2005

Interactive computer music: a performer's guide to issues surrounding Kyma with live clarinet input

Roland Anton Karnatz
Louisiana State University and Agricultural and Mechanical College

INTERACTIVE COMPUTER MUSIC : A PERFORMER'S GUIDE TO ISSUES SURROUNDING KYMA WITH LIVE CLARINET INPUT

A Written Document

Submitted to the Graduate Faculty of the
Louisiana State University and
Agricultural and Mechanical College
in partial fulfillment of the
requirements for the degree of
Doctor of Musical Arts

in

The School of Music

by
Roland Anton Karnatz
Bachelor of Arts in Music, Victorian College of the Arts, 1990
Graduate Diploma in Music, Victorian College of the Arts, 1992
Master of Music, Virginia Commonwealth University, 2000
May 2005

Table of Contents

List of Figures

Abstract

Musicians are familiar with interaction in rehearsal and performance of music. Technology has become sophisticated and affordable to the point where interaction with a computer in real time performance is also possible. The nature of live interactive electronic music has blurred the distinction between the formerly exclusive realm of composition and that of performance. It is quite possible for performers to participate in the genre but currently little information is available for those wishing to explore it.

This written document contains a definition of interaction, discussion on how it occurs in traditional music-making and a brief history of the emergence of live interaction in computer music. It also discusses the concept of live interaction, its aesthetic value, and highlights the possibilities of live interactive computer music using clarinet and the Kyma system, revealing ways a performer may maximize the interactive experience. The document, written from a player's perspective, contains descriptions of possible methods of interaction with Kyma and live clarinet input divided into two areas: the clarinet can be used as a controller and the clarinet can be used as a source of sound. Information upon technical issues such as the speaker system, performance-space acoustics and diffusion options, possible interactive inputs, and specifically on microphone choices for clarinet is provided.

There is little information for musicians contemplating the use of Kyma; specifically clarinetists will find in this paper a practical guide to many aspects of live electronic interaction and be better informed to explore the field. This area has the potential to expand not only our performing opportunities, but might increase economic development. Application of interactive music technology can be used in a traditional recital and for collaborative work with other art forms, installation projects and even music therapy.

Knowledge of these programs also opens possibilities for sound design in theatre, film and other commercial applications.

Chapter 1. Introduction

Musicians are familiar with interaction in the rehearsal and performance of traditional acoustic music. The "interactive loop" of action and reaction among musicians is found in situations as diverse as a performer collaborating with a composer, a conductor with an orchestra, or members of a wind quintet or a jazz ensemble with each other. Computer technology has become sophisticated and affordable to the point where musical interaction between performers and computers in performance is also possible. Similar interactive relationships to those found in traditional acoustic music can be simulated with a computer. A musician can influence the way a computer reacts and in turn is influenced by the output of the computer. It is also possible to interact with a computer in ways not found in traditional music; Todd Winkler postulates that these "interactive techniques may suggest a new musical genre, where the computer's capabilities are used to create new musical relationships that may exist only between humans and computers in a digital world."[1]

Little information is available for the performer wishing to explore this field. The purpose of this paper, in conjunction with a lecture recital, is to highlight the possibilities of live interactive computer music using clarinet and Kyma, and reveal ways a performer may maximize the interactive experience.

Carla Scaletti and Kurt Hebel began developing Kyma while working as students at the University of Illinois.[2] Scaletti describes Kyma as a language for specifying, manipulating and combining sounds. The first version appeared in 1986 and there have been numerous

---

[1] Todd Winkler, *Composing Interactive Music: Techniques and Ideas Using Max* (Cambridge, Mass.: MIT Press, 1998), 4.
[2] Carla Scaletti, *Kyma Sound Design Environment* (Champaign, Illinois: Symbolic Sound Corporation, 1997), 9.

revisions.[3] It has a graphic user interface, and, given some knowledge of electronic manipulation of sound, its use is relatively intuitive. Kyma allows a clarinetist to augment playing skills in a traditional sense with composition, improvisation and "conducting the ensemble."

This document begins with a definition of interaction, noting its occurrence in traditional music-making, the emergence of live interaction in computer music, and efforts to categorize the genre. The document continues with information on current applications of live interaction in music, a discussion of possible future directions of these types of interaction and also addresses some aesthetic issues. A chapter is devoted to the choices of possible interactive inputs and information about suitable microphones for the clarinet. The remainder of the document is devoted to the description of the interactive possibilities with Kyma and clarinet, divided into two areas; the clarinet as a controller and the clarinet as a sound source.

Sources on Live Interaction

A review of available literature shows that relatively little has been written about live interaction; most authors have written from a compositional or computer programming perspective and not as an acoustic-instrument performer. The rapid development of technology has rendered most publications more than ten years old obsolete, since older technology bears little resemblance to that currently available. In fact, no one has published information addressing issues of interaction between Kyma and a performer. The most closely related work is that of Todd Winkler, *Composing Interactive Music: Techniques and Ideas*

[3] Carla Scaletti, "Computer Music Languages, Kyma, and the Future," *Computer Music Journal* 26, no.4 (2002): 73.

Using Max(Max is a similar program used in live interaction.).[4] Composer Robert Rowe also works with Max but concentrates on manipulations of musical material via MIDI information and shows little concern for modifying instrumental sound.[5] Camurri,[6] Godøy[7] and others are publishing work in artificial intelligence and interpretation of physical gesture which is only distantly related to the topic. There are also a number of general books with specific chapters dedicated to electronic and computer music, (Chadabe,[8] Manning[9] and Emmerson[10]) but they do they do not explore the subject in great depth.

Contemporary Music Review[11] published a particularly interesting collection of articles dealing with the aesthetics of live interactive music from a composer's perspective. Some issues relevant to this topic are discussed in the paper, however, a full exploration of the subject of aesthetics is too substantial and outside the scope of this document.

A clarinetist has few resources on the theory and practicality of interactive computer music. This document provides the background and practical knowledge to begin exploring live interaction with Kyma.

---

[4] Todd Winkler, *Composing Interactive Music: Techniques and Ideas Using Max* (Cambridge, Mass.: MIT Press, 1998).
[5] Robert Rowe, *Interactive Music Systems: Machine Listening and Composing* (Cambridge, Mass.: MIT Press, 1993).
[6] A. Camurri, P. Coletta, M. Ricchetti, and G. Volpe. "Expressiveness and Physicality in Interaction" *Journal of New Music* 29, no.3 (2000).
[7] Rolf Inge Godøy, "Gestural Imagery in the Service of Musical Imagery," in *Music and Artificial Intelligence : Second International Conference, 2002*. Edinburgh, Scotland: 12-14 September 2002, edited by Christina Anagnostopoulou. (Berlin: Springer-Verlag, 2002).
[8] Joel Chadabe, *Electric Sound: The Past and Promise of Electronic Music.* (Upper Saddle River, N.J.: Prentice-Hall, 1997).
[9] Peter Manning, *Electronic and Computer Music* (New York: Oxford University Press, 2004).
[10] Simon Emmerson, ed., *Music, Electronic Media and Culture* (Burlington, V.T.: Ashgate, 2000).
[11] Marc Battier, ed., *The Aesthetics of Live Electronic Music* (Overseas Publishers Association, published by license under the Harwood Academic Publishers imprint, 1999).

Chapter 2. Interaction

A Two-Way Street

In his introduction to the concept of interaction, Todd Winkler succinctly states, "Interaction is a two-way street."[12] In traditional music performance one often thinks of this phenomenon in terms of communication and collaboration with others; interaction requires both action and reaction. As Winkler says, "Music has always been an interactive art in which musicians respond to each other as they play."[13] Performers react to aural and visual stimuli, continuously altering and shaping their response and simultaneously creating the material to which they respond.

Many variants of interaction can be found in traditional music practice. The relationship between a conductor and orchestra is one where the orchestra responds to visual signals from the conductor who, in turn, is influenced by the aural and visual feedback of the orchestra in coordination with the score. A wind quintet involves complex interaction among all five players. For any member of the group, there are four sets of aural and visual signals with which to interact individually or collectively. Even an unaccompanied musician interacts with the audience and the acoustics of the performance space. A performer is aware of, and will respond to audience recognition, encouragement and reaction to the performance. Both solo performers and ensembles interact with the reverberation of the performance space and will constantly adjust dynamics and timing of events according to aural feedback.

Electronic Music and Interaction

The ability to record sound has radically changed many of the interactive situations found between performers and audience in traditional music. In the playback of a recording,

[12] Todd Winkler, *Composing Interactive Music: Techniques and Ideas Using Max* (Cambridge, Mass.: MIT Press, 1998), 3.
[13] Ibid., 4.

there is no longer live interaction between the performers and the "audience". Instead, the listener has a very basic form of interactive control over the time, place and volume of the recording playback.

Even the recording of some styles of traditional acoustic music can now be done with little interaction. Multitrack recording techniques have enabled performers to play their parts at completely different times and even in different places without any interaction. It is no longer a two-way street; issues of interaction are replaced with those of coordination found also in the performance of works for instrument and tape.

In its early forms, electronic music generally did not involve live interaction. Works for tape alone dispense with the necessity of a performer and present only limited options of playback. The ability to record enabled Pierre Schaefer to assemble sounds taken out of the context of their natural existence. His *Etude aux Chemins de Fer* (1948) was the first such assemblage and marked the beginning of what Schaefer referred to *musique concrète.*[14] Technical development quickly enabled wax phonograph and gramophone recording technology to be replaced by the more advanced magnetic tape. In 1949, stereo tape was introduced, and the first commercial splicing block became available.[15]

Two schools of thought developed as to the possible sources of the original sound material for early works for tape; Schaeffer's *musique concrète* approach used existing sounds and manipulated them through physical processes such as splicing, looping, and altering tape speed. On the other hand, composers in Cologne, led by Karlheinz Stockhausen, were drawn towards the aesthetic of *electronische musick*, the synthesizing of sound with sine wave

[14] Joel Chadabe, *Electric Sound: The Past and Promise of Electronic Music* (Upper Saddle River, N.J.: Prentice-Hall, 1997), 26.
[15] Ibid., 31.

generators, filtering, modulation and other processes.[16]

In 1957, the first computer generated sounds were created by Max Mathews at Bell Telephone Laboratories.[17] The program, Music 1, could produce monophonic sounds using only a single waveform; pitch, loudness and duration were the only controllable parameters. By the early 1960's, Mathews developed Music 3, the first general-purpose program for sound synthesis.[18]

Interaction with a computer was made possible through the ability to program, however, it was an extremely limited, non-real-time process throughout the 1960's. A composer's input was processed in successive stages of calculation and accumulated as samples on a magnetic tape. This tape was subsequently processed onto a normal audio tape. F. Richard Moore, who was responsible for transferring samples to sound, described the situation: "People from Princeton used to drive their tapes up and leave them with us. Then we'd convert them … and they'd show up two weeks later and hear their music. A two week turnaround was pretty good…"[19] Moore further described the Computer Audio Research Laboratory system at the University of California at San Diego; by the late 1970's it could process a continuous run of sound up to twenty minutes long and, when the system was not busy, "would take anywhere between ten and one hundred seconds to compute one second of sound".[20]

As processing speed increased, the computer's response time to input decreased. Various live interactive instruments were developed. The CEMS and the SalMar Construction

[16] Ibid., 38.
[17] Ibid., 108.
[18] Charles Dodge and Thomas Jerse, *Computer Music* (USA: Schirmer, 1997), 16.
[19] F.Richard Moore, quoted in Joel Chadabe, *Electric Sound: The Past and Promise of Electronic Music* (Upper Saddle River, N.J.: Prentice-Hall, 1997), 113.
[20] Joel Chadabe, *Electric Sound: The Past and Promise of Electronic Music* (Upper Saddle River, N.J.: Prentice-Hall, 1997), 123.

appeared in the early 1970's and were the first *interactive composing instruments*.[21] Around 1974 the Synclavier, the first digital synthesizer, was developed by Jon Appleton and Sydney Alonso.[22] During the 1980's, numerous computer programs provided the ability to synthesize sound and would react to input in real time. These included; SAL by Salvatore Martirano, Cypher by Robert Rowe, and M by David Zicarelli.

Interaction in Computer Music- Approaches to Categorization

Joel Chadabe describes well the emerging recognition of interaction:

> In summary, during the 1970s and 1980s many composers and performers gained experience in creating and working with real-time interactive processes. By the mid-1990s, a substantial body of knowledge had been formed and the idea had become credible. And people began to say that there were two reasons to use electronics in making music. One was to access sound. The other was interaction.[23]

Non-real-time musical interaction with computers began with the first sounds made on computer. With increased processing speeds, processes originally used for tape music became flexible and quick enough for real time interaction to be a viable possibility. Guy Garnett describes interactive pieces as "works wherein the performer in some way controls the electronics or the electronics affect the performer's sounds."[24] Robert Rowe uses the term *interactive computer music system* and defines it as one "whose behavior changes in response to musical input."[25]

The ability for computers to engage in live interaction requires a system that processes an input and generates an output without apparent delay or latency. If the input is from a source that has the ability to modify its behavior according to stimuli, then both action and

[21] Ibid., 291.
[22] Jean-Claude Risset, "Composing in Real-Time?" *Contemporary Music Review* 18:3 (1999): 32.
[23] Joel Chadabe, *Electric Sound: The Past and Promise of Electronic Music* (Upper Saddle River, N.J.: Prentice-Hall, 1997), 323.
[24] Guy E Garnett, “The Aesthetics of Interactive Computer Music,” *Computer Music Journal* 25:1 (2001): 21.
[25] Robert Rowe, *Interactive Music Systems : Machine Listening and Composing.* (Cambridge, Mass.: MIT Press, 1993), 1.

reaction can occur. Attempts to categorize the nature of interaction have led to several conflicting approaches, but ultimately any system is characterized by the processes applied to its input. Description of these interactive processes is either a description of the relationship between performer and computer, or a description of the actual processes in the system.

Composer Joel Chadabe categorizes interaction according to the role a performer plays in the process. His three basic models are: "(1) a performer is a performer who performs someone else's composition, (2) a performer is a composer who composes by controlling an algorithm in real time, and (3) a performer is an improviser who improvises by controlling an algorithm in real time while reacting to new information generated by the instrument."[26]

Robert Rowe, a prominent leader in the field of live interaction, is concerned predominantly with communication of *musical* data (notes, intervals, harmonic relationships and rhythms) and its processing. Rowe talks of interactive music systems presenting live realization of formularized generative musical processes; processes well-established as traditional compositional techniques in genres ranging from Guido d'Arezzo's chant to twentieth century serialism.[27]

In categorizing the nature of the interaction between computer and performer, Rowe presents what he terms "a multi dimensional approach".[28] At one level, programs can be score driven; a collection of predetermined events or stored music fragments are aligned with input sound. Accompaniment programs that follow a performer while realizing the accompaniment are in this category. Alternately, Rowe's category of performance-driven programs does not anticipate the realization of a particular score.[29]

[26] Joel Chadabe, "The Performer is Us," *Contemporary Music Review* 18:3 (1999): 29.
[27] Robert Rowe, "Aesthetics of Interactive Music," *Contemporary Music Review* 18:3 (1999): 84.
[28] Ibid., 7-8.
[29] Ibid., 7.

Another dimension of Rowe's categorization of the nature of interaction is the differentiation between instrument paradigm and player paradigm systems. In an instrument paradigm system, the computer processes the input as an extension of the musical instrument, and the result is thought of as a solo. A player paradigm results in the construction of a separate entity which will vary in the degree to which it interacts with the live input and produces something more like a duet.

Highlighting his own approach to interactive computer music, Todd Winkler, another prominent leader in the field, presents a different method of categorization. Unlike Rowe, Winkler displays a more *sound-based* approach, extracting and manipulating elements from the input of raw sound data. The term "sound sculpture" would be applicable in describing Winkler's approach to interaction. Winkler presents a more detailed categorization of the type of interaction between humans and computers by relating it to interactive models in traditional music: the conductor model, chamber music model, improvisation model and free improvisation model.[30]

In his conductor model, he notes the conductor is master-controller of an orchestra, acting as the single source for directing the players by controlling time-flow, dynamic change and acoustic balance. Performers can control interaction with a computer in a similar way, through the computer's real-time analysis of the input. In the conductor model, score-following techniques require the computer to follow the performer's input, matching it to a stored version of the score. This is then coordinated with the computer's performance score.

In addition to accompaniment programs, devices such as Max Mathew's Radio Baton also closely resemble the conductor model. Batons containing low frequency radio

[30] Todd Winkler, *Composing Interactive Music: Techniques and Ideas Using Max* (Cambridge, Mass.: MIT Press, 1998), 23-27.

transmitters are waved in a manner similar to conducting over a surface containing receivers. The signals are used to control tempo, dynamics and other musical aspects of a score stored in a computer.[31]

From this point, Winkler departs from Rowe's categorization and divides the latter's *performance-driven* category into several areas. His chamber music model involves the complex type of interaction found in a string quartet, for example, where musicians reciprocally influence each other's output. In interaction with a computer, the performer and computer share control of events, and the taking and yielding of this is a dynamic element of a performance.

Winkler's improvisation model refers to the type of interaction in a jazz ensemble which occurs through a formal structure and shared conceptual framework. Computers can recognize patterns, identify such musical attributes as scale types and chord progressions, and be programmed to respond through the simulation of shared assumptions and implied rules of the genre. The interactive performance is a unique and unpredictable event occurring within scripted boundaries.

The free improvisation model offers a complex level of interaction involving the spontaneity, expressiveness and unpredictability of the genre. The computer may produce music that is not obviously related to the performers input and the interaction is obscured. Neither the computer or performer may be completely "in charge," but each will have some influence on the other's reaction.

Indicative of this genre's relative youth, there exists various organizational approaches to live interaction and a lack of any formalized standard. The systems of categorization

[31] Joel Chadabe, *Electric Sound: The Past and Promise of Electronic Music* (Upper Saddle River, N.J.: Prentice-Hall, 1997), 231.

described above highlight individual approaches and differing styles of composition. Neither approach to interaction need be employed exclusively, but the basic differences between manipulation of either musical data or sound-based input will distinguish work in this field for some time. Organization of the genre will also be affected by the emerging new environments in which interaction will be present; no longer is it the exclusive domain of composers.

Interactive Environments and the Future

Interaction with a computer was originally the domain of composers. As processing speeds increased, live interaction became a viable possibility; performers became involved in the process and an audience was able to witness it. With improved interfaces, interactive technology has become considerably more user-friendly and enabled performers with limited knowledge of the original programming language access to the process of interaction. The performer can now play the role of composer and make compositional decisions as to the form and content of an interactive work.

The place of the listener may also change with new interactive technology. These new associations are reflected by Morton Subotnick who lists three environments where interaction can now take place: public performances, installations, and home media.[32] The possibility that a new relationship will emerge between composer, performer and listener is suggested in writings as diverse as that of Jacques Attali and Glenn Gould.

Jacques Attali's predictions on music include interaction and the emergence of composition as the central driving force behind its existence. The final two chapters of his 1985 book, *Noise: The Political Economy of Music,* are titled "Repeating" and

[32] Morton Subotnick, "Interactive Performance Environment" *Contemporary Music Review* 18:3 (1999): 114.

"Composition". [33] In these chapters, Attali outlines his ideas on the current state of music in society and the possible direction it will take. His central theory is that the listener is gaining increasing control over music. Repetition, equating to mass production, is the environment in which noise, or music, is easily copied, distributed and purchased. Rather than purchasing a ticket to attend a concert at a pre-determined time, one takes the music home and has control over the time, place and volume it will be played. An extension of this state of repetition is to bypass the purchase of specific music and gain even greater control through composition. “The listener is the operator.”[34] Attali goes further, predicting the reflection of this in the economy, when the “bulk of commodity production then shifts to the production of tools allowing people to create the conditions for taking pleasure in the act of composing.”[35]

Glenn Gould also heralds the possibilities of interaction. “As this medium [electronics] evolves, as it becomes available for situations in which the quite properly self-indulgent participation of the listener will be encouraged, those venerable distinctions about the class structure within the musical hierarchy – distinctions that separated composer and performer and listener – will become outmoded.”[36]

This phenomenon is demonstrated in Morton Subotnick’s *All My Hummingbirds Have Alibis*, (1993) the first musical composition created specifically as a multi-media CD-ROM.[37] The listener is required to interact by selecting the ordering of the sections and which visual elements will be seen during the performance. Similarly, Peter Gabriel’s *Xplora1* presents the listener with the ability to re-mix the track levels in real time and Todd Rundgren’s *No World*

---

[33] Jacques Attali, Brian Massumi trans., *Noise: The Political Economy of Music* (Minneapolis: University of Minnesota Press, 1985)
[34] Ibid., 135.
[35] Ibid., 145
[36] Glenn Gould, *The Glenn Gould Reader* (New York: Knopf, 1984), 351.
[37] Joel Chadabe, *Electric Sound: The Past and Promise of Electronic Music* (Upper Saddle River, N.J.: Prentice-Hall, Inc., 1997), 332.

Order allows interaction with tempo-related controls.[38] The electronica band *Oval* now produces "algorithmic software to be used by its audiences to compose their own Oval music."[39] Sampling and remixing programs such as *Acid* and *Live* allow people with no traditional training in composition to participate in the production of a variety of loop-based music styles.

The interactive nature of these products will continue to become more complex and further obscure the boundary between listener, performer, and composer. The skills and knowledge required for live interactive performance are similar to those needed in other environments where interaction takes place, and it is possible for a musician to participate in the design, creation and use of interactive systems in these new environments.

---

[38] Ibid., 333

[39] Christopher Cox and Daniel Warner, *Audio Culture* (New York: Continuum, 2004), 208.

Chapter 3. Aesthetics

Performance Practice, Compositional Practice and Interpretation

While a full exploration of the aesthetics of live interaction is beyond the scope of this document, a discussion and summary of some published views is relevant. At first glance as a performer, live interaction solves a lot of problems; it maintains the dynamic nature of traditional live performance and expands the possibilities of electronic music from simply a medium performed from recording. Some published writings, notably by composers rather than performers or audience, express concerns over issues related to the genre. A substantial source of writings on the aesthetics of live electronic music is an issue of *Contemporary Music Review* in which "composers give insight into how they see its impact on the audience, the performance situation and ultimately on the music."[40]

The major concern expressed by several writers was the lack of performance, interpretative and compositional traditions in electronic music, and that these will never develop to the comparable level of traditional music. Jean-Claude Risset makes the point that technology changes so fast that systems become obsolete in the progress. Rather than adapting their composition to a new system, composers tend to write a new piece. Consequently there is no chance to develop performance tradition and in Risset's words, “It brings the risk of a perishable, memoriless electronic art.”[41]

Jon Appleton notes that live electronic music, unlike tape music, has the potential to provide a different interpretation at each performance in much the same way as traditional music. He suggests the problem however, is that most people rarely hear the same work

[40] Marc Battier, "Introduction", *Contemporary Music Review* 18:3 (1999): 1.
[41] Jean-Claude Risset, “Composing in Real-time?,” *Contemporary Music Review* 18:3 (1999): 35.

twice, listening only to its novelty and seeking a new experience each time.[42] Marco Stroppa adds to this the argument that in traditional music, interpretation is "simply defined as the performer's own contribution to what is notated in the score…", while in electronic music the score "hardly exists", and therefore the notion of interpretation may not either.[43]

Interactive electronic music may not have a score traditionally notated on paper, particularly when the live input is improvised; however, the score does exist in a different form. Each algorithmic process in a live interactive work can be viewed as some form of physical representation. A performer or composer can analyze these algorithms that process input, and also see some method of organization of those processes over time. As with a score of traditional music, any process can be revised, edited or moved around the time frame. Likewise, the realization and interpretation of an interactive work involves a processing, by the performer, of a composer's representation of musical ideas – the same process as found in traditional music.

The lack of repeat performances resulting in no tradition of interpretation is not a phenomenon associated solely with electronic music. It is the nature of any new art music that most will only receive a single performance before an audience moves on to the next experience. Interpretive tradition has not developed to the extent it has with traditional music because more time is needed for a repertoire with recognized masterpieces to be established.

Tape music, a precursor to live interactive music, was regarded as a fixed statement, free of interpretation. This has changed remarkably with the recognition of interpretation through sound diffusion in the performance of tape music. During playback, the performer has become the person moving the sound about the space. Each performance will be entirely

[42] Jon Appleton, "Reflections of a Former Performer of Electroacoustic Music," *Contemporary Music Review* 18:3 (1999): 15-19.
[43] Marco Stroppa, "Live Electronics or…Live Music?" *Contemporary Music Review* 18:3 (1999): 53-54.

different, and the final applause is in part for the diffuser's role in realizing the work. Interpretation is recognized to such an extent that competitions are now held where performers are judged on their diffusion of the same work under the same conditions. In time, live interactive music will follow a similar path with the development of some form of interpretive aesthetic.

Other Aesthetic Considerations

Marco Stroppa contends that "the available algorithms for real-time processing are limited and sometimes yield quite predictable, stereotyped timbres."[44] Most sound processing algorithms developed in non-real-time programs are now available for use in real-time applications. There is a stable, common set of processes widely recognized, understood, and described in detail in books such as *Computer Music* by Charles Dodge and Thomas Jerse[45] and *Computer Music Tutorial* by Curtis Rhodes[46]. These are tools analogous to compositional processes in traditional music, and likewise provide a source of infinite variation. Just as in traditional music, resources can be manipulated and arranged in more or less predictable and inspired ways.

Other writers acknowledge some strengths of interactive music through the powers of computer processing. Chris Chafe writes, "Digital technology provides a means for endless variations of any extent, it also provides the precision to exactly repeat any pattern, no matter how random."[47] The possibility to execute this ability live is beyond the realm of traditional acoustic performance. A similar power lies in the ability of interactive systems to "…generate material based on analysis of improvisation as easily as they can on analyses of

[44] Ibid., 48.
[45] Charles Dodge and Thomas Jerse, *Computer Music* (USA.:Schirmer, 1997).
[46] Curtis Roads, *The Computer Music Tutorial* (Cambridge, Mass.: MIT Press, 1996).
[47] Chris Chafe, "Interplay(er) Machines," *Contemporary Music Review* 18:3 (1999): 94.

notated music."[48] This unique situation has simultaneously extended the domain of composition and increased the creative responsibility of the performer and computer program.

David Zicarelli highlights the fact that the ability of interactive systems to respond immediately has also affected the process of creativity. "You can start with little or no concept - and with immediate feedback you can decide in what direction to go next."[49] While this may not always be the most effective way to arrive at the form of a composition, immediate results from experimentation allow extensive exploration of the manipulation of every detail in an algorithm.

Common to both traditional music and interactive music, the live element offers the challenge of risk for the performer and the engagement of the audience witnessing this challenge. Likewise, the visual spectacle presented by interactive music, absent in the performance of tape music, allows an audience to respond to the familiarity of a performer onstage.

The final aesthetic issue to be considered in this chapter is the degree to which an audience perceives interaction and subsequently their understanding of the relationship between input and output of an interactive system. A particularly complex issue, there is little research in the area of perceptual correlation between an action/ input and response/ output. In traditional music, the audience is familiar with many of the interactive loops they witness between performers. The elapsed time between action and response, and the relationship of the content between these is clear: a conductor gives a downbeat and everyone plays, a member of a chamber group performs a rubato and all follow, a singer breathes and the accompanist prepares.

---

[48] Robert Rowe, "Aesthetics of Interactive Music," *Contemporary Music Review* 18:3 (1999): 53-54.
[49] David Zicarelli, quoted in Joel Chadabe, *Electric Sound: The Past and Promise of Electronic Music* (Upper Saddle River, N.J.: Prentice-Hall, Inc., 1997), 317.

Interactive computer systems can appropriate the familiar relationships found in performance of traditional music. A one to one correspondence - for example, when the pitch of the output changes directly according to the pitch of the input - is generally an easily perceived relationship. The events are contemporaneous and have direct correlation through pitch.

It is arguable that there are also interactive relationships in traditional music of which a listener is not aware; an interpretation of phrasing by one player may be a reaction to that presented by another earlier in the piece and may not be apparent to the audience. Interactive computer systems can present unfamiliar, obscure and indirect correspondence between action and response. The time elapsed between events is an important factor in audience perception. Interaction can occur where the reaction is not contemporaneous to the action. An input may be processed to an output, but presented much later in the work; the listener is likely to loose the connection between the two events.

The occurrence of two events in close proximity that are related simply by time and not content can also leave some ambiguity over the relationship. Perceived interaction may well be obtained by outputs that do not respond at all to the input, and yet the alleatoric coordination of events gives some impression that interaction occurs.

The triggering of an unrelated output event by a specific input event, while very controlled, may or may not be obvious interaction to the audience. A pre-designated level of pitch from the input, for example, can be programmed as the control signal for another sound to start, stop or change. The computer's response to a specific note will be perfectly clear for the performer, but, in the midst of many notes, an audience will not connect that particular note to the subsequent response.

When using interactive computer systems, compositional decisions on the meaningfulness of the interaction and the need of the audience to perceive it must be made. There are infinite variations upon the relationship of action to reaction. The participation of the performer in the interactive loop and the witnessing of this by an audience is one of this genre's greatest strengths.

Chapter 4. Practical Considerations

Inputs

The response from an interactive music system is reliant upon the input the system receives and how it is programmed to process that input. Through the interface of a sensor, such as a microphone, video camera or pressure sensor, computers can receive input analogous to many of the human senses. The type and detail of data from sensors can vary from a simple on-off switch to complex audio and video signals. It is possible to enter input from any source that can be measured in real time – for example the temperature of a room – however, the ability of that source to interact and the sophistication of its response must be taken into account; there must be meaningful action and reaction.

Devices that track and measure the *physical actions* of a performer can be categorized into two groups according to their output. The first group of sensors, often called controllers, provide a musical interface and transform musical data to the computer. Devices such as MIDI (musical instrument digital interface) keyboards have a similar physical interface to a piano and process the player's physical actions into MIDI data. Similarly, an electronic wind instrument mimics the fingerings and mouthpiece characteristics of single reed instruments and outputs MIDI data. MIDI , a protocol for transforming musical data is described by Robert Rowe as "… an abstract representation of audio, which provides no information about the timbre or change in dynamic of a note."[50]

The second group of sensors that measure physical action output a changing voltage which can be scaled or processed accordingly. Currently available sensors include a dual-axis accelerometer, pressure sensor, simple on/off switch and bend/flex sensor.[51] Many of these

---

[50] Robert Rowe, *Machine Musicianship* (Cambridge, Mass.: MIT Press, 2001), 29.
[51] Electrotap, http://www.electrotap.com/sensors/ [commercial website]; Internet; accessed 1 December 2004.

devices are very small and light, enabling simple attachment to an acoustic instrument, or the body of a player. A switch on the instrument, for example, expands the options for control, while a flex sensor could be attached to a finger to input information concerning the physical nature of playing.

A common characteristic of physical sensor devices is the fact that they produce no acoustic output. All sound originates from the output of the computer, and there are no issues of balance between the acoustic and processed sounds. When this type of sensor is used in interactive music, the absence of an actual input sound requires the audience to find connections between their visualization of the physical action of the performer and the aural reaction of the computer. This is analogous to the interaction between the visual signals of a conductor and the aural response of an orchestra.

Input from visual data is possible with remote sensors using ultrasound, infra-red and light-beam systems to detect proximity and movement. The two-dimensional scaled output from these can be used in a manner similar to the output of physical action sensors. Video signal is a particularly detailed representation of visual data and it is possible to use this in limited ways for interaction. It is, however, difficult to isolate and track a particular parameter from a video signal and present meaningful interaction of it with sound, while also addressing the aesthetic issue of audience perception and understanding of that interaction.

Simon Emmerson discusses biophysical interfaces that have "remained on the fringes of experimental music throughout both analogue and digital developments, [but] may yet move to the forefront of music interface technology."[52] As with many of these sensors, meaningful interaction between the input source and the aural reaction may be difficult to achieve.

---

[52] Simon Emmerson, ed., *Music, Electronic Media and Culture* (Burlington, V.T.: Ashgate, 2000), 200.

The capture and digitizing of sound has a comparatively long history. It is possible to input a very detailed audio signal, although there are still limitations for some processes. Pitch tracking, for example, is generally limited to a monophonic audio signal. If the audio signal consists of more than one pitch, the computer will treat the data as a single tone creating errors in the tracking; sometimes there is even difficulty identifying the fundamental from its overtones. Pitch tracking algorithms can therefore seem unreliable. While the algorithms analyzing audio input are not sufficiently developed to differentiate and quantify the existence of more than one fundamental tone, the same complex signal can be processed in full for pitch-shifting because similar detail of analysis is not required.

All input systems currently available have limitations and there is no single all-inclusive representation of *music* for input into a computer.[53] Recognition of musical attributes such as phrase delineation, meter or musical character remain difficult tasks with either audio or MIDI input. Overcoming this would require more advanced inputs that contained specific information on musical attributes, or alternatively, more sophisticated software to recognize musical attributes from the input systems currently available.

While MIDI presents very specific information, it is limited to only a small number of musical parameters. At the same time, musical parameters are sometimes extracted from digital audio signals. Inputs from non-musical sources such as physical sensors and video require analysis of the input data and ability of the source to interact in some meaningful way. Evident in the contrasting approaches to interactive computer music, the choice of a suitable input depends on how the system is programmed to interact; Robert Rowe manipulates predominantly musical data via MIDI input and Todd Winkler's sound-based manipulations are more suited to audio input.

[53] Robert Rowe, *Machine Musicianship* (Cambridge, Mass.: MIT Press, 2001), 31.

While selection of the most suitable method of input depends on the processes the system applies to the incoming data, there are several advantages for an instrumentalist to use audio input as the primary source for live interactive music. A musician already has remarkable control over a sound producing mechanism and is trained to produce fine nuance and expression via the *sound*. While it is difficult for a computer to recognize musical attributes, it is easy for it to represent sound; capturing the sound will allow possible transfer of other intangible musical qualities, as well as quantifiable ones such as pitch, timbre, and dynamics.

In traditional acoustic music, musicians interact in a sophisticated manner through predominantly aural stimuli. Transferring the sound itself, rather than the physical action of producing it, to the input of a computer system preserves many of these possibilities for interaction. Preserving an environment that allows a musician to perform on his/her accustomed instrument is the most efficient and flexible single-input solution. The one additional component necessary is a microphone.

Capture of the Clarinet's Sound

A crucial element in a live interactive system with clarinet is the capture of the instrument's sound. The microphone required is one that can accurately record the clarinet within an often noisy environment. Microphones are traditionally used for either amplification of a live performer, or for studio recording and their properties differ accordingly.

A microphone used for amplification is designed to pick up the sound of a specific instrument, usually within the context of a noisy environment. The microphone's directional nature (the position in space from where the sound is captured) is of more importance than the fidelity of the signal. Alternately, a microphone designed for recording purposes must pick up

the sound in its vicinity as accurately as possible. Specific isolation of sound from a particular direction is of less importance. The ideal microphone for an interactive system requires both of the above properties. It is necessary to capture a detailed representation of the clarinet sound for subsequent processing, but also to avoid the processed signal from the speakers feeding back into the input.

A microphone can sense vibrations from the surroundings in two ways. Contact microphones work by picking up vibrations from a solid object, in this case an instrument, and cannot sense vibrations transmitted through the air; therefore, contact microphones pick up a very specific source of audio and are excellent for live amplification. There are few problems with feedback. The standard contact microphone readily available and used by many clarinetists is a product from the Barcus Berry company. This can either be attached to the reed, or into a hole drilled in a spare clarinet barrel used specifically for that purpose. Unfortunately the sound captured from any one particular place on the instrument is not an accurate representation of what one hears.

Dynamic and condenser microphones record noise from the surrounding air. There are many variations available, ranging from designs for recording, to specific application on a particular instrument for amplification. The clarinetist can play in front of a recording quality microphone on a stand, but the slightest movement of the instrument around the microphone changes the volume and quality of the captured sound. Feedback is also a problem in this situation. Fixing a small condenser microphone to the instrument solves the unintended variations in volume and increases mobility of the player. Attached to the bell of the clarinet, with directional pick-up, a microphone will capture only the sound emanating from that area.

Again, this does not provide the best sound, as some frequencies are projected differently from tone holes higher up the instrument.

The *WS* model from Applied Microphone Technology (AMT) is an effective solution to capture a better sound from the clarinet while still avoiding feedback.[54] This system consists of two condenser microphones attached to a clamp on the bell. One microphone picks up sound from the vicinity of the bell while the other is positioned on a flexible arm above the keys; the mix of the two provides a more realistic sound.

The AMT system seems to be the best option currently available. F.Gerrard Errante, a prominent figure in contemporary clarinet performance, summed up the choices: "The market seems to have settled on Barcus Berry and AMT. There is no question in my mind that the quality of the Applied Microphone Technology condenser mics is superior to that of the Barcus Berry contact mics."[55] Likewise, the composer/performer Burton Beerman advocates the AMT product as his preferred choice for clarinet at this time.[56]

Speakers, Diffusion and Acoustics

Several other practical issues at the output end of an interactive computer system present some opportunity for control. The number and quality of speakers available at a recital hall is variable and unless one owns such equipment, use of the available resources cannot be avoided. While this situation is improving and may be standardized in the future, possible options for performance range from stereo to eight or more channels. A decision at the early compositional stage needs to be made concerning how sound diffusion will be achieved. One

---

[54] Applied Microphone Technologies, http://www.appliedmic.com/ , [commercial website]; Internet; accessed 10 January 2005.
[55] F. Gerrard Errante, "re:microphones for clarinet," [e-mail]; from fgerrante@aol.com; received at rkarna1@lsu.edu, 17 October 2004.
[56] Burton Beerman, "re:microphones," [e-mail]; from bbeerman@bgnet.bgsu.edu; received at rkarna1@lsu.edu, 1 October 2004.

can have a fixed work in stereo, a work with optional live diffusion requiring a designated diffuser, or in some cases, diffusion can be pre-programmed in the system.

The acoustics of particular halls will also have a significant effect on a performance. Considerable time must often be spent adjusting the balance between processed sound, live acoustic sound and amplified live sound. Natural reverberation may be strong enough in some halls to require adjustment of any processed reverb, however, certain frequencies may also need attenuating to balance the sound in differing acoustic spaces. It is difficult for the performer onstage to be aware of these acoustic issues; a second person should be considered to operate the mixing desk from within the audience and make the necessary adjustments.

Conclusion

For a performer wishing to participate in interactive computer music, use of his/her accustomed instrument as the primary source of input for the system is the most practical solution. However, the search for improved microphones, speaker systems and perfect acoustics will continue, finances permitting, indefinitely. Constant advances in technology and design present new options which are evaluated by the user and adopted for their improvements. For a clarinetist, knowledge of microphones, speaker systems and sound diffusion is valuable not only in the field of interactive computer music but can be applied in related areas such as recording techniques and the commercial music industry. Likewise, knowledge of Kyma's use in live interaction is also valuable for application in sound design for film, video games and even acoustic research.

Chapter 5. Kyma

Kyma Explained

The following information is designed to give the reader sufficient knowledge for understanding interactive approaches to Kyma outlined later in this chapter. Carla Scaletti's *Kyma X Revealed!* provides a complete explanation of the workings of Kyma.[57] Many digital sound-synthesis terms are also used in describing this subject. While their use has been kept to a minimum, and the aural result of the process has been described if not self-evident, a detailed explanation of the processes can be found in textbooks such as *Computer Music* by Charles Dodge and Thomas Jerse[58] and *Computer Music Tutorial* by Curtis Roads.[59]

Kyma can be described in many ways; the title of the user manual refers to it as *Kyma Sound Design Environment*,[60] but it can also be seen as a language for specifying, manipulating and combining sounds, as multiple ways of viewing data, or as a system of hardware and software.

The physical components of Kyma form a system that consists of a digital signal processing unit called the Capybara,[61] connected to a computer via a firewire interface. The Kyma program runs on the computer and provides a graphic user-interface to the Capybara. The system supports up to eight channels of input and output, to and from the Capybara. Input signal to the Capybara must be at line level; microphone signals first need to be passed through a microphone pre-amplifier or mixer. Output signal from the Capybara must be passed through an amplifier before distribution to speakers or headphones.

---

[57] Carla Scaletti, *Kyma X Revealed!* (Champaign Illinois: Symbolic Sound Corporation, 2003).
[58] Charles Dodge and Thomas Jerse, *Computer Music* (USA.:Schirmer, 1997).
[59] Curtis Roads, *The Computer Music Tutorial* (Cambridge, Mass.: MIT Press, 1996).
[60] Carla Scaletti, *Kyma Sound Design Environment* (Champaign Illinois: Symbolic Sound Corporation, 1997).
[61] A capybara is a large South American aquatic guinea pig.

"Kyma is a language for specifying, manipulating and combining sounds…"[62] In her 2002 article, *Computer Music Languages, Kyma, and the Future*, Carla Scaletti stresses her definition of Kyma as a language. "A language provides one with a finite set of 'words' and a 'grammar' for combining these words into phrases, sentences, and paragraphs to express an infinite variety of ideas."[63] Scaletti believes that Kyma displays characteristics which contribute to a successful language. Like any language Kyma developed in response to a need (in sound design,) and has a large community of users. As with a spoken language, it is able to express new ideas, and does not impose a particular style on the user. Kyma is also a successful language because, in Scaletti's words, "…the underlying data structure can support extensions and multiple interpretations without violating the original model."[64]

In the above definition of Kyma, Scaletti is deliberate in her use of the term *sound*. "A *Sound* is defined to be a Sound S, a unary function of another Sound *f(S)*, or an *n*-ary function of two or more Sounds *f(s1, s2,…,sn).*"[65] A *Sound* may be a source of sound from an audio input, (Figure 1) an audio file from the hard drive, a noise generator, or a mixer combining any number of sources. (Figure 2) Kyma makes no distinction between live audio input, audio files or synthetically generated signals; they all act as sources and can be manipulated using the same sets of functions.

Figure 1. Graphic representation of a source of audio.

[62] Ibid., 9.
[63] Carla Scaletti, "Computer Music Languages, Kyma, and the Future." *Computer Music Journal* 26, no.4 (2002): 69.
[64] Ibid., 70.
[65] Ibid., 73

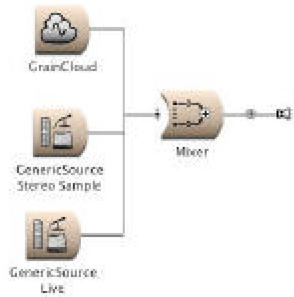

Figure 2. Mixer with a combination of inputs: Each icon represents a Sound.

A Sound may also be a *function* applied to a sound such as equalization. (Figure 3) A function is an algorithm that processes a sound source. Arbitrarily long chains of functions of functions can be constructed.

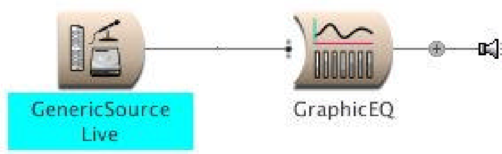

Figure 3. A function applied to a Sound.

Kyma graphically represents each Sound with an expandable, tree-like signal chart flowing from left to right, providing a very intuitive way of viewing these processes. A chain of processes can be viewed as a single entity or expanded into its individual components (Figure 4). Each prototype in Figure 4 has maintained its original title although an alternate name could be applied to one or all of them.

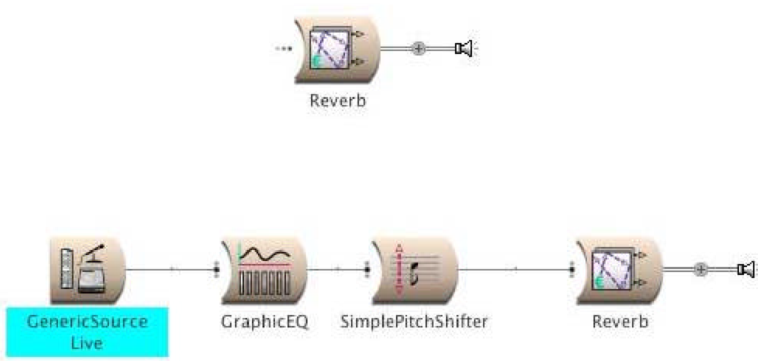

Figure 4. Compressed and expanded views of a Sound.

Scaletti uses the terms *Sound* and *function* for the purpose of her definitions. In Kyma's user interface however, each icon in the above figures represents a *prototype*. The prototypes are the "words" of the Kyma language. Like a language with nouns, verbs and adjectives, some prototypes produce audio "as is" while others execute functions or processes that need to be employed as part of a sequence. Each prototype calculates a specific algorithm. A library of available prototypes is arranged according to purpose; sources and generators, mixing, filters, spectral analysis, and additive synthesis are some of the categories. It is the infinite combinations of prototypes, and internal manipulation of each, that present compositional possibilities.

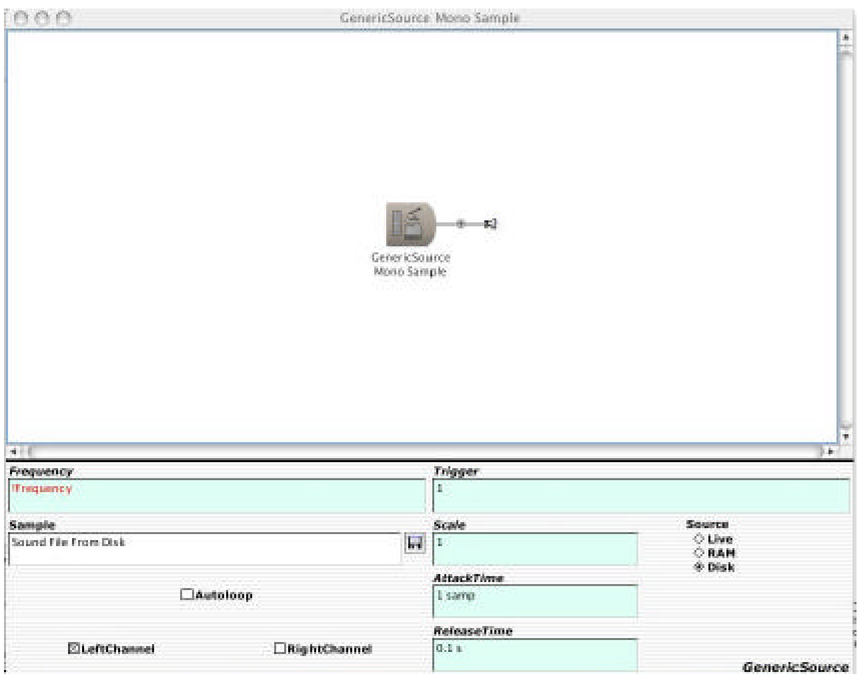

Figure 5. Prototype with parameter fields displayed.

A Sound is viewed in the *Sound editor window*. The top half of the window shows a graphical representation of the signal flow, while the lower half displays the "internal workings" or parameter fields of a selected prototype. All parameters in Figure 5 can be manipulated in some manner: frequency, trigger, sample, scale, source, attack time, release

time, channel selection and looping option. Values in parameter fields preceded by an exclamation mark and written in red text are *hot parameters*. These are variables that can be altered in real time during playback of the Sound. When a prototype is playing, hot parameters take their values from onscreen graphical sliders, are controlled via another Sound, or the desired values for the parameter can be graphed over time.

Pre-combined sequences of prototypes form a Sound Library. These are analogous to "factory presets" found on many synthesizers. From the Sound Library and the prototypes, it is possible for a Kyma user to explore a large range of options by manipulating parameters of the existing Sounds rather than designing new Sounds. Many Sounds will provide good interactive possibilities with clarinet by simply substituting their input for the live-input prototype.

Another way to view and manipulate Sounds is to arrange them on Kyma's multitrack timeline (Figure 6). The timeline's cursor is the visual representation of the passage of time, lacking in the graphic representation of a Sound in the Sound editor screen. Like a traditional written score, sound is arranged over time in this view.

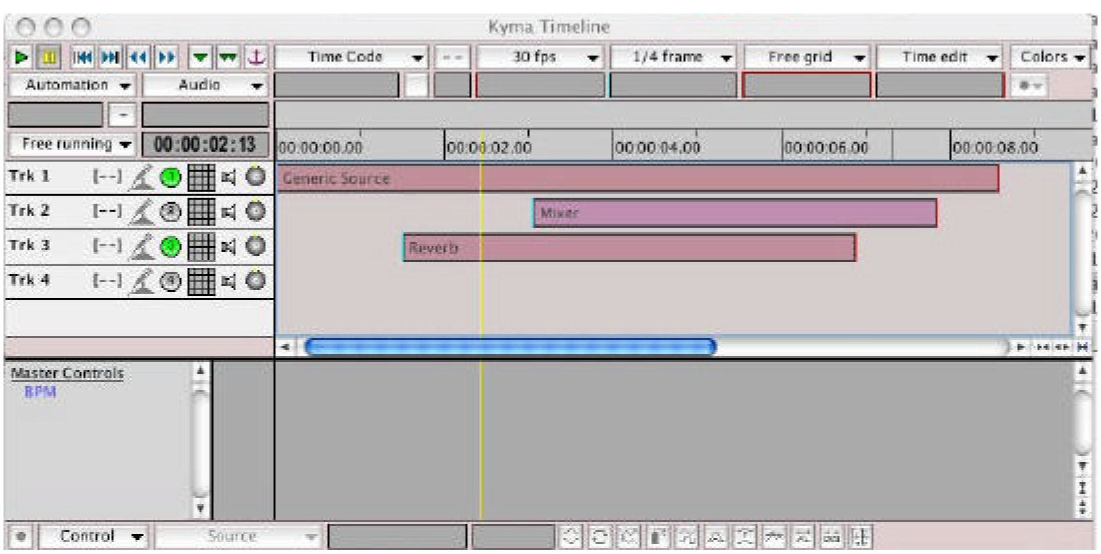

Figure 6. Kyma timeline view.

Before a timeline is played it must be compiled. Each Sound is a program and Kyma schedules the realization of these on the multiple processors in the Capybara. It also compiles any hot parameters for real-time evaluation on the Capybara.

Use of the Kyma System is relatively intuitive; the manual suggests that one can learn the basics of Kyma with 24 hours of study.[66] Kyma's graphic user-interface and high-level language allows one to manipulate algorithms with little knowledge of the underlying programming language required to specify them. Musicians unfamiliar with text-based programming language can explore interaction efficiently.

This explanation is extremely brief and represents only a small part of the workings of Kyma, however it is sufficient background knowledge to understand the possible ways the clarinet can be used to interact with the program.

Interaction with Kyma: The Clarinet as a Controller

For a clarinetist, the interactive possibilities with Kyma divide into two categories: the clarinet as the controller, and the clarinet as a source of sound. The categorization is a result of my work with Kyma. To approach interaction with these categories in mind is helpful in the compositional process; each provides a different context. The performer and listener perceive and understand action and reaction differently for each case. An interactive composition may contain representatives of both categories or may be limited to the type of interaction found in a single category. The following are examples I designed from prototypes that are particularly suited for use with clarinet input.

The SoundToGlobalControl[67] prototype (Figure 7), provides an interface that may be inserted into a parameter field of another sound. SoundToGlobalControl takes a source of

[66] Carla Scaletti, *Kyma Sound Design Environment* (Champaign Illinois: Symbolic Sound Corporation, 1997), 89.

[67] Prototype titles in Kyma contain no spaces between each word.

audio, or the output of another function as its input and generates a corresponding output expressed as either a single figure or a continuous controller stream between 0 and 1. The output values can be scaled by simple addition or multiplication to cover any range. To all other Sounds in Kyma, the output of the SoundToGlobalControl prototype appears the same as the value of a hot parameter.

Examples of the use of the SoundToGlobalControl prototype in a parameter field of another sound will be presented later. The following is a description of how the clarinet, via an AmplitudeFollower, FrequencyTracker or EnergyAtFrequency prototype may provide input for SoundToGlobalControl.

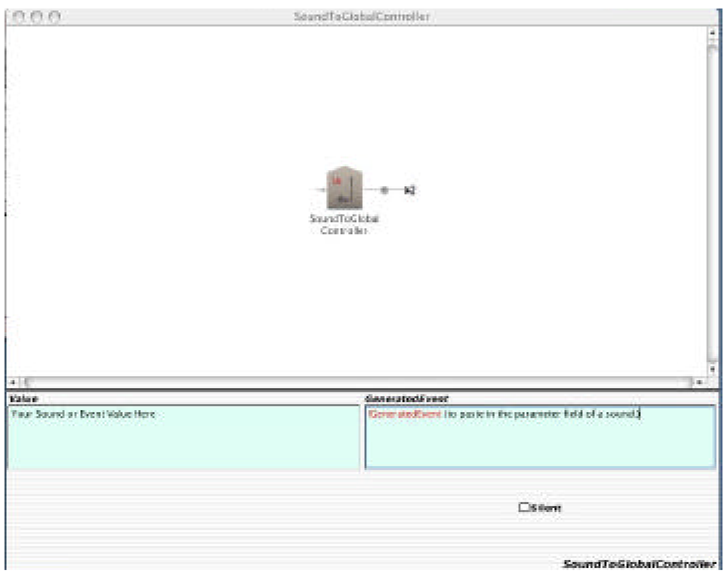

Figure 7. SoundToGlobalControl prototype.

In Figure 8, the audio input from the clarinet (graphically represented by the "Clarinet Input") is passed to the AmplitudeFollower prototype. The AmplitudeFollower passes its output on to the SoundToGlobalControl. A value between 0 and 1 is produced that corresponds to the loudness of any pitch the clarinet plays. The AmplitudeFollower produces reliable results; the main adjustment involves scaling the softest sound that will be played by the clarinet to 0 and the loudest to 1.

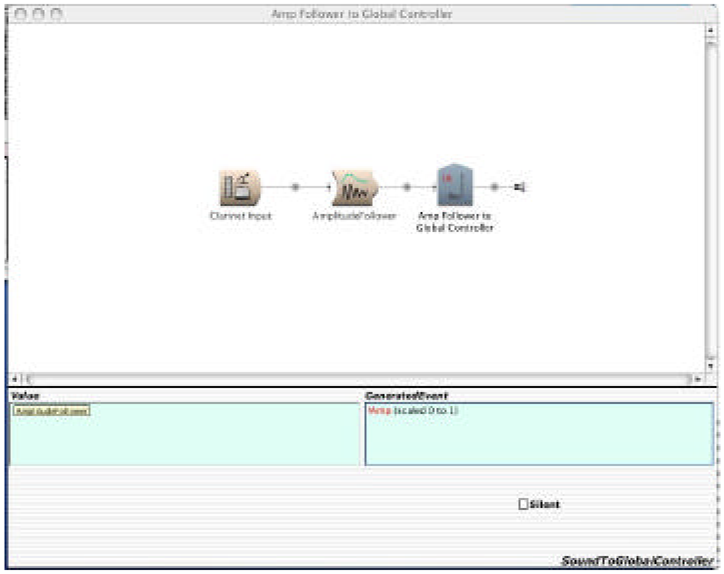

Figure 8. AmplitudeFollower to SoundToGlobalControl.

Alternatively, a FrequencyTracker prototype may be employed as the input to the SoundToGlobalControl. (Figure 9) In this instance, the *pitch* of the clarinet is mapped onto a 0 to 1 scale. The user must specify the expected lowest and highest notes in parameter fields of the FrequencyTracker. A narrow input frequency range results in more accurate tracking. There are several ways to express pitch in Kyma, but the most convenient for clarinet input is *midi note number*; middle *C* is note number 60 and each semitone is a whole number above or below this. The clarinet's low *E* is midi note number 50 while a third register *G* is note number 93. Frequency tracking can be somewhat unpredictable; pitches in the third register of the clarinet and fast changes between notes are difficult to track. If these situations are avoided, frequency tracking remains an effective option.

In the next example, the clarinet sound is passed through EnergyAtFrequency and Threshold prototypes before input to the SoundToGlobalControl. The EnergyAtFrequency prototype, whose parameters are shown in the lower half of the Sound editor screen in Figure 10, outputs an amplitude envelope showing the amount of energy at a specified frequency. This becomes the input for a Threshold prototype. The output of a Threshold is 1 when its

input amplitude exceeds the specified threshold; otherwise it is 0. Therefore, the SoundToGlobalControl will output a single event; the value of 1 when the clarinet input is at the note specified, and 0 when any other note is played.

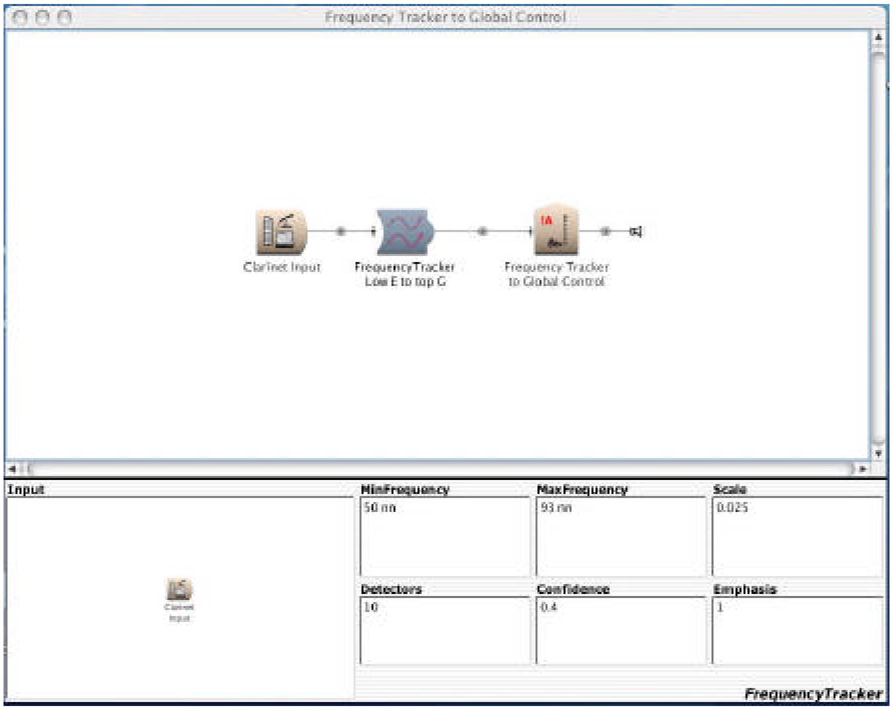

Figure 9. FrequencyTracker to SoundToGlobalControl.

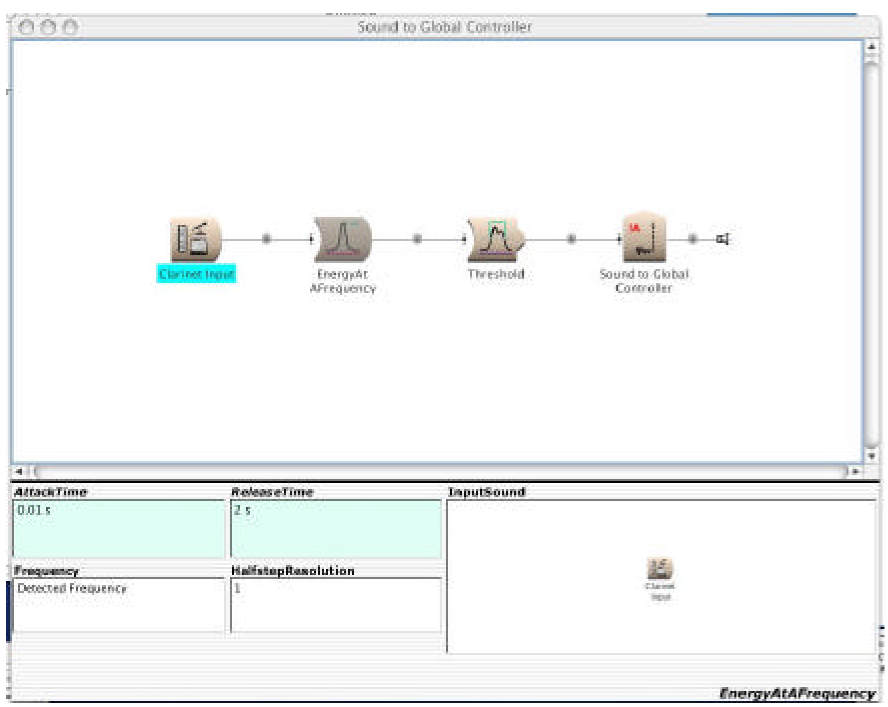

Figure 10. EnergyAtFrequency to GlobalControl.

The examples above show the two types of output from SoundToGlobalControl: either a continuous controller stream corresponding to the changing amplitude or pitch of the clarinet, or a single value of 0 or 1 depending on detection of a specified pitch. To operate as a hot parameter, the SoundToGlobalControl prototype is copied and pasted into the parameter field of another Sound. There are countless situations where this can occur as most prototypes accept hot parameters in many of their parameter fields. The following selected examples show a variety of ways the clarinet may act as a controlling element of other Sounds via the SoundToGlobalControl.

Control of the pan parameter of a Pan prototype is possible with the SoundToGlobalControl. In the pan parameter, a value of 0 positions the sound entirely in the left speaker, and a value of 1 positions it entirely in the right. Values between those extremes cause the input source appear as if it were placed somewhere in-between the two speakers. Figure 11 shows a FrequencyTracker input to the SoundToGlobalControl and therefore placement in space of "A Sound" in Figure 11 will correspond to the pitch of the clarinet. A high note on the clarinet, for example, would produce an output of near 1 and cause "A Sound" to be placed mostly in the right speaker. Similarly use of an AmplitudeFollower to SoundToGlobalControl in this instance would cause "A Sound" to move in space according to the loudness of the clarinet.

The pitch of the clarinet can control the frequency and speed of playback of "A Sound" in Figure 12 through the use of a SoundToGlobalControl in the Frequency parameter of the Sound. For the example shown in Figure 12, the frequency units specified in the parameter are Hertz (Hz). In this case some simple mathematical adjustment of the original 0 to 1 output scale is made since the maximum 1Hz tone from the SoundToGlobalControl

would be below the minimum audible frequency of 20Hz. Multiplication of the SoundToGlobalControl output by 100, for example, will scale the range from 0 to 100 Hz. Offsetting the scale by addition of say 200, ensures the minimum frequency will be 200 Hz. (Click here for Audio Example 1.)

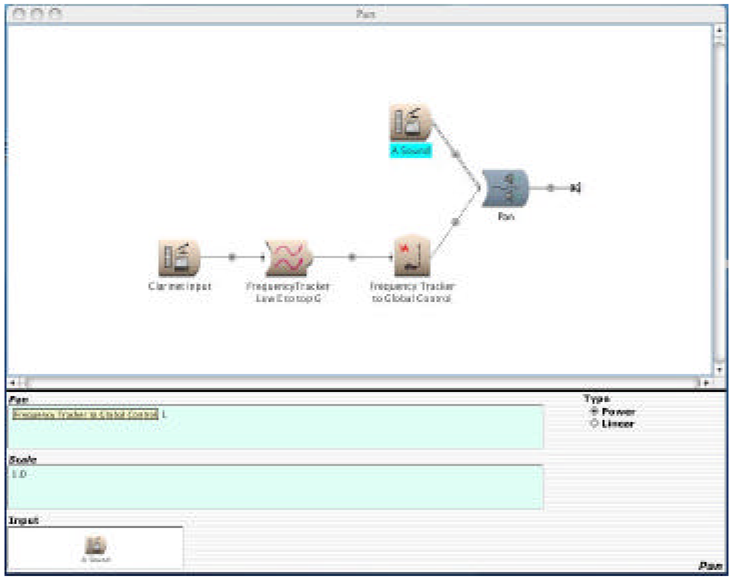

Figure 11. FrequencyTracker to SoundToGlobalControl controlling position in space of a sound.

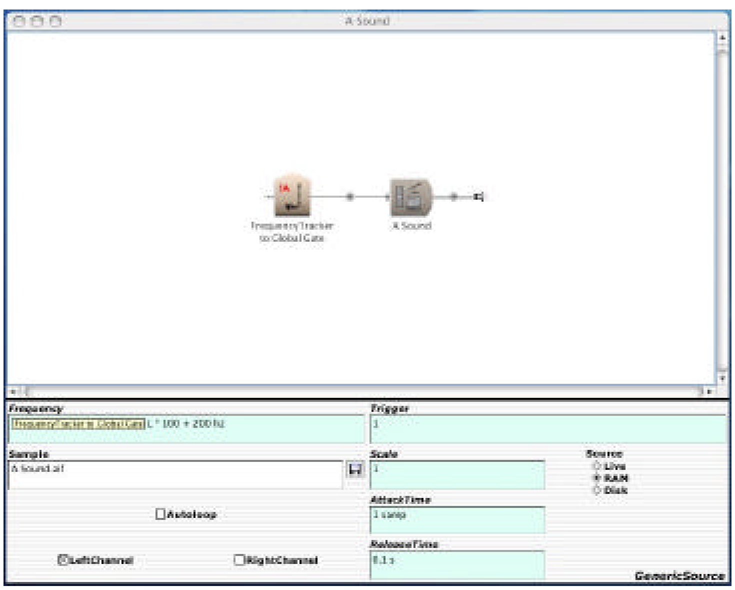

Figure 12. Frequency of a Sample controlled by a FrequencyTracker via SoundToGlobalControl.

Another interactive use of the Sound to Global Control prototype is to control the position in time of the resynthesis of a spectrum file. Figure 13 shows a SumOfSines prototype in which an AmplitudeToGlobalControl has been pasted into the Time Index parameter. The Time Index is like a "pointer" to a particular snapshot of time in a file. Moving the pointer through a file, in either direction, reads it backwards or forwards at the rate at which the pointer is moved. The amplitude of the clarinet controls the position of the pointer. In this prototype, the beginning of the sound file is at –1, and the end at +1. Some mathematical manipulation can expand a 0 to 1 scale to a –1 to +1 scale; multiply by 2 and subtract 1. A crescendo on the clarinet causes the SumOfSines prototype to move the pointer from –1 to +1 and thus read the file from beginning to end. A fast crescendo causes it to read the file quickly and a decrescendo results in a backwards rendition. (Click here for Audio Example 2.)

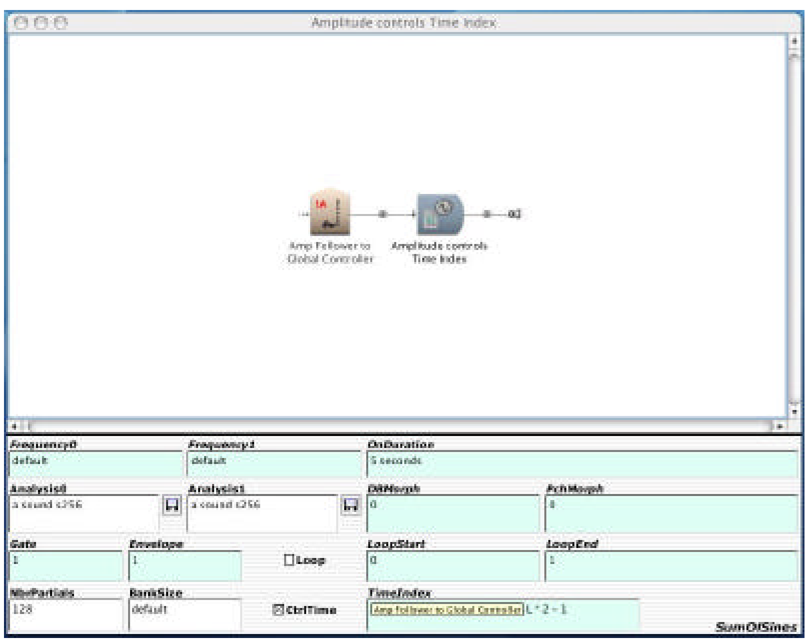

Figure 13. Amplitude controls the Time Index of a sound file.

Figure 14 shows, via a SoundToGlobalControl, the use of a specific pitch on the clarinet to trigger the start of another Sound. The EnergyAtFrequency prototype, via the

SoundToGlobalControl has an output of 1 if the specified note is played by the clarinet and 0 if not. The GenericSource prototype will play a sound file every time the value of the *Trigger* parameter is greater than 0. A specified note on the clarinet becomes the "switch" to start another sound by pasting this SoundToGlobalControl into the trigger parameter of a GenericSource prototype. (Click here for Audio Example 3.)

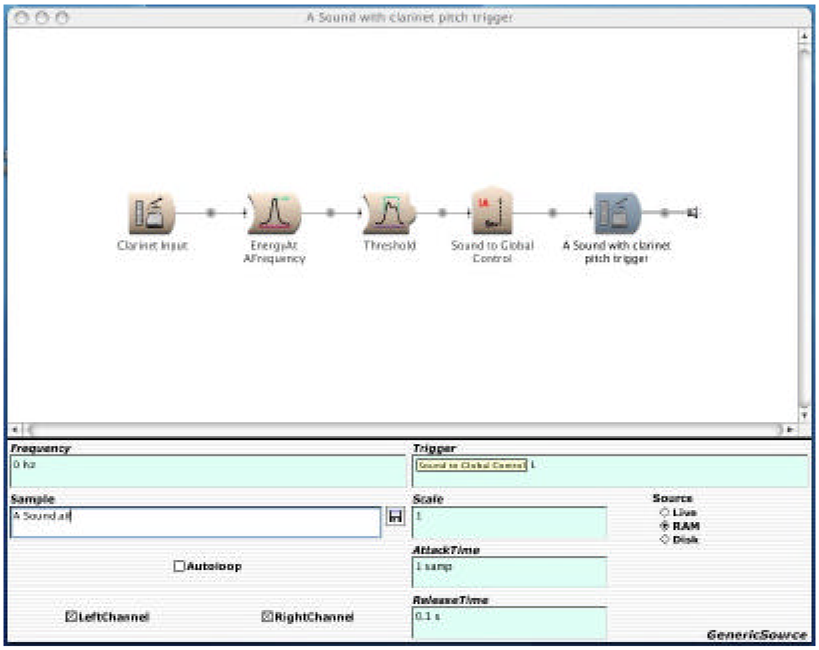

Figure 14. SoundToGlobalControl as a Trigger.

The final example of the use of SoundToGlobalControl is with the WaitUntil prototype (Figure 15). WaitUntil can be placed in a Kyma timeline and will halt the progression of time at that point until the value of the Resume parameter is a positive integer. This can be controlled via the EnergyAtFrequency prototype so that a specific pitch produces the desired positive integer. Employed in a timeline, it allows a player to control the progression from one section to another through, in this example, a specific pitch. Amplitude can also act as the trigger if one sets a threshold so that above a certain dynamic, the WaitUntil is triggered.

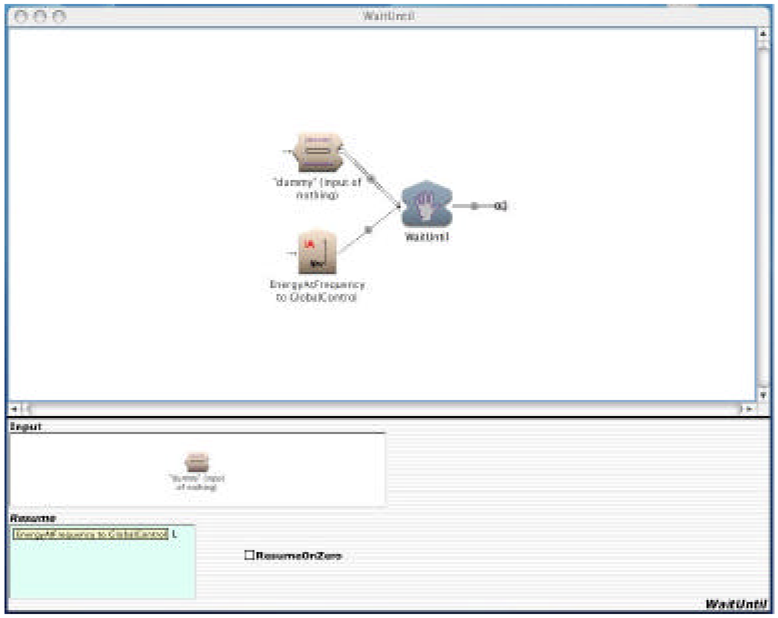

Figure 15. WaitUntil.

In conclusion, when the clarinet is used as a control device, the listener perceives a distinct type of interaction in which it is not the *sound* of the instrument, but rather a *parameter* of that sound such as pitch or amplitude that controls a response. The SoundToGlobalControl provides a versatile interface for this to happen. The number of possible parameters in which this prototype can be used presents unlimited options for experimentation with this type of interaction.

<u>The Clarinet as a Sound Source</u>

When the clarinet is a *sound source*, interaction is perceived differently to that when it is a controller. As a sound source, the listener perceives a processing of the actual clarinet sound rather than an interaction between a parameter of the sound and another event. There are countless ways a clarinet can provide the sound source for interaction with Kyma. Many of the processes that have been available as guitar effects pedals such as reverberation, flanging and delay are prototypes in Kyma. There are also more complex processes such as spectral analysis and resynthesis, which have only recently been available for real-time

interaction and provide effects less familiar to a listener. The following are selected examples of a range of processes applied to the clarinet input.

A SimplePitchShifter applied to the clarinet input results in a direct response to the input; an interaction which is understood clearly by the listener (Figure 16). There is little latency between action and reaction, the timbre of the input and output are similar, and the process of transposing the pitch is not obscure. A hot parameter in the Interval parameter field controls shifting the input's pitch up or down; a shift of 1 is relative to a half step, although any increment can be used. (Click here for Audio Example 4.)

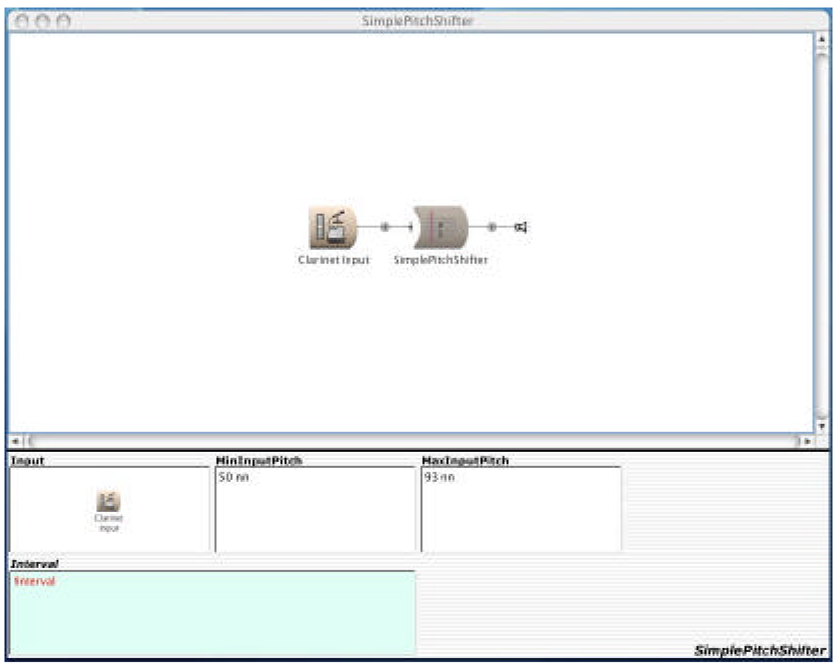

Figure 16. SimplePitchShifter.

Figure 17 is an example of ring modulation showing the parameter fields of the Oscillator in the Sound. Selected from the collection of prototypes, this example has been altered with the simple modification of the input to a live source. The clarinet input is the *carrier frequency* and its amplitude value, pasted into the frequency parameter of an oscillator, provides the *modulating frequency*. These are multiplied together in the final prototype to produce a modified clarinet sound. Although classic ring modulation produces

sidebands with no carrier frequency, in this example the carrier frequency is still present as the live clarinet sound. A change in pitch on the clarinet will produce a different set of sidebands; likewise, a change in dynamic of the clarinet on any held pitch will result in a change in modulating frequency. Both action and reaction in this process are contemporaneous and it is clear to a listener that interaction is taking place and one sound is affected by the other. (Click here for Audio Example 5.)

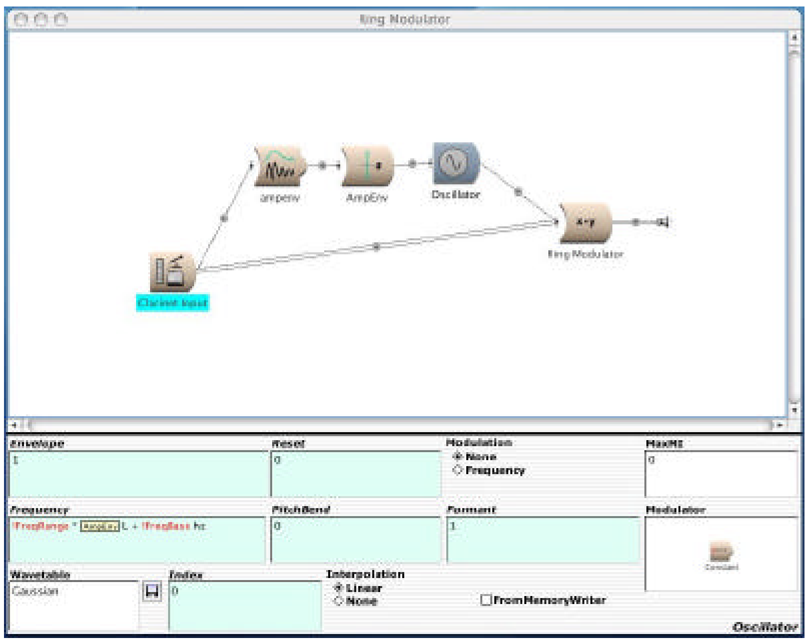

Figure 17. Clarinet Input with RingModulation.

Figure 18 shows a complex Sound from the prototype collection entitled "Freeze & Scramble", which requires analysis and resynthesis of the input. The Clarinet Input is subjected to a live spectral analysis and via a ChannelCrosser, the amplitude and frequency information is separated. Both are processed in a DelayWithFeedback (freezeAmp/ freezeFreq), rejoined, and passed on to a SpectrumModifier. In modifying the output of a spectral source, the SpectrumModifier selects or rejects tracks of the spectrum according to criteria defined in parameter fields, and then it optionally scales and offsets each frequency

and/or amplitude value of the selected tracks. This spectrum is resynthesized through an oscillator bank.

The effect is perceived by a listener as the title describes; the clarinet sound is frozen in time, scrambled, and parts of it replayed. The player has control over the general range of the output frequency and amplitude as this is related to the pitch and dynamic of the clarinet input; this interaction will be perceived by a listener. On the other hand, the "rando" prototype, pasted in the FreqScale and AmpScale parameter fields, contributes a randomness to the amplitudes and frequencies within the specified range and over which the player has no control. (Click here for Audio Example 6.)

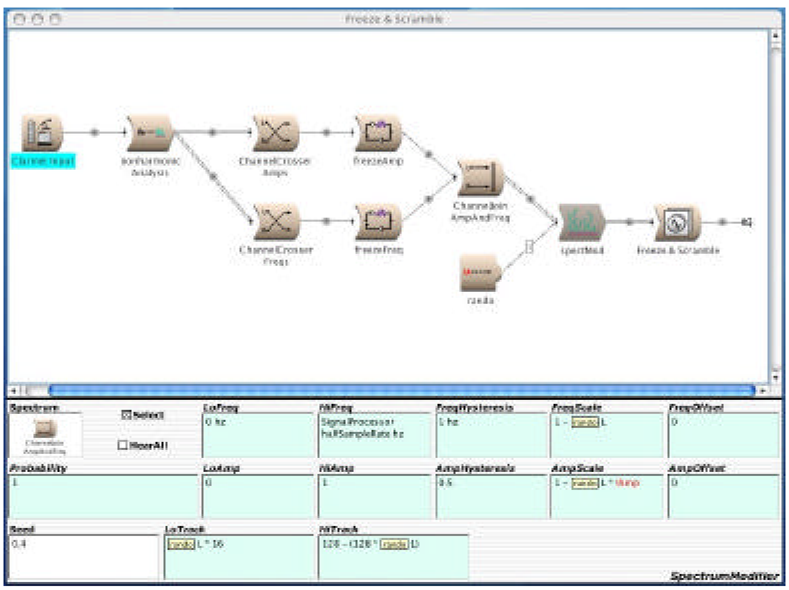

Figure 18. Freeze & Scramble showing parameter fields for the SpectrumModifier.

Morphing from one sound to another through the use of analysis and resynthesis techniques can result in a distinctive effect. Spectral parameters of the clarinet sound can morph into those from a seemingly unrelated sound. The listener hears a sound that contains elements from two sources; the interaction is one of transformation between two different sounds rather than a processing on a single sound.

In Figure19, the clarinet input is subjected to a LiveSpectralAnalysis, which produces amplitude and frequency envelopes for controlling a bank of oscillators. In this example, information on the amplitude and frequency of a second sound is provided by a SpectrumInRam prototype. The SpectrumInRam reads the spectral analysis of a previously analyzed file and outputs the spectrum as a sequence of amplitude and frequency pairs. The Interpolate prototype combines the two input sources in a controlled manner. The LeftInterp parameter field controls the amplitude envelopes while the RightInterp controls the frequency envelopes. In this example, the LeftInterp parameter field is set at 1, resulting in the reproduction of the amplitude envelopes from only the SpectrumInRam. The RightInterp is controlled by a hot parameter and can morph the frequency envelopes between the two inputs. The clarinet pitch can be superimposed gradually over the other sound source without changing the amplitude envelope.

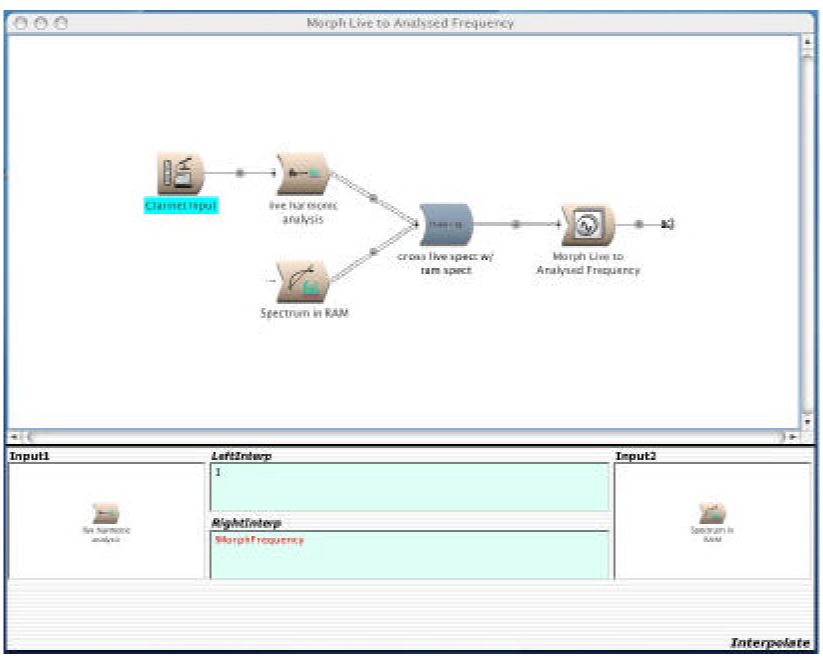

Figure 19. Morphing live frequency to another sound.

This example shows that a similar effect can be achieved using either the clarinet as a controller or the clarinet as a sound source. A "fully morphed" frequency in the SumOfSines

achieves the same affect as a FrequencyTracker controlling frequency of a Sound via a SoundToGlobalControl. The added capability of the SumOfSines is that by morphing, one can move gradually between the "real" frequency of the sample and the morphed version from the clarinet. (Click here for Audio Example 7.)

There are countless processes that can be applied to the clarinet sound. Kyma allows one to experiment with all prototypes, usually by substituting the default input for a GenericSource (live clarinet) prototype. Through intuition, a user can quickly determine if a certain process has potential application with clarinet. The final step in the compositional process is to distribute the Sounds over time and adjust their hot parameters.

Chapter 6. Application of Interactive Techniques: *Prepared Brahms*

The following is a description of how I used Sounds designed in Kyma for an interactive composition with clarinet. *Prepared Brahms,* for electronically prepared clarinet, uses specific pitches and ranges of pitches to trigger processes and explores interaction using the clarinet both as a controller and as a sound source. In this work, the clarinet is electronically "prepared" in a way similar to a prepared piano in which objects are placed between the strings so that each key sounds a different timbre or effect. Rather than physically modifying tone holes and mechanisms on the instrument, Sounds in Kyma are triggered by specific pitches on the clarinet. Through an EnergyAtFrequency/ Threshold/ SoundToGlobalControl, a pitch on the clarinet triggers that note as the input to a process.

Any number of pitches can be assigned as triggers for a single process by joining SoundToGlobalControl prototypes and their triggered clarinet input in a Mixer prototype (Figure 20). The input to a function can also be triggered by a group of adjacent pitches by adjusting the HalfstepResolution parameter in the EnergyAtFrequency prototype. A figure of 2, for example, will give a range of a whole tone either side of the specified pitch.

While the trigger remains "on" when the specified notes are played, there are also adjustment possibilities on the attack and decay parameters of the of clarinet input. Adjustment of a GenericSource prototype's decay parameter will affect the amount of time clarinet sound is allowed into the process *after* the input has moved to other non-trigger pitches. This "bleeding" of a process to other pitches can make the interaction more obscure, however, it also allows for a more natural decay and avoids the abrupt cut-off produced when a non-trigger note is played.

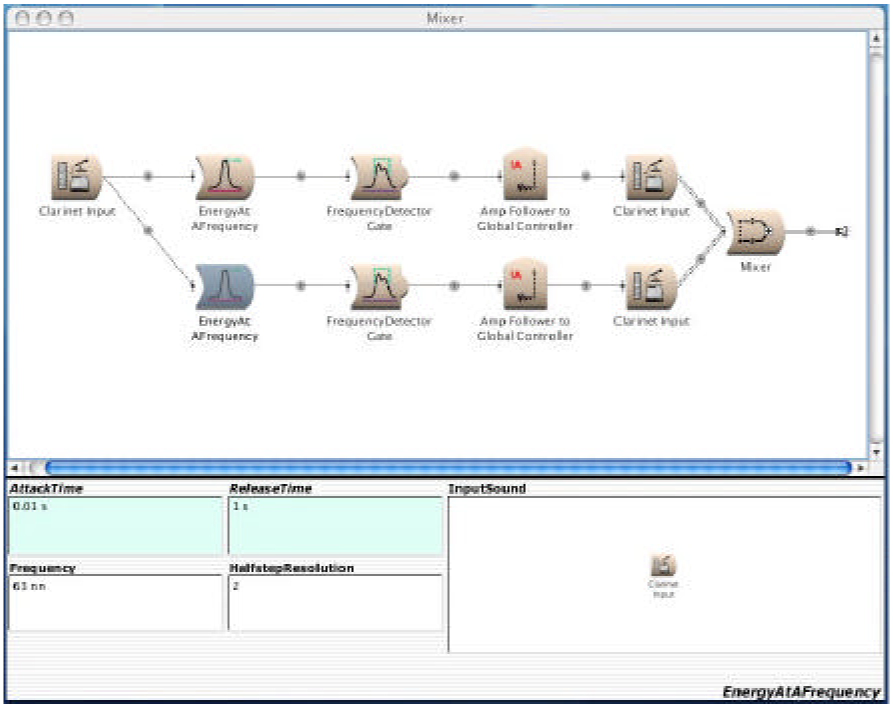

Figure 20. A mix of two separate triggered inputs.

Figure 21 is a view of the first 8 seconds of the timeline showing all the Sounds (except for track 4 which enters later). The annotations on the colored bars are the titles of the Sounds and the pitches and resolution at which the input is triggered. The yellow line (the timeline's cursor) is paused at the first Sound, WaitUntil. In performance, this allows complete control over the beginning of the work. Kyma compiles the processes and will begin playing immediately the condition is reached that triggers the WaitUntil. In this case it is the first pitch of the piece.

The graph in the lower half of the window of Figure 21 shows the shifting of the Formant parameter of the Sound in the first track. This is an example of a hot parameter that is graphed.

Figure 22 shows the timeline with the Interval parameter of the SimplePitchShifter graphed to alter over time. Note numbers 50 and 60 trigger the pitch shift, but the amount it is shifted varies between an octave (12 increments on the graph) up or down, depending on

where the timeline is at that specific moment. The listener's perception of interaction is that of a process, in this case a pitch shift, rather than expecting a specific interval from the trigger note. (Click here for Audio Example 8.)

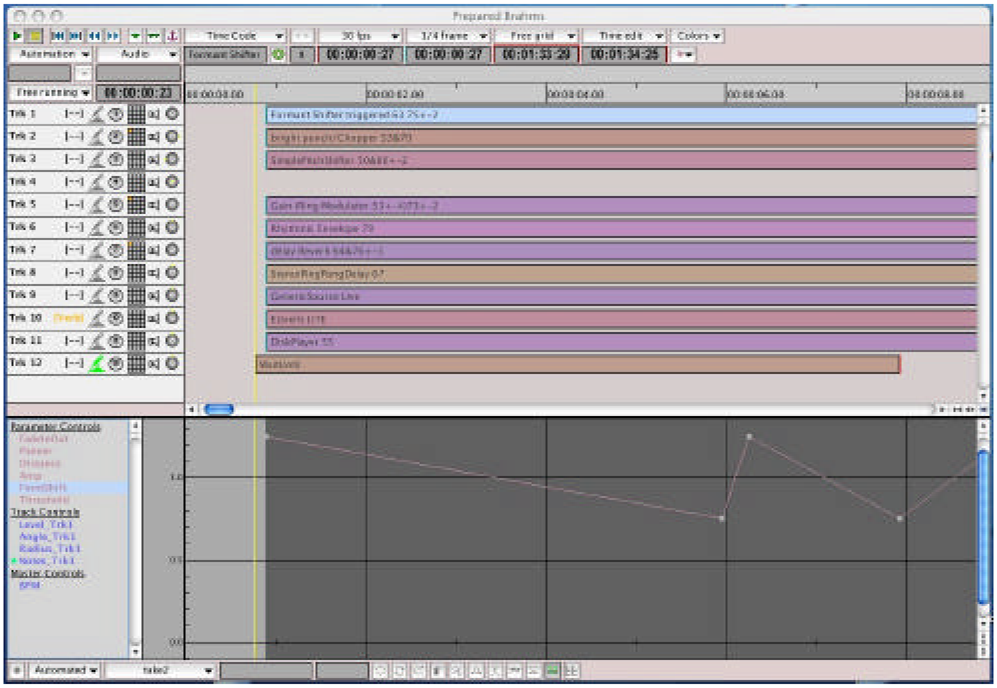

Figure 21. Timeline view of *Prepared Brahms.*

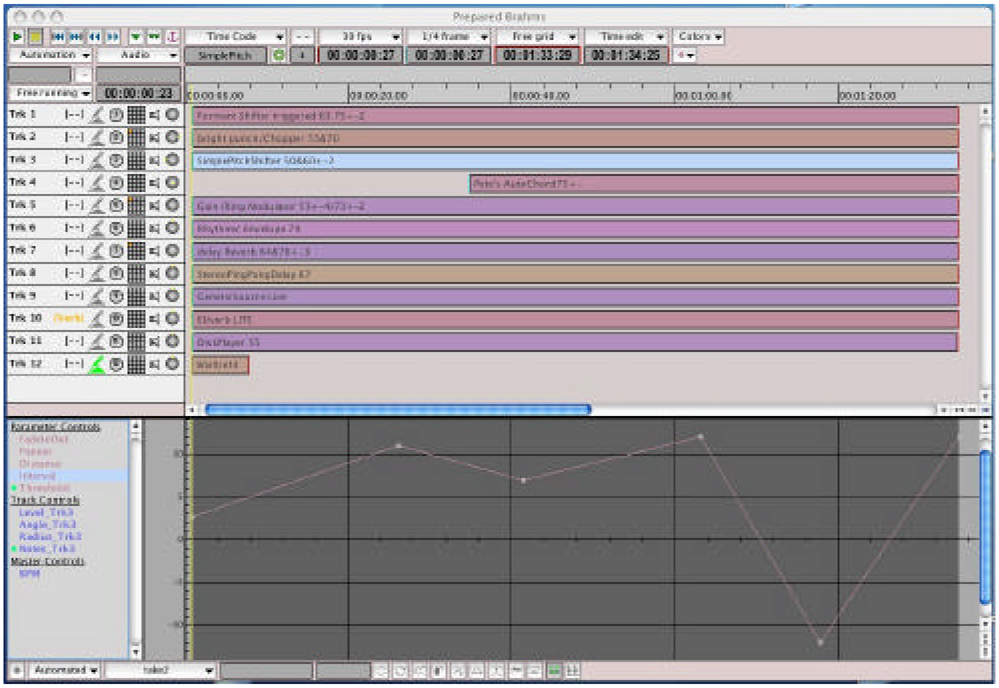

Figure 22. Timeline view showing the graphing of the track 3 pitch shift parameter.

The interactive techniques used in *Prepared Brahms* show only some of numerous applications of the clarinet as either a controller or a sound source. This work provides an interesting example that extends the concept of a prepared instrument into the electronic world using the clarinet as both controller and source. There are unlimited ways a composer can combine Sounds and explore interactive possibilities; approaching the use of the clarinet (or any instrument) with the two categories in mind is an efficient beginning to the compositional process.

Chapter 7. Conclusion

This document has provided information for a clarinetist interested in interactive computer music. It addresses relevant issues surrounding the genre and specific information on practical application with regard to Kyma. As there is little information of this nature, the contents represent considerable time in the trial and error of such things as microphone choice or use of particular prototypes in Kyma, that need not be repeated. Indicative of the relative newness of the genre, many areas related to the topic present opportunity for considerably more research; that of the listener's perception of interaction is outlined below.

Categorization of the use of the clarinet as either a controller or a source of sound is the unique perspective of an instrumentalist working with Kyma. This document highlights the difference between the two uses of the instrument; these categories provide a method of approaching interactive composition with reference to both the instrumentalist's and listener's perceptions of interaction.

A clarinetist's traditional education does not cover the understanding and use of microphones for the instrument. While some clarinetists are using microphones for sound reinforcement in jazz music, few players are knowledgeable of the requirements and products available for live interactive computer music. This document has provided a basic understanding of the different properties of microphones and options for use in interactive systems. While new products may come on the market, the particular requirements for use with interactive systems will remain the same.

The experience of live interactive music is intricately connected to the actual process of interaction. For a performer it is the exploration of the process between input and output, while for a listener, it is the recognition of the relationships between action and reaction.

There is little published research about the listener's perception of interaction with respect to interactive computer music and yet the presentation of ways in which a clarinetist can interact with Kyma is most clearly explained through the listener's likely perception of that interaction. This field presents some interesting opportunities for research. Study of the listener's perception of interaction would enhance the understanding of interactive computer music and provide a new set of tools to work in conjunction with the traditional analysis of musical parameters such as form, harmony and pitch content.

This genre presents an engaging, dynamic performance possibility and currently has only a handful of practitioners. User-friendly programs such a Kyma make it feasible for almost any performer to master the technology and augment their performance opportunities. Many potential employment opportunities exist in the performance of interactive music and additionally in the application of this technology in the related areas such as installations, multimedia, collaborative works and sound design.

Bibliography

Appleton, Jon. "Reflections of a Former Performer of Electroacoustic Music." *Contemporary Music Review* 18:3 (1999): 15-19.

Attali, Jacques. Brian Massumi trans. *Noise: The Political Economy of Music*. Minneapolis: University of Minnesota Press, 1985.

Baird, B., D. Blevins, and N. Zahler. "Artificial Intelligence and Music – Implementing an Interactive Computer Performer." *Computer Music Journal* 17, no.2 (1993): 73-79.

Ballora, Mark. "Get With The Interaction." [Article on-line]; available from; http://emusician.com/mag/square_one/emusic_interaction/ ; Internet: accessed 16 November 2004.

Battier, Marc, ed. *The Aesthetics of Live Electronic Music.* Overseas Publishers Association, published by license under the Harwood Academic Publishers imprint, 1999

Bertini, G and L. Tarabella. "About The Role of Mapping in Gesture-Controlled Live Computer Music." *Lecture Notes in Computer Science. Computer Music Modeling and Retrieval* 2771 (2004): 217-224

Burke, P., L. Polansky, and D. Rosenboom. "HMSL (Heirachial Music Specification Language) – A Theoretical Overview." *Perspectives in New Music* 28, no.2 (1990): 136-178.

Camurri, A., P. Coletta, M. Ricchetti, and G. Volpe. "Expressiveness and Physicality in Interaction." Journal of New Music 29, no.3 (2000): 187-198.

Camurri, Antonio. "Expressive Gesture," in *Music and Artificial Intelligence : Second International Conference, 2002*. Edinburgh, Scotland: 12-14 September 2002, edited by Christina Anagnostopoulou. Berlin: Springer-Verlag, 2002.

Chadabe, Joel. *Electric Sound: The Past and Promise of Electronic Music.* Upper Saddle River, N.J.: Prentice-Hall, Inc., 1997.

________. "The Performer is Us." *Contemporary Music Review* 18:3 (1999): 25-29.

Chafe, Chris. "Interplay(er) Machines." *Contemporary Music Review* 18:3 (1999): 89-97.

Cox, Christoph and Daniel Warner editors. *Audio Culture. Readings in Modern Music.* Continuum New, York, 2004.

Dodge, Charles and Thomas Jerse. *Computer Music*. USA.:Schirmer, 1997.

Druhan, Mary Alice. “A Performer's Guide to Multimedia Compositions for Clarinet and Visuals : A Tutorial Focusing on Works by Joel Chabade, Merrill Ellis, William O. Smith, and Reynold Weidenaar.” D.M.A. diss., Louisiana State University, 2003.

Emmerson, Simon. *The Language of Electroacoustic Music*. London: Macmillan, 1986.

________. *Music, Electronic Media and Culture.* Aldershot, England & Burlington, VT, Ashgate, 2000.

Errante, F Gerrard. *Contemporary Clarinet Repertoire :Clarinet & Electronics.* Mentone, CA.: Mill Creek Publications, 1993.

Garnett, GE. “The Aesthetics of Interactive Computer Music.” *Computer Music Journal* 25, no.1 (2001)

Godøy, Rolf Inge. “Gestural Imagery in the Service of Musical Imagery,” in *Music and Artificial Intelligence : Second International Conference, 2002*. Edinburgh, Scotland: 12-14 September 2002, edited by Christina Anagnostopoulou. Berlin: Springer-Verlag, 2002.

Gould, Glenn. *The Glenn Gould Reader.* New York: Knopf, 1984.

Haus, Goffredo, ed. *The Computer Music and Digital Audio Series - Music Processing,* vol. 9, *Applications of Artificial Intelligence Methodologies and Tools for Musical Description and Processing,* by A. Camurri. Madison, Wisconsin: A-R Editions, Inc, 1993.

Lippe, C. “Real-Time Interactive Digital Signal Processing: A View of Computer Music.” *Computer Music Journal* 20, no.4 (1996): 21-24.

Manning, Peter. *Electronic and Computer Music.* New York: Oxford University Press, 2004.

Pope, Stephen T. *The Well-Tempered Object : Musical Applications of Object-Oriented SoftwareTechnology.* Cambridge, Mass.: MIT Press, 1991.

Risset, Jean-Claude. "Composing in Real-time?" *Contemporary Music Review* 18:3 (1999): 31-39.

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

Rowe, Robert. "The Aesthetics of Interactive Music Systems." *Contemporary Music Review* 18:3 (1999): 83-87.

________. *Interactive Music Systems : Machine Listening and Composing.* Cambridge, Mass.: MIT Press, 1993.

________. *Machine Musicianship* Cambridge, Mass.: MIT Press, 2001.

Scaletti, Carla. “Composing Sound Objects in Kyma.” *Perspectives of New Music* 27, no.1 (1989): 42-69.

________. “Computer Music Languages, Kyma, and the Future.” *Computer Music Journal* 26, no.4 (2002):

________. “Kyma : An Interactive Graphic Environment for Object-Oriented Music Composition and Real-Time Software Sound Synthesis Written in Smalltalk –80” M.Comp.Sc. diss., University of Illinois at Urbana-Champaign, 1989.

________. “The Kyma Platypus Computer Music Work Station.” *Computer Music Journal* 13, n0.2 (1989): 23-38

________. *Kyma Sound Design Environment* Champaign, Illinois: Symbolic Sound Corporation, 1997.

________. *Kyma X Revealed!* Champaign, Illinois: Symbolic Sound Corporation, 2003.

Stroppa, Marco. "Live Electronics or…Live Music? Towards a Critique of Interaction." *Contemporary Music Review* 18:3 (1999): 41-77.

Subotnick, Morton. "The Use of Computer Technology in an Interactive or 'Real Time' Performance Environment." *Contemporary Music Review* 18:3 (1999): 113-117.

Taylor, Timothy Dean. *Strange Sounds : Music, Technology & Culture*. New York: Routledge, 2001.

Winkler, Todd. *Composing Interactive Music: Techniques and Ideas Using Max*. Cambridge, Mass.: MIT Press, 1998.

________. “Interactive Signal Processing for Acoustic Instruments.” In *Proceedings of the 1991 International Computer Music Conference, Montreal.*

Appendix: Permission to Use Screen Shots of Kyma

Sent by: Symbolic Sound Corporation <symsound@SymbolicSound.com>
To: Roland A Karnatz <rkarna1@lsu.edu>
cc:
Subject: Re: Permission for screen shots to be used in a thesis

Dear Roland,

Thanks for your email. There is no problem with including screen shots of your Sounds in your dissertation. BTW, once your dissertation is finished, will you have a copy on your website? (I'd love to read it).

I'm happy to hear that you're finding Kyma useful for your work!

Look forward to meeting you next month.

Best regards,

-Carla

>I am an Australian clarinettist finishing my DMA at Louisiana State
>University. My minor is in electro-acoustic composition; Steve Beck is my
>professor and I've spent quite a bit of time working with Kyma here.
>
>My final lecture recital/ written document is "Interactive Computer Music:
>A Performer's Guide to Kyma with Live Clarinet Input." I would like to use
>screen shots of various Sounds in Kyma to illustrate ways of interaction
>for the written document - maybe 25 or so. I'm not sure if formal
>permission is required to do this, but I would appreciate it anyway.
>
>Thank you
>
>Roland Karnatz
>
>PS. Kyma has been the single best "discovery" in my doctoral studies!
>
>I'm really looking forward to your visit in March.
--
==
| Symbolic Sound Corporation |
| P. O. Box 2549 * Champaign, IL 61825-2549 * USA |
| Tel: +1-217-355-6273 * Fax: +1-217-355-6562 |
| Email: symsound@SymbolicSound.com |
| URL: http://www.SymbolicSound.com |

Vita

Roland Anton Karnatz is an Australian musician. As a clarinetist he has performed with the Baton Rouge Symphony, Acadiana Symphony, Natchez Opera Festival Orchestra, Williamsburg Symphonia and Williamsburg Ballet Orchestra (USA), the International Chamber Ensemble and Ad Libitum Sinfonietta (Italy), and the Malvern Symphony Orchestra (Melbourne). Roland has composed interactive works for Kyma with clarinet, bass clarinet, Theremin, didgeridoo and voice. He has been sound designer for several theatre productions and artistic director of many multimedia collaborative events. He is currently Technical Director for the Operafestival di Roma. Roland received the degree of Bachelor of Arts in Music from the Victorian College of the Arts in 1990, a Graduate Diploma in Music from the Victorian College of the Arts in 1992, and the degree of Master of Music from Virginia Commonwealth University in 2000.