, Martin. "The Question Concerning Technology." *Basic Writings.* Edited by David Farrell Krell. New York: Harper & Row, 1977.

Herber, Norbert. "The Composition-Instrument: Emergence, Improvisation, and Interaction in Games and New Media." In *From Pac-Man to Pop Music: Interactive Audio in Games and New Media,* edited by Karen Collins. Burlington, VT: Ashgate, 2008.

Harkins, Paul. "Microsampling: From Akufen's Microhouse to Todd Edwards and the Sound of UK Garage." In *Musical Rhythm in the Age of Digital Reproduction.* Edited by Anne Danielsen, 177-94. Burlington, VT: Ashgate, 2012.

Harper, Richard H.R. *Texture: Human Expression in the Age of Communications Overload.* Cambridge, MA: The MIT Press, 2010.

Harwood, John. *The Interface: IBM and the Transformation of Corporate Design, 1945-1976.* Minneapolis and London: University of Minnesota Press, 2011.

Hayles, N. Katherine. *Electronic Literature: New Horizons for the Literary.* Notre Dame, IN: University of Notre Dame Press, 2008.

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

________. *How We Think: Digital Media and Contemporary Technogenesis.* Chicago: University of Chicago Press, 2012.

Heatherly, Ben and Logan Howard. "Video Game Controllers." Accessed June 8, 2015. http://andrewd.ces.clemson.edu/courses/cpsc414/spring14/papers/group2.pdf.

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

Helmreich, Stefan. "An anthropologist underwater: Immersive soundscapes, submarine cyborgs, and transductive ethnography." *American Ethnologist* 34 no. 4 (2007): 621-41.

Henriques, Julian. *Sonic Bodies: Reggae Sound Systems, Performance Techniques, and Ways of Knowing*. London and New York: Continuum, 2011.

Higgins, Hannah B., and Douglas Kahn. *Mainframe Experimentalism: Early Computing and the Foundations of the Digital Arts.* Berkeley and Los Angeles: University of California Press, 2012.

Hookway, Branden. *Interface.* Cambridge and London: The MIT Press, 2014.

Huhtamo, Erkki. *Illusions in Motion: Media Archeology of the Moving Panorama and Related Spectacles.* Cambridge, MA: MIT Press, 2013.

Huizinga, Johan. *Homo Ludens: A Study of the Play-Element in Culture.* Boston: The Beacon Press, 1950.

Hunley, Alex. "Hamlet's Blackberry – William Powers." *Full Stop,* August 24, 2011. Accessed May 15, 2015, http://www.full-stop.net/2011/08/24/reviews/alexhunley/hamlets-blackberry-william-powers/.

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

"Incredible amazing awesome Apple." Kanal von justanotherguy84, uploaded September 15, 2009. Accessed March 22, 2016, https://www.youtube.com/watch?v=Nx7v815bYUw.

Ingold, Tim. *The Perception of the Environment*. New York and London: Routledge, 2000.

Ingram, Matthew. "Switched-On." *Loops* 1 (2009): 129-137.

"Interactivity and Feedback." In *iOS Human Interface Guidelines,* last updated November 5, 2015. Accessed December 30, 2015, https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/InteractivityInput.html.

Intothewoods.tv, "Far From Home #9 – Daedelus Live Arcade Set." Uploaded April 18, 2012. Accessed March 30, 2016. https://vimeo.com/40608297.

"iPhone 3g Commercial 'There's An App For That' 2009." *CommercialKid,* uploaded February 4, 2009. Accessed April 30, 2016, https://www.youtube.com/watch?v=szrsfeyLzyg.

"Is it live or is it Memorex." ukclassictelly, uploaded May 14, 2012. Accessed April 4, 2016, https://www.youtube.com/watch?v=lhfugTnXJV4.

Jacko, Julie A. *The Human-Computer Interaction Handbook,* 3rd edition. Boca Raton, FL: CRC Press, 2012.

Jacob, Robert J.K., et al. "Reality-based interaction: a framework for post-WIMP interfaces." In *CHI 2008 Proceedings: Post-WIMP,* Florence, Italy, April 5-10 (2008).

James, Robin. "Aesthetics and Live Electronic Music Performance." *Cyborgology,* December 13, 2013. Accessed January 5, 2016. http://thesocietypages.org/cyborgology/2013/12/13/aesthetics-and-live-electronic-music-performance/.

Jenkins, Henry. *Confronting the Challenges of Participatory Culture: Media Education for the 21st Century.* Cambridge, MA: The MIT Press, 2009.

________. *Convergence Culture: Where Old and New Media Collide.* New York: New York University Press, 2006.

Jenkins, Henry, Sam Ford, and Joshua Green. *Spreadable Media: Creating Value and Meaning in a Networked Culture.* New York and London: New York University Press, 2013.

Johnson, Steven. *Interface Culture.* New York: Harper Collins, 1997.

Juul, Jesper. *Half-Real: Video Games Between Real Rules and Fictional Worlds.* Cambridge and London: The MIT Press, 2011.

________. *A Casual Revolution: Reinventing Video Games and Their Players.* Cambridge, MA: The MIT Press, 2009.

Kachka, Boris, Rebecca Milzoff and Dan Reilly. "10 Tricks That Musicians and Actors Use in Live Performances." *Vulture,* April 3, 2016. Accessed April 18, 2016, http://www.vulture.com/2016/03/live-performance-tricks.html.

Kane, Brian. *Sound Unseen: Acousmatic Sound in Theory and Practice.* New York: Oxford University Press, 2014.

Kant, Immanuel. *Critique of Judgment.* Translated by Werner S. Pluhar. Indianapolis, IN: Hackett, 1987.

Kartomi, Margaret. "The Classification of Musical Instruments: Changing Trends in Research from the Late Nineteenth Century, with Special Reference to the 1990s." *Ethnomusicology* 45, no. 2 (2001).

Katz, Mark. *Capturing Sound: How Technology Has Changed Music.* Berkeley and Los Angeles: University of California Press, 2004.

________. *Groove Music: The Art and Culture of the Hip-Hop DJ.* New York: Oxford University Press, 2012.

________. "The Amateur in the Age of Mechanical Music." In *The Oxford Handbook of Sound Studies.* Edited by Trevor Pinch and Karin Bijsterveld, 459-479. New York: Oxford University Press, 2012.

Kay, Alan. "Computer Software." *Scientific American* 251, no. 3 (1984): 52-9.

Kay, Alan and Adele Goldberg. "Personal Dynamic Media." In *The New Media Reader*. Edited by Noah Wardrip-Fruin and Nick Montfort, 393-404. Cambridge: The MIT Press, 2003.

Keil, Charles. "The Theory of Participatory Discrepancies: A Progress Report." *Ethnomusicology* 39, no. 1 (1995): 1-19.

Kelly, Heather. "Do video games cause violence?" *CNN,* August 17, 2015. Accessed April 18, 2016, http://money.cnn.com/2015/08/17/technology/video-game-violence/.

Kelty, Chris. *Two Bits: The Cultural Significance of Free Software.* Durham, NC: Duke University Press, 2008.

Kincaid, Jason. "RJDJ Generates an Awesome, Trippy Soundtrack for Your Life." *TechCrunch,* October 13, 2008. Accessed April 11, 2016, http://techcrunch.com/2008/10/13/rjdj-generates-an-awesome-trippy-soundtrack-for-your-life/.

Kirschenbaum, Matthew G. *Mechanisms: New Media and the Forensic Imagination.* Cambridge, MA: The MIT Press, 2008.

Kitchin, Rob, and Martin Dodge. *Code/Space: Software and Everyday Life.* Cambridge and London: The MIT Press, 2011.

Kittler, Friedrich. *Discourse Networks 1800/1900.* Stanford, CA: Stanford University Press, 1990.

________. *Gramophone, Film, Typewriter*. Translated by Geoffrey Winthrop-Young and Michael Wutz. Stanford, CA: Stanford University Press, 1999.

Krug, Steve. *Don't Make Me Think, Revisited: A Common Sense Approach to Web and Mobile Usability.* San Francisco, CA: New Riders, 2014.

Kuniavsky, Mike. *Smart Things: Ubiquitous Computing User Experience Design.* Burlington, MA: Morgan Kaufmann, 2010.

LaBelle, Brandon. *Acoustic Territories: Sound Culture and Everyday Life.* New York: Continuum, 2010.

Langton, Christopher. "Artificial Life." In *Artificial Life, SFI Studies in the Sciences of Complexity,* edited by Christopher Langton, 1-47. New York: Addison-Wesley, 1989.

Larson, Brad. "Rapid Prototyping in Swift Playgrounds." *objc.io* 16 (September 2014). Accessed April 12, 2016. https://www.objc.io/issues/16-swift/rapid-prototyping-in-swift-playgrounds/.

Latour, Bruno. *Reassembling the Social: An Introduction to Actor-Network-Theory.* Oxford and New York: Oxford University Press, 2005.

________. "Technology is Society Made Durable." In *A Sociology of Monsters? Essays on Power, Technology and Domination, Sociological Review Monograph*. Edited by John Law, 103-31. London: Routledge, 1991.

Laurel, Brenda. *Computers as Theatre.* Boston, MA: Addison-Wesley, 1993.

"Lecture: Skream (Melbourne 2006)." Red Bull Music Academy, uploaded September 24, 2011. Accessed March 31, 2016. https://vimeo.com/29535666.

Le Guin, Elisabeth. *Boccherini's Body: An Essay in Carnal Musicology.* Berkeley and Los Angeles: University of California Press, 2006.

________. "'Cello-and-Bow Thinking': Boccherini's Cello Sonata in Eb Major, 'fuori catalog'." *Echo: A Music-Centered Journal* 1, no. 1 (1999). Accessed March 30, 2016. http://www.echo.ucla.edu/Volume1-Issue1/leguin/leguin-article.html.

Ledesma, Eduardo. "The Poetics and Politics of Computer Code in Latin America: Codewor, Code Art, and Live Coding." *Revista de Estudios Hispánicos* 49, no. 1 (2015): 91-120.

Leman, Marc. *Embodied Music Cognition and Mediation Technology.* Cambridge and London: The MIT Press, 2008.

Lessig, Lawrence. *Remix: Making Art and Commerce Thrive in the Hybrid Economy.* New York: Penguin, 2008.

Levy, Steven. *Hackers: Heroes of the Computer Revolution.* Garden City, NY: Doubleday, 1984.

Licklider, J.C.R. "Man-Computer Symbiosis." *IRE Transactions on Human Factors in Electronics* 1 (1960): 4-11.

Lin II, Rong-Gong. "ER doctors: Drug-fueled raves too dangerous and should be banned." *Los Angeles Times,* August 10, 2015. Accessed April 18, 2016, http://www.latimes.com/local/lanow/la-me-ln-why-some-er-doctors-want-to-end-raves-in-los-angeles-county-20150810-story.html.

Lipkin, Nadav. "Examining Indie's Independence: The Meaning of 'Indie' Games, the Politics of Production, and Mainstream Co-optation." *Loading…* 7, no. 11 (2012).

Livecoding.tv. "Livecoding: Watch People Code Products Live." Accessed April 5, 2016, https://www.livecoding.tv.

Lyon, Eric. *Designing Audio Objects for Max/MSP and Pd.* Middleton, WI: A-R Editions, 2012.

Lunenfeld, Peter. "Mia Laboro: Maker's Envy and the Generative Humanities." Book Presence in a Digital Age Conference. University of Utrecht, 2012.

________. "The Maker's Discourse." Critical Mass: the Legacy of Hollis Frampton, Invited Conference. University of Chicago, 2010.

________. *The Secret War Between Downloading and Uploading: Tales of the Computer as Culture Machine.* Cambridge, MA: The MIT Press, 2011.

Lunning, Frenchy, ed. *Mechademia 5: Fanthropologies.* Minneapolis: University of Minnesota Press, 2010.

Luta, Primus. "Musical Objects, Variability and Live Electronic Performance." *Sounding Out!,* August 12, 2013. Accessed January 5, 2016. http://soundstudiesblog.com/2013/08/12/musical-objects-variability/.

________. "Toward a Practical Language for Live Electronic Performance." *Sounding Out!,* April 29, 2013. Accessed January 5, 2016. http://soundstudiesblog.com/2013/04/29/toward-a-practical-language-for-live-electronic-performance/.

MacColl, Ian and Ingrid Richardson. "A cultural somatics of mobile media and urban screens: Wiffiti and the IWALL prototype." *Journal of Urban Technology* 15, no. 3 (2008): 99-116.

"Machinedrum: Sacred Frequencies." *Ableton,* September 25, 2012. Accessed April 28, 2015. https://www.ableton.com/en/blog/machinedrum/.

Mackenzie, Adrian. "Transduction: Invention, Innovation and Collective Life." Accessed April 19, 2013, http://www.lancs.ac.uk/staff/mackenza/papers/ transduction.pdf.

Magnusson, Thor. “Affordances and Constraints in Screen-Based Musical Instruments.” In *Proceedings of the 4th Nordic Conference on Human-Computer Interaction: Changing Roles,* 441-4. New York: ACM Press, 2006.

Magnusson, Thor. “Algorithms as Scores: Coding Live Music.” *Leonardo Music Journal* 21 (2011): 19-23.

“Making the iPhone: The Design Tao of Jony Ive.” *Bloomberg,* September 19, 2015. Accessed October 5, 2015, http://www.bloomberg.com/news/videos/b/41d8c560-9bc6-479c-a3b2-6dfd2b038aca.

Malloch, Joseph, David M. Birnbaum, Elliot Sinyor, and Marcelo M. Wanderley. “Towards a New Conceptual Framework for Digital Musical Instruments.” *Proceedings of the 9th International Conference on Digital Audio Effects* (2006): 49-52.

Manovich, Lev. *The Language of New Media.* Cambridge and London: The MIT Press, 2002.

________. *Software Takes Command.* New York and London: Bloomsbury, 2013.

Manzo, V.J. *Max/MSP/Jitter for Music: A Practical Guide to Developing Interactive Music Systems for Education and More.* Oxford and New York: Oxford University Press, 2011.

Martin, Chase Bowen and Mark Deuze. “The Independent Production of Culture: A Digital Games Case Study.” *Games and Culture* 4, no. 3 (2009): 276-295.

Mateas, Michael. “Procedural Literacy: Educating the New Media Practitioner.” In *Beyond Fun: Serious Games and Media,* edited by Drew Davidson. Pittsburgh, PA: ETC Press, 2008. Accessed December 9, 2014. http://press.etc.cmu.edu/node/205.

Mauro-Flude, Nancy. “The Intimacy of the Command Line.” *Scan: Journal of Media Arts Culture* 10, no. 2 (2013). Accessed April 18, 2016, http://scan.net.au/scn/journal/vol10number2/Nancy-Mauro-Flude.html.

Mauss, Marcel. “Body Techniques.” In *Sociology and Psychology Essays,* translated by Ben Brewster, 120-123. New York and London: Routledge, 1979.

McCarty, Willard. “Humanities Computing: Essential Problems, Experimental Practice.” *Literary and Linguistic Computing* 17, no. 1 (April 1, 2002): 103-125.

McGann, Jerome. *Radiant Textuality: Literature After the World Wide Web.* New York: Palgrave, 2001.

McGonigal, Jane. *Reality is Broken: Why Games Make Us Better and How They Can Change the World.* New York: The Penguin Press, 2011.

McLean, Alex, Julian Rohrhuber and Nick Collins, eds. Special Issue on Live Coding. *Computer Music Journal* 38, no. 1 (2014).

McLuhan, Marshall. *The Medium is the Massage: An Inventory of Effects.* Corte Madera, CA: Gingko Press, 2001.

________. *Understanding Media: The Extensions of Man.* New York: McGraw-Hill, 1964.

McPherson, Tara. "Why Are the Digital Humanities So White? or Thinking the Histories of Race and Computation." In *Debates in the Digital Humanities,* edited by Matthew K. Gold. Minneapolis: University of Minnesota Press, 2012.

Mechaber, Ezra. "President Obama is the First President to Write a Line of Code." *The White House Blog,* December 10, 2014. Accessed December 13, 2014. http://www.whitehouse.gov/blog/2014/12/10/president-obama-first-president-write-line-code.

Merrin, William. "The Rise of the Gadget and Hyperludic Me-dia." *Cultural Politics* 10, no. 1 (2014).

Microsoft. "Get Your Start With an Hour of Code." Accessed December 12, 2014, http://www.microsoft.com/about/corporatecitizenship/en-us/youthspark/youthsparkhub/hourofcode/.

Millard, Drew. "Interview: Rustie used to produce like how gamers game." *Pitchfork*, May 31, 2012. Accessed November 10, 2012. http://archive.is/dzPoJ.

Miller, Kiri. *Playing Along: Digital Games, YouTube and Virtual Performance.* New York and London: Oxford University Press, 2012.

Miller, Paul D. and Svitlana Matviyenko, eds. *The Imaginary App.* Cambridge, MA: The MIT Press, 2014.

Mills, Mara. "Do Signals Have Politics? Inscribing Abilities in Cochlear Implants." In *The Oxford Handbook of Sound Studies,* edited by Trevor Pinch and Karin Bijsterveld, 320-46. Oxford and New York: Oxford University Press, 2012.

Mitchell, William J. *The Reconfigured Eye: Visual Truth in the Post-Photographic Era.* Cambridge, MA: The MIT Press, 1994.

"Moldover: Performance and Controllerism." *Ableton* blog, November 28, 2013. Accessed November 12, 2015. https://www.ableton.com/en/blog/moldover-performance-and-controllerism/.

Molina, Brett. "Obama seeks research into violent video games." *USA Today,* January 16, 2013. Accessed April 18, 2016, http://www.usatoday.com/story/tech/gaming/2013/01/16/obama-gun-violence-video-games/1839879/.

Monome. "grid." Last updated October 25, 2015. Accessed January 5, 2016. http://market.monome.org/collections/primary/products/grid.

Montano, Ed. "'How do you know he's not playing Pac-Man while he's supposed to be DJing?': Technology, Formats and the Digital Future of DJ Culture." *Popular Music* 29, no. 3 (2010): 397-416.

Montfort, Nick. *10 PRINT CHR$(205.5+RND(1));:GOTO 10.* Cambridge, MA: The MIT Press, 2013.

"Monu/Mental." *Unity3D,* January 23, 2014. Accessed April 11, 2016, https://unity3d.com/showcase/case-stories/monument-valley.

Moretti, Franco. "Conjectures on World Literature." *New Left Review* 1 (2000). Accessed April, 17, 2016. https://newleftreview.org/II/1/franco-moretti-conjectures-on-world-literature.

Morozov, Evgeny. *To Save Everything, Click Here: The Folly of Technological Solutionism.* New York: Public Affairs, 2013.

Morrogh, Earl. *Information Architecture: An Emerging 21st Century Profession.* Upper Saddle River, NJ: Prentice Hall, 2003.

Moseley, Roger. "Playing Games With Music (and Vice Versa): Ludomusicological Perspectives on *Guitar Hero* and *Rock Band*." In *Taking it to the Bridge: Music as Performance,* edited by Nicholas Cook and Richard Pettengill. Ann Arbor: University of Michigan Press, 2013.

Moulier-Boutang, Yann. *Cognitive Capitalism.* Boston: Polity, 2011.

Muggs, Joe. "Flying Lotus: Cosmic Drama." *Resident Advisor,* March 24, 2010. Accessed April 20, 2015. http://www.residentadvisor.net/feature.aspx?1175.

Mumford, Lewis. *Technics & Civilization.* Chicago: The University of Chicago Press, 1934.

Murray, Janet. *Inventing the Medium: Principles of Interaction Design as a Cultural Practice.* Cambridge, MA: The MIT Press, 2012.

Myers, David. "The Video Game Aesthetic: Play as Form." In *Video Game Theory Reader 2.* Edited by Bernard Perron and Mark J.P. Wolf. New York: Routledge, 2009.

Nadin, Mihai. "Emergent Aesthetics: Aesthetic Issues in Computer Arts." *Leonardo* 2 (1989): 43-8.

Nancy, Jean-Luc. *Listening.* New York: Fordham University Press, 2002.

Nash, Chris, and Alan F. Blackwell. “Flow of Creative Interaction with Digital Music Notations.” In *The Oxford Handbook of Interactive Audio,* edited by Karen Collins, Bill Kapralos, and Holly Tessler. New York: Oxford University Press, 2013.

Native Instruments. “Maschine Studio.” Last modified February 1, 2016. Accessed March 30, 2016. http://www.native-instruments.com/en/products/maschine/production-systems/maschine-studio/.

Nevile, Ben. “An Interview with Kim Cascone.” *Cycling 74,* September 13, 2005. Accessed December 11, 2014. https://cycling74.com/2005/09/13/an-interview-with-kim- /.

Newman, Michael Z. “Indie Culture: In Pursuit of the Authentic Autonomous Alternative.” *Cinema Journal* 48, no. 3 (2009): 16-34.

Ngak, Chenda. “Tupac Coachella hologram: Behind the technology.” *CBS News,* November 9, 2012. Accessed April 18, 2016, http://www.cbsnews.com/news/tupac-coachella-hologram-behind-the-technology/.

Nichols, Bill. “The Work of Culture in the Age of Cybernetic Systems.” *Screen* 21, no. 1 (1988): 22-46.

NIME 2014. “Proceedings.” Last updated May 12, 2015. Accessed January 5, 2016. http://www.nime2014.org/technical-programme/proceedings/.

Noland, Carrie. *Agency and Embodiment: Performing Gestures/ Producing Culture.* Cambridge, MA: Harvard University Press, 2009.

Norman, Donald A. *The Design of Everyday Things.* New York: Basic Books, 1988.

Nusselder, André. *Interface Fantasy: A Lacanian Cyborg Ontology.* Cambridge and London: The MIT Press, 2009.

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

Nystrom, Christine. *Towards a Science of Media Ecology: The Formulation of Integrated Conceptual Paradigms for the Study of Human Communication Systems.* Doctoral dissertation, New York University (1973).

O’Neill, Susan. “Flow Theory and the Development of Musical Performance Skills.” *Bulletin of the Council for Research in Music Education* 141 (1999): 129-34.

Oliveros, Pauline. *Deep Listening: A Composer's Sound Practice.* Lincoln, NE: iUniverse, 2005.

Orr, David W. *The Nature of Design: Ecology, Culture, and Human Intention.* New York: Oxford University Press, 2002.

Ortner, Sherry. *Not Hollywood: Independent Film at the Twilight of the American Dream*. Durham and London: Duke University Press, 2013.

Paradiso, Joseph A., and Sile O'Modhrain. "Current Trends in Electronic Music Interfaces—Guest Editors' Introduction." *Journal of New Music Research* 32, no. 4 (2003): 345-349.

Park, Tae Hong. "An Interview With Max Mathews." *Computer Music Journal* 33, no. 3 (2009): 9-22.

Pask, Andrew. "Mini Interview: Jonny Greenwood." *Cycling 74,* January 2, 2014. Accessed March 30, 2016. https://cycling74.com/2014/01/02/mini-interview-jonny-greenwood/.

Patel, Nilay. "The Technics SL-1200 turntable returns in two new audiophile models." *The Verge,* January 5, 2016. Accessed April 18, 2016, http://www.theverge.com/2016/1/5/10718234/technics-sl1200-sl1200g-sl1200gae-turntable-new-models-announced-release-date-ces-2016.

Pattison, Louis. "Video Game Loops & Aliens: Flying Lotus Interviewed." *The Quietus,* May 18, 2010. Accessed November 13, 2015. http://thequietus.com/articles/04269-flying-lotus-interview-cosmogramma.

Pearson, Matt. *Generative Art: A Practical Guide Using Processing.* Shelter Island, NY: Manning Publications, 2011.

Perron, Bernard, and Mark J.P. Wolf, eds. *The Video Game Theory Reader 2.* New York: Routledge, 2008.

Phelan, Peggy. *Unmarked: The Politics of Performance.* New York: Routledge, 1993.

Pinch, Trevor and Frank Trocco. *Analog Days: The Invention and Impact of the Moog Synthesizer.* Cambridge, MA: Harvard University Press, 2004.

Pinch, Trevor and Karin Bijsterveld, eds. *The Oxford Handbook of Sound Studies.* New York: Oxford University Press, 2011.

Pink, Sarah. *Doing Sensory Ethnography*. London: Sage Publications, 2009.

Polanyi, Michael. *The Tacit Dimension*. Chicago: University of Chicago Press, 1966.

Postman, Neil. "The Reformed English Curriculum." In *High School 1980: The Shape of the Future in American Secondary Education,* edited by Alvin C. Eurich. New York: Pitman, 1970.

Powers, William. *Hamlet's Blackberry: Building a Good Life in the Digital Age.* New York: Harper Perennial, 2011.

"Proteus' Soundscapes: The Similarities Between Procedural Generation and Improvisation." *The Phonograph,* January 6, 2012. Accessed December 12, 2014. http://www.thephonograph.co.uk/2012/01/06/proteus-soundscapes-the-similarities-between-procedural-generation-and-improvisation/.

Puckette, Miller. "Max at Seventeen." *Computer Music Journal* 26, no. 4 (2002): 31-43.

________. "The Deadly Embrace Between Music Software and Its Users." In *Electroacoustic Music Beyond Performance: Proceedings of the Electroacoustic Music Studies Network Conference,* Berlin, 2014.

________. "The Patcher." In *Proceedings of the International Computer Music Asssociation*, 420-9. San Francisco: International Computer Music Association, 1988.

Raessens, Joost. *Homo Ludens 2.0: The Ludic Turn in Media Theory.* Utrecht: Utrecht University, 2012.

Raessens, Joost and Jeffrey Goldstein. *Handbook of Computer Game Studies*. Cambridge, MA: The MIT Press, 2006.

Ramsay, Stephen. *Reading Machines: Toward an Algorithmic Criticism.* Champaign: University of Illinois Press, 2011.

Ranciere, Jacques. "The Distribution of the Sensible: Politics and Aesthetics." In *The Politics of Aesthetics,* 12-30. London: Continuum, 2004.

Raskin, Jef. *The Humane Interface: New Directions for Designing Interactive Systems.* Boston, MA: Addison-Wesley Professional, 2000.

Reas, Casey, and Chandler McWilliams. *Form+Code in Design, Art, and Architecture.* New York: Princeton Architectural Press, 2010.

Reich, Steve. *Writings on Music: 1965-2000.* Oxford and New York: Oxford University Press, 2002.

Rémi. "The Maturity of Visual Programming." *craft ai,* September 29, 2015. Accessed April 11, 2016, http://www.craft.ai/blog/the-maturity-of-visual-programming/.

"Report: 90% Of Waking Hours Spent Staring At Glowing Rectangles." *The Onion,* June 15, 2009. Accessed April 29, 2016, http://www.theonion.com/article/report-90-of-waking-hours-spent-staring-at-glowing-2747.

Resmini, Andrea and Luca Rosati. *Pervasive Information Architecture: Designing Cross-Channel User Experiences.* Burlington, MA: Morgan Kauffman, 2011.

Reynaldo, Shawn. "In the Studio: Machinedrum." *XLR8R,* August 27, 2012. Accessed April 25, 2016, https://www.xlr8r.com/gear/2012/08/in-the-studio-machinedrum/.

Reynolds, Simon. "Maximal Nation: Electronic Music's Evolution Toward the Thrilling Excess of Digital Maximalism." *Pitchfork,* December 6, 2011. Accessed April 20, 2015. http://pitchfork.com/features/articles/8721-maximal-nation/.

Rhys-Taylor, Alex. "The Essences of Multiculture: A Sensory Exploration of an Inner-City Street Market." *Identities: Global Studies in Culture and Power* 20, no. 4 (2013): 393-406.

Ribeiro, Márcio, Paulo Borba and Claus Brabrand. *Emergent Interfaces for Feature Modularization*. Gewerbestrasse: Springer International Publishing, 2014.

Richardson, Ingrid. "Faces, Interfaces, Screens: Relational Ontologies of Framing, Attention and Distraction." *Transformations* 18 (2010).

________. "Ludic Mobilities: The Corporealities of Mobile Gaming." *Mobilities* 5 (2010): 431-47.

________. "Mobile Technosoma: Some Phenomenological Reflections on Itinerant Media Devices." *The Fibreculture Journal* 6 (2005).

________. "Pocket Technospaces: The Bodily Incorporation of Mobile Media." *Continuum: Journal of Media and Cultural Studies* 21 (2007): 205-15.

________. "Touching the Screen: A Phenomenology of Mobile Gaming and the iPhone." In *Studying Mobile Media: Cultural Technologies, Mobile Communication, and the iPhone*. Edited by Larissa Hjorth, Jean Burgess and Ingrid Richardson. New York: Routledge, 2012.

Richardson, Ingrid and Rowan Wilken. "Haptic Vision, Footwork, Place-Making: A Peripatetic Phenomenology of the Mobile Phone Pedestrian." *Second Nature: International Journal of Creative Media* 1 (2009): 22-41.

Rodgers, Tara. "Synthesizing Sound: Metaphor in Audio-Technical Discourse and Synthesis History." PhD diss., McGill University, 2011.

________. "Towards a Feminist Historiography of Electronic Music." In *The Sound Studies Reader,* edited by Jonathan Sterne, 475-89. New York and London: Routledge, 2012.

Roell, Craig H. *The Piano in America, 1890-1940.* Chapel Hill: University of North Carolina Press, 1989.

Romanyshyn, Robert. *Technology as Symptom and Dream.* New York: Routledge, 1989.

Rose, Frank. *The Art of Immersion: How the Digital Generation is Remaking Hollywood, Madison Avenue, and the Way We Tell Stories.* New York and London: W.W. Norton & Co., 2011.

Rosebush, Judson. "The Proceduralist Manifesto." *Leonardo* 22, no. 5 (1989): 55-6.

Rosenberg, Robert. "The Spatial Experience of Telephone Use." *Environment, Space, Place* 2, no. 2 (2010): 61-75.

Rosenfeld, Louis and Peter Morville. *Information Architecture for the World Wide Web.* Sebastopol, CA: O'Reilly & Associates, 1998.

Rovan, Joseph Butch. "Living on the Edge: Alternate Controllers and the Obstinate Interface." In *Mapping Landscapes for Performance as Research: Scholarly Acts and Creative Cartographies,* edited by Shannon Rose Riley and Lynette Hunter. New York: Palgrave Macmillan, 2009.

Ruffino, Paolo. "Independent Games: Narratives of Emancipation in Video Game Culture." *Loading…* 7, no. 11 (2012).

"Rural Legend." *Unity3D,* April 25, 2014. Accessed April 11, 2016, https://unity3d.com/showcase/case-stories/endless-legend.

Salen, Katie, and Eric Zimmerman. *Rules of Play: Game Design Fundamentals.* Cambridge and London: The MIT Press, 2004.

Sallis, Friedemann, ed. *Music Sketches.* Cambridge, UK: Cambridge University Press, 2015.

Sampson, Tony D. *Virality: Contagion Theory in the Age of Networks.* Minneapolis: University of Minnesota Press, 2012.

Schafer, R. Murray. *The Soundscape: Our Sonic Environment and the Tuning of the World.* Rochester, VT: Destiny Books, 1994.

Schloss, Joseph G. *Making Beats: The Art of Sample-Based Hip-Hop.* Middletown, CT: Wesleyan University Press, 2004.

Sevaldson, Birger. "Systems Oriented Design." Oslo School of Architecture and Design. Accessed April 12, 2016. http://www.systemsorienteddesign.net.

Sharples, Mike, et al. "Maker Culture: Learning By Making." In *Innovating Pedagogy 2013: Open University Innovation Report 2,* 33-5. Milton Keynes, UK: The Open University, 2013.

Shepard, Mark, ed. *Sentient City: Ubiquitous Computing, Architecture, and the Future of Urban Space.* Cambridge and London: The MIT Press, 2011.

Shneiderman, Ben. "Direct Manipulation: A Step Beyond Programming Languages." *Computer* 16, no. 8 (1983).

Simon, Bart. "Wii are out of Control: Bodies, Game Screens and the Production of Gestural Excess." March 5, 2009. Accessed November 12, 2015. http://itu.dk/~miguel/DesignReadings/Simon.pdf.

Sinnreich, Aram. *Mashed Up: Music, Technology, and the Rise of Configurable Culture.* Amherst, MA: University of Massachusetts Press, 2010.

Sisario, Ben and James C. McKinley Jr. "Drug Deaths Threaten Rising Business of Electronic Music Fests." *The New York Times,* September 9, 2013. Accessed April 18, 2016, http://www.nytimes.com/2013/09/10/arts/music/drugs-at-music-festivals-are-threat-to-investors-as-well-as-fans.html?_r=0.

Skjelten, Elisabeth. *Complexity and Other Beasts.* Oslo, Norway: The Oslo School of Architecture and Design, 2014.

Smalley, Denis. "Spectromorphology: Explaining Sound-Shapes." *Organised Sound* 2, no. 2 (1997): 107-26.

Smith, Jacob. *Vocal Tracks: Performance and Sound Media.* Berkeley and Los Angeles: University of California Press, 2008.

Spitz, Andrew. "Skube—A Last.fm & Spotify Radio." *Sound + Design,* October 16, 2012. Accessed December 12, 2014, http://www.soundplusdesign.com/?p=5516.

Stanyek, Jason and Benjamin Piekut. "Deadness: Technologies of the Intermundane." *The Drama Review* 54, no. 1 (Spring 2010): 14-38.

Stern, Craig. "What makes a game indie: a universal definition." *Sinister Design,* August 22, 2012. Accessed April 19, 2016, http://sinisterdesign.net/what-makes-a-game-indie-a-universal-definition/.

Sterne, Jonathan. *MP3: The Meaning of a Format.* Durham and London: Duke University Press, 2012.

________. *The Audible Past: Cultural Origins of Sound Reproduction.* Durham and London: Duke University Press, 2003.

________. "Sonic Imaginations." In *The Sound Studies Reader.* Edited by Jonathan Sterne. New York: Routledge, 2012.

"Steve Jobs introduces iMovie & iMac DV - Apple Special Event (1999)." EverySteveJobsVideo, uploaded December 21, 2012. Accessed March 9, 2016, https://youtu.be/ngW5qCBRwxk.

"Steve Jobs introduces rainbow iMacs & Power Mac G3 - Macworld SF (1999)." EverySteveJobsVideo, uploaded December 21, 2013. Accessed March 9, 2016, https://youtu.be/NuCYHrSig94.

Stinson, Jim. "Real-Time Sound Effects: The Foley Way." *Videomaker,* July 1, 1999. Accessed April 18, 2016, http://www.videomaker.com/article/7220-real-time-sound-effects-the-foley-way.

Stockwell, Stephen and Adam Muir. "The Military-Entertainment Complex: A New Facet of Information Warfare." *The Fibreculture Journal* 1 (2003). Accessed April 26, 2016, http://one.fibreculturejournal.org/fcj-004-the-military-entertainment-complex-a-new-facet-of-information-warfare.

Stone, Allucquère Rosanne. *The War of Desire and Technology at the Close of the Mechanical Age.* Cambridge: The MIT Press, 1996.

Suits, Bernard. *The Grasshopper: Games, Life, and Utopia.* Ontario: Broadview Press, 2005.

Sweetser, Penny. *Emergence in Games.* Boston: Course Technology, 2008.

Tanaka, Atau, et al. "Gestural Musical Affordances." In *Proceedings of the 9th Sound and Music Computing Conference,* 318-25. Copenhagen, 2012.

Tanaka, Atau. "Mapping Out Instruments, Affordances, and Mobiles." In *Proceedings of the 2010 Conference on New Interfaces for Musical Expression,* 15-8. Sydney, 2010.

Tarakajian, Sam. "An Interview with Ducky." *Cycling 74,* November 18, 2013. Accessed December 11, 2014. https://cycling74.com/2013/11/18/an-interview-with-ducky/.

Taylor, Timothy. "The Commodification of Music at the Dawn of the Era of 'Mechanical Music'." *Ethnomusicology* 51, no. 2 (2007): 281-305.

________. *The Sounds of Capitalism: Advertising, Music, and the Conquest of Culture.* Chicago and London: University of Chicago Press, 2012.

Taylor, Timothy D., Mark Katz, and Tony Grajeda, eds. *Music, Sound, and Technology in America: A Documentary History of Early Phonograph, Cinema, and Radio.* Durham and London: Duke University Press, 2012.

Terranova, Tiziana. *Network Culture: Politics for the Information Age.* London: Pluto Press, 2004.

"The Human Voice Is Human on the New Orthophonic Victrola." *David Sarnoff Library digital archive*, March 27, 1927. Accessed April 29, 2016, http://digital.hagley.org/cdm/ref/collection/p16038coll11/id/12569.

Théberge, Paul. *Any Sound You Can Imagine: Making Music/Consuming Technology.* Hanover, NH: Wesleyan University Press, 1997.

________. "The Network Studio: Historical and Technological Paths to a New Ideal in Music making." *Social Studies of Science* 34, no. 5 (2004): 759-81.

Thompson, Emily. *The Soundscape of Modernity: Architectural Acoustics and the Culture of Listening in America, 1900-1933*. Cambridge, MA: The MIT Press, 2004.

Tidwell, Jenifer. *Designing Interfaces: Patterns for Effective Interaction Design.* Sebastopol, CA: O'Reilly Media, 2005.

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

Tufte, Edward. "iPhone Resolution by Edward Tufte." Edward Tufte, uploaded November 25, 2008. Accessed March 25, 2016, https://www.youtube.com/watch?v=YslQ2625TR4.

Turing, Alan. "Computing Machinery and Intelligence." *Mind: A Quarterly Review of Psychology and Philosophy* 59, no. 236 (1950): 433-60.

Turkle, Sherry. *Alone Together: Why We Expect More From Technology and Less From Each Other.* New York: Basic Books, 2011.

Turner, Amber Leigh. "The history of flat design: How efficiency and minimalism turned the digital world flat." *The Next Web,* March 13, 2014. Accessed April 25, 2016, http://thenextweb.com/dd/2014/03/19/history-flat-design-efficiency-minimalism-made-digital-world-flat/.

Unger, Russ and Carolyn Chandler. *A Project Guide to UX Design: For User Experience Designers in the Field or in the Making.* Upper Saddle River, NJ: Pearson Education, 2009.

United States Senate. "Record Labeling: Hearing before the Committee on Commerce, Science, and Transportation." United States Senate, Ninety-ninth Congress, First Session on Contents of Music and the Lyrics of Records, September 19, 1985. Accessed April 28, 2016, http://www.joesapt.net/superlink/shrg99-529/.

Upbin, Bruce. "Why Los Angeles Is Emerging As The Next Silicon Valley." *Forbes,* August 28, 2012. Accessed April 29, 2016, http://www.forbes.com/sites/ciocentral/2012/08/28/why-los-angeles-is-emerging-as-the-next-silicon-valley/

Upton, Brian. *The Aesthetic of Play*. Cambridge, MA: The MIT Press, 2015.

Van Den Boomen, Marianne. *Transcoding the Digital: How Metaphors Matter in New Media.* Amsterdam: Institute of Network Cultures, 2014.

Van der Hoef, John. "Other Ways to Play: Independent Gaming's Aesthetics, Representation, and Industry." *Press Start to Drink: Video Games, Academics, Drinks,* October 22, 2008. Accessed April 18, 2016. http://pressstarttodrink.blogspot.com/2008/10/other-ways-to-play-independent-gamings.html.

Van Dijk, Jan. *The Network Society: Social Aspects of the New Media.* Thousand Oaks, CA: SAGE Publications, 1999.

Van Veen, Tobias and Bernardo Alexander Attias. "Off the Record: Turntablism and Controllerism in the 21st Century (Part 1)." *Dancecult* 3, no. 1 (2011).

Verbeek, Peter-Paul. *What Things Do: Philosophical Reflections on Technology, Agency, and Design.* University Park, PA: The Pennsylvania State University Press, 2000.

Vercellone, Carlo. "The new articulation of wages, rent and profit in cognitive capitalism." Queen Mary University School of Business and Management, February 29, 2008.

von Hornbostel, Erich M. and Curt Sachs. "Systematik der Musikinstrumente: Ein Versuch." *Zeitschrift für Ethnologie* 4 (1914).

Vreeland, Rich. "Philosophy of Music Design in Games – Fez." Hunter Short, uploaded April 20, 2013. Accessed March 30, 2016. https://www.youtube.com/watch?v=Pl86ND_c5Og.

Wang, Ge. "Ocarina: Designing the iPhone's Magic Flute." *Computer Music Journal* 38, no. 2 (Summer 2014).

Wardrip-Fruin, Noah. *Expressive Processing: Digital Fictions, Computer Games, and Software Studies.* Cambridge and London: The MIT Press, 2009.

Watts, Alan. *Nature, Man, and Woman.* New York: Vintage Books, 1991.

Weatherspoon, Josh. Interview by Author. Digital Recording. Los Angeles, CA. October 23, 2014.

Weiser, Mark. "The Computer for the 21st Century." *Scientific American* (September 1991).

Weiss, Allen S. *Breathless: Sound Recording, Disembodiment, and the Transformation of Lyrical Nostalgia.* Middletown, CT: Wesleyan University Press, 2002.

Weiss, Jeff. "Flying Lotus' Nocturnal Visions." *LA Weekly,* October 4, 2012. Accessed June 8, 2015. http://www.laweekly.com/music/flying-lotus-nocturnal-visions-2611706.

Whalen, Zach. "Play Along: An Approach to Videogame Music." *Game Studies* 4, no. 1 (2004). Accessed April 20, 2015. http://www.gamestudies.org/0401/whalen/.

"When Will the Bass Drop." *NBC,* n.d. Accessed April 18, 2016, http://www.nbc.com/saturday-night-live/video/digital-short-when-will-the-bass-drop/2783137.

Whitelaw, Mitchell. "System Stories and Model Worlds: A Critical Approach to Generative Art." In *Readme 100: Temporary Software Art Factory,* edited by Olga Goriounova, 135-54. Norderstedt: BoD, 2005.

Wiener, Norbert. *Cybernetics: Or, Control and Communication in the Animal and the Machine.* Cambridge and London: The MIT Press, 1965.

________. "Men, Machines, and the World About." In *Medicine and Science.* Edited by I. Galderston, 13-28. New York: International Universities Press, 1954.

________. *The Human Use of Human Beings: Cybernetics and Society.* Boston, MA: Houghton Mifflin, 1950.

Wilken, Rowan. "From *Stabilitas Loci* to *Mobilitas Loci:* Networked Mobility and the Transformation of Place." *The Fibreculture Journal* 6 (2005).

Williams, Raymond. *Television, Technology and Cultural Form*. New York and London: Routledge, 2003.

Wing, Jeanette. "Computational Thinking." *Communications of the ACM* 49, no. 3 (2006): 33-5.

Winner, Langdon. "Do Artifacts Have Politics?" *Daedalus* 109, no. 1 (Winter, 1980): 121-136.

Wishart, Trevor. *On Sonic Art,* edited by Simon Emmerson. Amsterdam: Harwood Academic Publishers, 1996.

Wodtke, Christina. *Information Architecture: Blueprints for the Web.* San Francisco, CA: New Riders, 2009.

Wojcicki, Susan. "Closing the Tech Industry Gender Gap." *Huffington Post,* January 27, 2016. Accessed April 26, 2016, http://www.huffingtonpost.com/susan-wojcicki/tech-industry-gender-gap_b_9089472.html.

Wooller, Rene, Andrew Brown, et al. "A Framework for Comparison of Processes in Algorithmic Music Systems." In *Proceedings from Generative Arts Practice,* 109-24. Sydney: Creativity and Cognition Studios Press, 2005.

Wozniak, Steve. *iWoz*. New York: W.W. Norton & Company, 2006.

Wurman, Richard Saul. *Information Architects.* New York: Graphis Inc., 1997.

Young, John. “Sound Morphology and the Articulation of Structure in Electroacoustic Music.” *Organised Sound* 9, no. 1 (2004): 7-14.

________. “The Extended Environment.” In *Proceedings of the 1994 International Computer Music Conference,* 23-6. Aarhus: Danish Institute of Electroacoustic Music, 1994.

Zak, Albin J. *The Poetics of Rock: Cutting Tracks, Making Records.* Berkeley: University of California Press, 2001.

Zattra, Laura. “The identity of the work: agents and processes of electroacoustic music.” *Organised Sound* 11, no. 2 (2006).

Zicarelli, David. “An Interview with William Kleinsasser.” *Cycling 74,* September 13, 2005. Accessed December 11, 2014. https://cycling74.com/2005/09/13/an-interview-with-william-kleinsasser/.

________. “How I Learned to Love a Program That Does Nothing.” *Computer Music Journal* 26, no. 4 (2002).

Zittrain, Jonathan. *The Future of the Internet—And How to Stop It.* New Haven and London: Yale University Press, 2008.